



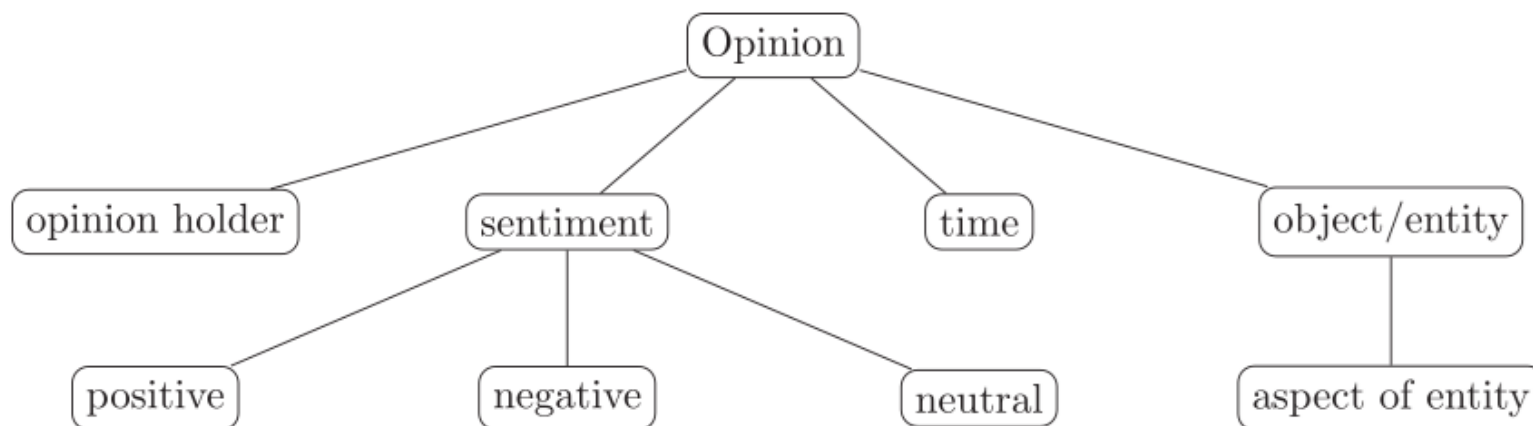
ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

# Lesson 6: OPINION MINING

# Content

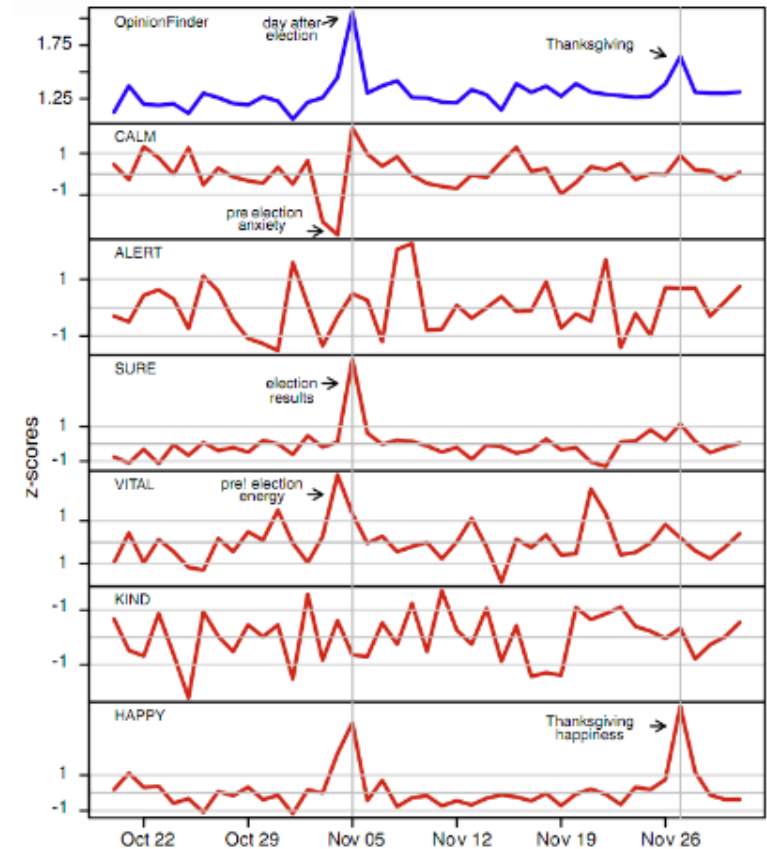
1. Problems in opinion mining
2. Unsupervised Sentiment Analysis
3. Supervised Sentiment Analysis

