# 데이터 마이닝 특강
# Practice session

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

Junseok Park

2018-08-30

KAIST

# 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

| *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 |

## Association Rules

{Diaper} → {Beer}
{Milk, Bread} → {Eggs, Coke}
{Beer, Bread} → {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

| *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 |

- **Support count ($\sigma$) or absolute support**
  - Frequency of occurrence of an itemset
    - e.g., $\sigma$({Milk, Bread, Diaper}) = 2

- **(Relative) support**
  - Fraction of transactions that contain an itemset
    - e.g., s({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→Y, where X and Y are itemsets
    - e.g., {Milk, Diaper}→{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:

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

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

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

# Association Rule Mining Task

- **Given a set of transactions *T*, the goal of association rule mining is to find all rules such that**
  - support ≥ *minsup* threshold
  - confidence ≥ *minconf* threshold

- **Brute-force approach:**
  - List all possible association rules
  - Compute the support and confidence for each rule
  - Prune rules that fail the *minsup* and *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:

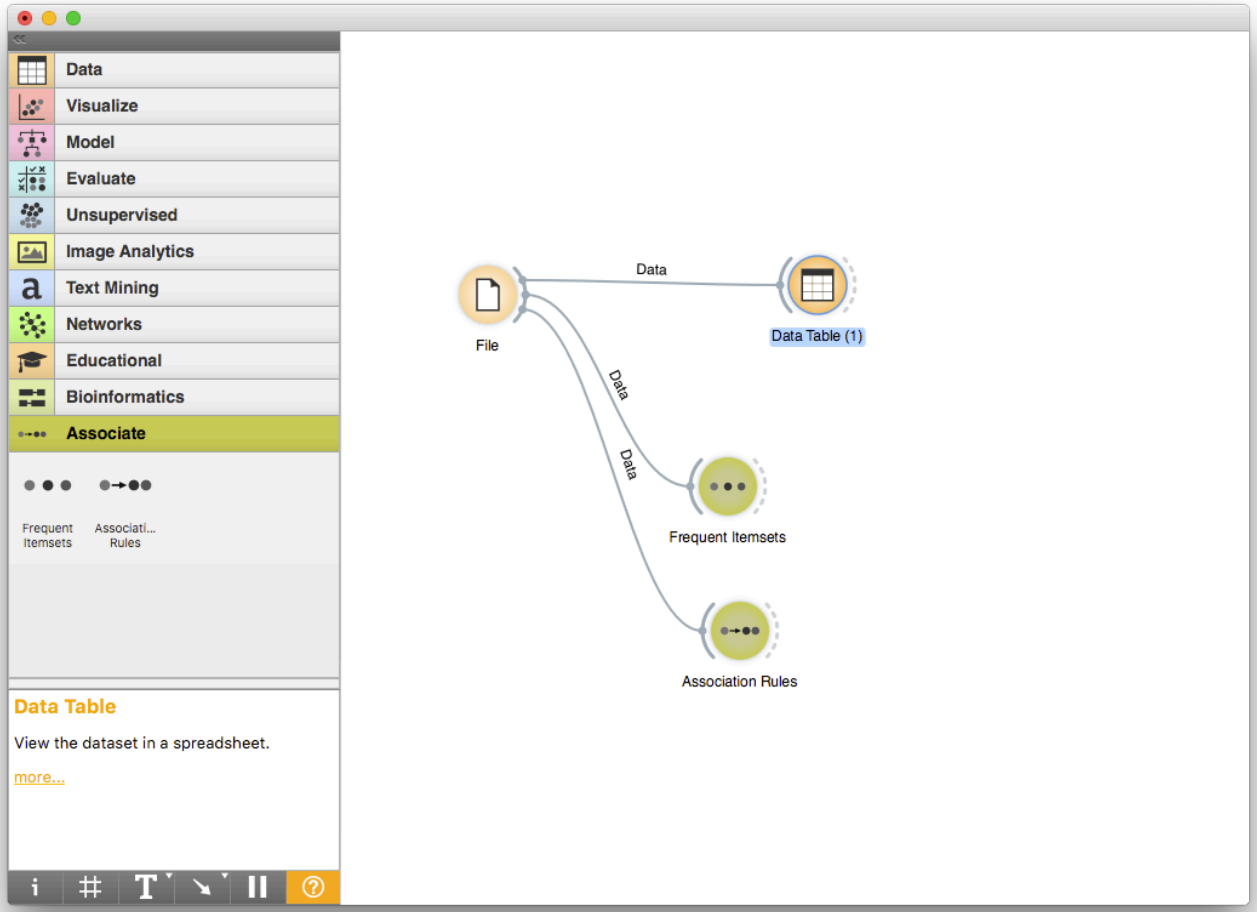
{Milk,Diaper} → {Beer} (s=0.4, c=0.67)
{Milk,Beer} → {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} → {Milk} (s=0.4, c=0.67)
{Beer} → {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} → {Milk,Beer} (s=0.4, c=0.5)
{Milk} → {Diaper,Beer} (s=0.4, c=0.5)

- **Observations:**
  - All the above rules are binary partitions of the same itemset: {Milk, Diaper, 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)

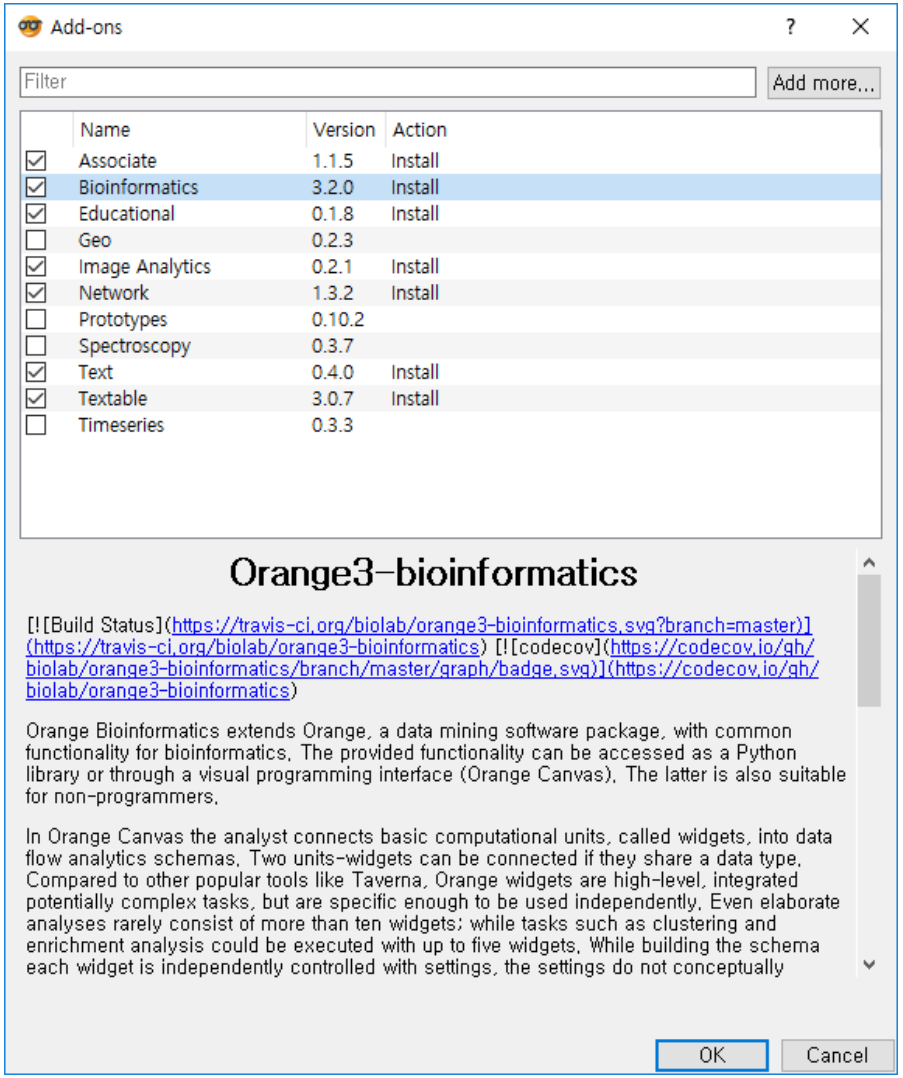
- **Frequent Itemset Generation**
  - Generating all items whose support ≥ *minsup*
- **Rule Generation**
  - Generating high confidence rules from each frequent itemset, where each rule is a binary partitioning of a frequent itemset
- **Frequent itemset generation is still computationally expensive**
- **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



