

# 데이터 마이닝 특강

## Practice session

[Association Rule Mining and Text mining for Social Trend Analysis]

Junseok Park

2018-08-30



# Classification review

---

- Practice of classification algorithms
  - Decision Tree
  - Multi Layer Perceptron
- Real-word data practice
  - MNIST

# Overview

---

- **Association Rule Mining**

- Introduction
- Practice

- **Text mining**

- Twitter

# **Association Rule Mining**

# Association Rule Mining

---

- Given a set of transactions, we find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction

Market-Basket transactions

<i>TID</i>	<i>Items</i>
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Association Rules

$\{\text{Diaper}\} \rightarrow \{\text{Beer}\}$

$\{\text{Milk, Bread}\} \rightarrow \{\text{Eggs, Coke}\}$

$\{\text{Beer, Bread}\} \rightarrow \{\text{Milk}\}$

# Definition: Frequent Itemset

---

- **Itemset**

- A collection of one or more items
  - e.g., {Milk, Bread, Diaper}
- $k$ -itemset
  - An itemset that contains  $k$  items

- **Support count ( $\sigma$ ) or absolute support**

- Frequency of occurrence of an itemset
  - e.g.,  $\sigma(\{\text{Milk, Bread, Diaper}\}) = 2$

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

- **(Relative) support**

- Fraction of transactions that contain an itemset
  - e.g.,  $s(\{\text{Milk, Bread, Diaper}\}) = 2/5$

- **Frequent Itemset**

- An itemset whose support is greater than or equal to a  $minsup$  threshold

# Definition: Association Rule

---

- **Association Rule**

- An implication expression of the form  $X \rightarrow Y$ , where X and Y are itemsets
  - e.g.,  $\{\text{Milk, Diaper}\} \rightarrow \{\text{Beer}\}$

- **Rule Evaluation Metrics**

- Support (s)
  - Fraction of transactions that contain both X and Y
- Confidence (c)
  - Measures how often items in Y appear in transactions that contain X

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

## Example:

$$\{\text{Milk, Diaper}\} \Rightarrow \text{Beer}$$

$$s = \frac{\sigma(\text{Milk, Diaper, Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk, Diaper, Beer})}{\sigma(\text{Milk, Diaper})} = \frac{2}{3} = 0.67$$

# Association Rule Mining Task

---

- Given a set of transactions  $T$ , the goal of association rule mining is to find all rules such that
  - support  $\geq \text{minsup}$  threshold
  - confidence  $\geq \text{minconf}$  threshold
- Brute-force approach:
  - List all possible association rules
  - Compute the support and confidence for each rule
  - Prune rules that fail the  $\text{minsup}$  and  $\text{minconf}$  thresholds  
⇒ Computationally prohibitive!

# Mining Association Rules (1/2)

---

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

## Example of Rules:

$\{\text{Milk}, \text{Diaper}\} \rightarrow \{\text{Beer}\}$  ( $s=0.4, c=0.67$ )  
 $\{\text{Milk}, \text{Beer}\} \rightarrow \{\text{Diaper}\}$  ( $s=0.4, c=1.0$ )  
 $\{\text{Diaper}, \text{Beer}\} \rightarrow \{\text{Milk}\}$  ( $s=0.4, c=0.67$ )  
 $\{\text{Beer}\} \rightarrow \{\text{Milk}, \text{Diaper}\}$  ( $s=0.4, c=0.67$ )  
 $\{\text{Diaper}\} \rightarrow \{\text{Milk}, \text{Beer}\}$  ( $s=0.4, c=0.5$ )  
 $\{\text{Milk}\} \rightarrow \{\text{Diaper}, \text{Beer}\}$  ( $s=0.4, c=0.5$ )

- **Observations:**

- All the above rules are binary partitions of the same itemset:  $\{\text{Milk}, \text{Diaper}, \text{Beer}\}$
- Rules originating from the same itemset have identical support but can have different confidence
- Thus, we may decouple the support and confidence requirements

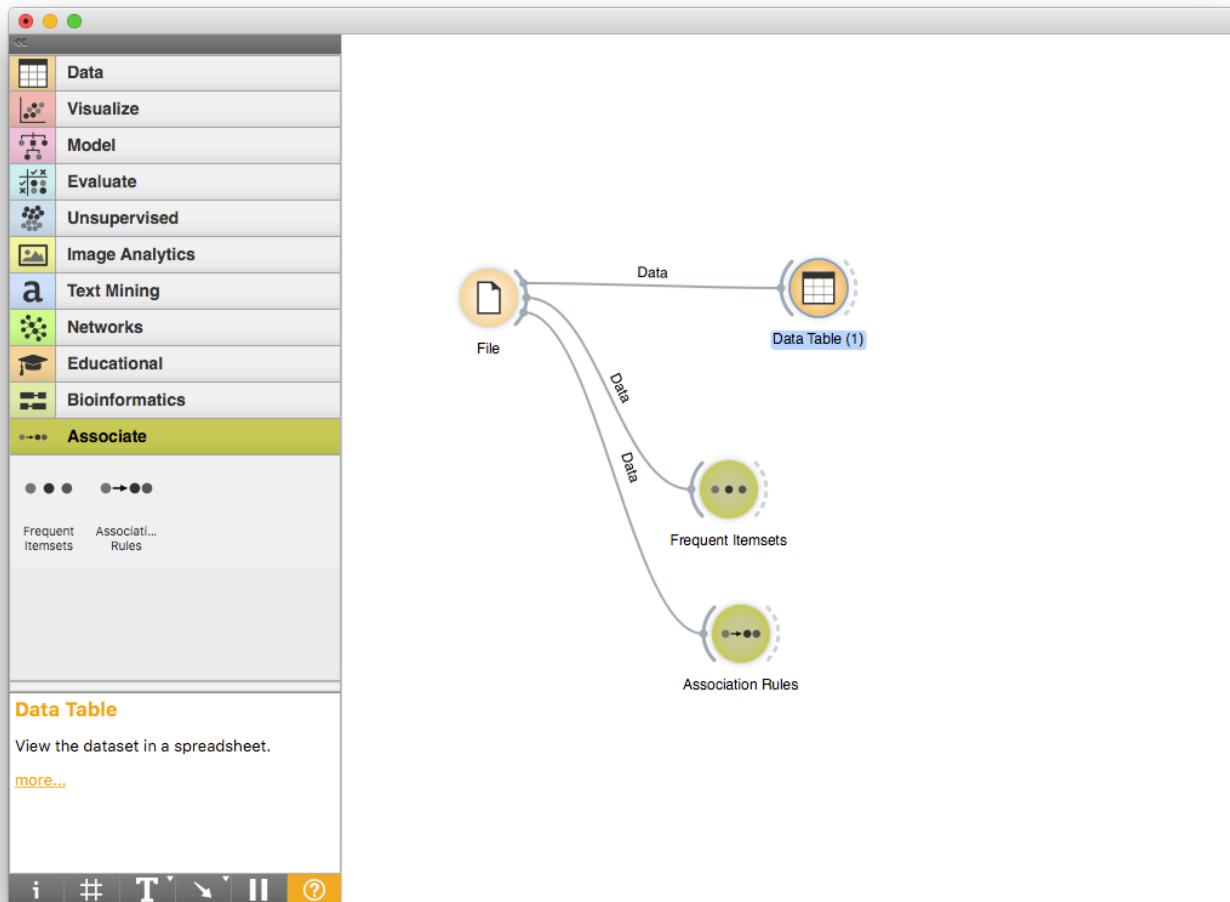
# Mining Association Rules (2/2)

---

- **Frequent Itemset Generation**
  - Generating all items whose support  $\geq m_{insup}$
- **Rule Generation**
  - Generating high confidence rules from each frequent itemset, where each rule is a binary partitioning of a frequent itemset
- **Frequent itemset generation is still computationally expensive**
- **Association rules are a powerful way**
  - to improve your business by organizing your actual or online store, adjusting marketing strategies to target suitable groups
  - providing product recommendations and generally understanding your client base better

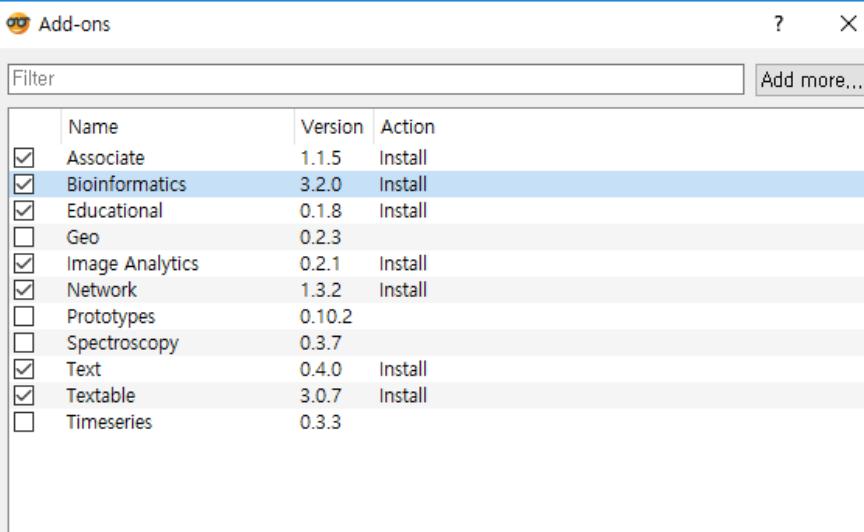
# Build schema

---



# Practice preparation

- Install add-ons



The screenshot shows the 'Add-ons' dialog box from the Orange3 software. The 'Bioinformatics' add-on is selected for installation, indicated by a checked checkbox in the 'Action' column. The table lists various add-ons with their names, versions, and actions (Install or Uninstall). The 'Bioinformatics' add-on is highlighted with a blue selection bar.

Name	Version	Action
Associate	1.1.5	Install
Bioinformatics	3.2.0	Install
Educational	0.1.8	Install
Geo	0.2.3	
Image Analytics	0.2.1	Install
Network	1.3.2	Install
Prototypes	0.10.2	
Spectroscopy	0.3.7	
Text	0.4.0	Install
Textable	3.0.7	Install
Timeseries	0.3.3	

**Orange3–bioinformatics**

[![Build Status](https://travis-ci.org/biolab/orange3-bioinformatics.svg?branch=master)](https://travis-ci.org/biolab/orange3-bioinformatics) [![codecov](https://codecov.io/gh/biolab/orange3-bioinformatics/branch/master/graph/badge.svg)](https://codecov.io/gh/biolab/orange3-bioinformatics)

Orange Bioinformatics extends Orange, a data mining software package, with common functionality for bioinformatics. The provided functionality can be accessed as a Python library or through a visual programming interface (Orange Canvas). The latter is also suitable for non-programmers.

