Naive Bayes text classification

Sumin Han (hsm6911@kaist.ac.kr)

Contents

Introduction

Artificial Intelligence

Rule-based AI

→ Long
Yellow
Little bent → Banana

Machine Learning

= Banana → Long
Yellow
Little bent → Banana

Artificial Intelligence

Rule-based AI

Long
Yellow
Little bent
or
White
Flat
Round

Manually add rule

Banana

Machine Learning

= Banana

Training

Long
Yellow
Little bent
or
White
Flat
Round

Create
own rule

Banana

Example: Starcraft AI

Rule-based AI

- Build Supply Depot
- Build Barrak
- Produce marines
- Build Factory
- …

Machine Learning

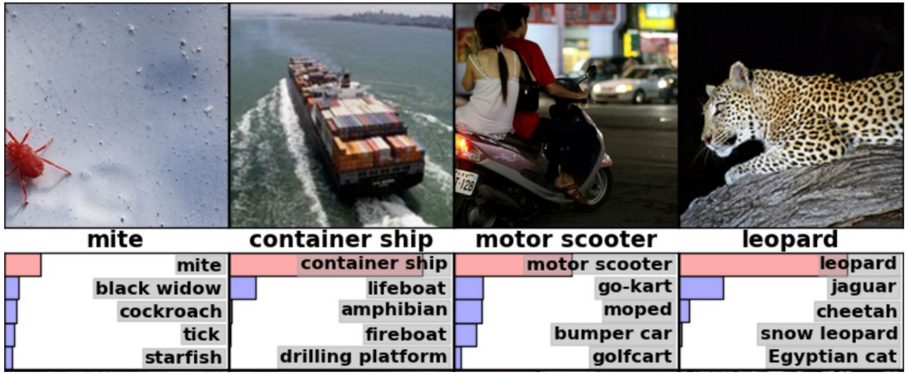
- Train AI using millions of replays
- Make its own build order
- Make its own decision in a certain situation

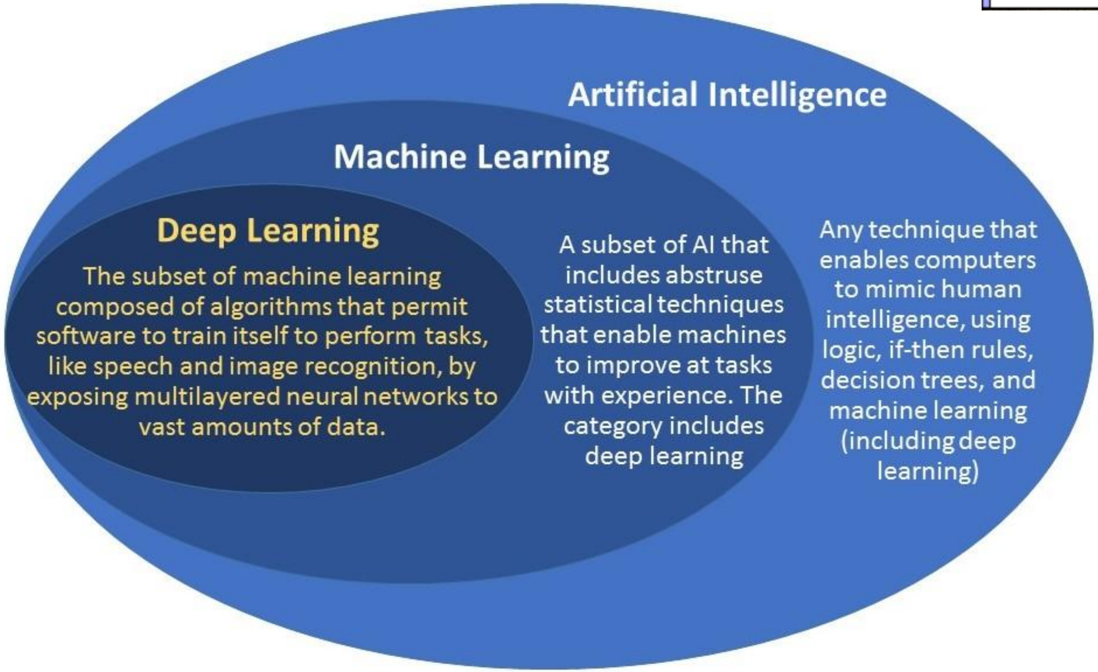
AI for RTS game is still here!