# 1. Problems in opinion mining



# Applications

- Customer service
- Advertising, marketing
- Social credit, personal finance
- National security
- Social policies



## Problem 1: Sentiment Analysis

Classify comments and reviews into one of three classes:

- Positive
- Negative
- Neutral

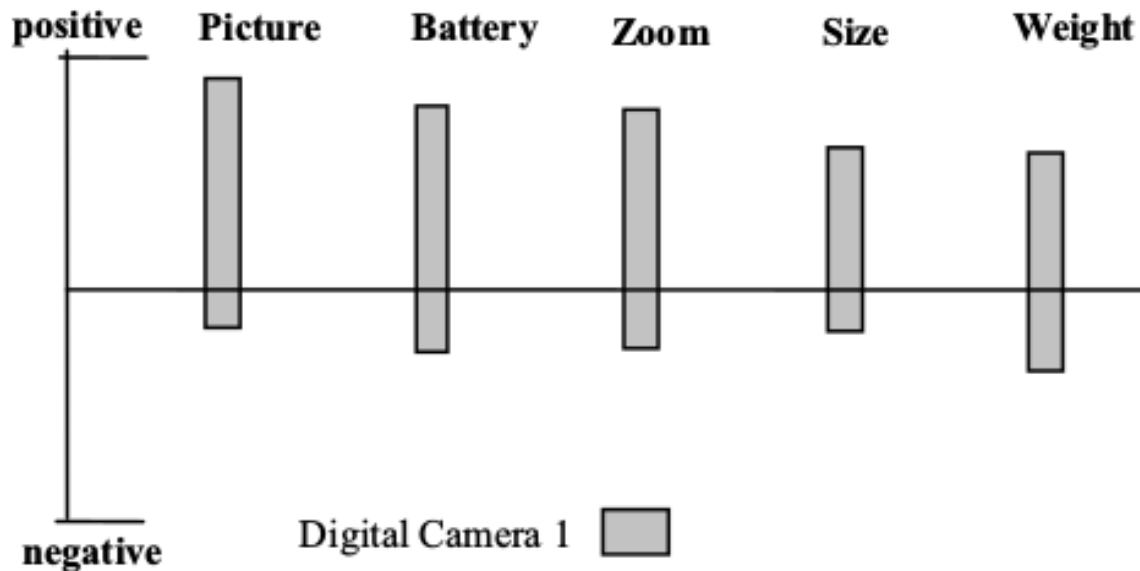


***“BPhone 3, chất đến từng chi tiết”***

## Problem 2: Opinion summarization

Includes two sub-problems:

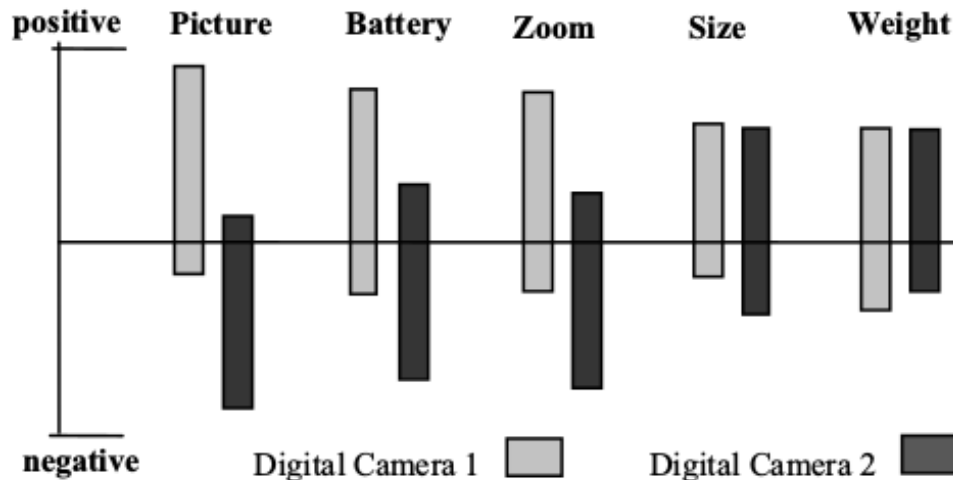
- Define aspect
- Categorize sentiment with each aspect