In Orange Canvas the analyst connects basic computational units, called widgets, into data flow analytics schemas. Two units–widgets can be connected if they share a data type. Compared to other popular tools like Taverna, Orange widgets are high-level, integrated potentially complex tasks, but are specific enough to be used independently. Even elaborate analyses rarely consist of more than ten widgets; while tasks such as clustering and enrichment analysis could be executed with up to five widgets. While building the schema each widget is independently controlled with settings, the settings do not conceptually

OK Cancel

# Dataset

---

- Market basket data

The screenshot shows the Weka Data Explorer window. At the top, there is a file selection bar with 'File: market-basket.tab' selected. Below this is an 'Info' panel containing the message: '5 instance(s), 6 feature(s), 0 meta attribute(s) Data has no target variable.' Underneath is a table titled 'Columns (Double click to edit)' showing the following data:

	Name	Type	Role	Values
1	Bread	C	categorical feature	1
2	Milk	C	categorical feature	1
3	Diapers	C	categorical feature	1
4	Beer	C	categorical feature	1
5	Eggs	C	categorical feature	1
6	Cola	C	categorical feature	1

At the bottom of the window are two buttons: 'Browse documentation datasets' and 'Apply'. A question mark icon and a help icon are also present at the bottom left.

# Overview data

---

Info

5 instances  
6 features (40.0% missing values)  
No target variable.  
No meta attributes

Variables

Show variable labels (if present)  
 Visualize numeric values  
 Color by instance classes

Selection

Select full rows

Restore Original Order

Send Automatically

? ⌂

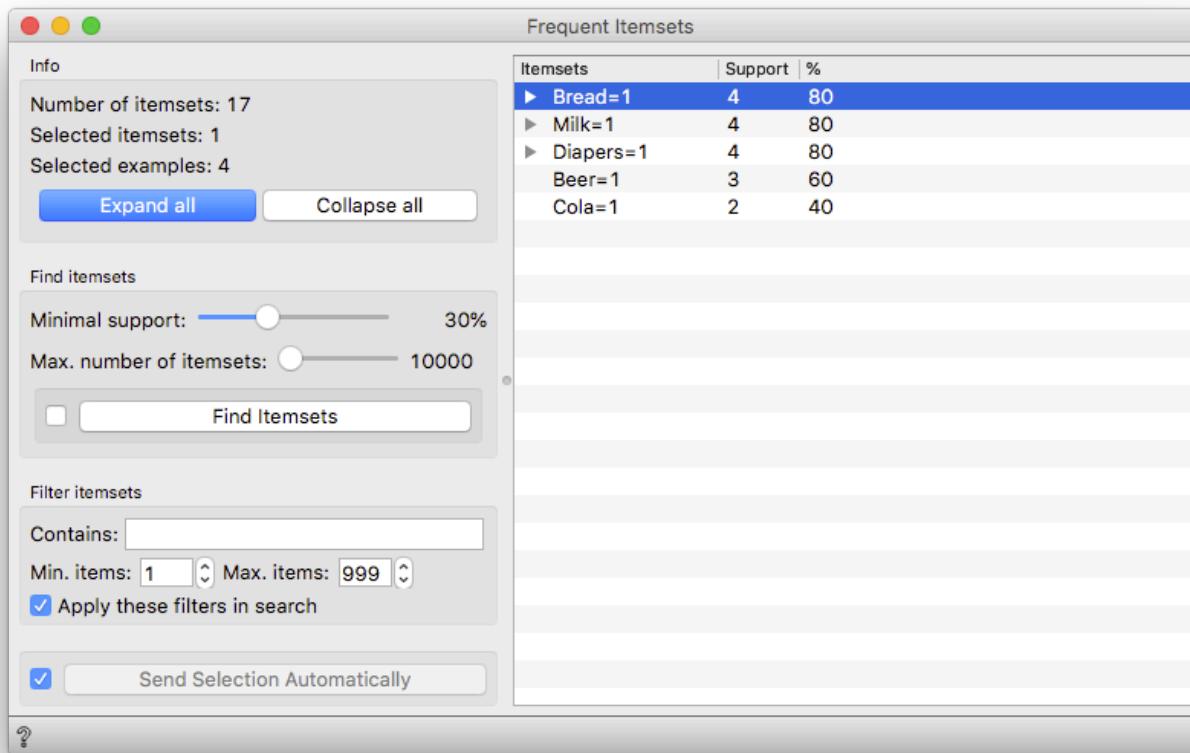
Data Table (1)

	Bread	Milk	Diapers	Beer	Eggs	Cola
1	1	1	?	?	?	?
2	1	?	1	1	1	?
3	?	1	1	1	?	1
4	1	1	1	1	?	?
5	1	1	1	?	?	1

# Frequent item set

---

- What is the our most important products ('bestsellers')?
- What really sells in your store?



# Association Rules

- how about some transaction flows?

Association Rules

Supp	Conf	Covr	Strg	Lift	Levr	Antecedent	Consequent
0.600	1.000	0.600	1.333	1.250	0.120		Beer=1 → Diapers=1
0.400	1.000	0.400	2.000	1.250	0.080	Bread=1, Beer=1	→ Diapers=1
0.400	1.000	0.400	2.000	1.250	0.080	Milk=1, Beer=1	→ Diapers=1
0.400	1.000	0.400	2.000	1.250	0.080	Cola=1	→ Milk=1
0.400	1.000	0.400	2.000	1.250	0.080	Cola=1	→ Diapers=1
0.400	1.000	0.400	2.000	1.250	0.080	Diapers=1, Cola=1	→ Milk=1
0.400	1.000	0.400	2.000	1.250	0.080	Milk=1, Cola=1	→ Diapers=1
0.400	1.000	0.400	1.500	1.667	0.160	Cola=1	→ Milk=1, Diapers=1
0.200	1.000	0.200	4.000	1.250	0.040	Bread=1, Milk=1, Beer=1	→ Diapers=1
0.200	1.000	0.200	4.000	1.250	0.040	Eggs=1	→ Bread=1
0.200	1.000	0.200	4.000	1.250	0.040	Eggs=1	→ Diapers=1
0.200	1.000	0.200	4.000	1.250	0.040	Diapers=1, Eggs=1	→ Bread=1
0.200	1.000	0.200	4.000	1.250	0.040	Bread=1, Eggs=1	→ Diapers=1
0.200	1.000	0.200	3.000	1.667	0.080	Eggs=1	→ Bread=1, Diapers=1
0.200	1.000	0.200	3.000	1.667	0.080	Eggs=1	→ Beer=1
0.200	1.000	0.200	4.000	1.250	0.040	Beer=1, Eggs=1	→ Bread=1
0.200	1.000	0.200	3.000	1.667	0.080	Bread=1, Eggs=1	→ Beer=1
0.200	1.000	0.200	2.000	2.500	0.120	Eggs=1	→ Bread=1, Beer=1
0.200	1.000	0.200	4.000	1.250	0.040	Beer=1, Eggs=1	→ Diapers=1
0.200	1.000	0.200	3.000	1.667	0.080	Diapers=1, Eggs=1	→ Beer=1
0.200	1.000	0.200	3.000	1.667	0.080	Eggs=1	→ Diapers=1, Beer=1
0.200	1.000	0.200	4.000	1.250	0.040	Diapers=1, Beer=1, Eggs=1	→ Bread=1
0.200	1.000	0.200	4.000	1.250	0.040	Bread=1, Beer=1, Eggs=1	→ Diapers=1

Info

Number of rules: 38  
Filtered rules: 38  
Selected rules: 0  
Selected examples: 0

Find association rules

Minimal support: 1%  
Minimal confidence: 90%  
Max. number of rules: 10000  
 Induce classification (itemset → class) rules  
 Find Rules

Filter rules

Antecedent

Contains:   
Min. items: 1  Max. items: 999

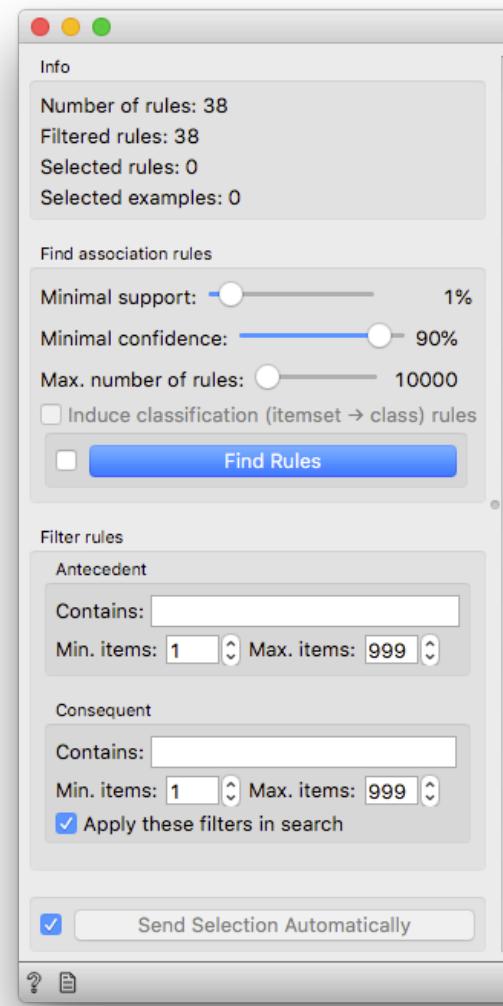
Consequent

Contains:   
Min. items: 1  Max. items: 999   
 Apply these filters in search

Send Selection Automatically

# Parameters

- **Support**
  - how often a rule is applicable to a given data set (rule/data)
- **Confidence**
  - how frequently items in Y appear in transactions with X or in other words how frequently the rule is true (support for a rule/support of antecedent)
- **Coverage**
  - how often antecedent item is found in the data set (support of antecedent/data)
- **Strength**
  - (support of consequent/support of antecedent)
- **Lift**
  - how frequently a rule is true per consequent item (data \* confidence/support of consequent)
- **Leverage**
  - the difference between two item appearing in a transaction and the two items appearing independently ( $\text{support} \times \text{data} - \text{antecedent support} \times \text{consequent support} / \text{data}^2$ )



# Self study data

---

- Foodmart 2000 dataset
  - <https://github.com/neo4j-examples/neo4j-foodmart-dataset/tree/master/data>
- Kaggle Apriori Algorithm
  - <https://www.kaggle.com/datatheque/association-rules-mining-market-basket-analysis>