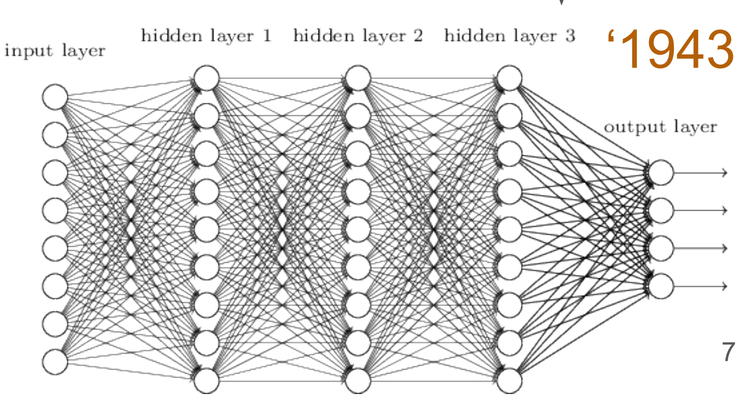
AlphaGo

Humans Are Still Better Than AI at StarCraft—for Now (October, 31th)
https://www.technologyreview.com/s/609242/humans-are-still-better-than-ai-at-starcraftfor-now/

Deep Learning (1986~)

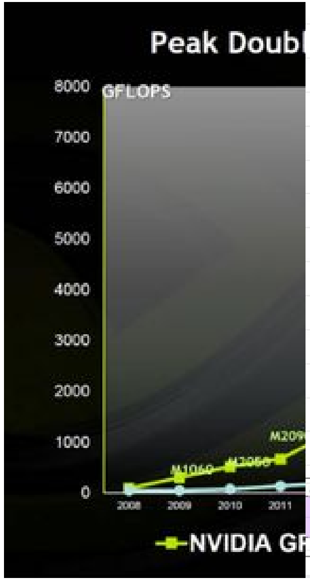






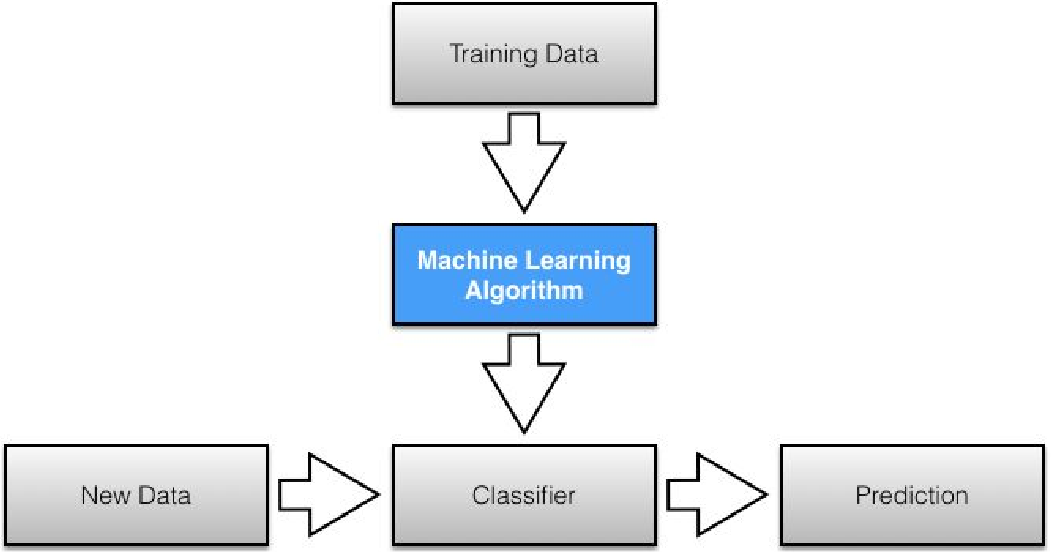
‘1943

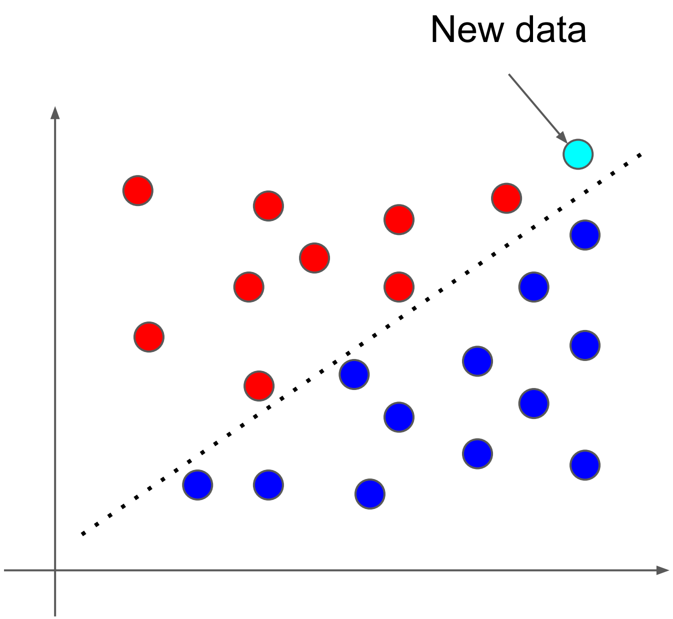
Computing power made Deep Learning feasible!



Naive Bayes Classification

Can a machine make a linear model to categorize new data into trained model?





Bayes' theorem

Breast cancer detection kit

Here’s a test kit for breast caner.

4 out of 1000 women have breast cancer. (prior probability: 0.004)

800 out of **1000 women with breast canccer** will get a positive result. (sensitivity: 0.8)