(A) Feature-based summary of opinions on a digital camera

## Problem 3: Comparative opinions

- Comparative opinions
  - Object A and object B
  - Object A and object B on aspect s
  - Object A with other objects



(B) Opinion comparison of two digital cameras

## Problems 4: Opinions Searching

- Searching for Opinions on an object
- search host architecture based

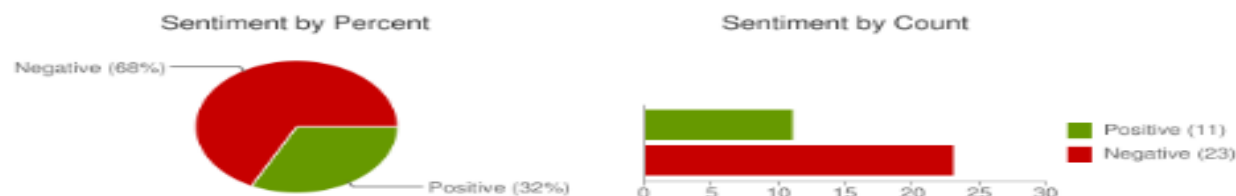
Type in a word and we'll highlight the good and the bad

"united airlines"

Search

[Save this search](#)

### Sentiment analysis for "united airlines"



**jljacobson:** OMG... Could @United airlines have worse customer service? W8g now 15 minut  
Posted 2 hours ago

**12345clumsy6789:** I hate United Airlines Ceiling!!! Fukn impossible to get my conduit in this d  
Posted 2 hours ago

**EMLandPRGbelgiu:** EML/PRG fly with Q8 united airlines and 24seven to an exotic destination  
Posted 2 hours ago

**CountAdam:** FANTASTIC customer service from United Airlines at XNA today. Is tweet more,  
Posted 4 hours ago



## Problem 5: Opinions Filtering

	Hype spam	Defaming spam
Good product	1	2
Bad product	3	4
Average product	5	6

## 2. Unsupervised Sentiment Analysis

### 2.1 Sentiment Analysis










	Example	Sentiment
<b>Introverted sentiment</b>	<i>It is an <u>honor</u> and <u>pride</u> for me to watch Vietnamese football playing at the World Cup</i>	<b>Positive</b>
<b>Extroverted sentiment</b>	<i>Nur Farahain is also known as a <u>friendly</u> and <u>sociable</u> teacher with students.</i>	<b>Positive</b>
<b>Mood</b>	<i>The contestant was <u>nervous</u>, collapsed on the table because of fatigue</i>	<b>Negative</b>
<b>Attitude</b>	<i>I wholeheartedly for my husband's family, but I'm still <u>hated</u> by my mother-in-law</i>	<b>Negative</b>
<b>Character</b>	<i>I consider myself quite <u>active</u> and know the piano.</i>	<b>Positive</b>

# Problem definition

- Requires **sentiment** recognition of a **subject** towards the **object** mentioned in the document.
- Simplify the problem given **subject** and **object** assumption

Document	Sentiment
<i>Logitech has a <b>good</b> battery, bought a B175 but the battery with the mouse has not been replaced for 3 years! Anyone who criticizes it, but I find the Logitech mouse <b>liked</b>!</i>	<b>Positive</b>
<i>The <b>expensive bad</b> product even <b>fake</b> the iPhone with a speaker underneath.</i>	<b>Negative</b>
<i>I'm using Logitech G502 and saw this one.....</i>	<b>Neutral</b>

# Methods of sentiment analysis

Methods	knowledge base request	Custom request by field	Training data request
Sentiment dictionary			
Unsupervised			
Supervised			