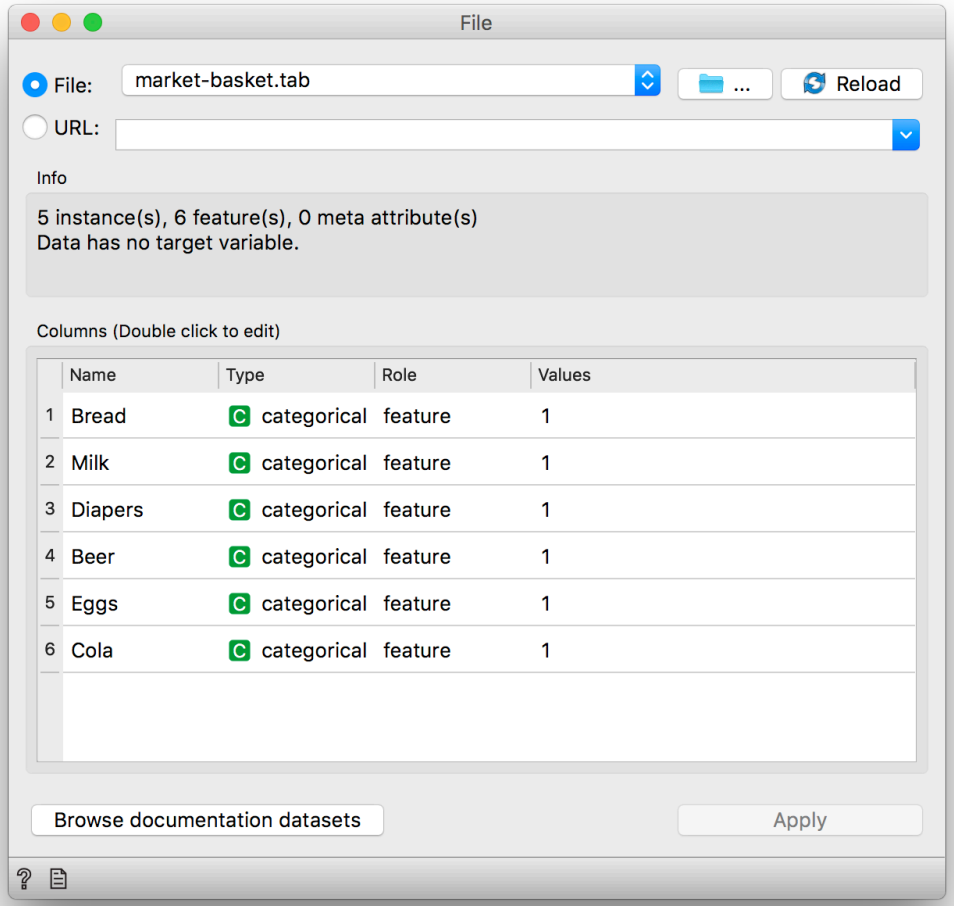
# Practice preparation

- **Install add-ons**

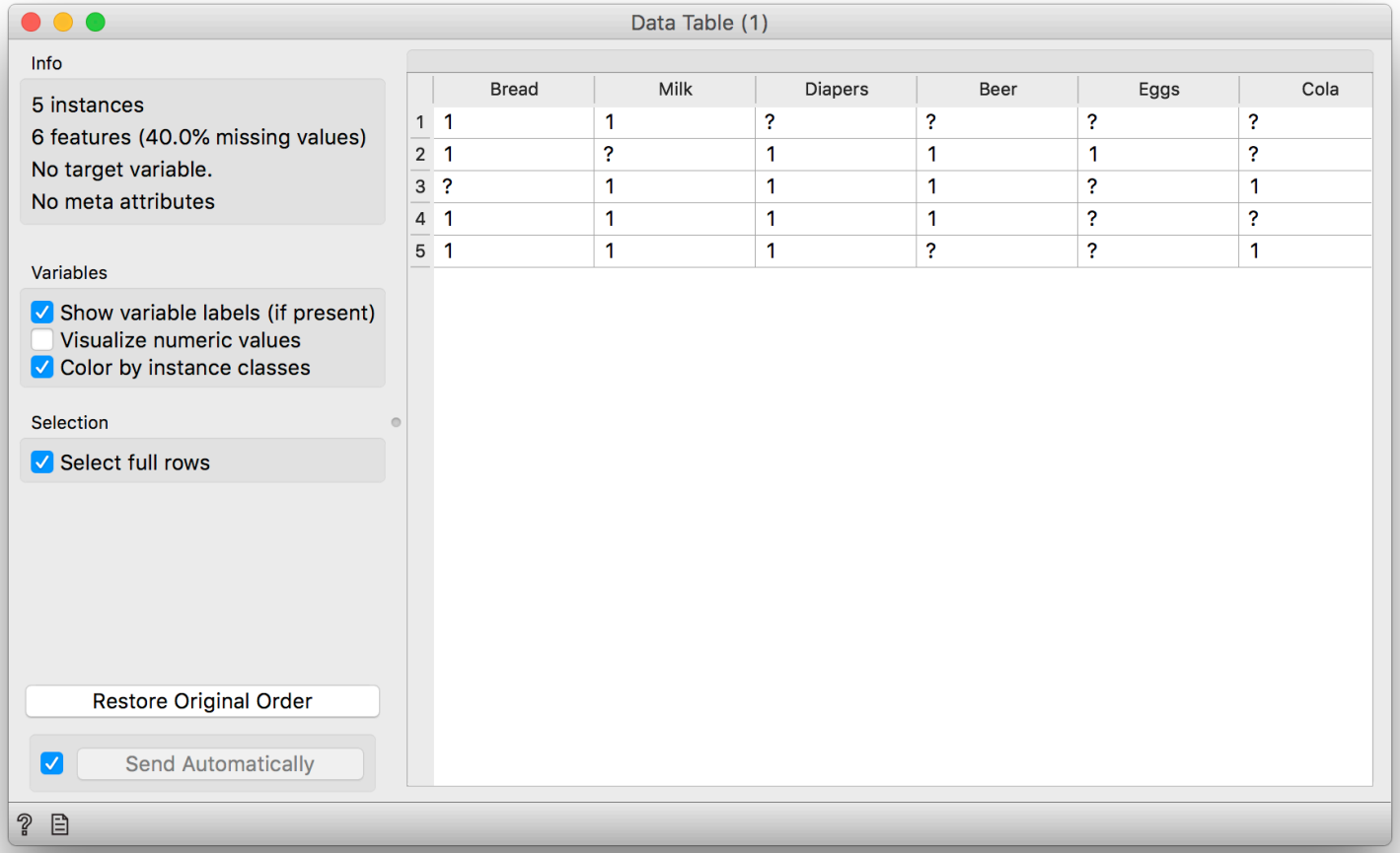


# Dataset

- **Market basket data**

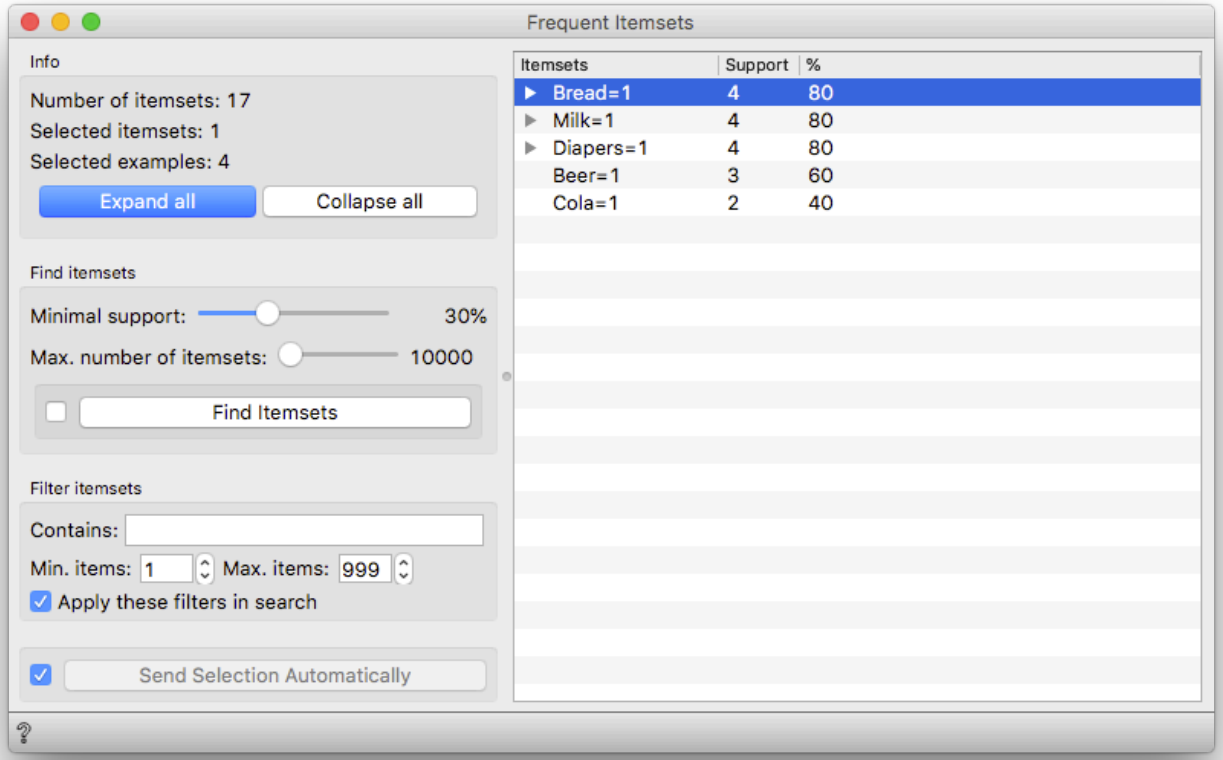


# Overview data



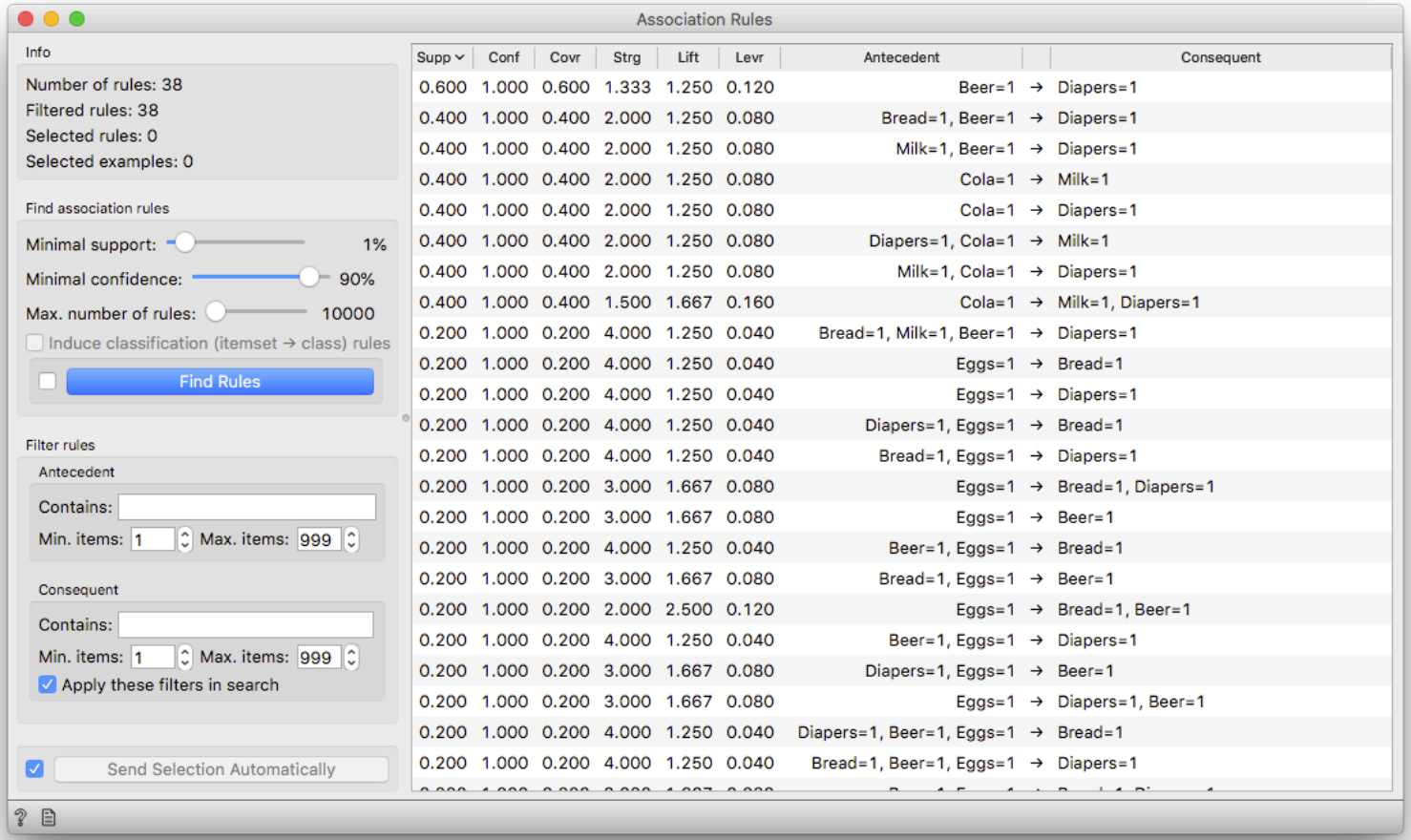
# Frequent item set

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



# Association Rules

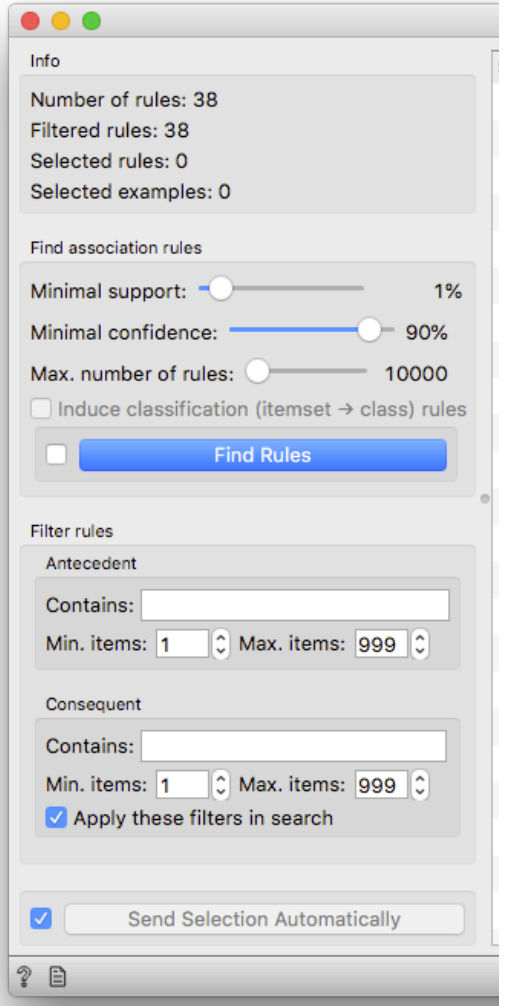
- **how about some transaction flows?**




| 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 |

# 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 (support*data - antecedent support * consequent support/data2)



# Self study data

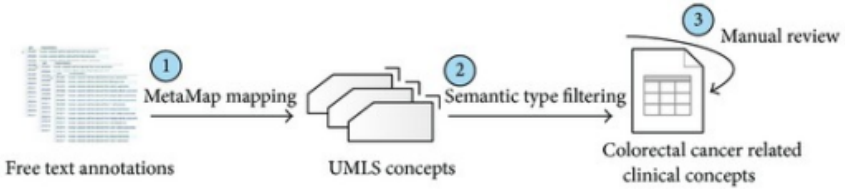
- **Foodmart 2000 dataset**
  - https://github.com/neo4j-examples/neo4j-foodmart-dataset/tree/master/data
- **Kaggle Apriori Algorithm**
  - https://www.kaggle.com/datatheque/association-rules-mining-market-basket-analysis

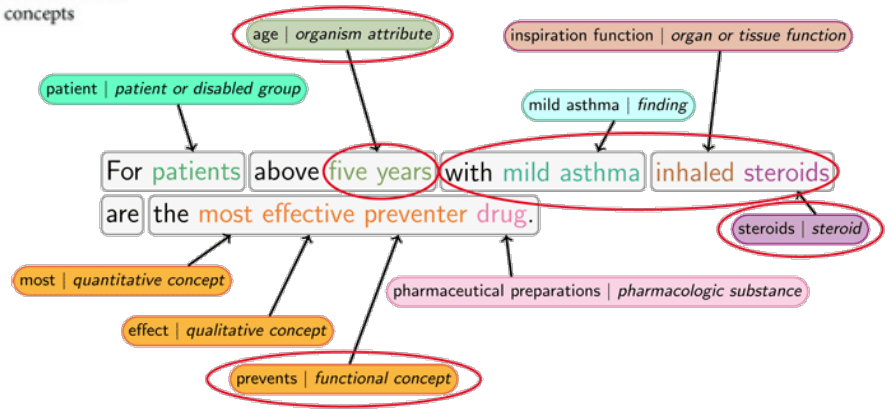
# Text Mining

# 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 | • It provides a UIMA wrapper for the OpenNLP opennlp.tools.chunker.Chunker class<br>• This wrapper can generate chunks of any kind as specified by the chunker model and the chunk creator | Chunker |
| 2 | Tokenizer Annotator | • Discovers tokens in the given text, following Penn Treebank tokenization rules.<br>• These tokens consist of words, punctuation, etc..... | TokenizerAnnotatorPTB |
| 3 | Context Dependent Tokenizer Annotator | • Find tokens based on context | ContextDependentTokenizerAnnotator |
| 4 | Dictionary Lookup Annotator DB | • UIMA annotator that identified entities based on lookup<br>• Specifies the maximum number of items to be returned from an lucene query | UmlsDictionaryLookupAnnotator |
| 5 | Status Annotator | • Analyze a list of tokens looking for a status pattern as specified by the class StatusIndicatorFSM | StatusrContextAnalyzer |
| 6 | Negation Annotator | • analyzes a list of tokens looking for a negation pattern as specified by the class NegationFSM | NegationContextAnalyzer |
| 7 | Extraction PreAnnotator | • UIMA annotator that prepares the CAS for extraction into DB2<br>• Performs some(final) updates to the CAS | ExtractionPrepAnnotator |
| 8 | Sentence Detector Annotator | • Wraps the OpenNLP sentence detector in a UIMA annotator<br>• Discovers sentence boundaries | SentenceDetector |
| 9 | Lookup Window Annotator | • Select pre-existing annotations in the CAS to create LookupWindow annotation from | org.apache.uima.java |
| 10 | Adjust Noun Phrase To Include Following NP | • To extend NP annotations to include prepositional phrases so that for the pattern NP PP NP, named entities that includes a word(s) from each of those NPs is found<br>• Adjust NP in NP NP to span both | ChunkAdjuster |

# Text mining pipeline example (2/3)

| No | Analysis Engine Name | Description | Class |
|---|---|---|---|
| 11 | Adjust Noun Phrase To Include Following PP NP | • Prevents NP annotations from only partially overlapping other NP annotations<br>• This annotator is written to be able to handle more general cases than NP PP NP<br>• Adjust NP in NP PP NP to spall all there | ChunkAdjuster |
| 12 | Simple Segment Annotator | • Creates a single Segment annotation, encompassing the entire document<br>• 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 | SimpleSgementAnnotator |
| 13 | POSTagger | • Load a file that contains the MaxEnt model used by the part of speech(POS) tagger | POSTagger |
| 14 | Lvg Annotator | • UIMA annotator that uses the UMLS LVG(Lexical Variant Generation) package to find the canonical form of WordTokens<br>• The package is also used to find one or more lemmas for a given WordToken along with its associated part of speech | LvgAnnotator (LVG) |
| 15 | Generic Cleartk Analysis Engine | | GenericCleartkAnalysisEngine |
| 16 | History Cleartk Analysis Engine | | HistoryCleartkAnalysisEngine |
| 17 | Polarity Cleartk Analysis Engine | | Polarity CleartkAnalysisEngine |
| 18 | Subject Cleartk Analysis Engine | | SubjectCleartkAnalysisEngine |
| 19 | Uncertainty Cleartk Analysis Engine | | UncertaintyCleartkAnalysisEngine |
| 20 | DependencyParser | • Provides a UIMA wrapper for the CLEAR dependency parser | ClearNLPDependencyParserAE (Ref link) |