# **Text Mining**

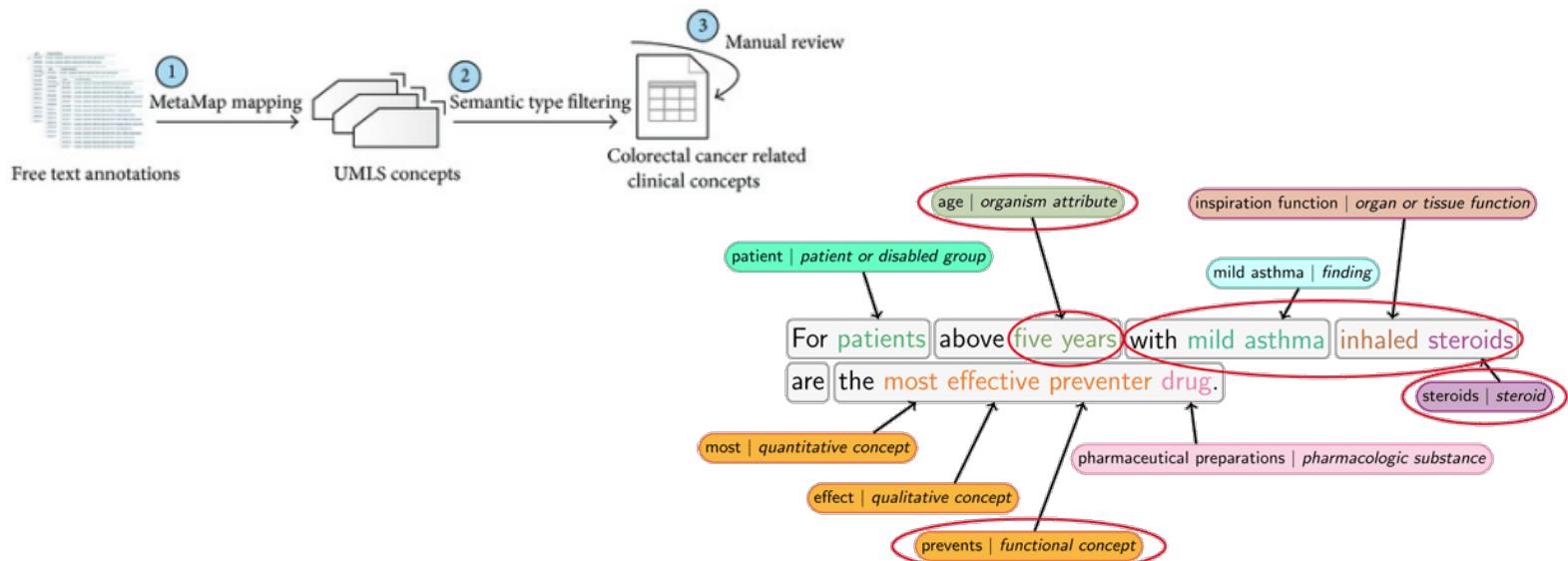
# Text mining

- Variety of forms text data

- web pages, emails, blogs, chats, research papers, books, news papers...
- they are mostly unstructured data

- Examples

- Metamap : knowledge-intensive approach based on symbolic, natural-language processing (NLP) and computational-linguistic techniques. Besides being applied for both IR and data-mining applications, MetaMap is one of the foundations of NLM's Medical Text Indexer (MTI) which is being used for both semiautomatic and fully automatic indexing of biomedical literature at NLM.



# Text mining pipeline example (1/3)

---

- Ctakes-Ytex

No	Analysis Engine Name	Description	Class
1	Chunker	<ul style="list-style-type: none"><li>• It provides a UIMA wrapper for the OpenNLP <code>opennlp.tools.chunker.Chunker</code> class</li><li>• This wrapper can generate chunks of any kind as specified by the chunker model and the chunk creator</li></ul>	<a href="#">Chunker</a>
2	Tokenizer Annotator	<ul style="list-style-type: none"><li>• Discovers tokens in the given text, following Penn Treebank tokenization rules.</li><li>• These tokens consist of words, punctuation, etc.....</li></ul>	<a href="#">TokenizerAnnotator</a> <a href="#">PTB</a>
3	Context Dependent Tokenizer Annotator	<ul style="list-style-type: none"><li>• Find tokens based on context</li></ul>	<a href="#">ContextDependentT</a> <a href="#">okenizerAnnotator</a>
4	Dictionary Lookup Annotator DB	<ul style="list-style-type: none"><li>• UIMA annotator that identified entities based on lookup</li><li>• Specifies the maximum number of items to be returned from an lucene query</li></ul>	<a href="#">UmlsDictionaryLook</a> <a href="#">upAnnotator</a>
5	Status Annotator	<ul style="list-style-type: none"><li>• Analyze a list of tokens looking for a status pattern as specified by the class <a href="#">StatusIndicatorFSM</a></li></ul>	<a href="#">StatusrContextAnal</a> <a href="#">yzer</a>
6	Negation Annotator	<ul style="list-style-type: none"><li>• analyzes a list of tokens looking for a negation pattern as specified by the class <a href="#">NegationFSM</a></li></ul>	<a href="#">NegationContextAn</a> <a href="#">alyzer</a>
7	Extraction PreAnnotator	<ul style="list-style-type: none"><li>• UIMA annotator that prepares the CAS for extraction into DB2</li><li>• Performs some(final) updates to the CAS</li></ul>	<a href="#">ExtractionPreAnno</a> <a href="#">tator</a>
8	Sentence Detector Annotator	<ul style="list-style-type: none"><li>• Wraps the OpenNLP sentence detector in a UIMA annotator</li><li>• Discovers sentence boundaries</li></ul>	<a href="#">SentenceDetector</a>
9	Lookup Window Annotator	<ul style="list-style-type: none"><li>• Select pre-existing annotations in the CAS to create <code>LookupWindow</code> annotation from</li></ul>	<a href="#">org.apache.uima.jav</a> <a href="#">a</a>
10	Adjust Noun Phrase To Include Following NP	<ul style="list-style-type: none"><li>• To extend NP annotations to include prepositional phrases so that for the pattern <code>NP PP NP</code>, named entities that includes a word(s) from each of those NPs is found</li><li>• Adjust NP in <code>NP NP</code> to span both</li></ul>	<a href="#">ChunkAdjuster</a>

# Text mining pipeline example (2/3)

---

No	Analysis Engine Name	Description	Class
11	Adjust Noun Phrase To Include Following PP NP	<ul style="list-style-type: none"><li>Prevents NP annotations from only partially overlapping other NP annotations</li><li>This annotator is written to be able to handle more general cases than NP PP NP</li><li>Adjust NP in NP PP NP to span all there</li></ul>	<a href="#">ChunkAdjuster</a>
12	Simple Segment Annotator	<ul style="list-style-type: none"><li>Creates a single Segment annotation, encompassing the entire document</li><li>For use prior to annotators that require a Segment annotation, when the input document is not in CDA/dose not have another annotator that creates Segment annotations</li></ul>	<a href="#">SimpleSegmentAnnotator</a>
13	POSTagger	<ul style="list-style-type: none"><li>Load a file that contains the MaxEnt model used by the part of speech(POS) tagger</li></ul>	<a href="#">POSTagger</a>
14	Lvg Annotator	<ul style="list-style-type: none"><li>UIMA annotator that uses the UMLS LVG(Lexical Variant Generation) package to find the canonical form of WordTokens</li><li>The package is also used to find one or more lemmas for a given WordToken along with its associated part of speech</li></ul>	<a href="#">LvgAnnotator (LVG)</a>
15	Generic Cleartk Analysis Engine		<a href="#">GenericCleartkAnalysisEngine</a>
16	History Cleartk Analysis Engine		<a href="#">HistoryCleartkAnalysisEngine</a>
17	Polarity Cleartk Analysis Engine		<a href="#">PolarityCleartkAnalysisEngine</a>
18	Subject Cleartk Analysis Engine		<a href="#">SubjectCleartkAnalysisEngine</a>
19	Uncertainty Cleartk Analysis Engine		<a href="#">UncertaintyCleartkAnalysisEngine</a>
20	DependencyParser	<ul style="list-style-type: none"><li>Provides a UIMA wrapper for the CLEAR dependency parser</li></ul>	<a href="#">ClearNLPDependencyparserAE (Ref link)</a>

# Text mining pipeline example (3/3)

---

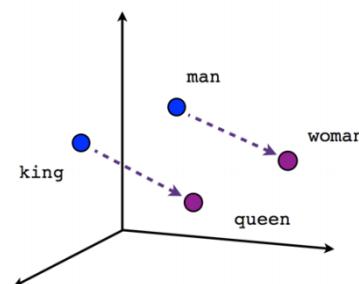
No	Analysis Engine Name	Description	Class
21	Semantic Role Labeler	<ul style="list-style-type: none"><li>Provides a UIMA wrapper for the <a href="#">ClearNLP Semantic Role Labeler</a></li><li>Before using this AnalysisEngine, user should run a Tokenizer, POS-tagger, Lemmatizer, and the CLEAR parser dependency parser</li></ul>	<a href="#">ClearNLPSemanticRoleLabelerAE</a>
22	Constituency Parser	<ul style="list-style-type: none"><li>Extends org.apache.uima.analysis_component.JCasAnnotator_ImplBase</li></ul>	<a href="#">ConstituencyParserAnnotator</a>

# **Word Embedding**

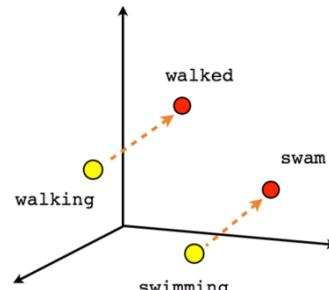
# Word2Vec

---

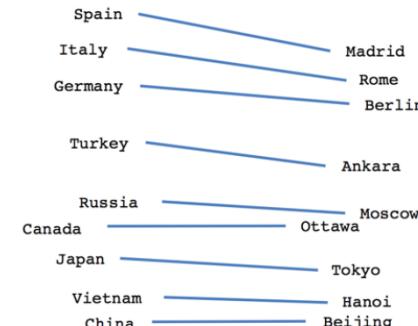
- Efficient estimation of word representations in vector space (2013), T. Mikolov et al
- 문장에 나오는 단어들의 위치로 학습시키자!
- Word Embedding method based on its location of a sentence
- E.g.) Significance of positive and inhibitory regulators in the TGF- $\beta$  signaling pathway in colorectal cancers.
  - Window Size = 1 : ([Significance, of] positive), ([of, positive], and), ([positive, and] inhibitory)



Male-Female



Verb tense



Country-Capital

Image from <https://www.tensorflow.org/tutorials/word2vec>

# Introduction

---

- **Bag of Word (Harris, 1954)**

- fixed-length vector representation for texts
- (Limitations) The word order is lost, it suffers from data sparsity and high dimensionality
- Example #1

## 2 Bag-of-Word

Index : 0 1 2 3 4 5 6 7 8 9

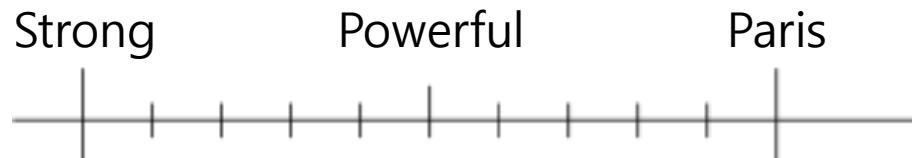
```
John likes to watch movies. Mary likes movie too.  
-> [1,2,1,1,2,0,0,0,1,1]  
John also likes to watch football game.  
-> [1,1,1,1,0,1,1,1,0,0]
```

