ML, NLP basics, and others

Akito Sakurai

Natural Language Processing

- Text classification is an example.
- The field of study that focuses on the interactions between human language and computers is called Natural Language Processing, or NLP for short. It sits at the intersection of computer science, artificial intelligence, and computational linguistics (Wikipedia).
- NLP is a way for computers to analyze, understand, and derive meaning from human language in a smart and useful way (source unknown).

NLP Applications

- Spell and Grammar Checking
  - Checking spelling and grammar
  - Suggesting alternatives for the errors
- Word Prediction
  - Predicting the next word that is highly probable to be typed by the user
- Information Retrieval
  - Finding relevant information to the user's query
- Information Extraction
  - Extracting important concepts from texts and assigning them to slot in a certain template
  - Includes named-entity recognition
- Text Categorization
  - Assigning one (or more) pre-defined category to a text
- Sentiment Analysis
  - Identifying sentiments and opinions stated in a text

NLP Applications

- Summarization
  - Generating a summary from one or more documents, sometimes based on a given query
- Speech recognition
  - Recognizing a spoken language and transforming it into a text
- Speech synthesis
  - Producing a spoken language from a text
- Question answering
  - Answering questions with a short answer
- Machine Translation
  - Translating a text from one language to another
- Spoken dialog systems
  - Running a dialog between the user and the system

Representations

- Suppose that we will consider just texts.
- A text is a series of characters, which we do not think good for processing, because characters convey almost nothing of semantics and syntax of the text tobeornottobethatisthequestion
  - (not necessarily so if we process many series of characters, which we now know)
- A word must be a unit of representation.
  - What is a word? Could we define it?
- Suppose that intuitive understanding of "word" is correct.
  - There are many preprocessing techniques for texts including deletion of stop words, word stemming, spelling corrections, etc.
- Suppose that we assign distinct integers to distinct words.

Representations

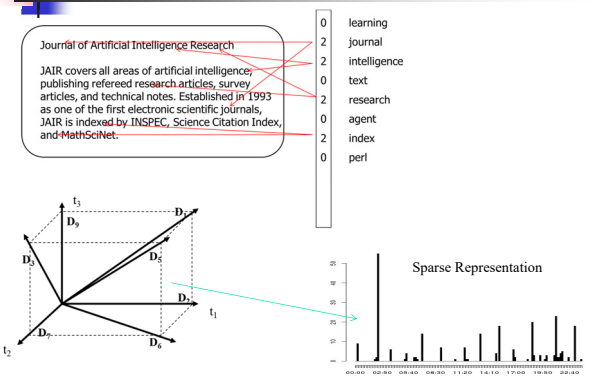
- Let us assign distinct integers to distinct words.
  - By doing so, we transform a series of words into a series of integers, e.g.,

    this film was just brilliant casting location scenery
    1 14 22 16 43 530 973 1622 1385 65
- Let us consider a series of words (integers) as a multi-set of words (integers).
  - Information on word order is lost. But we may try.
  - By doing so, we do not bother about indeterminate length of series. If we fix the vocabulary of words, the set is represented by a vector of the length.

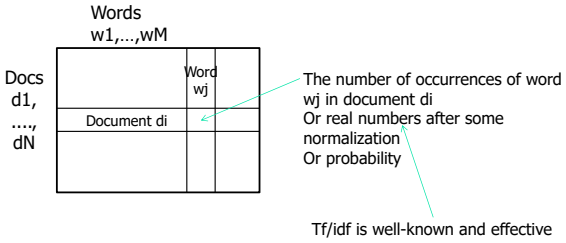
    1 3 0 1 2 0 0 1 9 0 ...
- How about just a set (do not consider word frequencies).

  1 1 0 1 0 0 0 1 1 0 ...