# Sentiment Analysis based on dictionary

thực\_sự là mình rất sợ trà\_sữa trân\_châu . hầu\_hết các cửa\_hàng toàn nhập nguyên\_liệu từ trung\_quốc với giá rất rẻ , vì mình có thằng bạn nó cũng làm quán trà\_sữa nó toàn lấy từ trung\_quốc . thế mới có lãi cao vì thuê mặt\_bằng rất đắt\_đỏ rồi . nên các bạn hãy cân\_nhắc có nên dùng trà\_sữa ko nhé

pos = 2

neg = 3

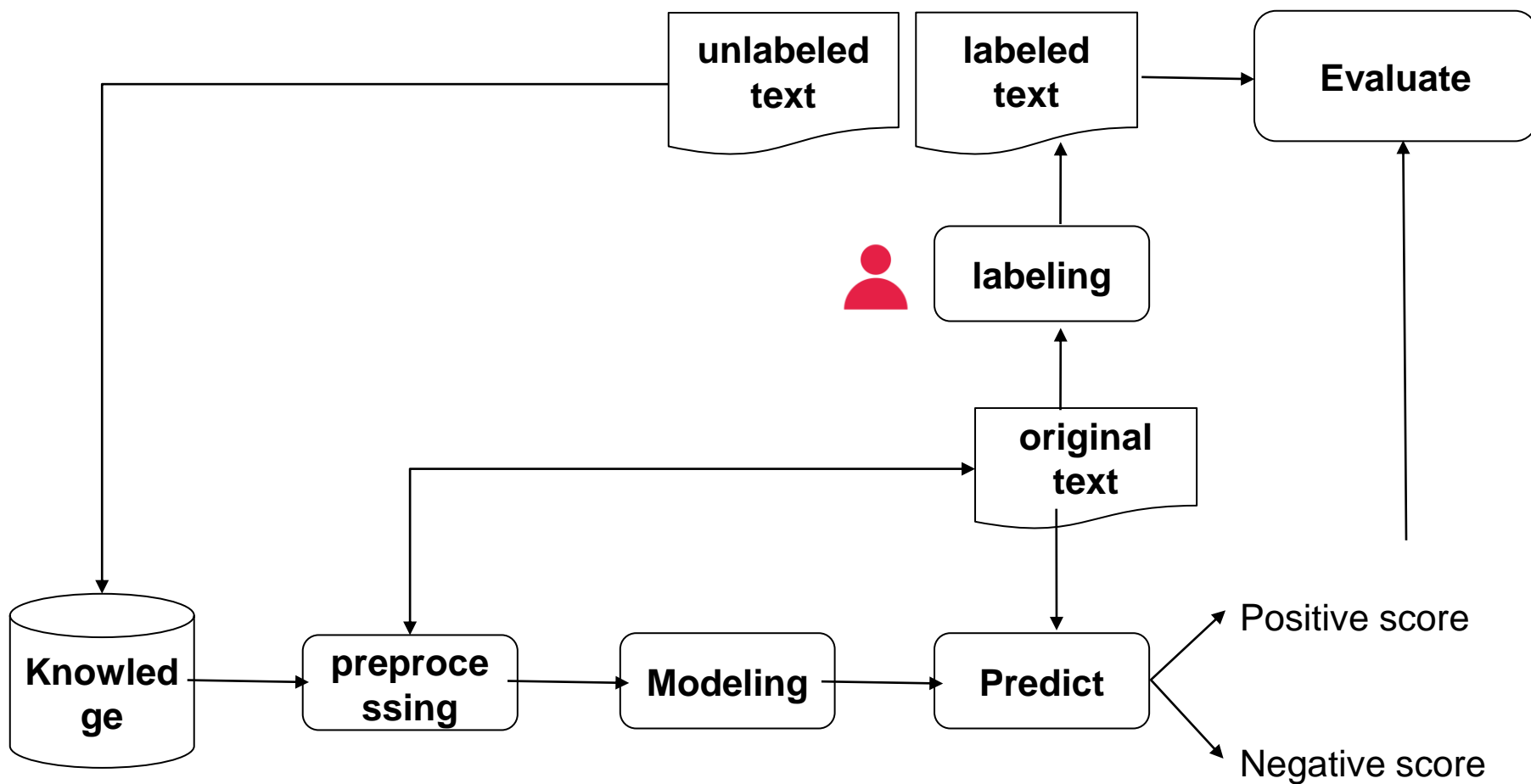
score = pos - neg = 2 - 3 = -1 < 0

Negative

## Sentiment lexicon

sợ	negative
rẻ	positive
lãi	positive
đắt đỏ	negative
cân nhắc	negative

# Supervised Sentiment Analysis



## 2.2 Unsupervised Sentiment Analysis

- P. Turney. “*Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews*”. ACL’02
- Algorithm:
  - B1. Extract opinion phrases
  - B2. Identify semantic/opinion orientation
  - B3. Determine the sentiment
- Apply to Vietnamese data