## 1

### Dictionary

```
{ "John": 0,  
  "likes": 1,  
  "to": 2,  
  "watch": 3,  
  "movies": 4,  
  "also": 5,  
  "football": 6,  
  "games": 7,  
  "Mary": 8,  
  "too": 9 }
```

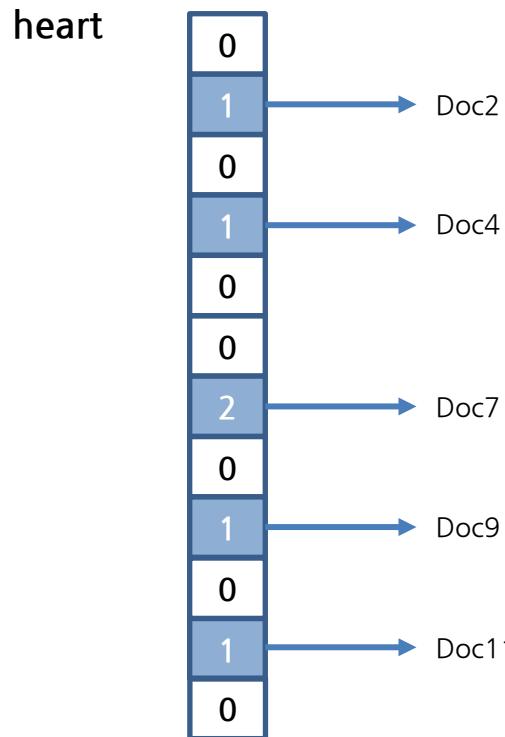
- Example #2



# Introduction

---

- **Vector Space Models (G. Salton, 1975)**
  - Represent an item (e.g. word) as a vector of numbers

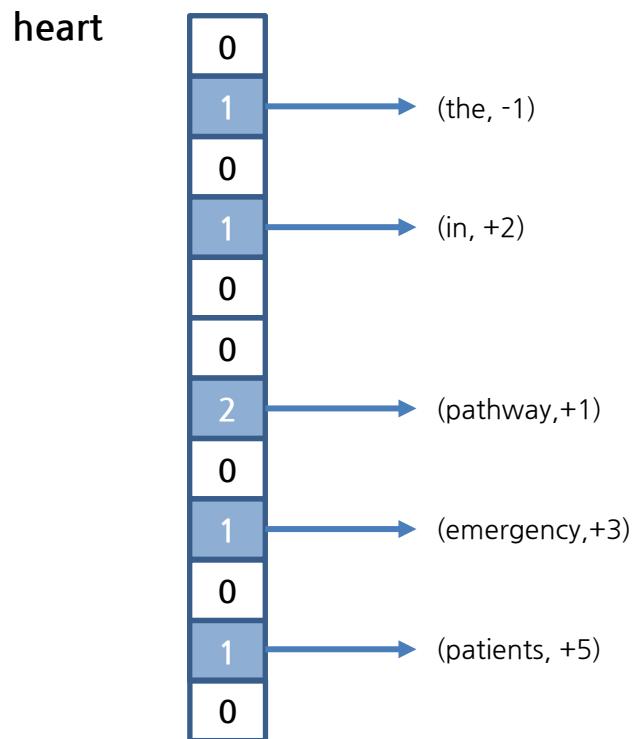


The vector can correspond to documents in which the word occurs

# Introduction

---

- **Vector Space Models (G. Salton, 1975)**
  - Represent an item (e.g. word) as a vector of numbers



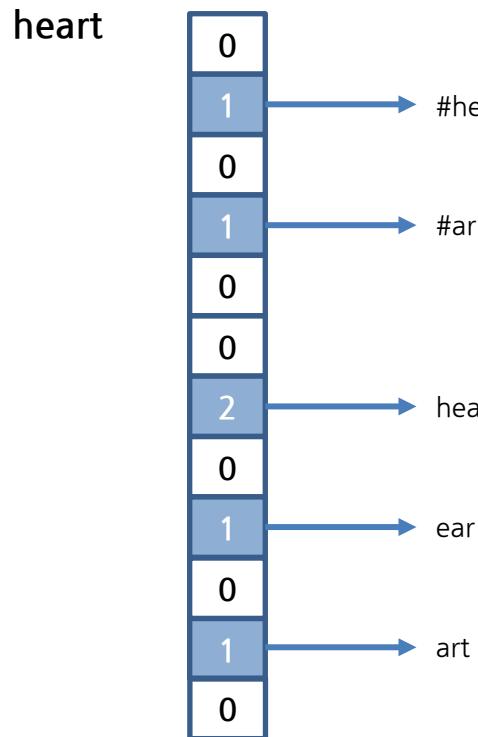
The vector can correspond to neighboring word context.

e.g., “Efficacy Evaluation of the HEART Pathway in  
Emergency Department Patients With Acute Chest  
Pain”  
-4                    -3                    -2                    -1                    0                    +1                    +2  
+3                    +4                    +5                    +6                    +7                    +8  
+9

# Introduction

---

- **Vector Space Models (G. Salton, 1975)**
  - Represent an item (e.g. word) as a vector of numbers



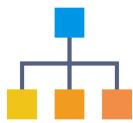
The vector can correspond to character trigrams in the word

# Similarity and Relatedness

---

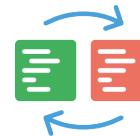
- Definition of Semantic Similarity and Relatedness<sup>[1]</sup>

## Semantic Similarity



Measures quantify how “alike” (or similar) two concepts are by determining their closeness in a hierarchy

## Semantics Relatedness



Information content based measures which are based on the probability of the concept occurring in the taxonomy

## • example

- (Similarity) Tokyo similar to Seoul?
  - Because they are both capital.
- (Relatedness) Apple similar to New York?
  - Because “Big Apple” is a nickname for New York.

[1] McInnes, B. T., et al. (2009). UMLS-Interface and UMLS-Similarity: open source software for measuring paths and semantic similarity. AMIA Annual Symposium Proceedings, American Medical Informatics Association.

# Embeddings

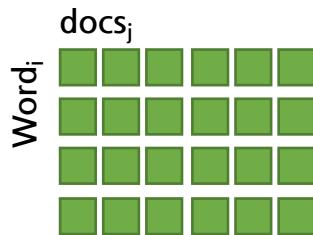
---

- The vectors we have been discussing so far very high-dimensional (thousands, or even millions) and sparse
- But there are techniques to learn lower-dimensional dense vectors for words using the same intuitions
- These dense vectors are called embeddings

# Dense Embeddings

- Matrix Factorization

- Factorize word-context matrix

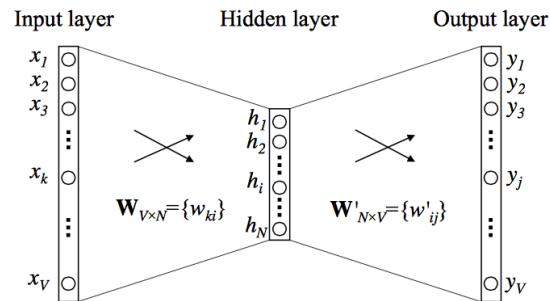


- Examples

- LDA (Word-Document)<sup>[1]</sup>
- GloVe (Word-NeighboringWord)<sup>[2]</sup>

- Neural Networks

- A neural network with a bottleneck, word and context as input and output respectively



- Example

- Word2Vec (Word-NeighboringWord)<sup>[3]</sup>

[1] Deerwester, S., et al. (1990). "Indexing by latent semantic analysis." Journal of the American society for information science 41(6): 391.

[2] Pennington, J., et al. (2014). Glove: Global Vectors for Word Representation. EMNLP.

[3] Mikolov, T., et al. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems.

# Practice for Word Embedding

---

- Just give me a code!

- Tokenization

```
// Split on white spaces in the line to get words
TokenizerFactory t = new DefaultTokenizerFactory();
t.setTokenPreProcessor(new CommonPreprocessor());
```

- Model Learning

```
int batchSize = 1000;
int iterations = 3;
int layerSize = 150;

log.info("Build model....");
Word2Vec vec = new Word2Vec.Builder()
    .batchSize(batchSize) //# words per minibatch.
    .minWordFrequency(5) //
    .useAdaGrad(false) //
    .layerSize(layerSize) // word feature vector size
    .iterations(iterations) // # iterations to train
    .learningRate(0.025) //
    .minLearningRate(1e-3) // learning rate decays wrt # words. floor learning
    .negativeSample(10) // sample size 10 words
    .iterate(iter) //
    .tokenizerFactory(tokenizer)
    .build();

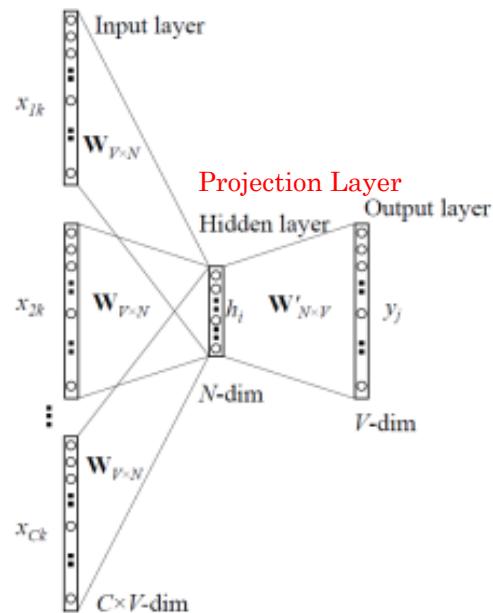
vec.fit();
```

Source from <https://deeplearning4j.org/kr/word2vec>

# Novelty of WordEmbedding

- $V \rightarrow \ln(V)$  : Complexity Reduction

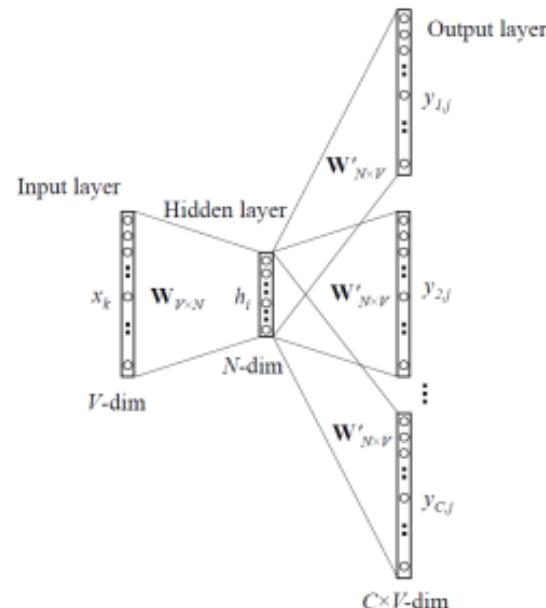
*CBOW Architecture*



*Calculation Amount*

- $C$ 개의 단어를 Projection 하는 데  $C \times N$
- Projection Layer에서 Output Layer로 가는 데  $N \times V$
- 전체 계산량 :  $C \times N + N \times V$