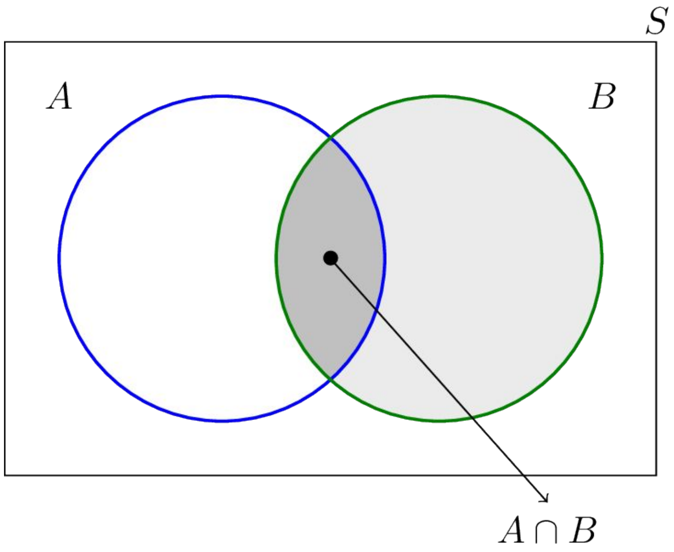
100 out of **1000 women without breast cancer** will get a positive result. (false alarm: 0.1)

*If my kit shows positive, what is the **probability** that I actual got cancer?*

Conditional probability

Probability that event A will occur when event X occured.

$$P(A|X) = \frac{P(A \cap X)}{P(X)}$$



$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Example:

A: event that dice showed n > 3

X: event that dice showed even number

$$P(A|X) = \frac{P(\{4,6\})}{P(\{2,4,6\})} = \frac{2}{3}$$

Bayes' theorem

Probability that kit shows positive result **when I have cancer**

Probability of having breast cancer

$$P(A|X) = \frac{P(X|A)P(A)}{P(X)}$$

Real probability that I have cancer when I have positive kit result.