# Text mining pipeline example (3/3)

| No | Analysis Engine Name | Description | Class |
|---|---|---|---|
| 21 | Semantic Role Labeler | • Provides a UIMA wrapper for the ClearNLP Semantic Role Labeler<br>• Before using this AnalysisEngine, user should run a Tokenizer, POS-tagger, Lemmatizer, and the CLEAR parser dependency parser | ClearNLPSemanticRoleLabelerAE |
| 22 | Constituency Parser | • Extends org.apache.uima.analysis_component.JCasAnnotator_ImplBase | ConstituencyParser Annotator |

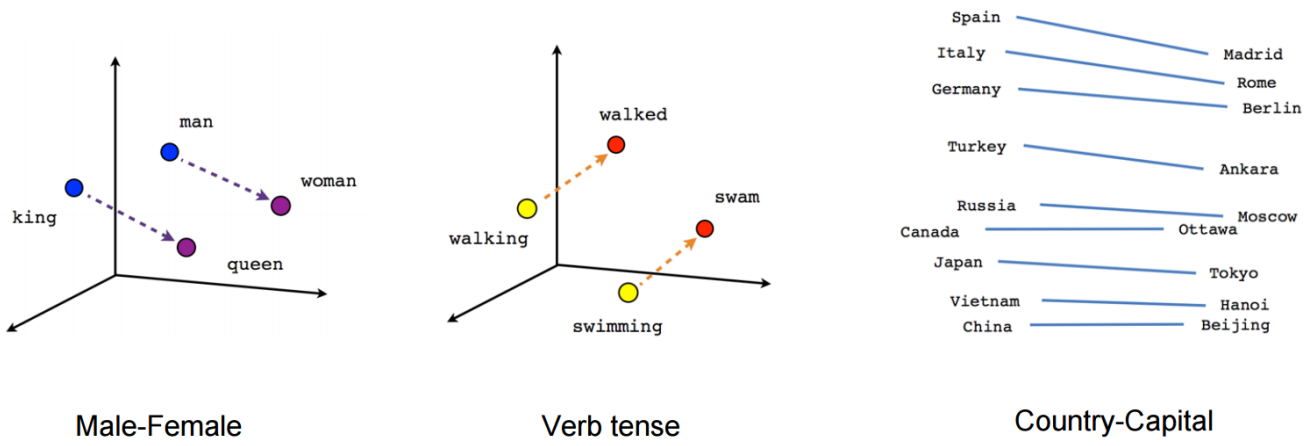
# Word Embedding

# 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-β 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

**1** **Dictionary**

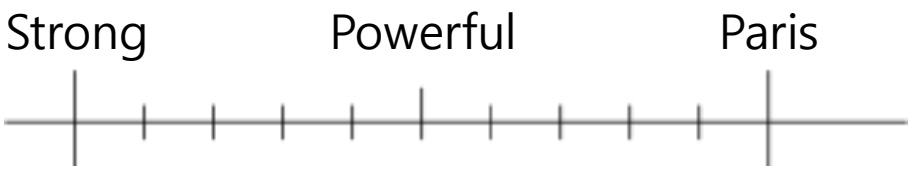
```
{ "John": 0,
  "likes": 1,
  "to": 2,
  "watch": 3,
  "movies": 4,
  "also": 5,
  "football": 6,
  "games": 7,
  "Mary": 8,
  "too": 9 }
```

**2** **Bag-of-Word**

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

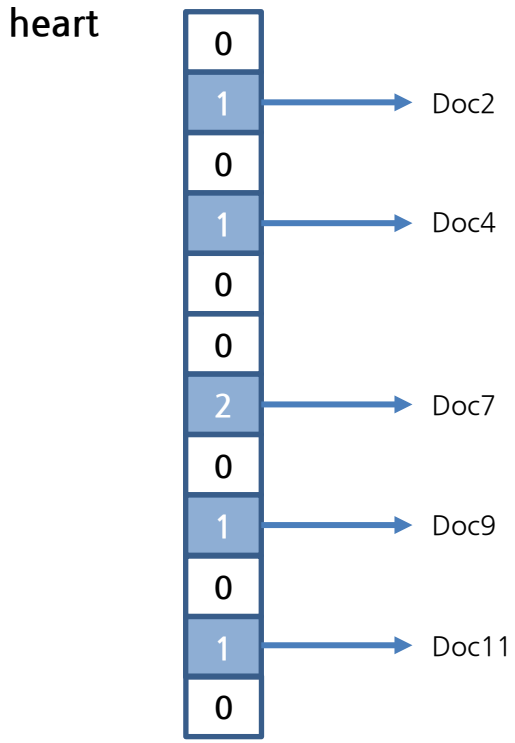
John likes to watch movies. Mary likes movie too.
-> [1,2,1,1,2,0,0,0,1,1]
John also likes to watch football game.
-> [1,1,1,1,0,1,1,1,0,0]

  - Example #2



# Introduction

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

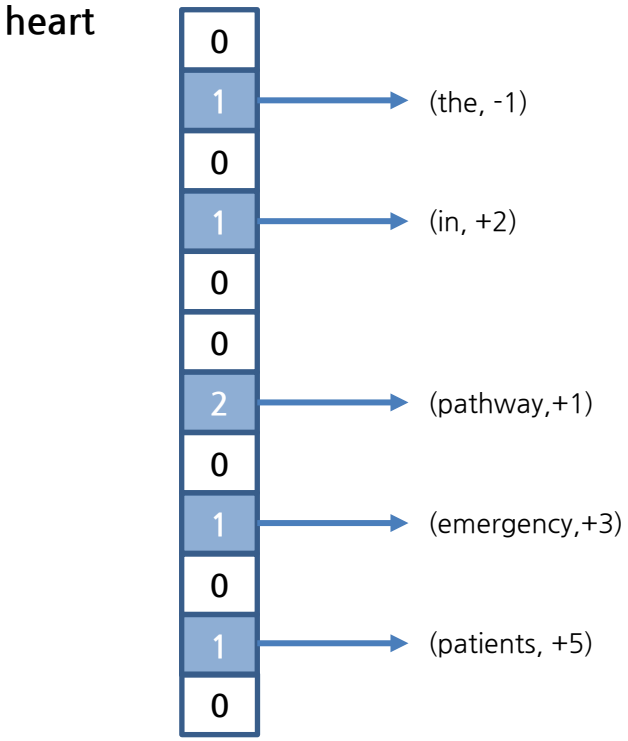



The vector can correspond to documents in which the word occurs

Some slides were referred from https://www.slideshare.net/BhaskarMitra3/a-simple-introduction-to-word-embeddings

# 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

Some slides were referred from https://www.slideshare.net/BhaskarMitra3/a-simple-introduction-to-word-embeddings

# 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

Some slides were referred from https://www.slideshare.net/BhaskarMitra3/a-simple-introduction-to-word-embeddings

# Similarity and Relatedness

- **Definition of Semantic Similarity and Relatedness[1]**

| Semantic Similarity | Semantics Relatedness |
| --- | --- |
|  |  |
| Measures quantify how “alike” (or similar) two concepts are by determining their closeness in a hierarchy | 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.

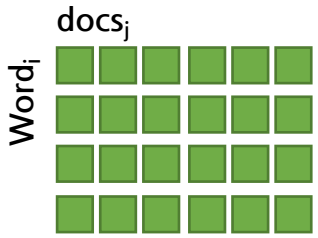
[1] McInnes, B. T., et al. (2009). UMLS-Interface and UMLS-Similarity: open source software for measuring paths and semantic similarity. AMIA Annual Symposium Proceedings, American Medical Informatics Association.

# 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
- This dense vectors are called embeddings

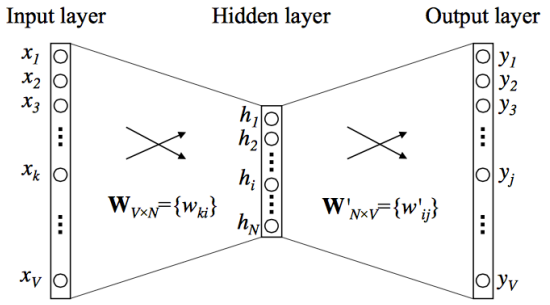
# Dense Embeddings

- **Matrix Factorization**
  - Factorize word-context matrix




  - Examples
    - LDA (Word-Document)[1]
    - GloVe (Word-NeighboringWord)[2]

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




  - Example
    - Word2Vec (Word-NeighboringWord)[3]