*Skip-gram Architecture*



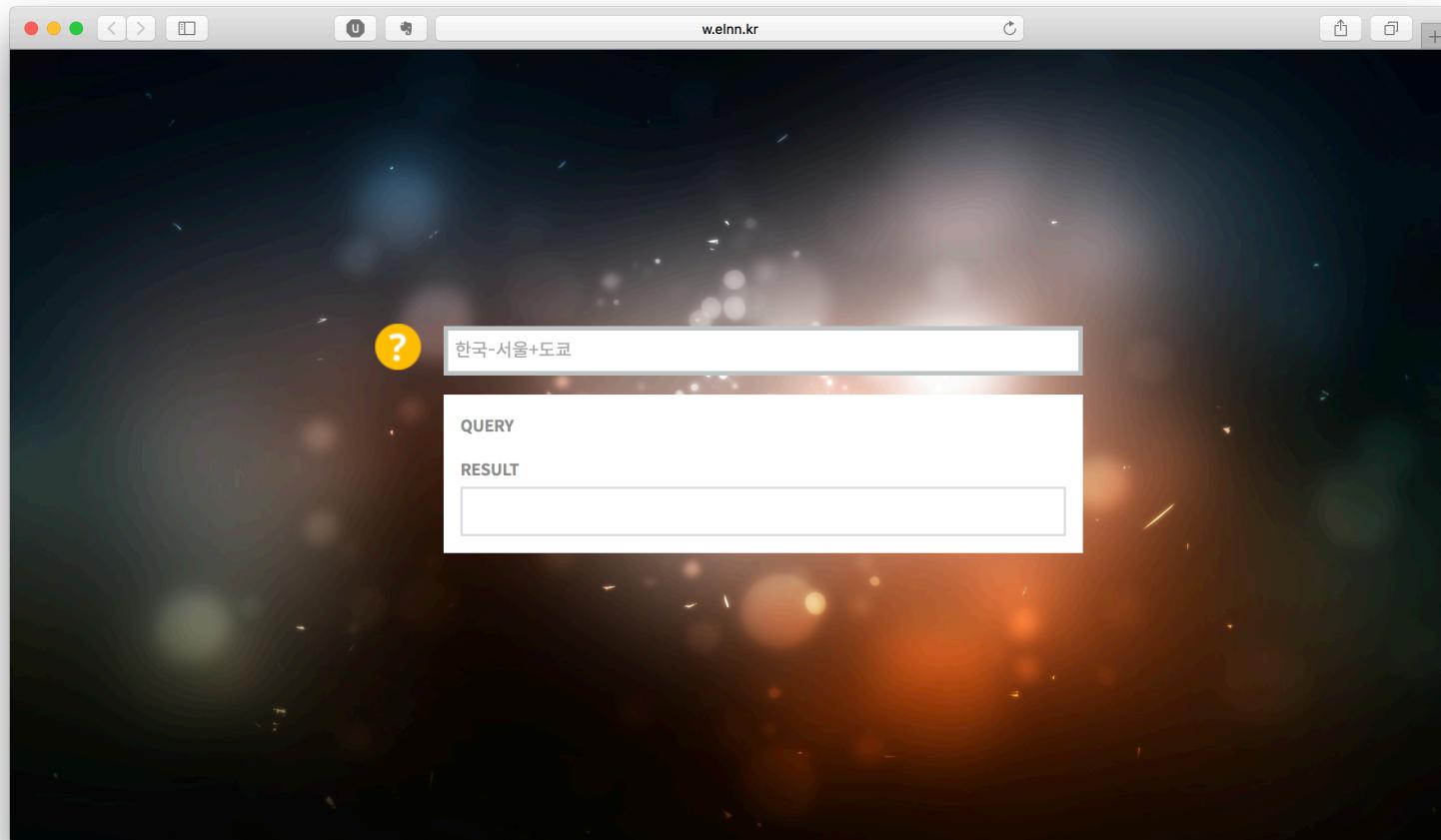
*Calculation Amount*

- 현재 단어를 Projection 하는 데  $N$
- Output을 계산하는 데에  $N \times V$
- 총  $C$ 개의 단어에 대해 진행하므로  $C$ 배
- 전체 계산량 :  $C(N + N \times V)$

# Demo

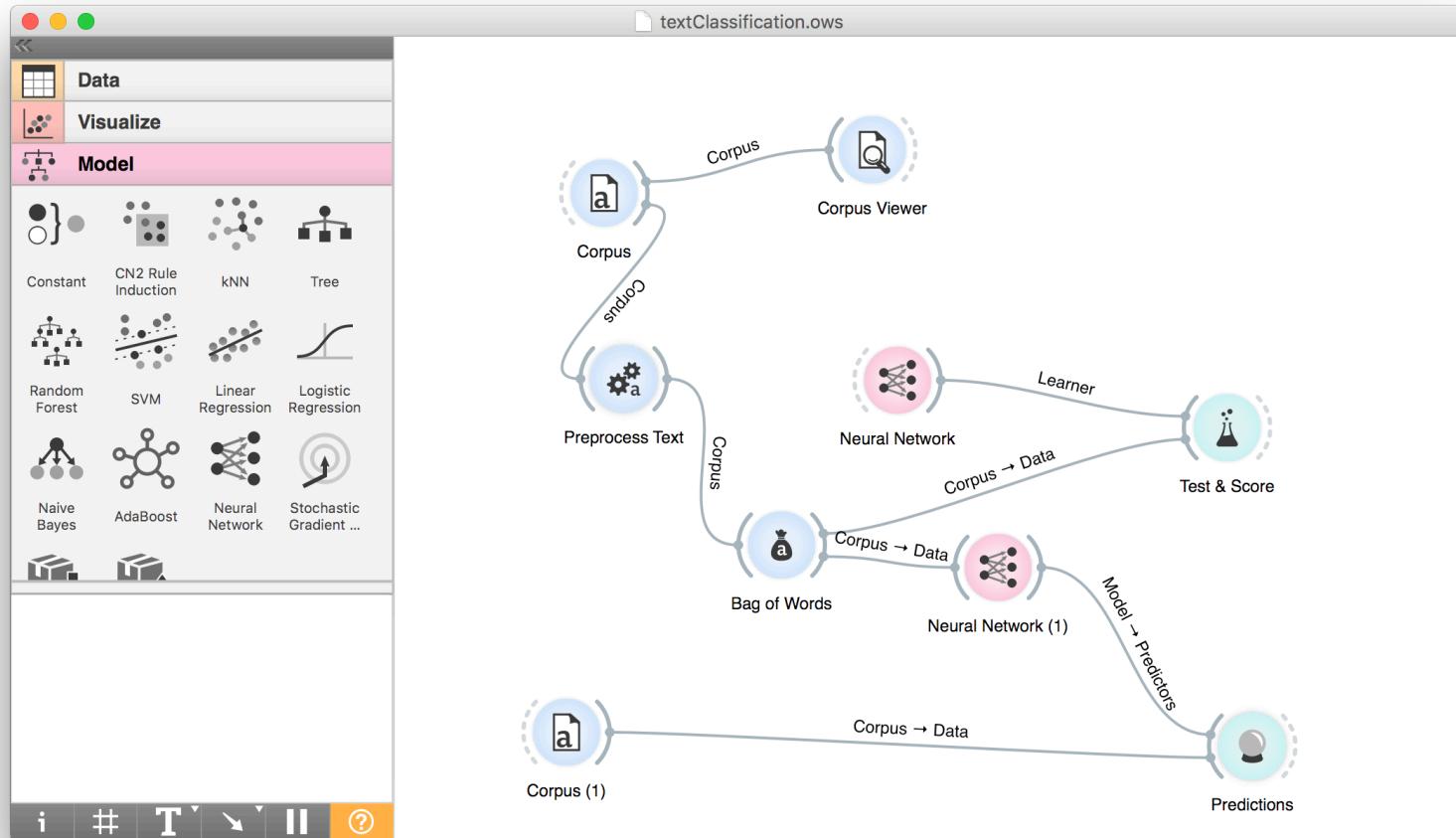
---

- <http://w.elnn.kr/search/>



# Build Schema

- Text Classification Playground



# **Twitter Example**

# Twitter API

- <https://apps.twitter.com/>

The screenshot shows a web browser window with multiple tabs open. The active tab is titled 'Twitter — Orange3 Text' and the URL is 'https://apps.twitter.com/'. The browser's address bar also displays the same URL. The main content area is titled 'Twitter Apps'. A yellow callout box contains text about developer accounts and app management. Below it, a message states 'You don't currently have any Twitter Apps.' At the bottom of the page, there are links for 'About', 'Terms', 'Privacy', and 'Cookies', along with a 'Tweet' button and a copyright notice for '© 2018 Twitter, Inc.'

As of July 2018, you must [apply for a Twitter developer account](#) and be approved before you may create new apps. Once approved, you will be able to create new apps from [developer.twitter.com](#).

For the near future, you can continue to manage existing apps here on [apps.twitter.com](#). However, we will soon retire this site and consolidate all developer tools, API access, and app management within the developer portal at [developer.twitter.com](#). You will be able to access and manage existing apps through that portal when we retire this site.

[Apply for a developer account](#)

You don't currently have any Twitter Apps.

About Terms Privacy Cookies © 2018 Twitter, Inc.

# Twitter API

twitter text mining orange | Twitter — Orange3 Text | Apply — Twitter Developer

Developer Use cases Products Docs More Apply

## Apply for developer access

STATUS: IN PROGRESS

User profile  
 Account details  
 Use case details  
 Terms of service  
 Email verification

### Interested in a developer account?

Some of our premium APIs are currently in Beta. By applying, you agree to receive emails from our team requesting feedback on your experience.

#### Select a user profile to associate

By default, this @username will be the admin of this developer account. If you are creating a developer account on behalf of your organization, you may wish to use your organization's @username as it is most likely to own the Apps you will use to access the API endpoints or warrant special permissions. You'll be able to invite teammates and re-assign roles later within your developer account settings.

#### Associate your current Twitter @username

PARK JUNSEOK  
@BapboLove

The phone number associated with this Twitter @username is not verified. You must add a valid phone number and verify it prior to applying for developer access.