Probability that kit shows positive

$P(X|A)$: sensitivity, $P(A)$: prior probability

$P(X|A) = 0.8$

: Probability that kit shows positive result when I have cancer

$P(A) = 0.004$

: Probability of having breast cancer

P(X): Probability that kit shows positive

$$P(X) = P(X \cap A) + P(X \cap \neg A)$$

$$P(X) = P(X \cap A) + P(X \cap \neg A)$$

$$= \underbrace{P(X|A)}_{0.8}\underbrace{P(A)}_{0.004} + \boxed{\underbrace{P(X|\neg A)}_{0.1}}\underbrace{P(\neg A)}_{0.996} \quad = 0.1028$$

P(X|~A) = 0.1

: Probability that kit shows positive though I don't have cancer.

P(~A) = 1 - P(A) = 0.996 $\boxed{\text{P(A|X) = P(X|A)*P(A)/P(X) = } \mathbf{3.11\ \%}}$

Likelihood

Candy Machine

	Red	Blue	Green
Candy Machine A	2	2	1
Candy Machine B	1	1	1

My kid brought (red, blue, green) = (4, 5, 1) candies for each kind.

Machine B itself looks more fancy and attractive,
so it has higher probability: P(B) = 0.6, P(A) = 0.4

Which candy machine did my kid used?

Definition

P(X) = Probability that my kid bring (5, 6, 1) candy combination.

P(A) = Probability that my kid used machine A
P(B) = Probability that my kid used machine B

P(A | X) = Probability that my kid used machine A when he brought (4, 5, 1)
P(B | X) = Probability that my kid used machine B when he brought (4, 5, 1)

$$P(A|X) = \frac{P(X|A)P(A)}{P(X)} \quad \textbf{vs.} \quad P(B|X) = \frac{P(X|B)P(B)}{P(X)}$$

We know...

$$P(A|X) : P(B|X) = \frac{P(X|A)P(A)}{P(X)} : \frac{P(X|B)P(B)}{P(X)}$$

You don't need to calculate this

$$= P(X|A)P(A) : P(X|B)P(B)$$

Likelihood

	Red	Blue	Green
Candy Machine A	**2**	**2**	**1**
Candy Machine B	**1**	**1**	**1**

Probability that I pick up **Red** candy from machine **A**: 2/5

Probability that I pick up **Blue** candy from machine **A**: 2/5

Probability that I pick up **Green** candy from machine **A**: 1/5

Likelihood (cont.)

	Red	Blue	Green
Candy Machine A	**2**	**2**	**1**
Candy Machine B	**1**	**1**	**1**

Probability that I pick up **4 Red** candies from machine **A**: (2/5)*(2/5)*(2/5)*(2/5)

Probability that I pick up **5 Blue** candy from machine **A**: (2/5)*(2/5)*(2/5)*(2/5)*(2/5)

Probability that I pick up **1 Green** candy from machine **A**: (1/5)

P(X | A) = (⅖)^4 * (⅖)^5 * (⅕) = 5.24288e-5

Likelihood (cont.)

	Red	Blue	Green
Candy Machine A	**2**	**2**	**1**
Candy Machine B	**1**	**1**	**1**

Probability that I pick up **4 Red** candies from machine **B**: (1/3)*(1/3)*(1/3)*(1/3)

Probability that I pick up **5 Blue** candy from machine **B**: (1/3)*(1/3)*(1/3)*(1/3)*(1/3)

Probability that I pick up **1 Green** candy from machine **B**: (1/3)

$P(X \mid B) = (\frac{1}{3})^4 * (\frac{1}{3})^5 * (\frac{1}{3}) = 1.69351e{-}5$

Compare!

$$P(A|X) : P(B|X) = \frac{P(X|A)P(A)}{P(X)} : \frac{P(X|B)P(B)}{P(X)}$$

You don't need to calculate this

$$= \underbrace{P(X|A)}_{5.24288e-5}\underbrace{P(A)}_{0.4} : \underbrace{P(X|B)}_{1.69351e-5}\underbrace{P(B)}_{0.6}$$

= 0.67361988 : 0.32638012

~= 2: 1

Text Categorization

Finally! we can make text categorization.