## Step1. Extract opinion phrases

- Identify language patterns with potential for opinion expression:
  - *NN+JJ: commonnoun + adjective ('máy mới')*
  - *RB+JJ: adverb + adjective ('rất tốt')*
  - *RB+VA: adverb + verb adjective ('rất khỏe')*
  - *RB+VB: adverb + verb ('rất muốn')*
  - *VB+RB: verb + adverb ('chạy mượt')*
- Require document to be POS tagged



## Step1. Extract opinion phrases (cont.)

first word	từ thứ hai
NN	JJ
RB	JJ/VA
RB	VB
VB	RB

*Thực\_sự là mình rất sợ trà\_sữa trân\_châu . Hầu\_hết các cửa\_hàng toàn nhập nguyên\_liệu từ trung\_quốc với giá rất rẻ , vì mình có thằng bạn nó cũng làm quán trà\_sữa nó toàn lấy từ trung\_quốc . Thế mới có lãi cao vì thuê mặt\_bằng rất đắt\_đỏ rồi . Nên các bạn hãy cân\_nhắc có nên dùng trà\_sữa ko nhé*

## Step1. Extract opinion phrases (cont.)

First word	Từ thứ hai
NN	JJ
RB	JJ/VA
RB	VB
VB	RB

*Thực\_sự là mình **rất/RB sợ/VB** trà\_sữa trân\_châu . Hầu\_hết các cửa\_hàng toàn nhập nguyên\_liệu từ trung\_quốc với giá **rất/RB rẻ/VA** , vì mình có thằng bạn nó cũng làm quán trà\_sữa nó toàn lấy từ trung\_quốc . Thế mới có **lãi/NN cao/JJ** vì thuê mặt\_bằng **rất/RB đắt\_đỏ/VA** rồi . Nên các bạn hãy cân\_nhắc có nên dùng trà\_sữa ko nhé*

# Step2. Identify Opinion Orientation

- For each extracted phrase  $t$ , necessary to determine opinion orientation of this phrase,  $SO(t)$
- Assumption:
  - ‘*tốt*’ : positive
  - ‘*kém*’: negative
- $SO(t) = \text{sim}(t, \text{‘tốt’}) - \text{sim}(t, \text{‘kém’})$

---

‘tốt’

$t$

‘kém’  
,

# Step2. Identify Opinion Orientation

- Determine the similarity of two phrases based on the likelihood of co-occurrence on a large corpus
  - Large Text Set: Web Text
  - Possibility co-occurrence: Pointwise Mutual Information (PMI)
  - $SO(t) = PMI(t; \text{'tốt'}) - PMI(t; \text{'kém'})$

# Step2. Identify Opinion Orientation

$$\text{PMI}(t_1; t_2) = \frac{\text{Pr}(t_1, t_2)}{\text{Pr}(t_1)\text{Pr}(t_2)} = \frac{\text{Pr}(t_1|t_2)}{\text{Pr}(t_1)} = \frac{\text{Pr}(t_2|t_1)}{\text{Pr}(t_2)}$$

$P(t_1)$ : Probability occurrence  $t_1$  in corpus

$P(t_1|t_2)$ : Probability occurrence  $t_1$  when  $t_2$  2occurrenced

$$P(t_1|t_2) = (\text{count}(t_1, t_2) + 1) / (\text{count}(t_2) + V)$$

$$P(t_1) = (\text{count}(t_1) + 1) / (\sum_{t'} \text{count}(t') + V)$$

V: Vocab size

## B3. Determine the sentiment

- Assume document  $d$  consists a set of opinion phrases  $T$  extracted from step 2
- For each  $t \in T$ , calculate  $SO(t)$
- Opinion orientation