[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

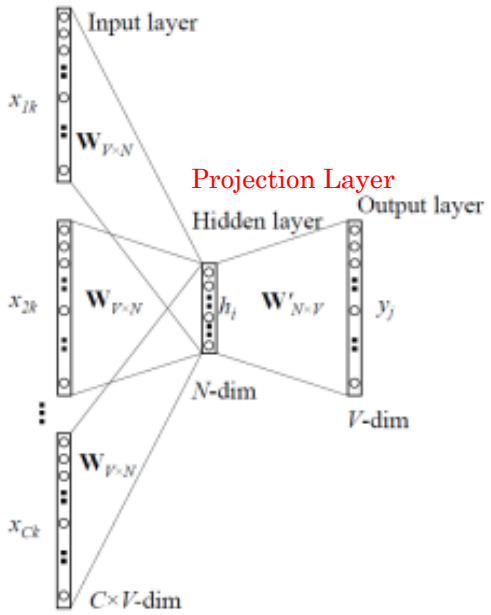
```
int batchSize = 1000;
int iterations = 3;
int layerSize = 150;

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 to ln(V) : Complexity Reduction**

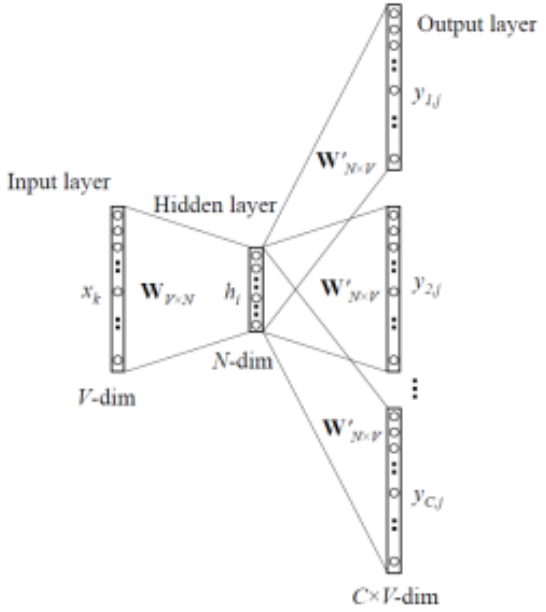
*CBOW Architecture*




**Calculation Amount**

- C개의 단어를 Projection 하는 데 CXN
- Projection Layer에서 Output Layer로 가는 데 NxV
- 전체 계산량 : CxN + NxV

*Skip-gram Architecture*

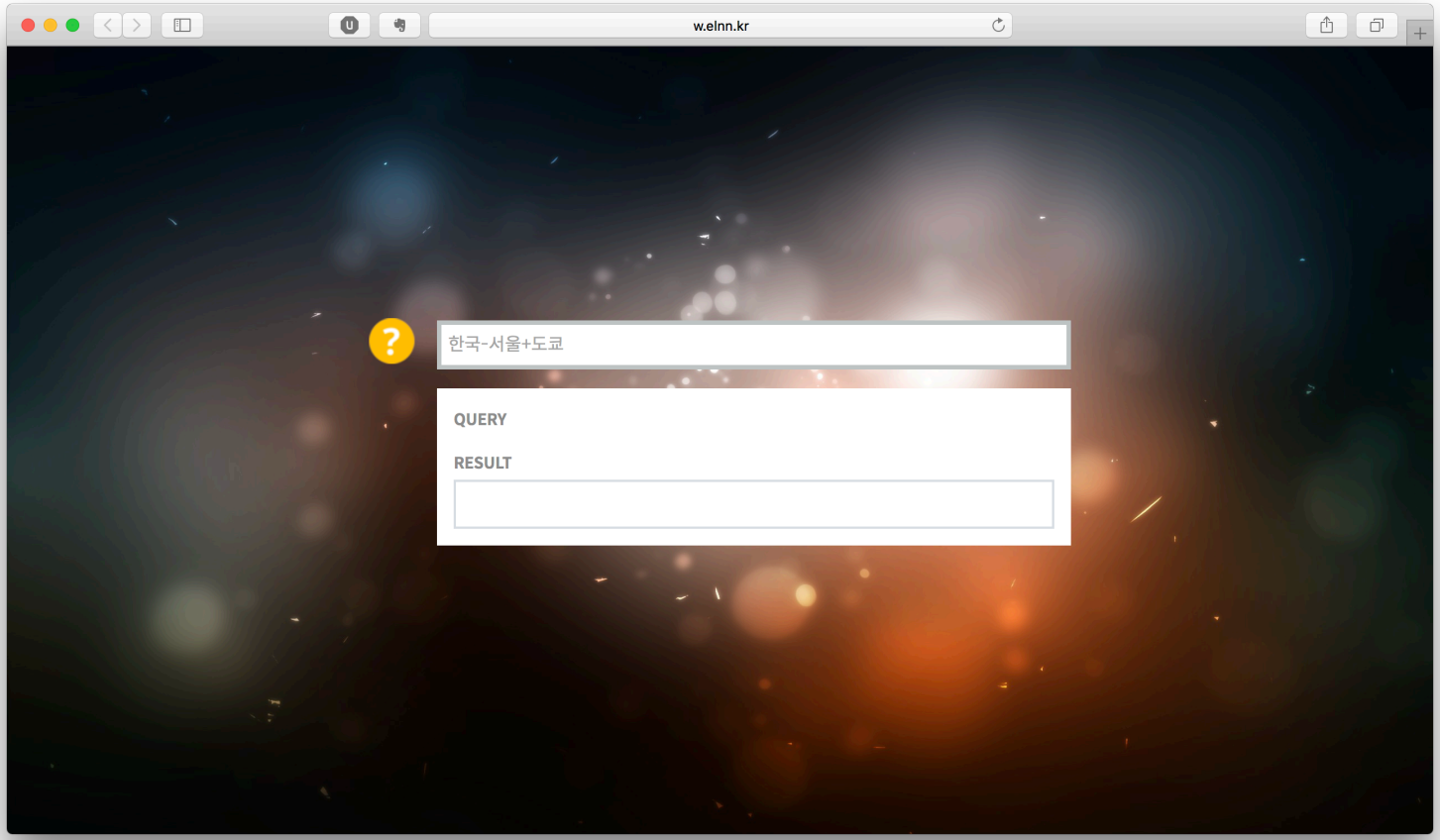



**Calculation Amount**

- 현재 단어를 Projection 하는 데 N
- Output을 계산하는 데에 NxV
- 총 C개의 단어에 대해 진행하므로 C배
- 전체 계산량 : C(N+NxV)

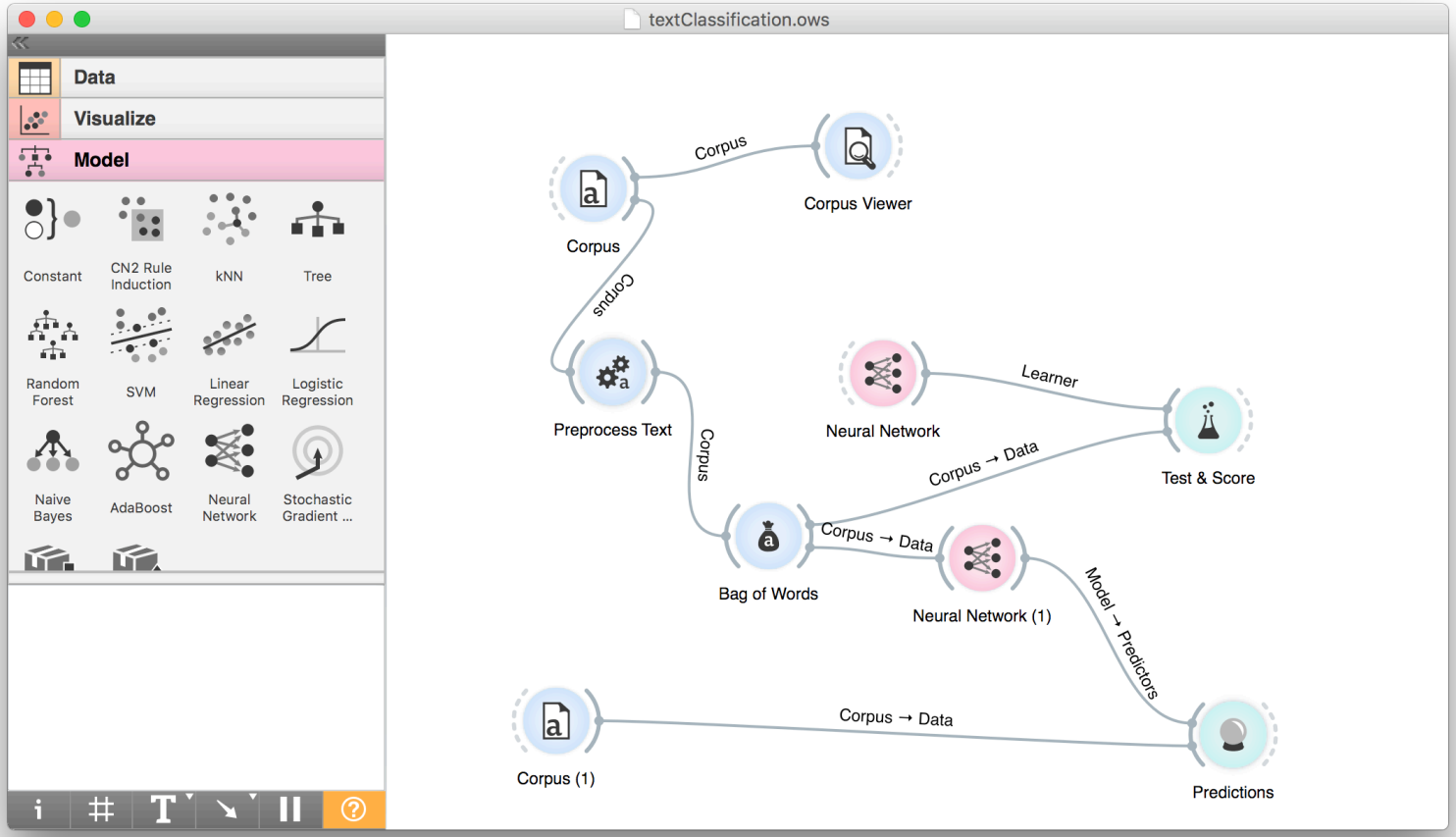
# Demo

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



# Build Schema

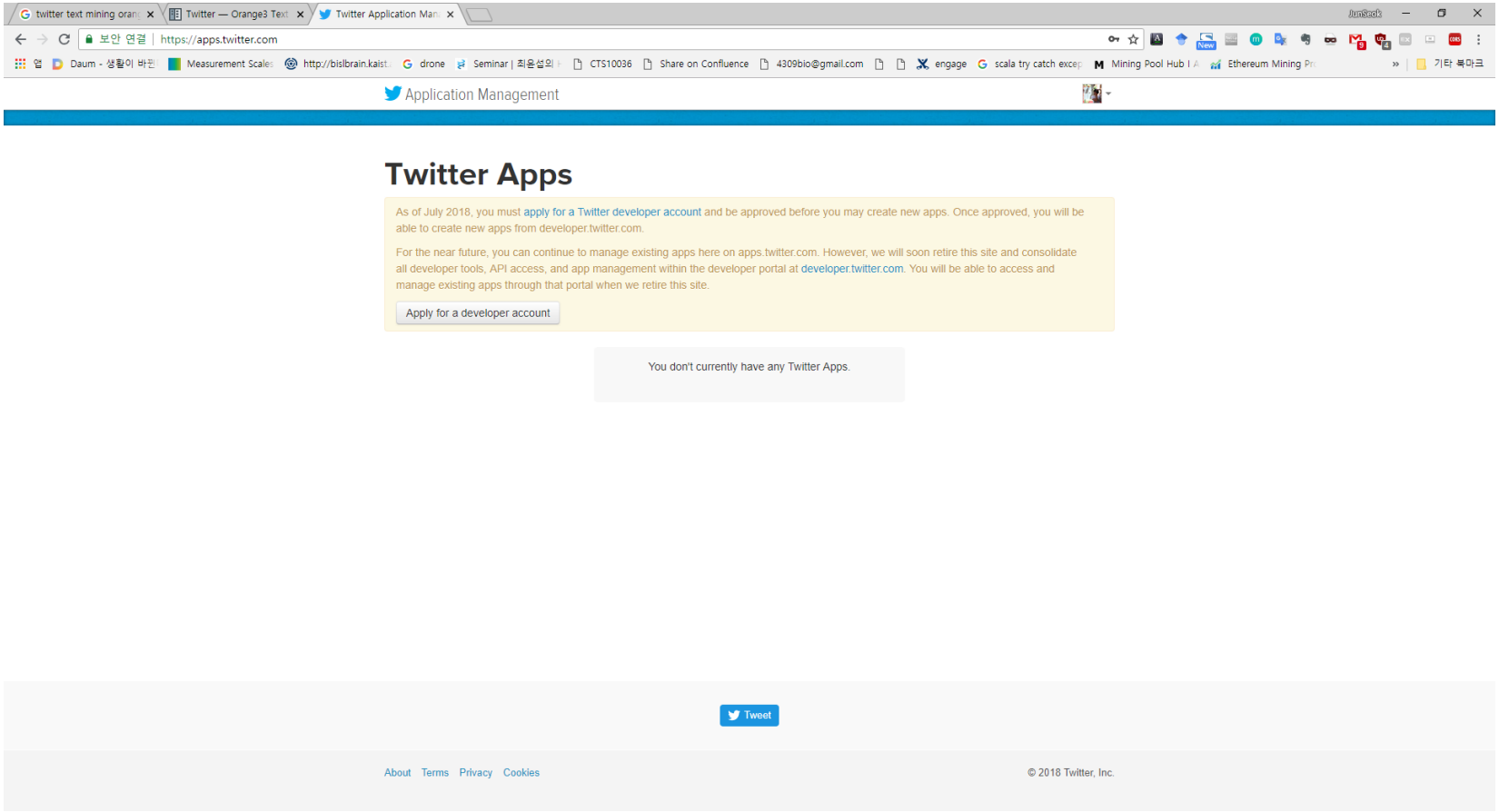
- **Text Classification Playground**



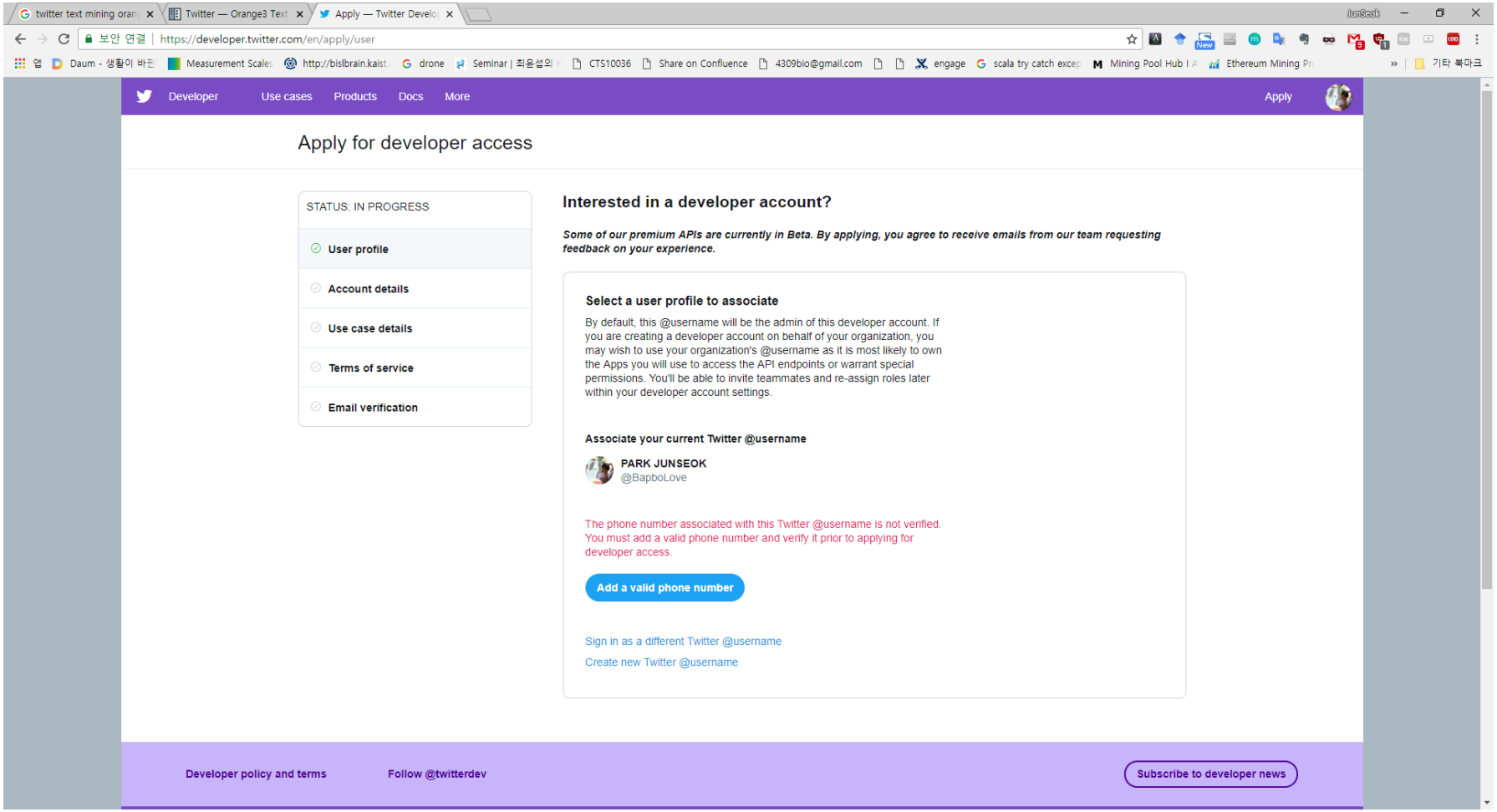
# Twitter Example

# Twitter API

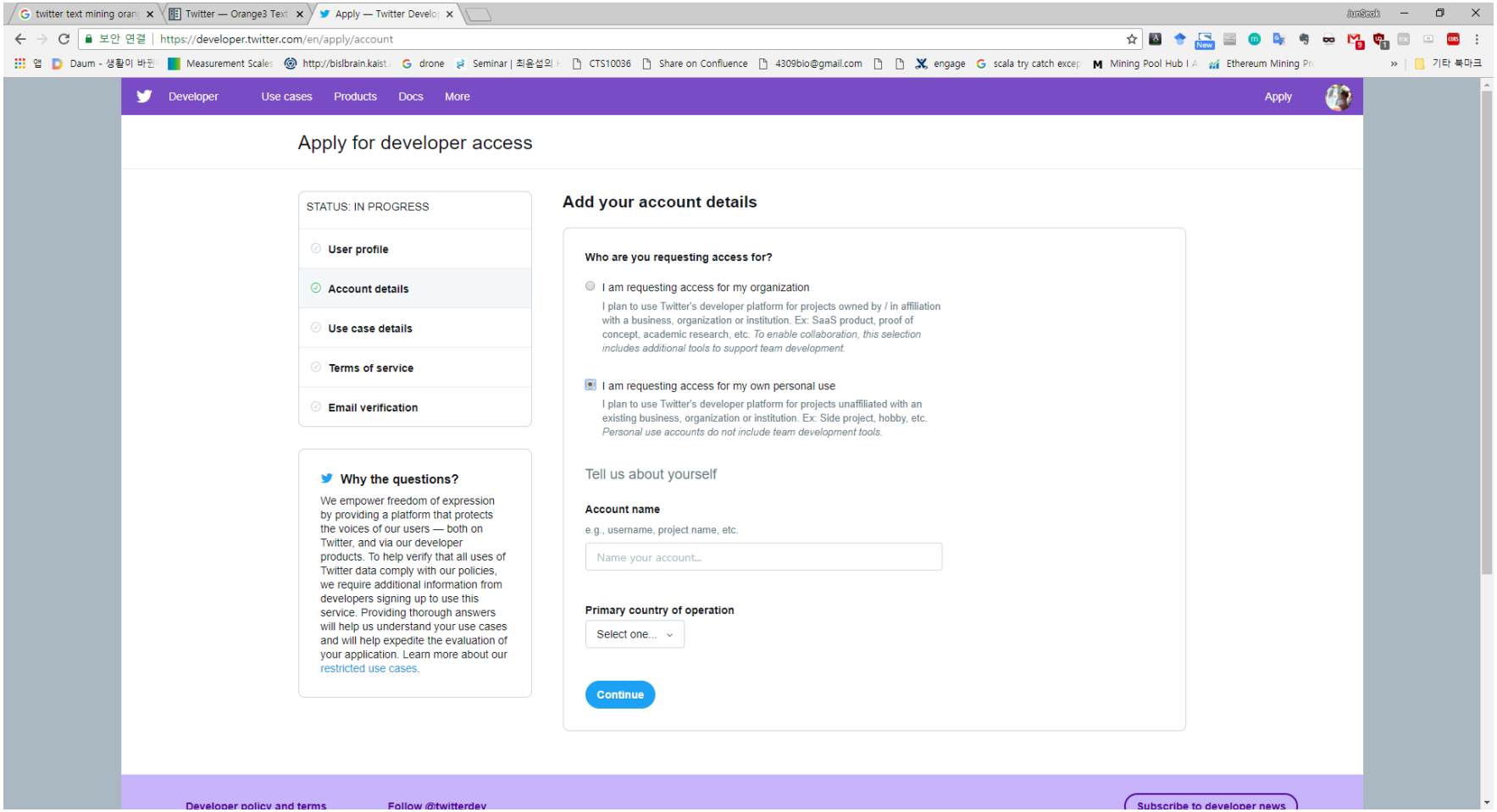
- **https://apps.twitter.com/**



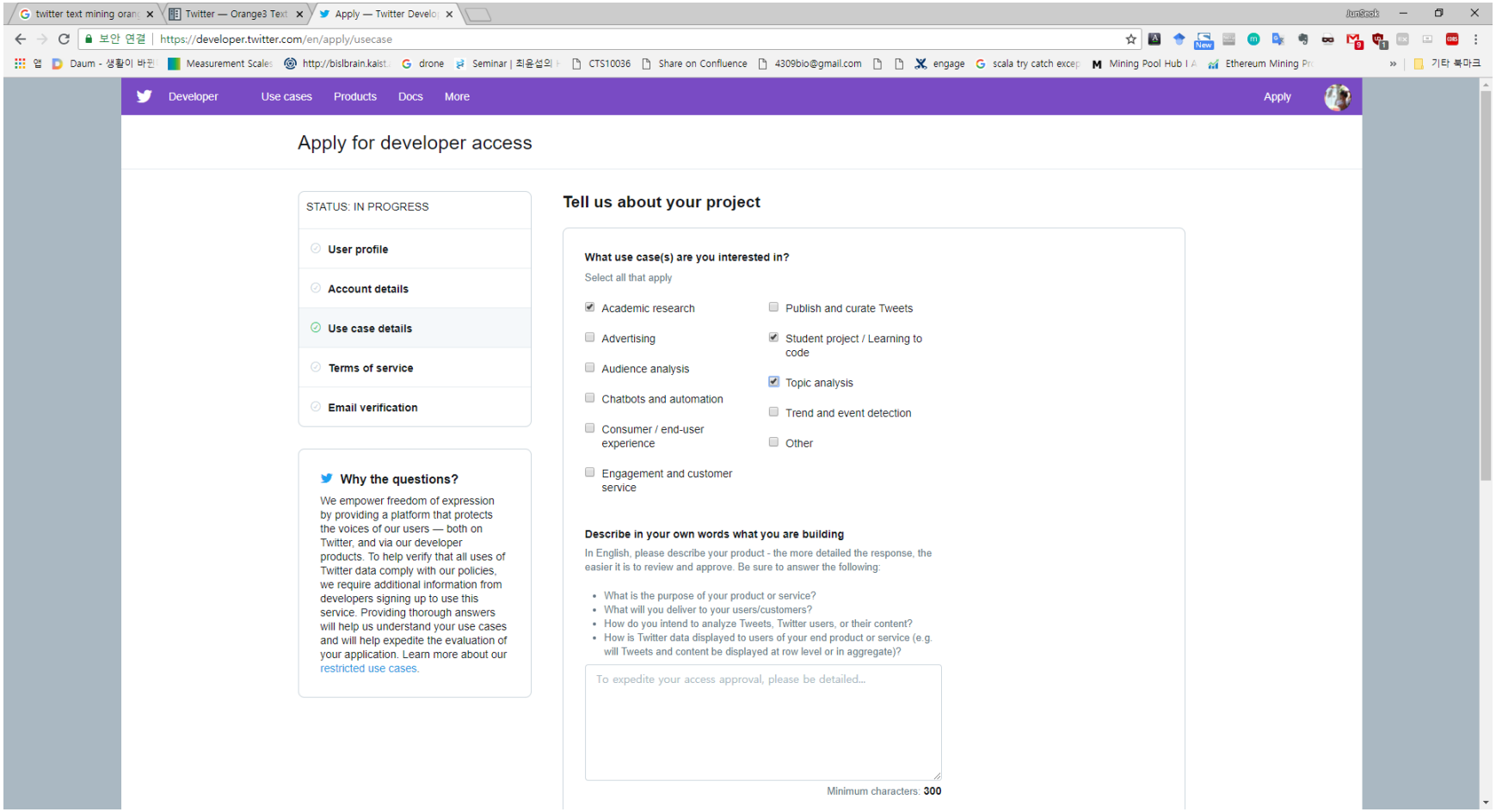
# Twitter API


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

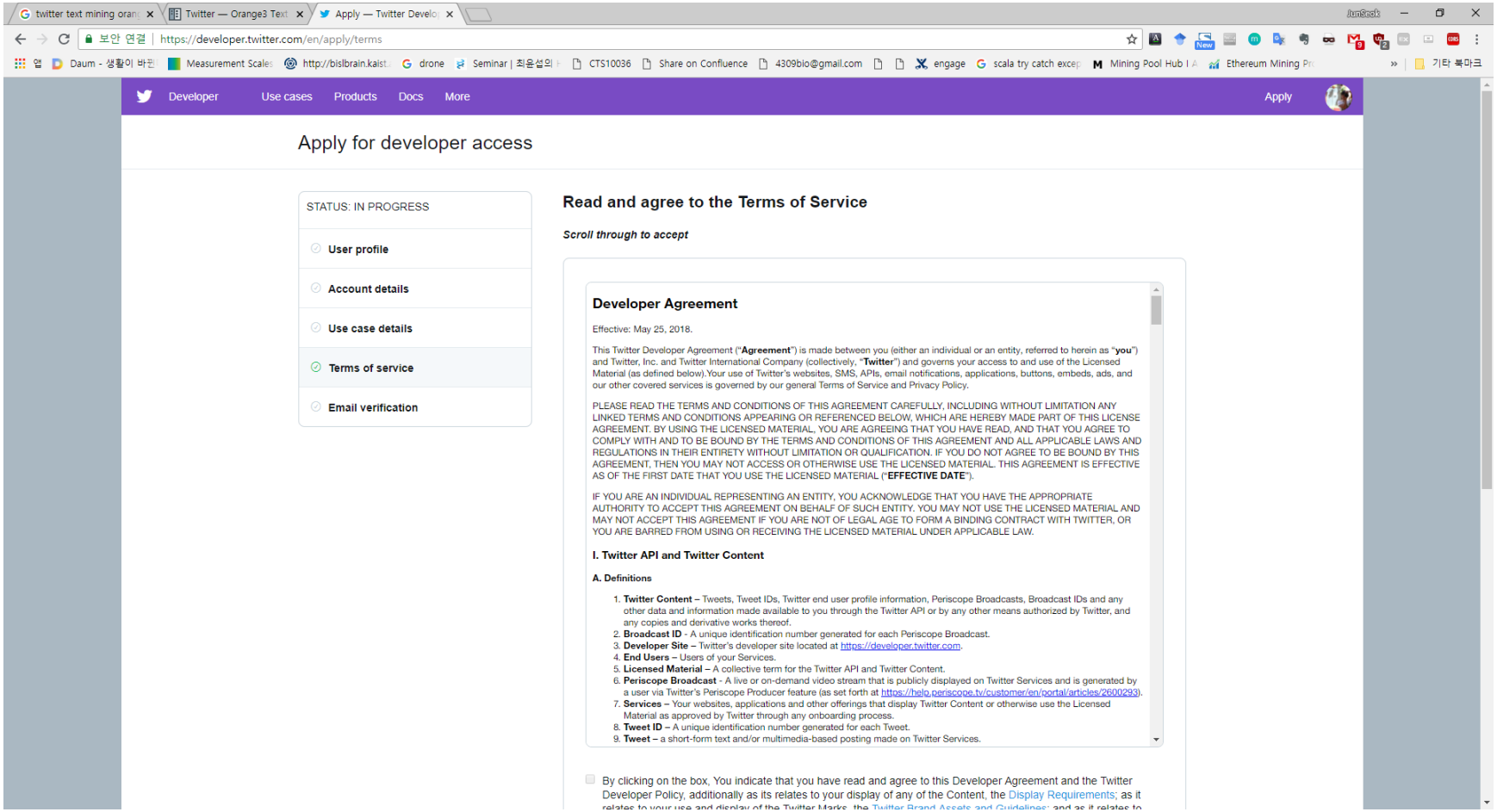
# Twitter API


twitter text mining oran
Twitter — Orange3 Text
Apply — Twitter Develo
https://developer.twitter.com/en/apply/account
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
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

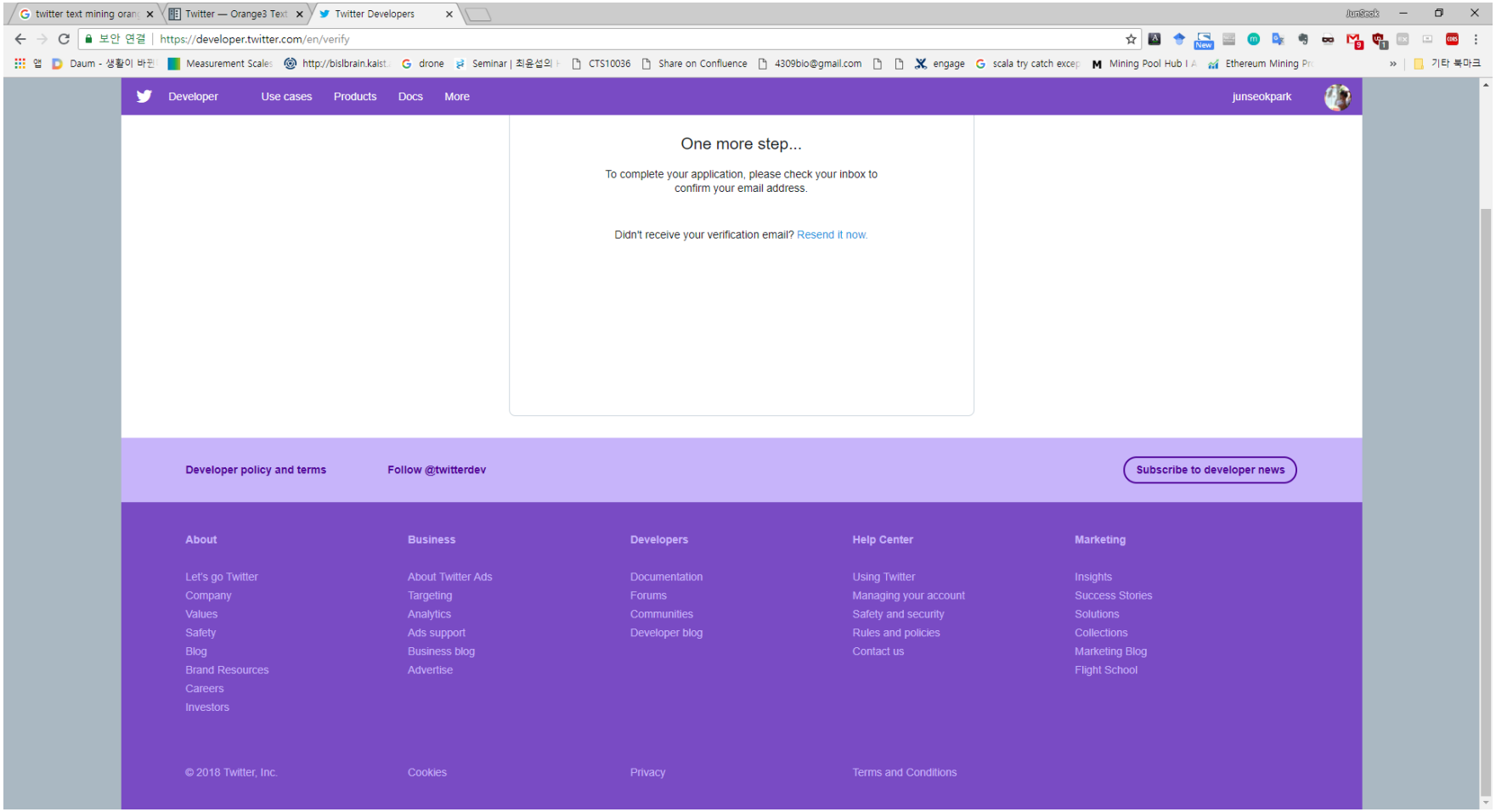
# Twitter API


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
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
Advertising
Audience analysis
Chatbots and automation