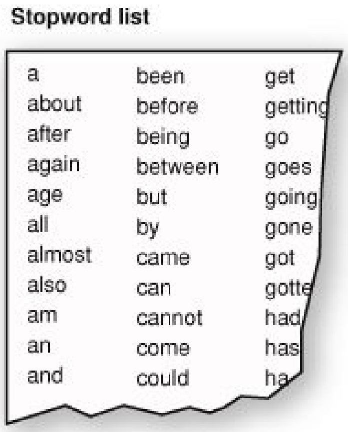
Training text (SpongeBob)

Today's the big day, Gary!
Look at me, I'm... ...naked! Gotta be in top physical condition for today, Gary.
I'm ready! I'm ready, I'm ready, I'm ready, I'm ready, I'm ready, I'm ready, I'm ready, I'm ready, I'm ready, I'm ready!
There it is. The finest eating establishment ever established for eating. The Krusty Krab, home of the Krabby Patty. With a 'Help Wanted' sign in the window! For years I've been dreaming of this moment! I'm gonna go in there, march straight to the manager, look 'im straight in the eye, lay it on the line and... I can't do this! Uh, Patrick!

Training text (Mr. Krabs)

Well lad, it looks like you don't even have your sea legs.
Well lad, well give you a test, and if you pass, you'll be on the Krusty Krew! Go out and fetch me... a, uh, hydrodynamic spatula... with, um, port-and-starboard-attachments, and, uh... turbo drive! And don't come back till you get one!
Carry on! We'll never see that lubber again.
That sounded like hatch doors! Do you smell it? That smell. A kind of smelly smell. A smelly smell that smells smelly. Anchovies.

Make Bag of words



Python dictionary[word]: count

SpongeBob: {'today': 2, "'s": 1, 'big': 1, 'day': 1, 'gary': 2, 'look': 2, "'m": 13, 'naked': 1, 'got': 1, 'ta': 1, 'top': 1, 'physical': 1, 'condition': 1, 'ready': 11, 'finest': 1, 'eating': 2, 'establishment': 1, 'ever': 1, 'established': 1, 'krusty': 1, 'krab': 1, 'home': 1, … }

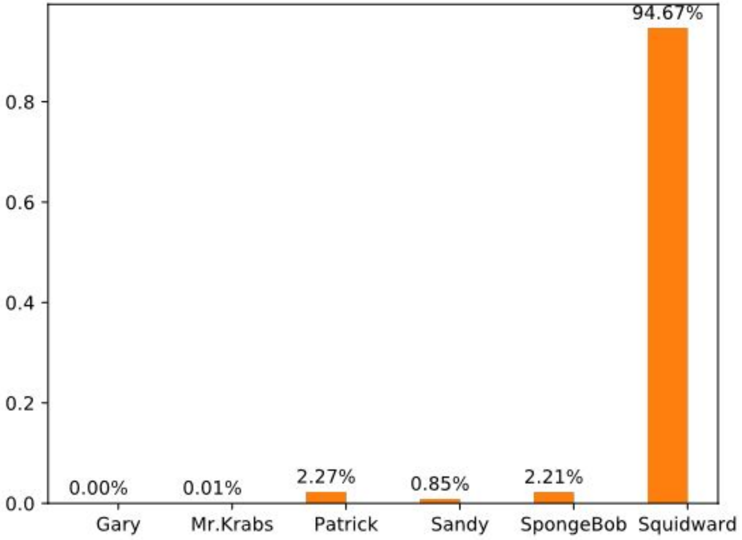
Mr. Krabs: {'well': 3, 'lad': 2, 'looks': 1, 'like': 2, "n't": 2, 'even': 1, 'sea': 1, 'legs': 1, 'give': 1, 'test': 1, 'pass': 1, "'ll": 2, 'krusty': 1, 'krew': 1, 'go': 1, 'fetch': 1, 'uh': 2, 'hydrodynamic': 1, 'spatula': 1, 'um': 1, 'port': 1, 'starboard': 1, 'attachments': 1, 'turbo': 1, … }

```python
import nltk
import re
#nltk.download() # if you are first time

special_chars_remover = re.compile("[^\w'|_]")
stpwd = nltk.corpus.stopwords.words('english')

def create_BOW(sentence):
  bow = {}
  sentence = remove_special_characters(sentence)
  sentence = sentence.lower()
  tokens = nltk.word_tokenize(sentence)

  for word in tokens:
    if len(word) < 1 or word in stpwd: continue
    word = word.lower()
    bow.setdefault(word, 0)
    bow[word] += 1
  return bow

def remove_special_characters(sentence):
  return special_chars_remover.sub(' ', sentence)

sent = input(">> ")
print(create_BOW(sent))
```