Add a valid phone number

Sign in as a different Twitter @username  
Create new Twitter @username

Developer policy and terms Follow @twitterdev

Subscribe to developer news

# Twitter API

The screenshot shows a web browser window with the URL <https://developer.twitter.com/en/apply/account>. The page is titled "Apply for developer access" and is part of the "Status: IN PROGRESS" section. The left sidebar lists steps: "User profile" (selected), "Account details", "Use case details", "Terms of service", and "Email verification". A box on the left explains the purpose of the questions. The main form asks "Who are you requesting access for?" with two options: "I am requesting access for my organization" (selected) and "I am requesting access for my own personal use". Below this, there's a section to "Tell us about yourself" with fields for "Account name" (e.g., username, project name, etc.) and "Primary country of operation" (with a dropdown menu). A "Continue" button is at the bottom. The footer includes links for "Developer policy and terms", "Follow @twitterdev", and "Subscribe to developer news".

STATUS: IN PROGRESS

User profile

Account details

Use case details

Terms of service

Email verification

**Why the questions?**

We empower freedom of expression by providing a platform that protects the voices of our users — both on Twitter, and via our developer products. To help verify that all uses of Twitter data comply with our policies, we require additional information from developers signing up to use this service. Providing thorough answers will help us understand your use cases and will help expedite the evaluation of your application. Learn more about our restricted use cases.

**Add your account details**

**Who are you requesting access for?**

I am requesting access for my organization

I plan to use Twitter's developer platform for projects owned by / in affiliation with a business, organization or institution. Ex: SaaS product, proof of concept, academic research, etc. *To enable collaboration, this selection includes additional tools to support team development.*

I am requesting access for my own personal use

I plan to use Twitter's developer platform for projects unaffiliated with an existing business, organization or institution. Ex: Side project, hobby, etc. *Personal use accounts do not include team development tools.*

**Tell us about yourself**

**Account name**  
e.g., username, project name, etc.

Name your account...

**Primary country of operation**

Select one... ▾

**Continue**

[Developer policy and terms](#) [Follow @twitterdev](#) [Subscribe to developer news](#)

# Twitter API

The screenshot shows a web browser window with the Twitter Developer API application page. The URL is <https://developer.twitter.com/en/apply/usecase>. The page has a purple header with navigation links: Developer, Use cases, Products, Docs, More, and a user profile icon. The main content area is titled "Apply for developer access" and "STATUS: IN PROGRESS". On the left, there's a sidebar with a list of steps: User profile (radio button selected), Account details, Use case details (radio button selected), Terms of service, and Email verification. Below this is a section titled "Why the questions?" which explains the purpose of the questions and links to restricted use cases. The main form area is titled "Tell us about your project" and asks "What use case(s) are you interested in?". It lists several options with checkboxes, some of which are checked: Academic research (checked), Advertising, Audience analysis, Chatbots and automation, Consumer / end-user experience, Publish and curate Tweets, Student project / Learning to code (checked), Topic analysis (checked), Trend and event detection, Other, and Engagement and customer service. Below this is a section titled "Describe in your own words what you are building" with instructions in English. A list of questions follows: What is the purpose of your product or service?, What will you deliver to your users/customers?, How do you intend to analyze Tweets, Twitter users, or their content?, and How is Twitter data displayed to users of your end product or service (e.g. will Tweets and content be displayed at row level or in aggregate?). At the bottom, there's a text area for comments with a character limit of 300.

STATUS: IN PROGRESS

User profile (radio button selected)

Account details

Use case details (radio button selected)

Terms of service

Email verification

Why the questions?

We empower freedom of expression by providing a platform that protects the voices of our users — both on Twitter and via our developer products. To help verify that all uses of Twitter data comply with our policies, we require additional information from developers signing up to use this service. Providing thorough answers will help us understand your use cases and will help expedite the evaluation of your application. Learn more about our [restricted use cases](#).

Tell us about your project

What use case(s) are you interested in?

Select all that apply

Academic research  Publish and curate Tweets

Advertising  Student project / Learning to code

Audience analysis  Topic analysis

Chatbots and automation  Trend and event detection

Consumer / end-user experience  Other

Engagement and customer service

Describe in your own words what you are building

In English, please describe your product - the more detailed the response, the easier it is to review and approve. Be sure to answer the following:

- What is the purpose of your product or service?
- What will you deliver to your users/customers?
- How do you intend to analyze Tweets, Twitter users, or their content?
- How is Twitter data displayed to users of your end product or service (e.g. will Tweets and content be displayed at row level or in aggregate)?

To expedite your access approval, please be detailed...

Minimum characters: 300

# Twitter API

The screenshot shows a web browser window with the URL <https://developer.twitter.com/en/apply/terms>. The page is titled "Apply for developer access". On the left, there is a sidebar with a status box showing "STATUS: IN PROGRESS" and a list of steps: "User profile" (radio button), "Account details" (radio button), "Use case details" (radio button), "Terms of service" (radio button, currently selected), and "Email verification" (radio button). The main content area is titled "Read and agree to the Terms of Service" with a sub-section "Developer Agreement". The "Developer Agreement" section contains the full text of the Twitter Developer Agreement, which is a legal document detailing the terms and conditions for using the Twitter API. At the bottom of this section, there is a checkbox labeled "By clicking on the box, You indicate that you have read and agree to this Developer Agreement and the Twitter Developer Policy, additionally as it relates to your display of any of the Content, the [Display Requirements](#); as it relates to your use and display of the Twitter Marks, the [Twitter Brand Assets and Guidelines](#); and as it relates to".

# Twitter API

A screenshot of a web browser showing the Twitter Developers verification page. The URL is https://developer.twitter.com/en/verify. The page has a purple header with the Twitter logo and navigation links: Developer, Use cases, Products, Docs, More, and a user profile for JunSeokPark. The main content area contains the text "One more step..." and instructions to check the inbox for a verification email. It also includes a link to resend the email if it's not received. At the bottom, there are links for Developer policy and terms, Follow @twitterdev, and a "Subscribe to developer news" button. The footer is purple and contains links for About, Business, Developers, Help Center, and Marketing categories.

Developer policy and terms    Follow @twitterdev    [Subscribe to developer news](#)

About    Business    Developers    Help Center    Marketing

Let's go Twitter    About Twitter Ads    Documentation    Using Twitter    Insights  
Company    Targeting    Forums    Managing your account    Success Stories  
Values    Analytics    Communities    Safety and security    Solutions  
Safety    Ads support    Developer blog    Rules and policies    Collections  
Blog    Business blog    Advertise    Contact us    Marketing Blog  
Brand Resources    Advertise     
Careers     
Investors

© 2018 Twitter, Inc.    Cookies    Privacy    Terms and Conditions

# Twitter API



JUN SEOK ▾ | ⓘ

## Verify your Twitter Developer Account

1분 전 오후 7:08

보낸 사람 [Twitter Developer Accounts >](#)

더 보기



Email verification

Hi PARK JUNSEOK!

Thanks for applying for a Twitter Developer account.

Please confirm your email address to complete your application.

[Confirm your email](#)

Thanks!

The Twitter Dev team

[developer.twitter.com](#) | @twitterdev

Twitter, Inc. 1355 Market Street, San Francisco, CA 94103

# Twitter API

The screenshot shows a web browser window with multiple tabs open. The active tab is 'Twitter Developers' at <https://developer.twitter.com/en/review>. The page displays a message: 'Application under review.' with a circular loading icon. Below it, text reads: 'Thanks! We've received your application and are reviewing it. We'll be in touch soon. We review applications to ensure compliance with our Terms of Service and Developer policies. [Learn more.](#)' Further down, it says: 'You'll receive an email when the review is complete. While you wait, check out our [documentation](#), explore our [tutorials](#), or check out our [community forums](#).'

At the bottom, there are links for 'Developer policy and terms', 'Follow @twitterdev', and 'Subscribe to developer news'. The footer contains sections for 'About', 'Business', 'Developers', 'Help Center', and 'Marketing' with various links.

# Build Schema

The screenshot shows the Text Mining application interface. On the left, a sidebar lists various tools under three categories: Text Mining, Networks, and Educational. Under Text Mining, the Twitter tool is selected, showing its configuration options. A central workspace displays a workflow diagram with nodes: Twitter, Corpus, Corpus Viewer, Preprocess Text, and Word Cloud, connected by dashed arrows labeled 'Corpus'.

**Text Mining**

- Corpus
- Import Docume...
- The Guardian
- NY Times
- Pubmed
- Twitter
- Wikipedia
- Preproc... Text
- Bag of Words
- Similarity Hashing
- Sentiment Analysis
- Tweet Profiler
- Topic Modelling
- Corpus Viewer
- Word Cloud
- Concord...
- GeoMap
- Word Enrichm...
- Duplicate Detection

**Networks**

**Educational**

**Twitter**

Load tweets from the Twitter API.

[more...](#)

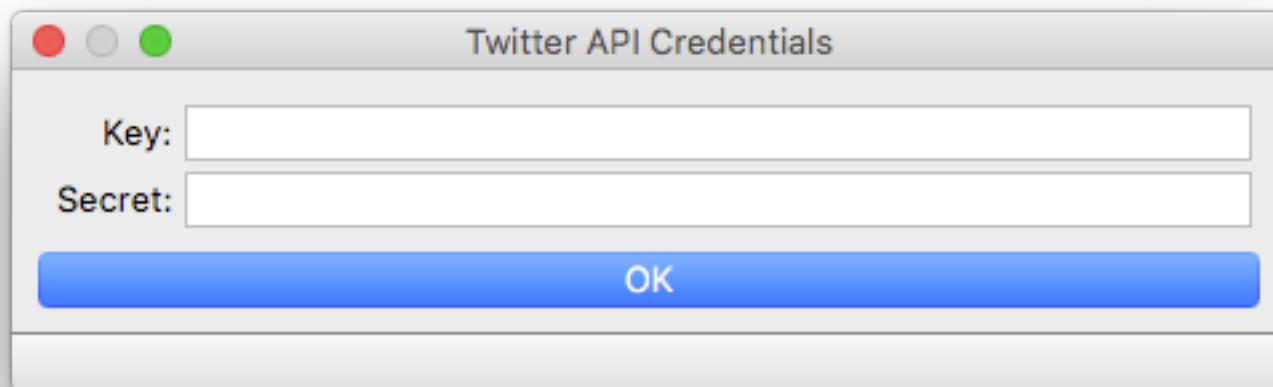
Workflow Diagram:

```
graph LR; Twitter((Twitter)) -- Corpus --> CorpusView((Corpus Viewer)); CorpusView -- Corpus --> PreprocessText((Preprocess Text)); PreprocessText -- Corpus --> WordCloud((Word Cloud));
```