Consumer / end-user experience
Engagement and customer service
Publish and curate Tweets
Student project / Learning to code
Topic analysis
Trend and event detection
Other
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

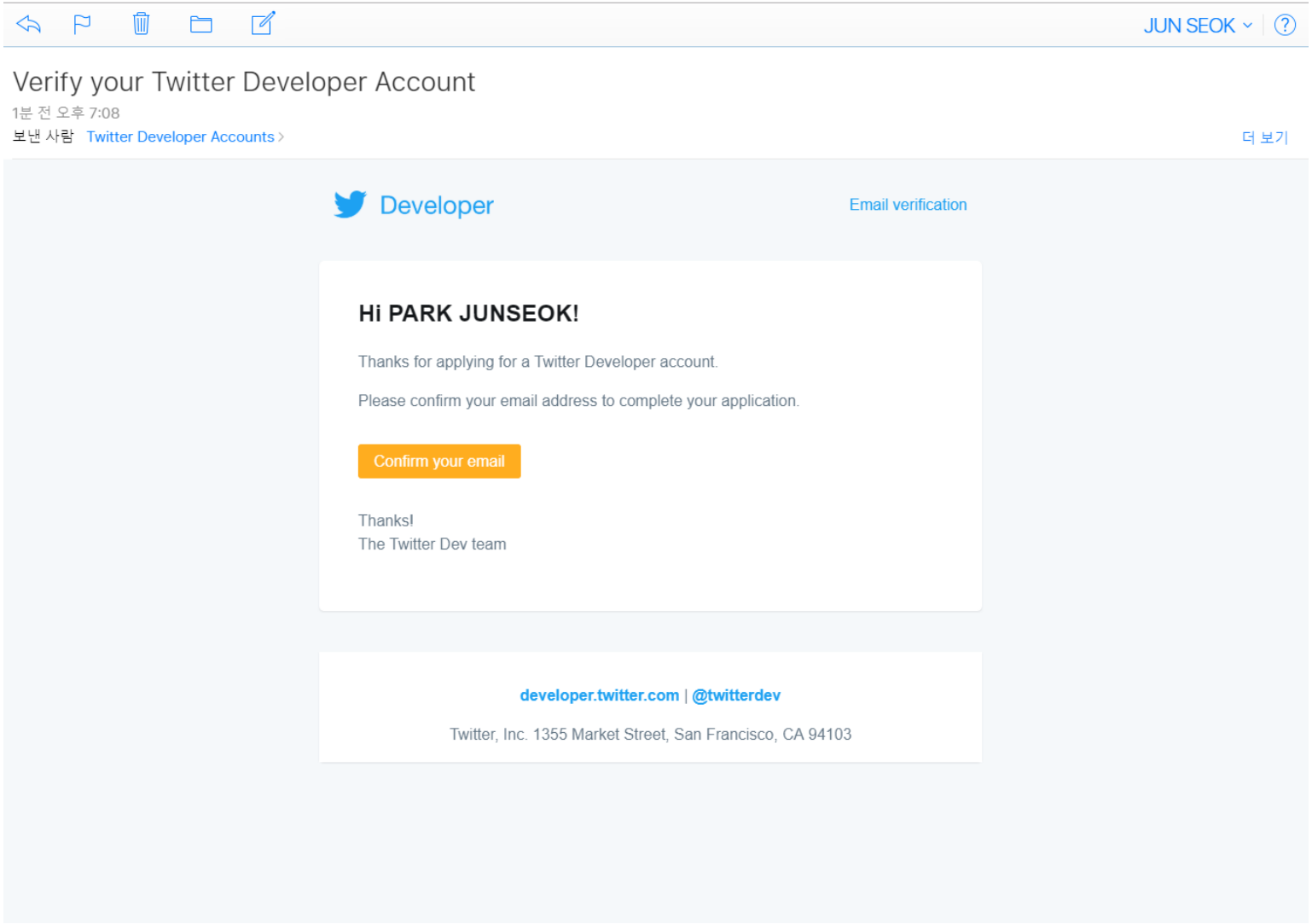
# Twitter API


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
Read and agree to the Terms of Service
Scroll through to accept
Developer Agreement
Effective: May 25, 2018.
This Twitter Developer Agreement ("Agreement") is made between you (either an individual or an entity, referred to herein as "you") and Twitter, Inc. and Twitter International Company (collectively, "Twitter") and governs your access to and use of the Licensed Material (as defined below).Your use of Twitter's websites, SMS, APIs, email notifications, applications, buttons, embeds, ads, and our other covered services is governed by our general Terms of Service and Privacy Policy.
PLEASE READ THE TERMS AND CONDITIONS OF THIS AGREEMENT CAREFULLY, INCLUDING WITHOUT LIMITATION ANY LINKED TERMS AND CONDITIONS APPEARING OR REFERENCED BELOW, WHICH ARE HEREBY MADE PART OF THIS LICENSE AGREEMENT. BY USING THE LICENSED MATERIAL, YOU ARE AGREEING THAT YOU HAVE READ, AND THAT YOU AGREE TO COMPLY WITH AND TO BE BOUND BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND ALL APPLICABLE LAWS AND REGULATIONS IN THEIR ENTIRETY WITHOUT LIMITATION OR QUALIFICATION. IF YOU DO NOT AGREE TO BE BOUND BY THIS AGREEMENT, THEN YOU MAY NOT ACCESS OR OTHERWISE USE THE LICENSED MATERIAL. THIS AGREEMENT IS EFFECTIVE AS OF THE FIRST DATE THAT YOU USE THE LICENSED MATERIAL ("EFFECTIVE DATE").