Run the code!

https://github.com/SuminHan/NLP-SpongeBob/tree/master/NaiveBayes

testing_sentence = **"I hate this job. I want to go home and play clarinet"**




```
def calculate_doc_prob(training_sentence, testing_sentence, alpha):
  logprob = 0
  training_model = create_BOW(training_sentence)
  testing_model = create_BOW(testing_sentence)
  '''
  Calculating the probability that training_model may produce
  testing_model.
  We use math.log, so note the use.
  Ex) 3 * 5 = 15
      log(3) + log(5) = log(15)
      5 / 2 = 2.5
      log(5) - log(2) = log(2.5)
  '''
  tot = 0
  for word in training_model:
      tot += training_model[word]
  for word in testing_model:
      if word in training_model:
          logprob += math.log(training_model[word])
          logprob -= math.log(tot)
      else:
          logprob += math.log(alpha)
          logprob -= math.log(tot)
  # log_prob = math.log(prob)
  return logprob
```

prevent Probability becomes 0

Tips & Reference

Log-likelihood keyness (antconc)

Word List Results 2 c

Word Types: 3800		Word Tokens: 29947
Rank	Freq a	Word
29	167	formula

Word List Results 1 d

Word Types: 14202		Word Tokens: 364385
Rank	Freq b	Word
178	294	formula

- Check

AntConc 3.4.4w (Windows) 2014

File Global Settings Tool Preferences Help

Corpus Files: Plankton.txt

Concordance | Concordance Plot | File View | Cluster

Types Before Cut: 3800 Types After Cut:

Rank	Freq	Keyness	Keyword
1	132	317.781	karen
2	167	303.777	formula

```
>>> def keyness(a, b, c, d):
...     a = float(a)
...     b = float(b)
...     c = float(c)
...     d = float(d)
...     E1 = c*(a+b) / (c+d)
...     E2 = d*(a+b) / (c+d)
...     ka = (a*math.log(a/E1))
...     kb = (b*math.log(b/E2))
...     return 2*(ka+kb)
...
>>> keyness(167, 294, 29947, 364385)
303.7765602866808
```

- Ref: http://ucrel.lancs.ac.uk/llwizard.html

Use Python 3.6

Try to Install **PyCharm** (https://www.jetbrains.com/pycharm/)

C:\> pip install numpy matplotlib nltk
(if you need any library to import, just execute on cmd prompt, **windows** + **R**)

C:\> python
>>> import nltk
>>> nltk.download()

Download NaiveBayes.zip to checkout my example:

⇒ https://github.com/SuminHan/NLP-SpongeBob/blob/master/NaiveBayes.zip

Other raw data is on https://github.com/SuminHan/NLP-SpongeBob, take a look.

Good luck with your project!