# Twitter Connection

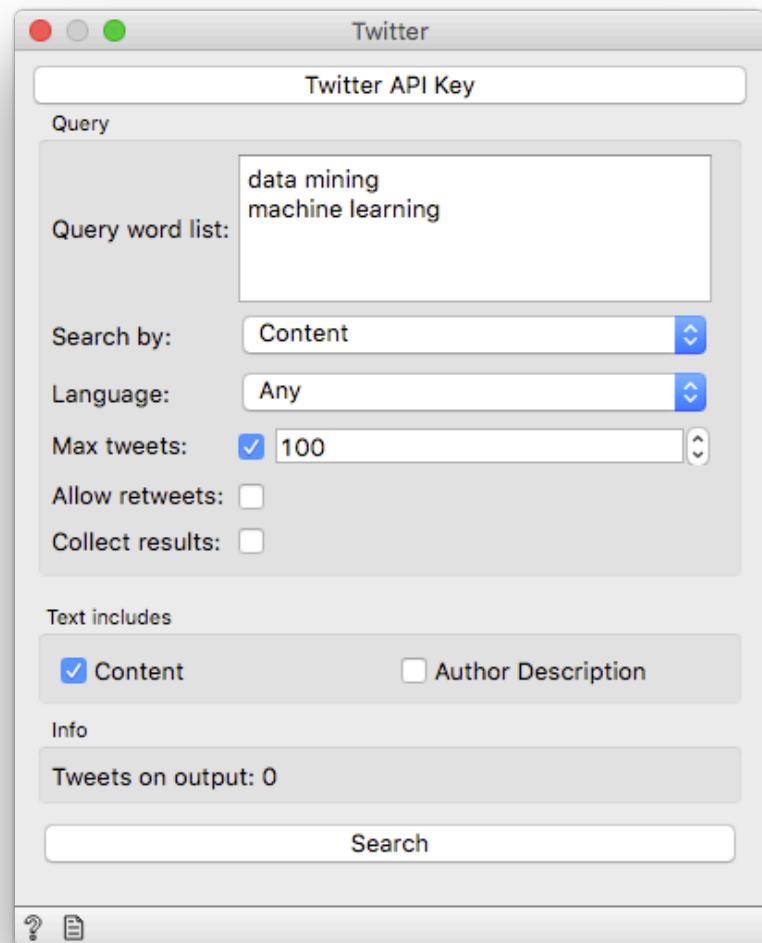
---

- Connect to twitter
  - insert Twitter key and secret



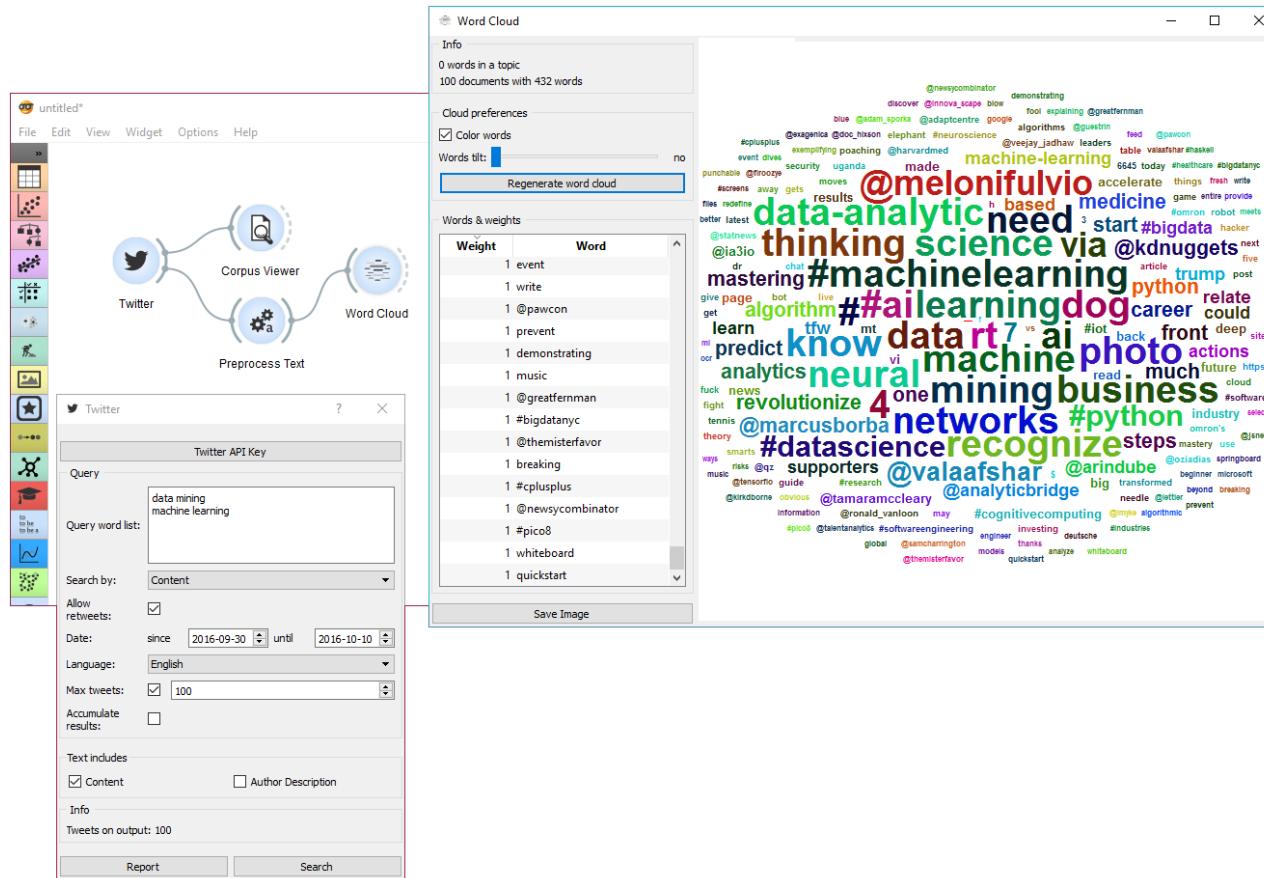
# Twitter Parameters

- **Query word list**
  - list desired queries, one per line. Queries are automatically joined by OR.
- **Search by**
  - specify whether you want to search by content, author or both. If searching by author, you must enter proper Twitter handle (without @) in the query list.
- **Allow retweets**
  - if ‘Allow retweets’ is checked, retweeted tweets will also appear on the output. This might duplicate some results.
- **Date**
  - set the query time frame. Twitter only allows retrieving tweets from up to two weeks back.
- **Language**
  - set the language of retrieved tweets. Any will retrieve tweets in any language.
- **Max tweets**
  - set the top limit of retrieved tweets. If box is not ticked, no upper bound will be set - widget will retrieve all available tweets.
- **Accumulate results**
  - if ‘Accumulate results’ is ticked, widget will append new queries to the previous ones. Enter new queries, run *Search* and new results will be appended to the previous ones.



# Simple query

- tweets containing either ‘data mining’ or ‘machine learning’ in the content and allow retweets.
  - limit our search to only a 100 tweets in English.



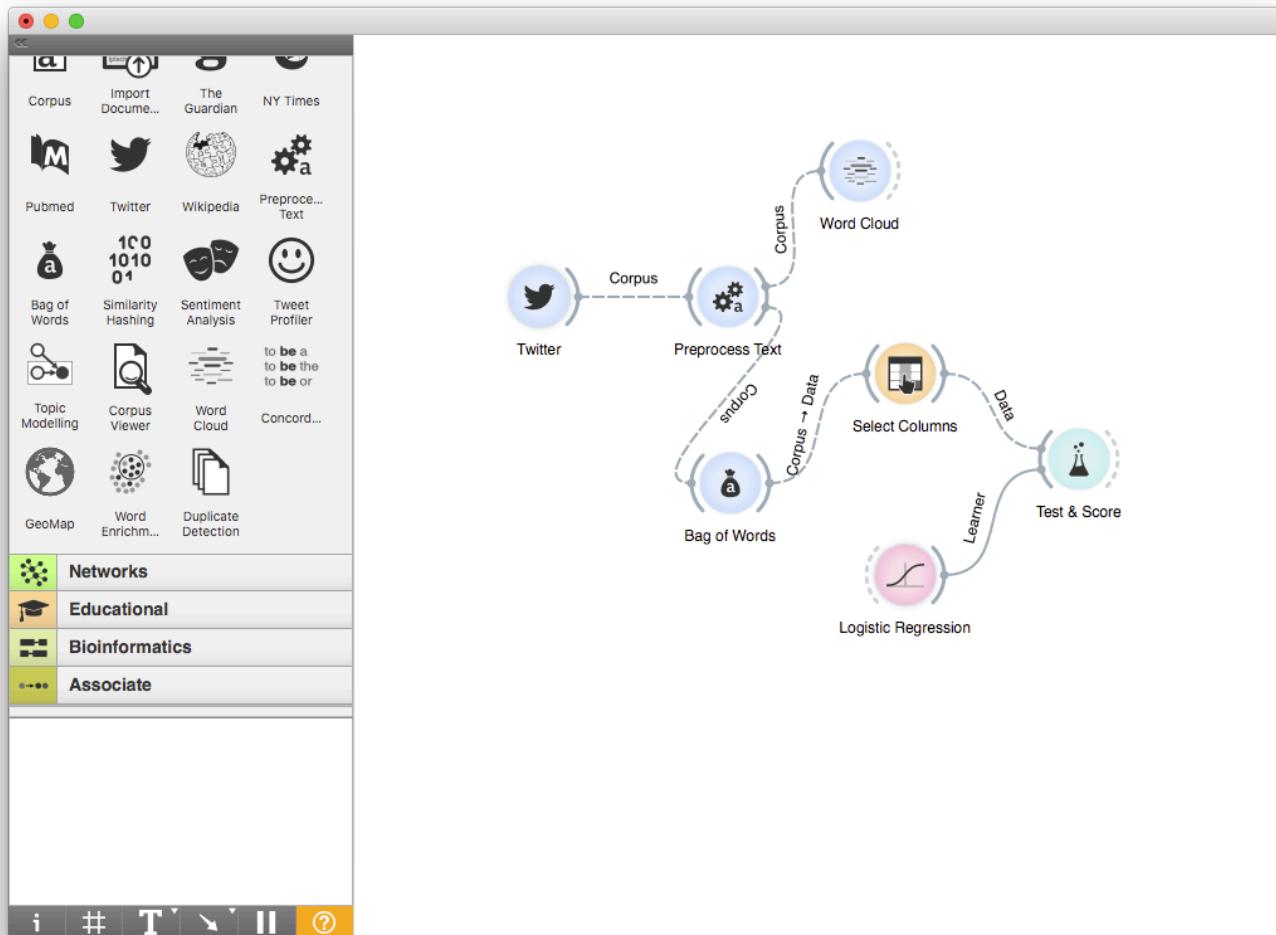
# Self study

---

- **checking the output in [Corpus Viewer](#) to get the initial idea**
  - preprocessing the tweets with lowercase, url removal, tweet tokenizer and removal of stop word and punctuation.