IF YOU ARE AN INDIVIDUAL REPRESENTING AN ENTITY, YOU ACKNOWLEDGE THAT YOU HAVE THE APPROPRIATE AUTHORITY TO ACCEPT THIS AGREEMENT ON BEHALF OF SUCH ENTITY. YOU MAY NOT USE THE LICENSED MATERIAL AND MAY NOT ACCEPT THIS AGREEMENT IF YOU ARE NOT OF LEGAL AGE TO FORM A BINDING CONTRACT WITH TWITTER, OR YOU ARE BARRED FROM USING OR RECEIVING THE LICENSED MATERIAL UNDER APPLICABLE LAW.
I. Twitter API and Twitter Content
A. Definitions
1. Twitter Content – Tweets, Tweet IDs, Twitter end user profile information, Periscope Broadcasts, Broadcast IDs and any other data and information made available to you through the Twitter API or by any other means authorized by Twitter, and any copies and derivative works thereof.
2. Broadcast ID - A unique identification number generated for each Periscope Broadcast.
3. Developer Site – Twitter's developer site located at https://developer.twitter.com.
4. End Users – Users of your Services.
5. Licensed Material – A collective term for the Twitter API and Twitter Content.
6. Periscope Broadcast - A live or on-demand video stream that is publicly displayed on Twitter Services and is generated by a user via Twitter's Periscope Producer feature (as set forth at https://help.periscope.tv/customer/en/portal/articles/2600293).
7. Services – Your websites, applications and other offerings that display Twitter Content or otherwise use the Licensed Material as approved by Twitter through any onboarding process.
8. Tweet ID – A unique identification number generated for each Tweet.