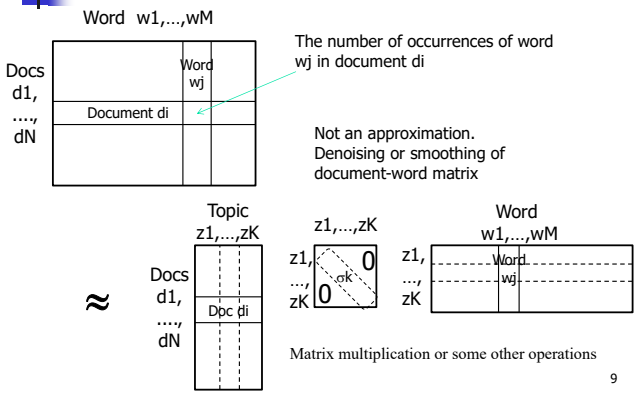
Vector space model



Document-word matrix



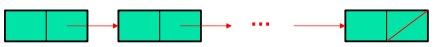
Topic model



A note for language features

list

- What it means vary in computer languages (as is usual).
- A sequence of objects that
  - Can be accessed by index
  - Accept commands like append/merge/is_empty
  - Can be depicted as (linked list):



array

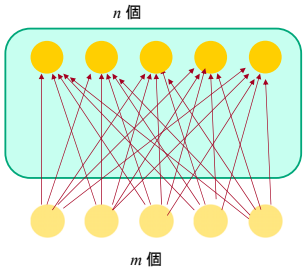
- A array of memory cells that occupy consecutive area, which
  - Can be accessed by index with high speed
  - Is fixed in size and location.
  - Cannot accept commands like append/merge
  - Can be depicted as:

Formal/actual parameters

- When you call a function *f* to get its value for some value, you have to specify the value; which is called actual parameter e.g. *f*(3), sin(π) or *f*(*x*) for a variable *x*.
  - Generally *f*(*x*) is different from *f*(*y*)
- When you define a function f, you have to specify the relation of formal parameters to the output value such as: $f(x) \stackrel{\text{def}}{=} x\text{^}2$
  - Generally $f(x) \stackrel{\text{def}}{=} x\text{^}2$ is equivalent to $f(y) \stackrel{\text{def}}{=} y\text{^}2$

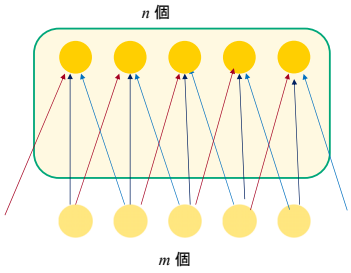
Dense (FC) vs. filter (conv.)

Fully (densely) connected layer



Degree of freedom
= the number of parameters
= (*m*+1)**n*

Convolutional (filter) layer

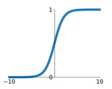


Degree of freedom
= the number of parameters
= 3+1 (for a threshold)

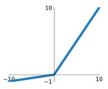
Activation functions

Activation functions

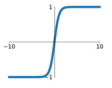
Sigmoid
$\sigma(x) = \frac{1}{1+e^{-x}}$



Leaky ReLU
$\max(0.1x, x)$

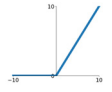


tanh
$\tanh(x)$

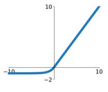


Maxout
$\max(w_1^T x + b_1, w_2^T x + b_2)$

ReLU
$\max(0, x)$



ELU
$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$

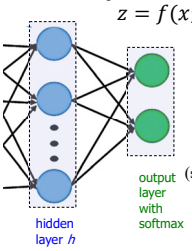


Softmax and crossentropy

Softmax and crossentropy loss

Output without softmax

$z = f(x; W)$



Output with softmax

$$\frac{e^{z_k}}{\sum_j e^{z_j}}$$

(supposedly probability distribution)

target $p(x)$: 0 0 0 1 0

actual $q(x)$: .1 .2 .1 .6 0

The distance between two distributions is often measured by Kullback Leibler divergence $D_{KL}(p \parallel q)$. When p is fixed and q that minimizes $D_{KL}(p \parallel q)$ is to be found, the problem is equivalent to the minimization of cross-entropy $H(p,q)$. This is why cross-entropy is used as a loss function.

Note that $D_{KL}(p||q) = -H(p) + H(p,q)$ where

$$D_{KL}(p||q) = \sum_x p(x) \log p(x)/q(x)$$

$$H(p,q) = -\sum_x p(x) \log q(x)$$

$$H(p) = -\sum_x p(x) \log p(x)$$

R or python

R or python

- "Both" is recommended.
  - You may choose one, of course.
- R is good for conventional machine learning.
- Python is good for deep learning.

- R is a language for statistical applications.
  - Good sense for usability
- Python is a language for computer scientist.
  - But is used commonly for DNN frameworks.