Reference

[1] Elice: https://academy.elice.io/courses/214/lectures

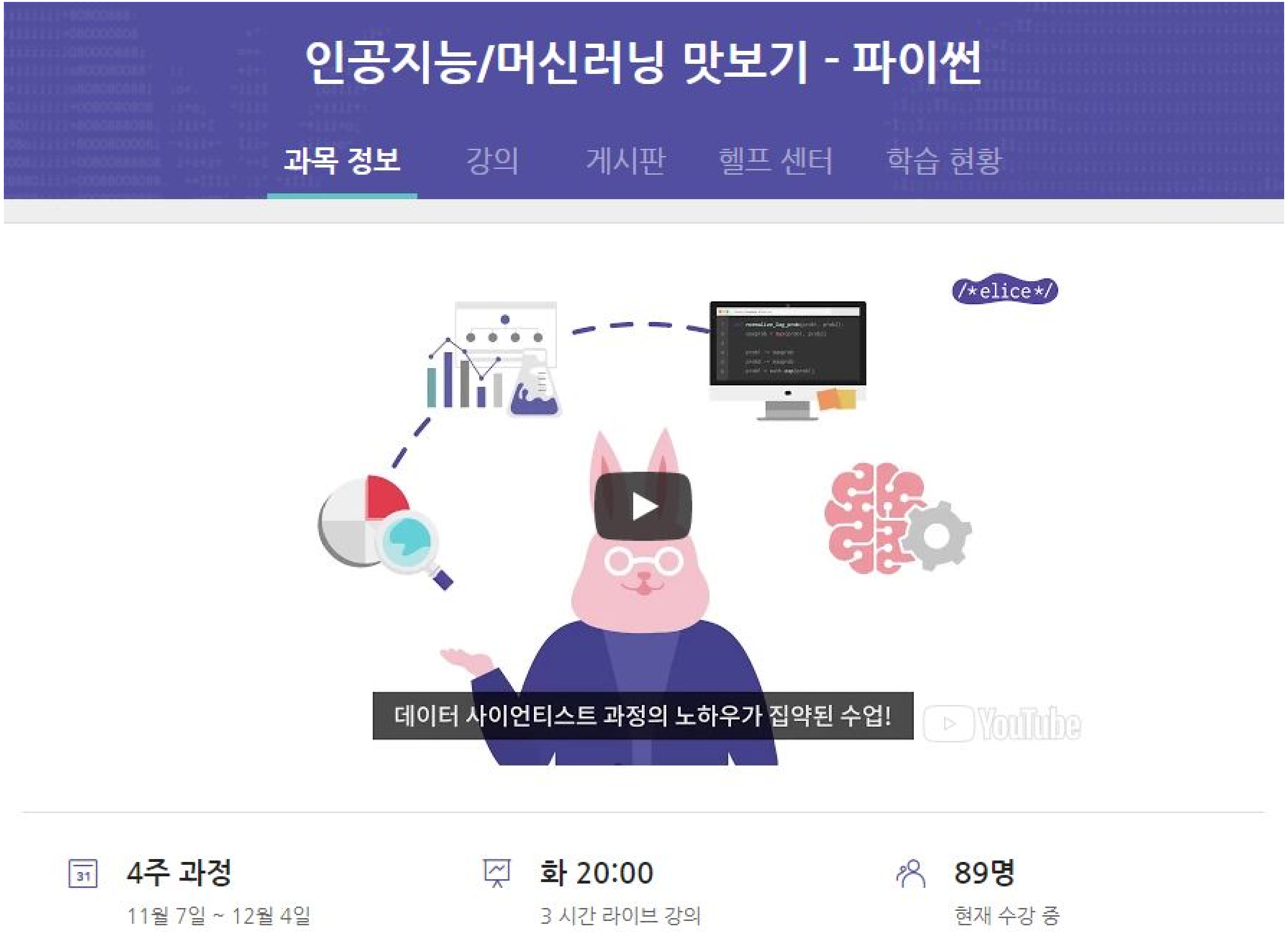
[2] Naive Bayes: http://sebastianraschka.com/Articles/2014_naive_bayes_1.html

[3] Intro to TensorFlow (Korean): https://github.com/golbin/TensorFlow-Tutorials

[4] SpongeBob Project: https://github.com/SuminHan/NLP-SpongeBob

Elice Lecture

https://elice.io/



Mail me if you have question

hsm6911@kaist.ac.kr