9. Tweet – a short-form text and/or multimedia-based posting made on Twitter Services.
By clicking on the box, You indicate that you have read and agree to this Developer Agreement and the Twitter Developer Policy, additionally as its relates to your display of any of the Content, the Display Requirements; as it

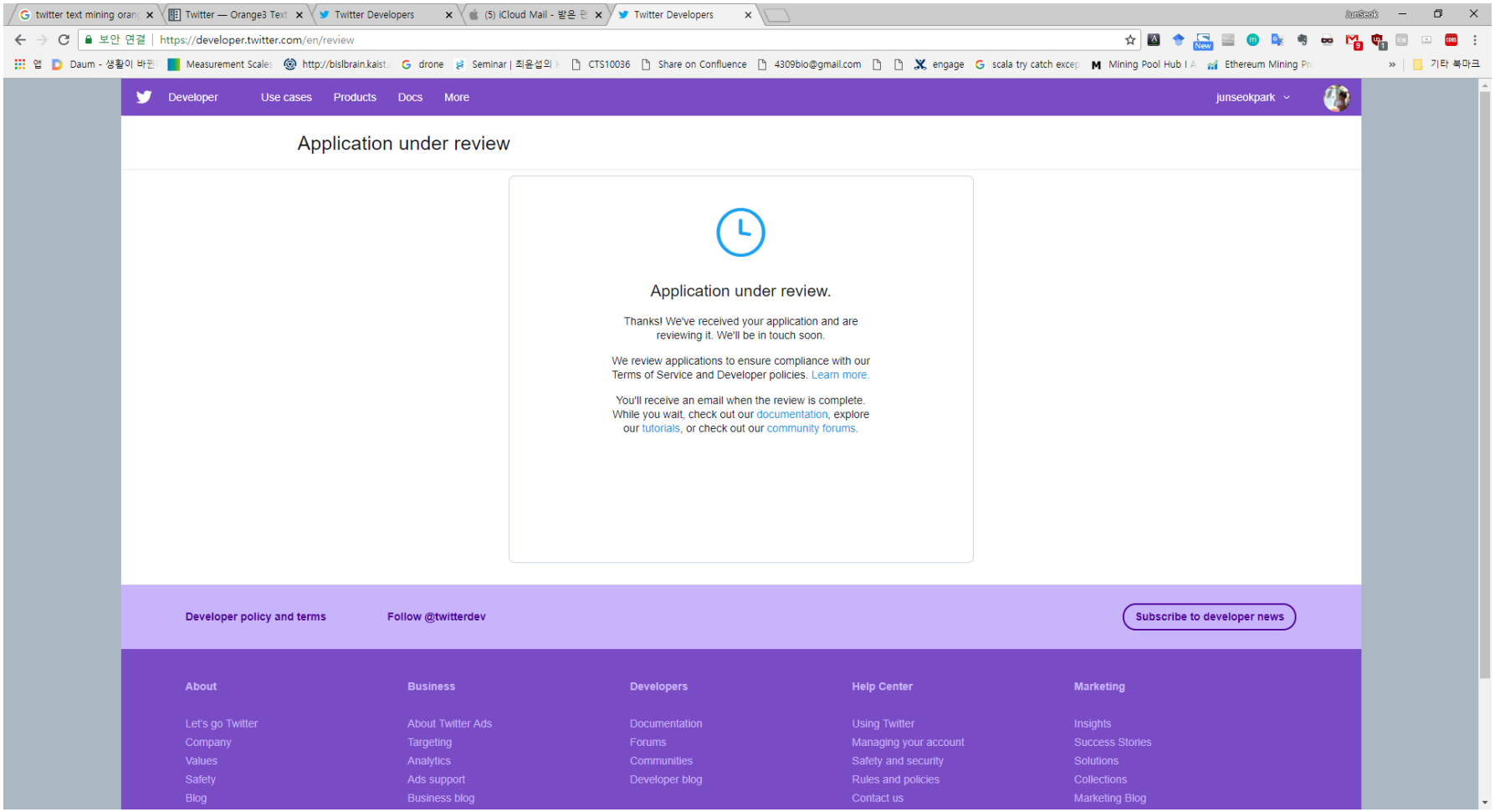
# Twitter API


Developer
Use cases
Products
Docs
More
junseokpark
One more step...
To complete your application, please check your inbox to confirm your email address.
Didn't receive your verification email? Resend it now.
Developer policy and terms
Follow @twitterdev
Subscribe to developer news
About
Let's go Twitter
Company
Values
Safety
Blog
Brand Resources
Careers
Investors
Business
About Twitter Ads
Targeting
Analytics
Ads support
Business blog
Advertise
Developers
Documentation
Forums
Communities
Developer blog
Help Center
Using Twitter
Managing your account
Safety and security
Rules and policies
Contact us
Marketing
Insights
Success Stories
Solutions
Collections
Marketing Blog
Flight School
© 2018 Twitter, Inc.
Cookies
Privacy
Terms and Conditions

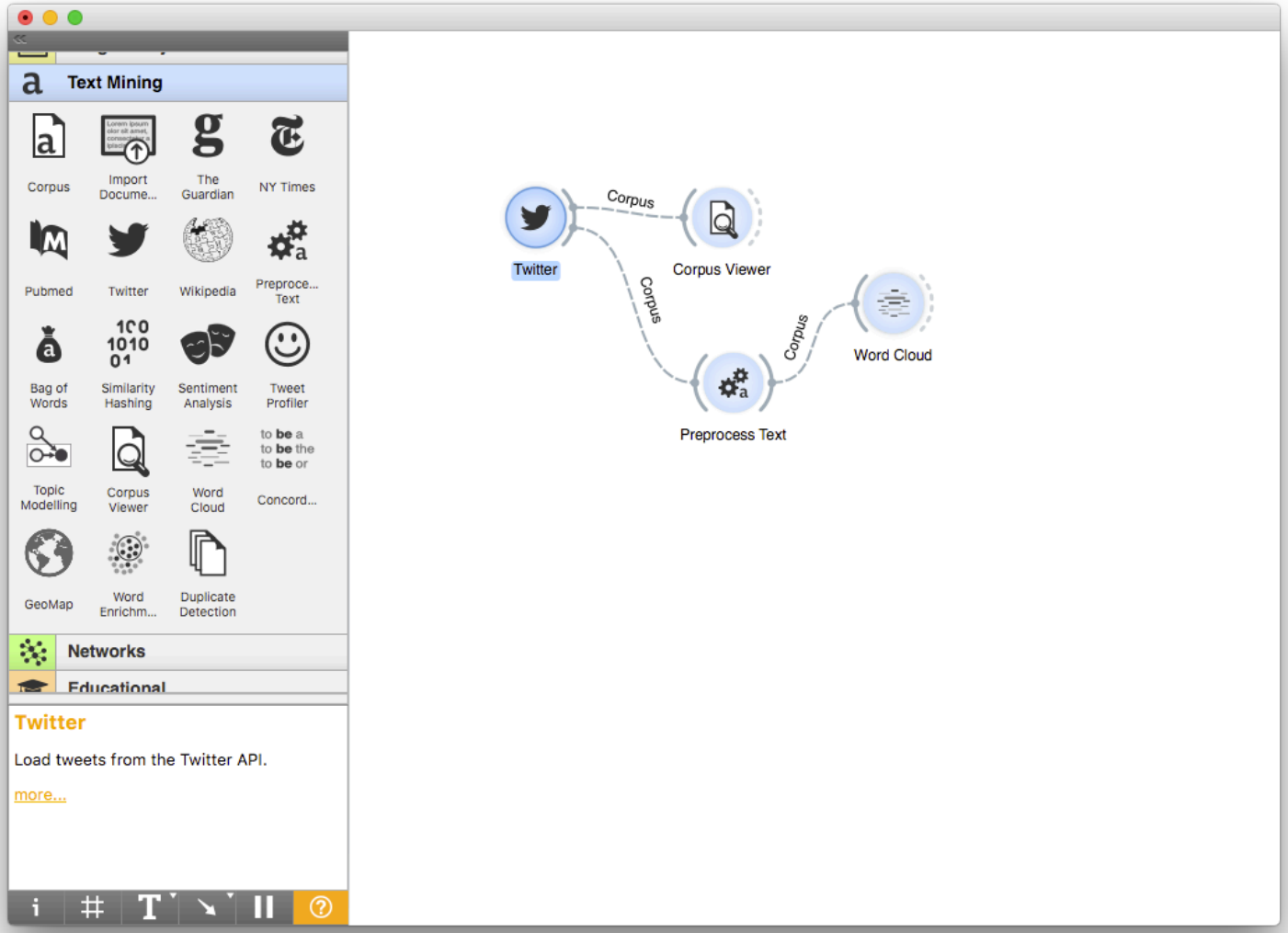
# Twitter API


JUN SEOK
Verify your Twitter Developer Account
1분 전 오후 7:08
보낸 사람 Twitter Developer Accounts
더 보기
Developer
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

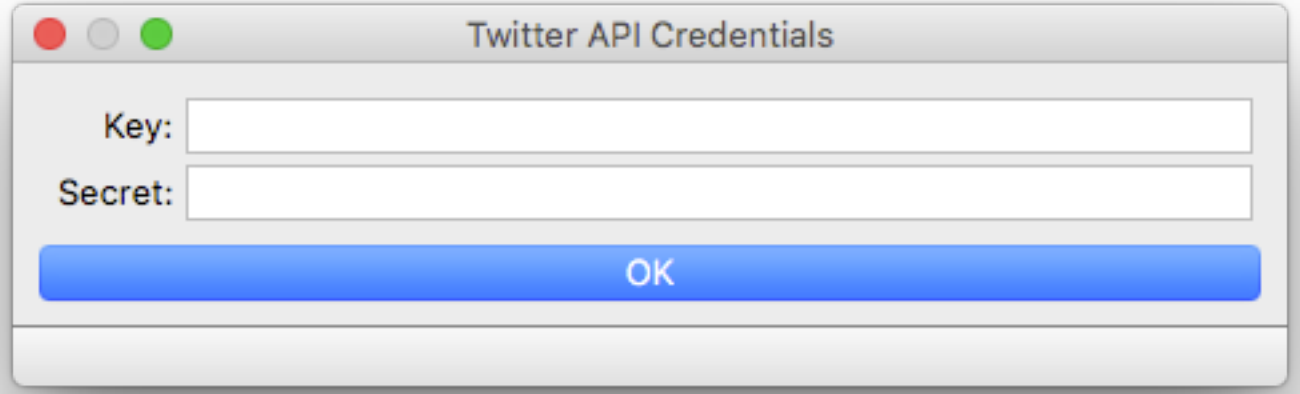

Developer
Use cases
Products
Docs
More
junseokpark
Application under review
Application under review.
Thanks! We've received your application and are reviewing it. We'll be in touch soon.