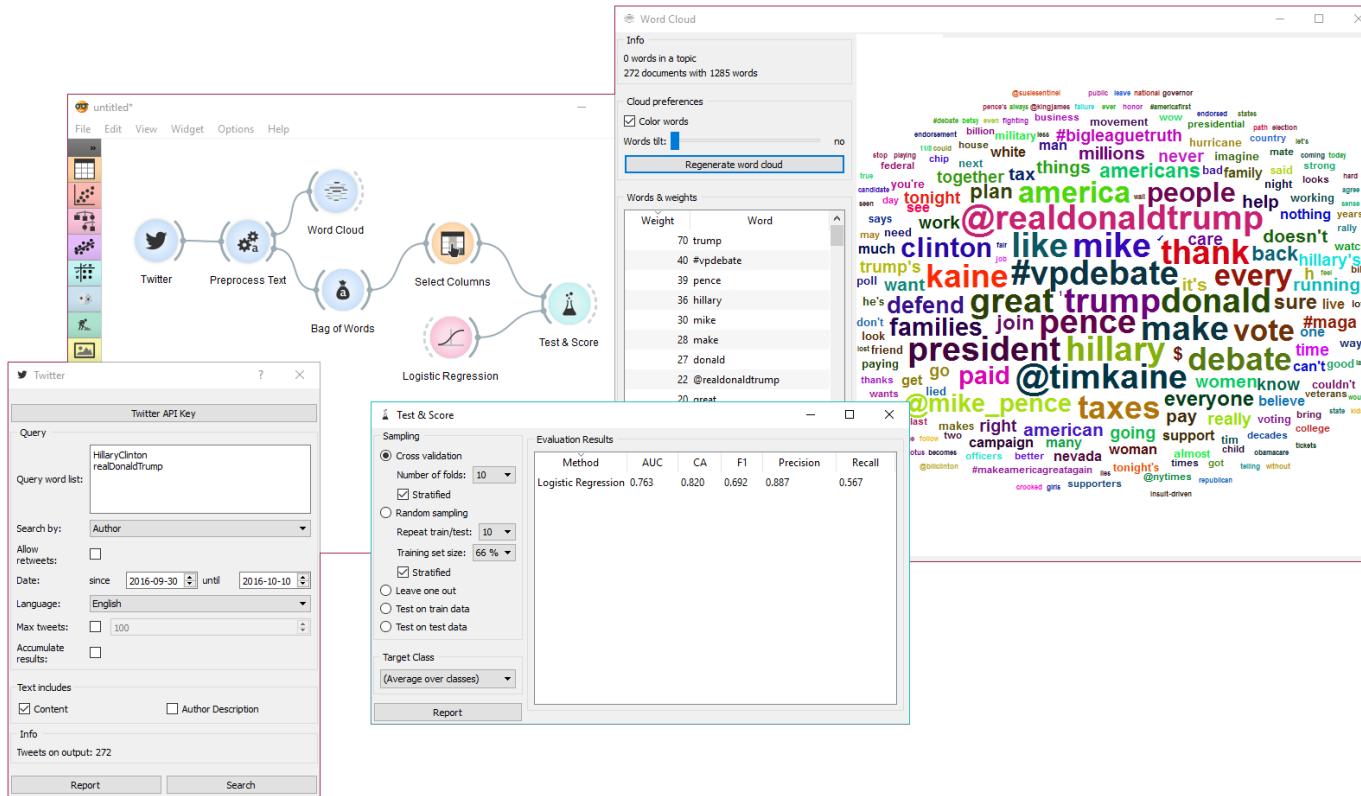
# Complex Example (1/2)

- Build Schema



# Complex Example

- querying tweets from Hillary Clinton and Donald Trump from the presidential campaign 2016

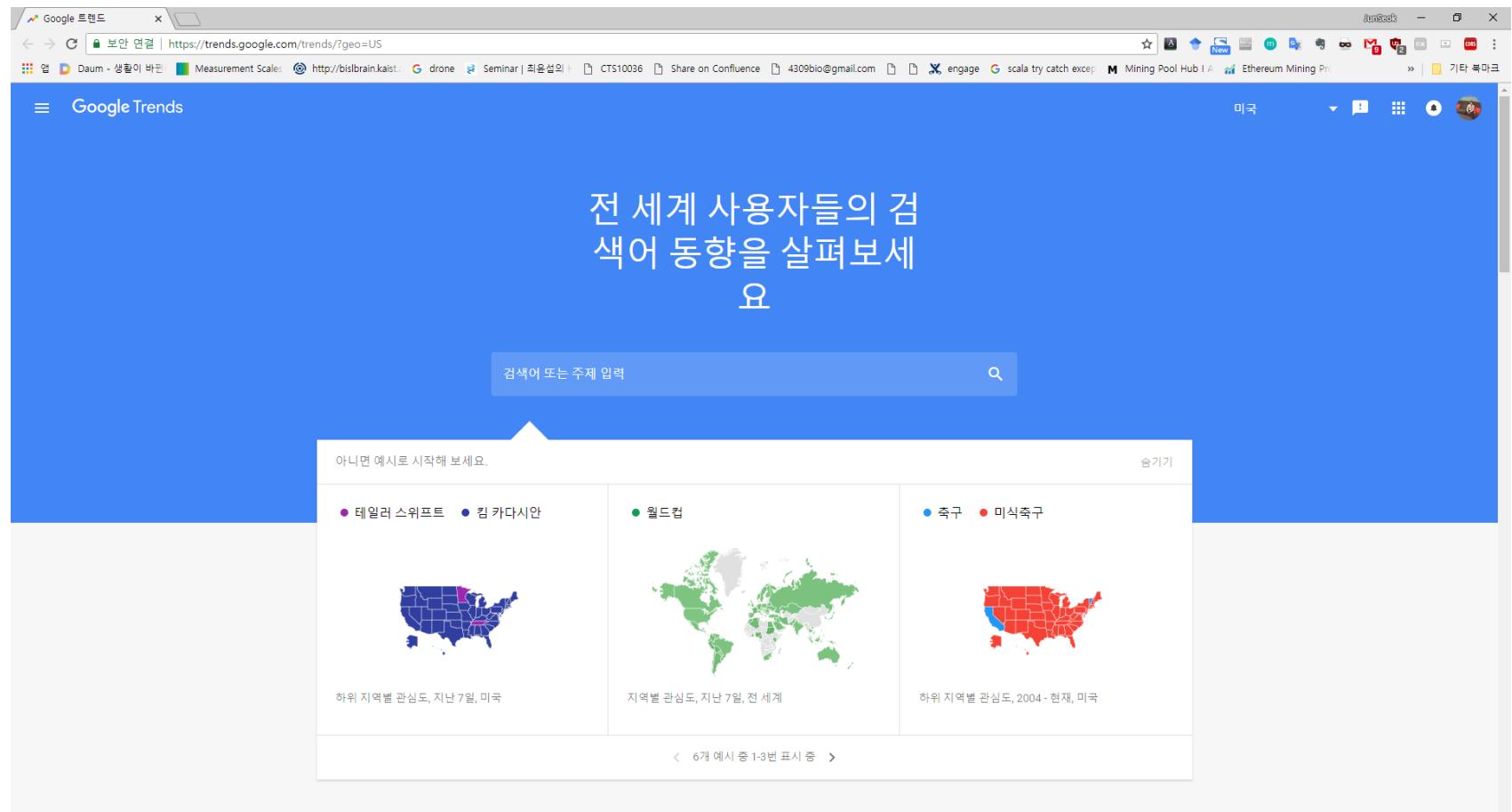


- Check the result by your self

# **Web-based datamining**

# Google trend

- Trend analysis



# Facebook marketing

- <https://bigfoot9.com>

The screenshot shows the BigFoot Text Analysis interface. On the left, a sidebar lists various analysis modules: OVERVIEW, ANALYSIS (Dashboard, KPI Board, Page Analysis, Post Analysis, Activity Analysis, User Analysis), Text Analysis (selected), Event Analysis, Comparison, Custom Report, Data Download, Comment Monitor, Best Influencer, Spread Health, and CONTENT SUITE (Event Live, Content Pool). The main area displays a word cloud centered around the year '2016' and the Korean word '새해' (New Year). A prominent red box labeled 'DEMO' is overlaid on the interface. A tooltip message in Korean states: '▲ 이 페이지는 테스트용 샘플 페이지이며 분석 데이터는 실제 데이터가 아닙니다.' (This page is a test sample page; the analyzed data is not real data). Below the word cloud, there are three summary cards: '210 Words', '7.5 Words Avg', and '167.11 Text length Avg'. At the top, there's a search bar and a navigation bar with links like 'Product & Shop', 'Content Suite', '한국어', and 'Login with Facebook'. The browser address bar shows the URL <https://bigfoot9.com/text>.

# Drug Research Tools

- <https://open.fda.gov/tools/>

The screenshot shows the 'Research tools' section of the openFDA website. The page title is 'Research tools'. Below it, a sub-section title is 'OpenFDA Powered Research Tools'. Six cards are displayed, each representing a different tool:

- Dashboard of drug adverse event reports**  
http://openfda.shinyapps.io/dash/  
This app brings together multiple UI elements—including charts and word clouds—to explore the whole space of adverse event reports. Concomitant drugs, common indications, and the nature of event reports can all be explored for the entire dataset or for custom searches.
- PRR Drug**  
https://openfda.shinyapps.io/RR\_D/  
This app uses the proportional reporting ratio (PRR) to show how common given adverse reactions are for a certain drug, compared with all other drugs. It can guide inquiry into the adverse reactions more likely to be associated with a given drug.
- PRR Event**  
https://openfda.shinyapps.io/RR\_E/  
This app uses the proportional reporting ratio (PRR) to show which drugs are more likely to be associated with a particular adverse reaction, compared with all other drugs. It can guide inquiry into drugs most likely to be associated with a given reaction.
- Dynamic PRR**  
https://openfda.shinyapps.io/dynprr/  
This app uses the proportional reporting ratio (PRR) to show how much more commonly associated a particular reaction is with a particular drug, over time, compared with other drugs.
- Change point analysis for adverse event**  
https://openfda.shinyapps.io/ChangePoint/  
This app shows changes over time in the average number of reports for particular drug and adverse reaction combinations. It can guide inquiry into increased or decreased adverse event reports, of particular reactions, or of particular drugs being reported.
- Drug adverse event report browser**  
https://openfda.shinyapps.io/reportview/  
This app is an interface that supports paging through adverse event reports one at a time, with report fields presented in easy to read tables.

# References

---

- <https://blog.biolab.si/2016/04/25/association-rules-in-orange/>
- <http://dm.kaist.ac.kr/kse525/>
- <https://orange3-associate.readthedocs.io/en/latest/widgets/associationrules.html#example>
- <https://github.com/biolab/orange3-associate>
- <https://orange3-text.readthedocs.io/en/latest/widgets/twitter.html>

**Thank you**