We review applications to ensure compliance with our Terms of Service and Developer policies. Learn more.
You'll receive an email when the review is complete. While you wait, check out our documentation, explore our tutorials, or check out our community forums.
Developer policy and terms
Follow @twitterdev
Subscribe to developer news
About
Let's go Twitter
Company
Values
Safety
Blog
Business
About Twitter Ads
Targeting
Analytics
Ads support
Business blog
Developers
Documentation
Forums
Communities
Developer blog
Help Center
Using Twitter
Managing your account
Safety and security
Rules and policies
Contact us
Marketing
Insights
Success Stories
Solutions
Collections
Marketing Blog

# Build Schema



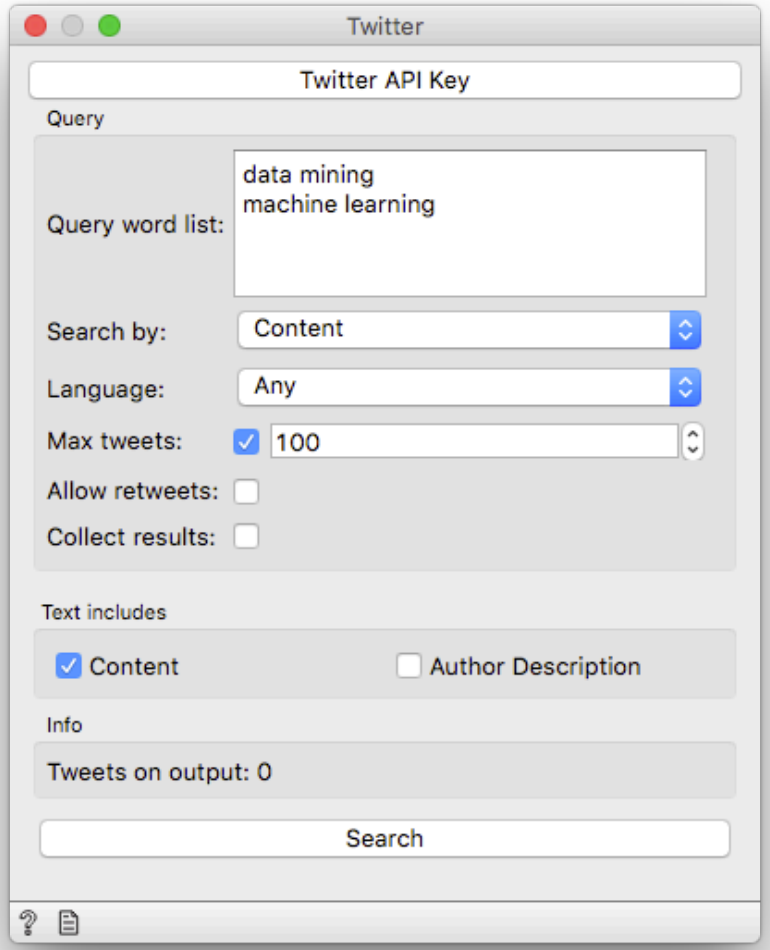
# Twitter Connection

- **Connect to twitter**
  - insert Twitter key and secret



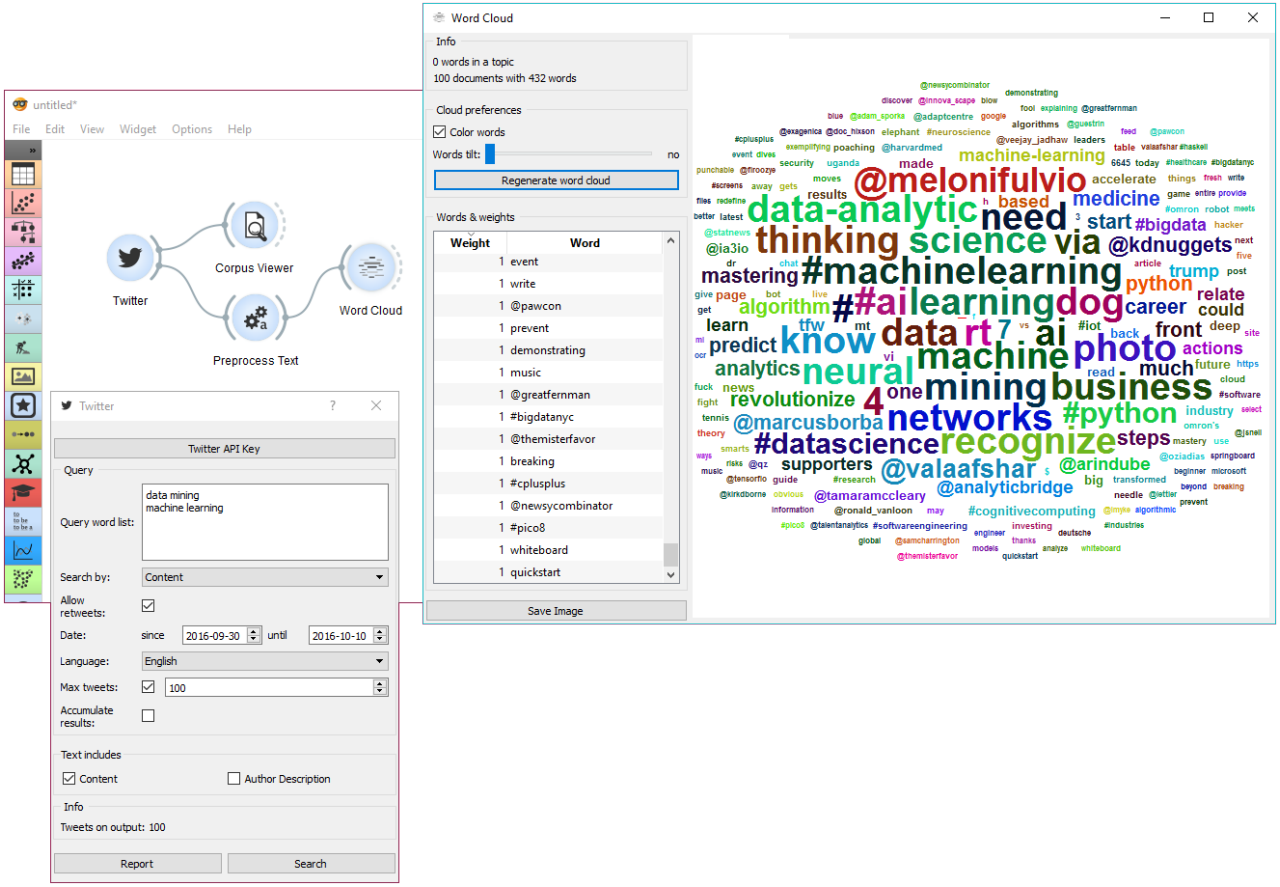
# Twitter Parameters

- ***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.

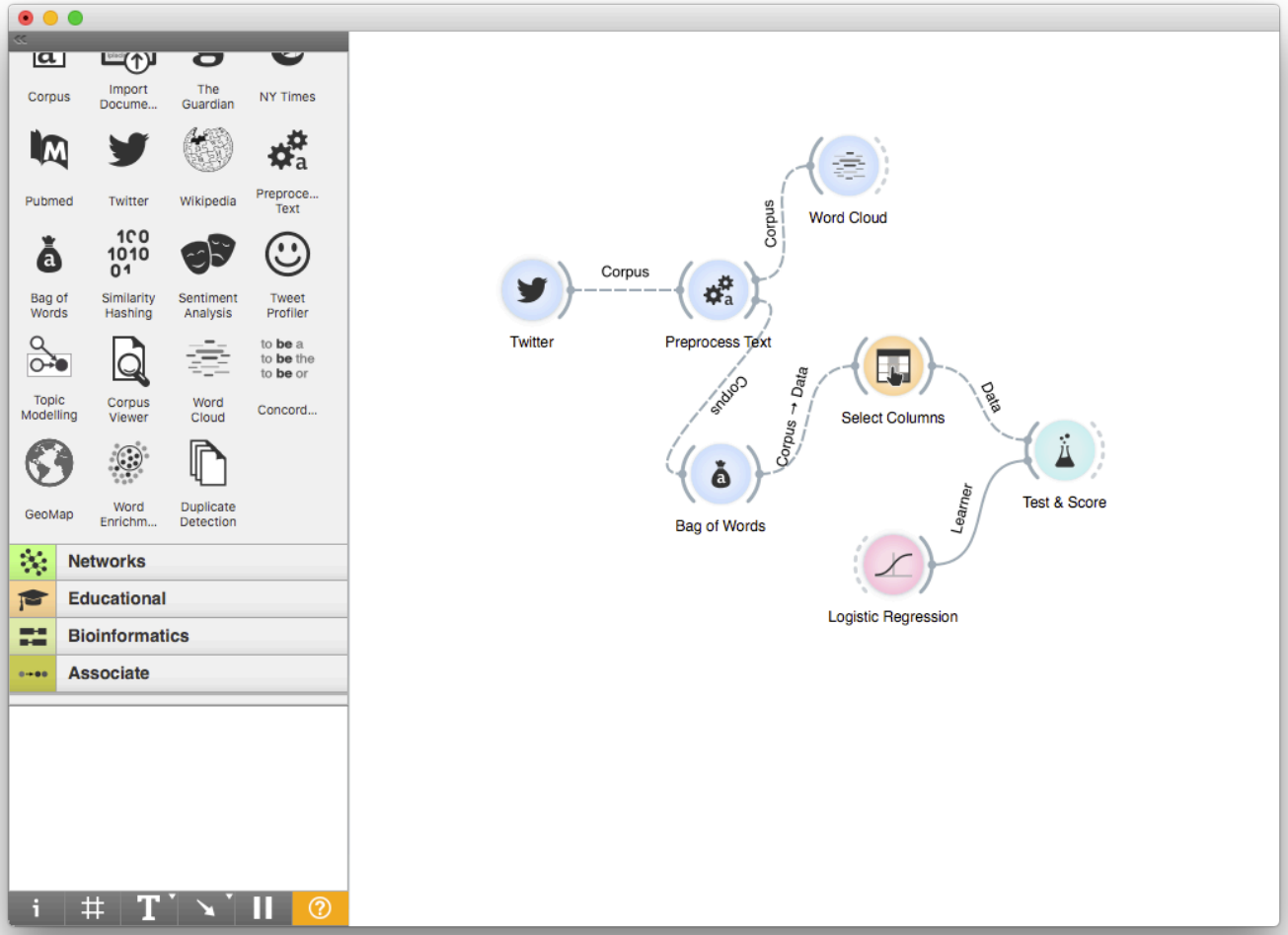


# Self study

- **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.

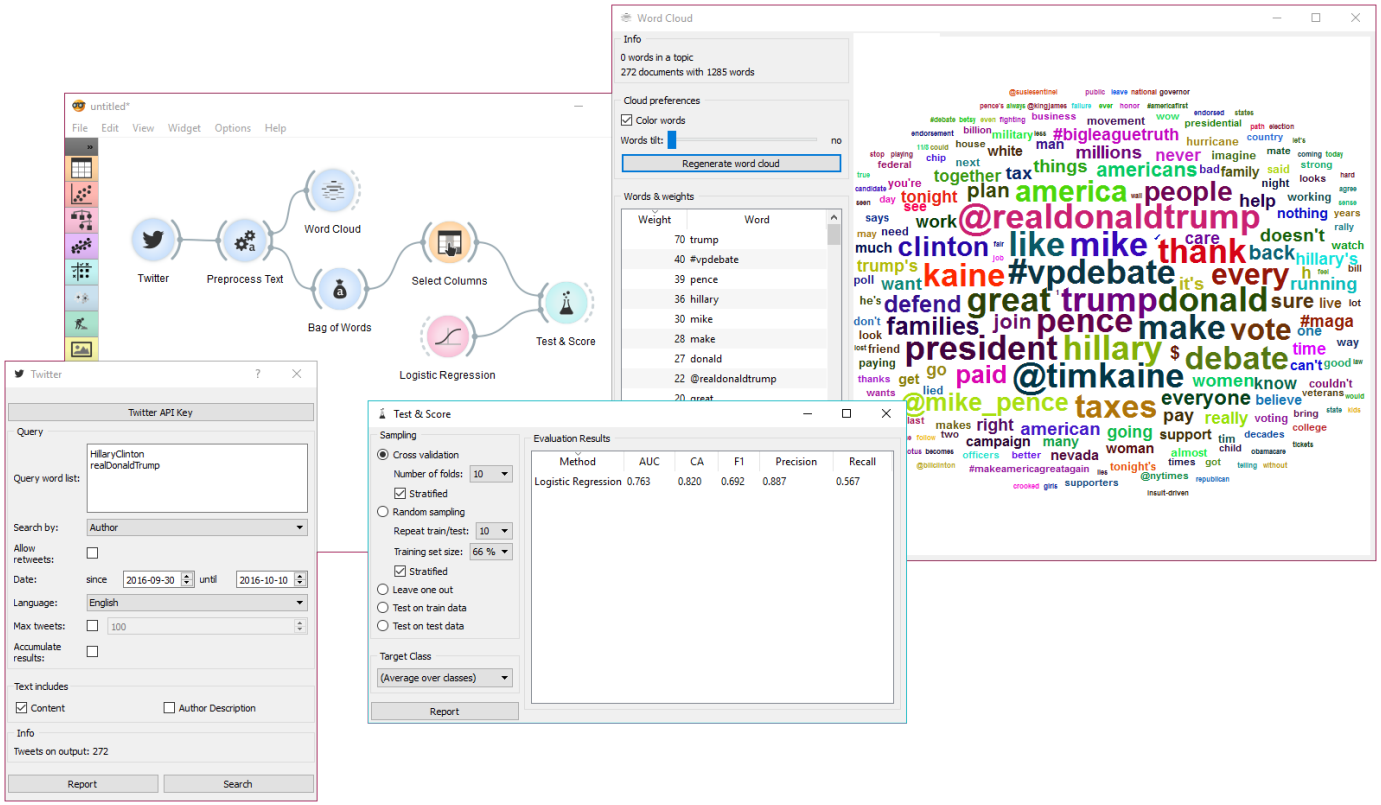
# Complex Example (1/2)

- **Build Schema**



# Complex Example

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

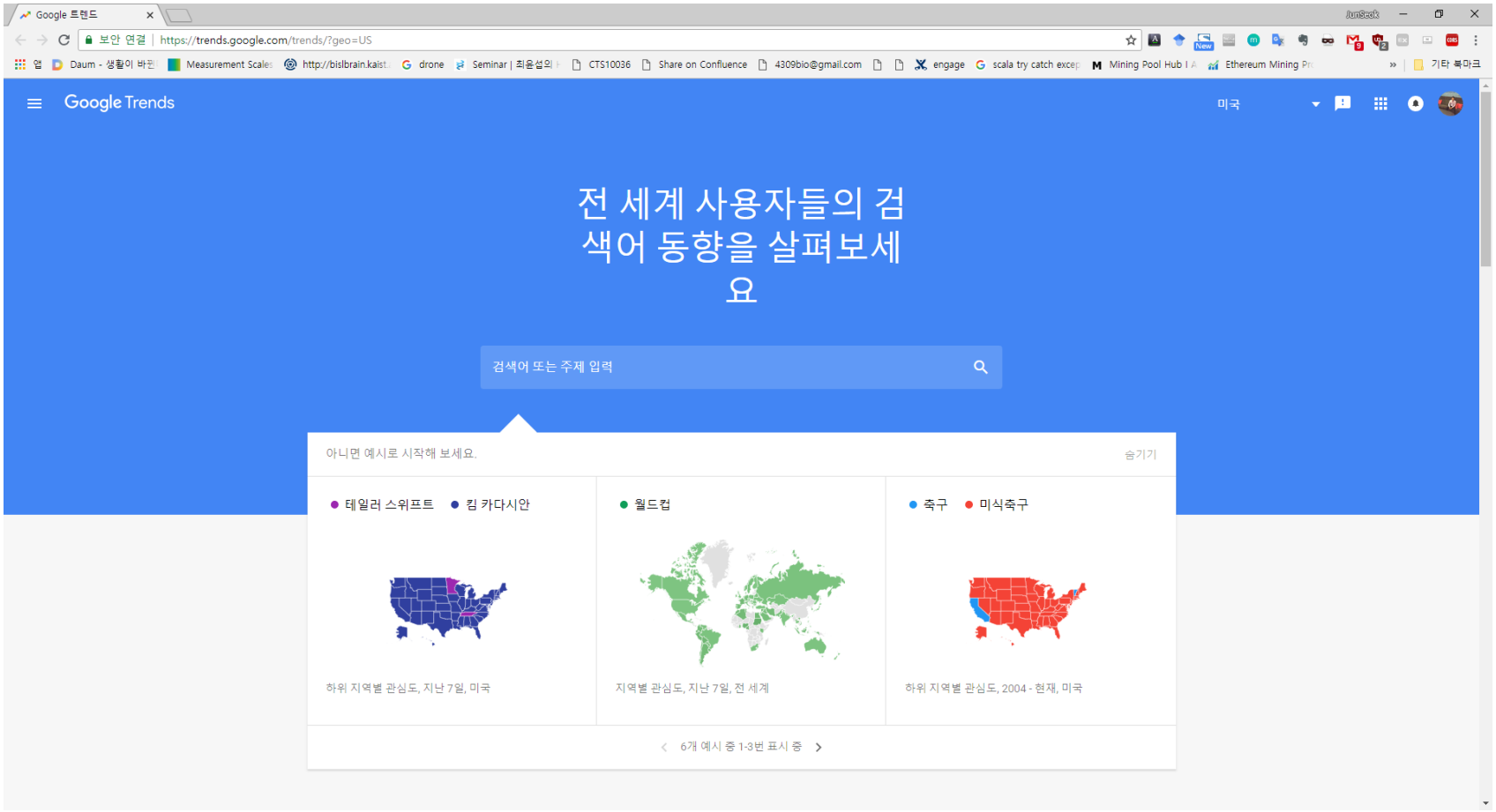



- Check the result by your self

# Web-based datamining

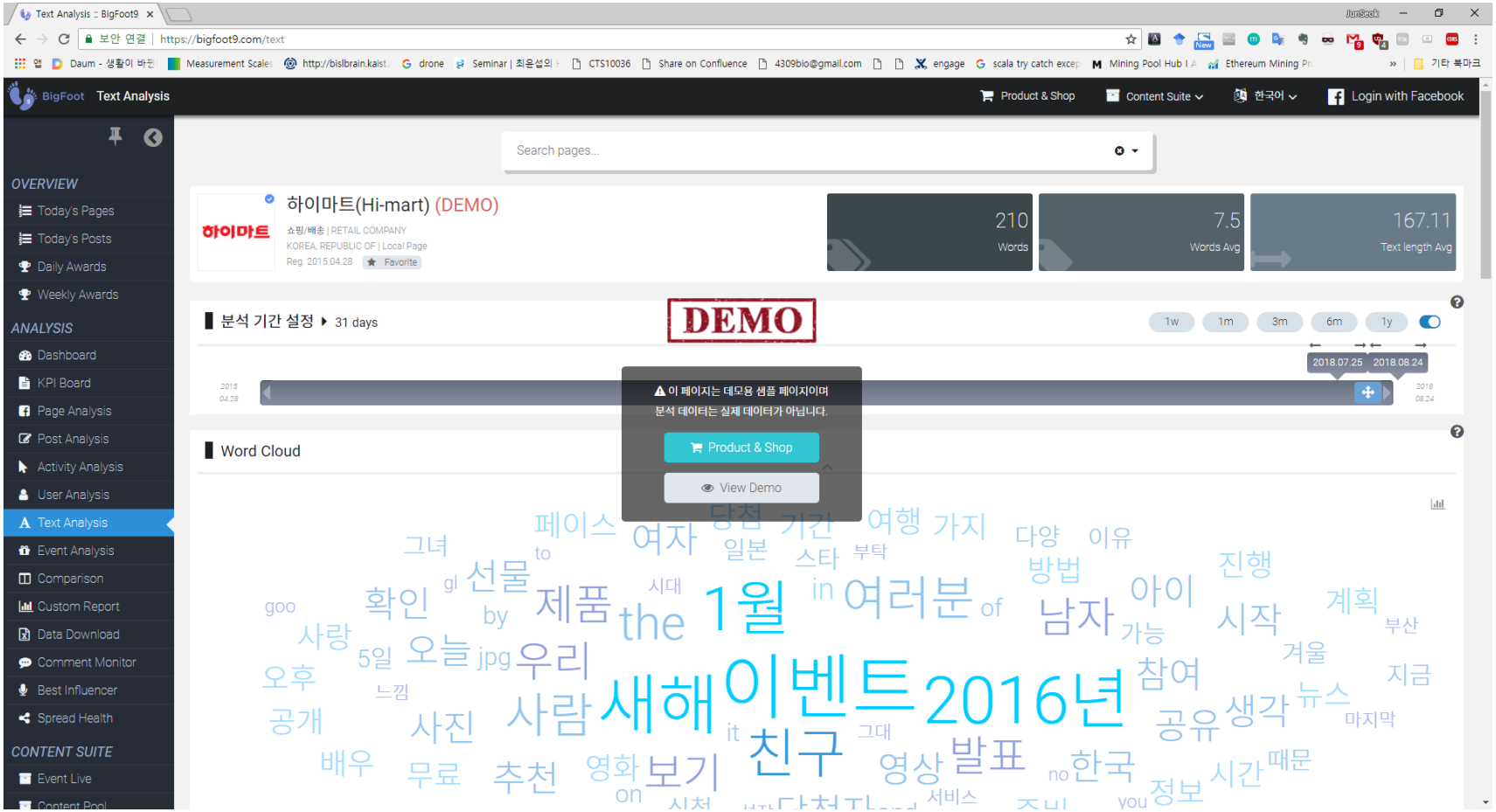
# Google trend

- **Trend analysis**



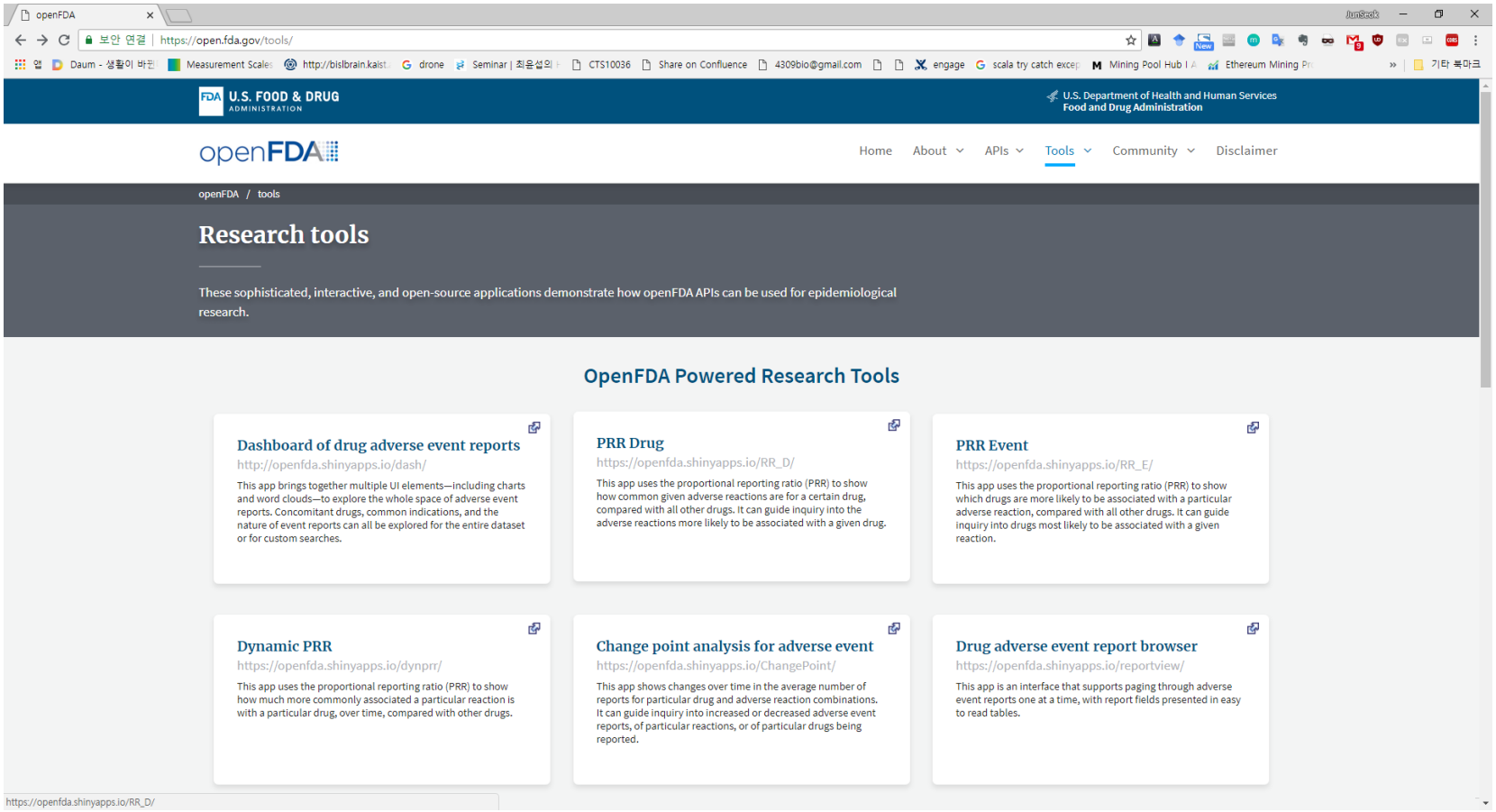
# Facebook marketing

- https://bigfoot9.com



# Drug Research Tools

- **https://open.fda.gov/tools/**



# 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