$$SO(d) = \sum_{t \in T} SO(t)$$

- $SO(d) > 0$ : Positive document
- $SO(d) < 0$ : Negative document

# 3. Supervised Sentiment Analysis

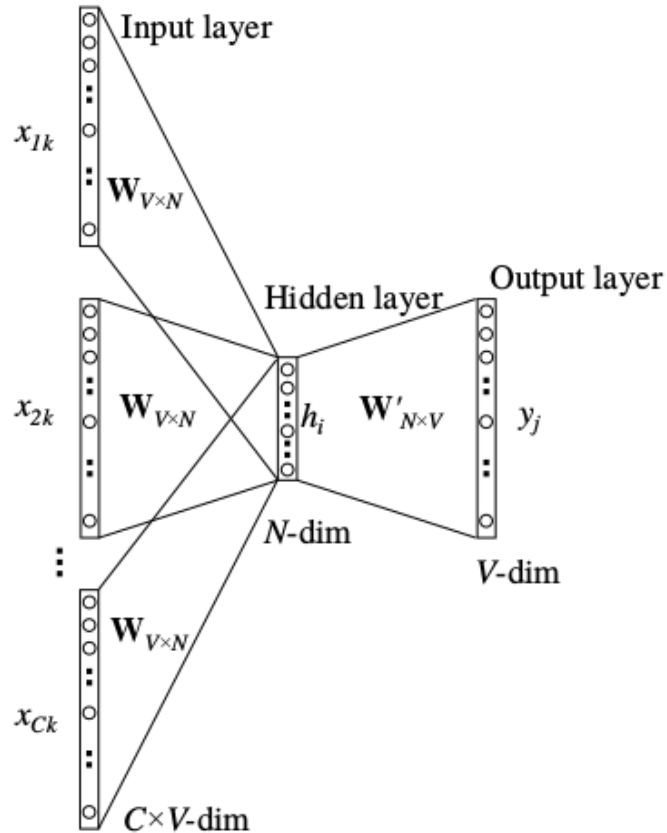
- Yoon Kim. “*Convolutional Neural Networks for Sentence Classification*”. EMNLP 2014
- Using CNN model to classify reviews
- Using pre-trained word embedding on a large data set as a word representation vector
- Concatenate the ordered word representation in the text to serve as a 2D input signal for the CNN

# Word2vec

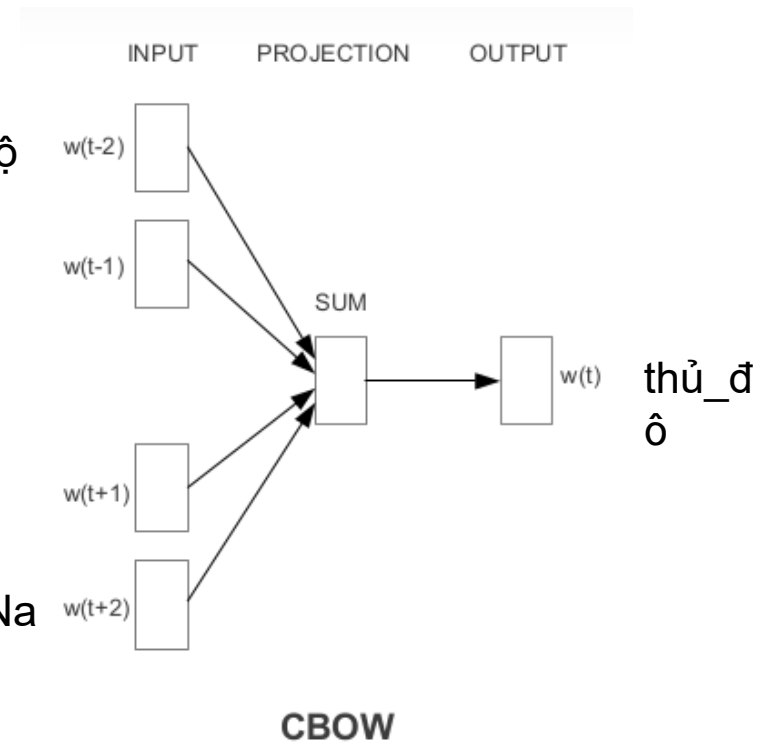
- Using a neural network to learn a language model task:
  - CBOW: Use surrounding words in a window to predict the center word
  - Skip-gram: Use focus words to predict surrounding words
- Leverage large amounts of learning data without labeling (!)
- Generates a vector representation of a word that exhibits some semantic relations.



# CBOW



Hà\_Nộ  
i  
l  
à  
  
củ  
a  
Việt\_Na  
m



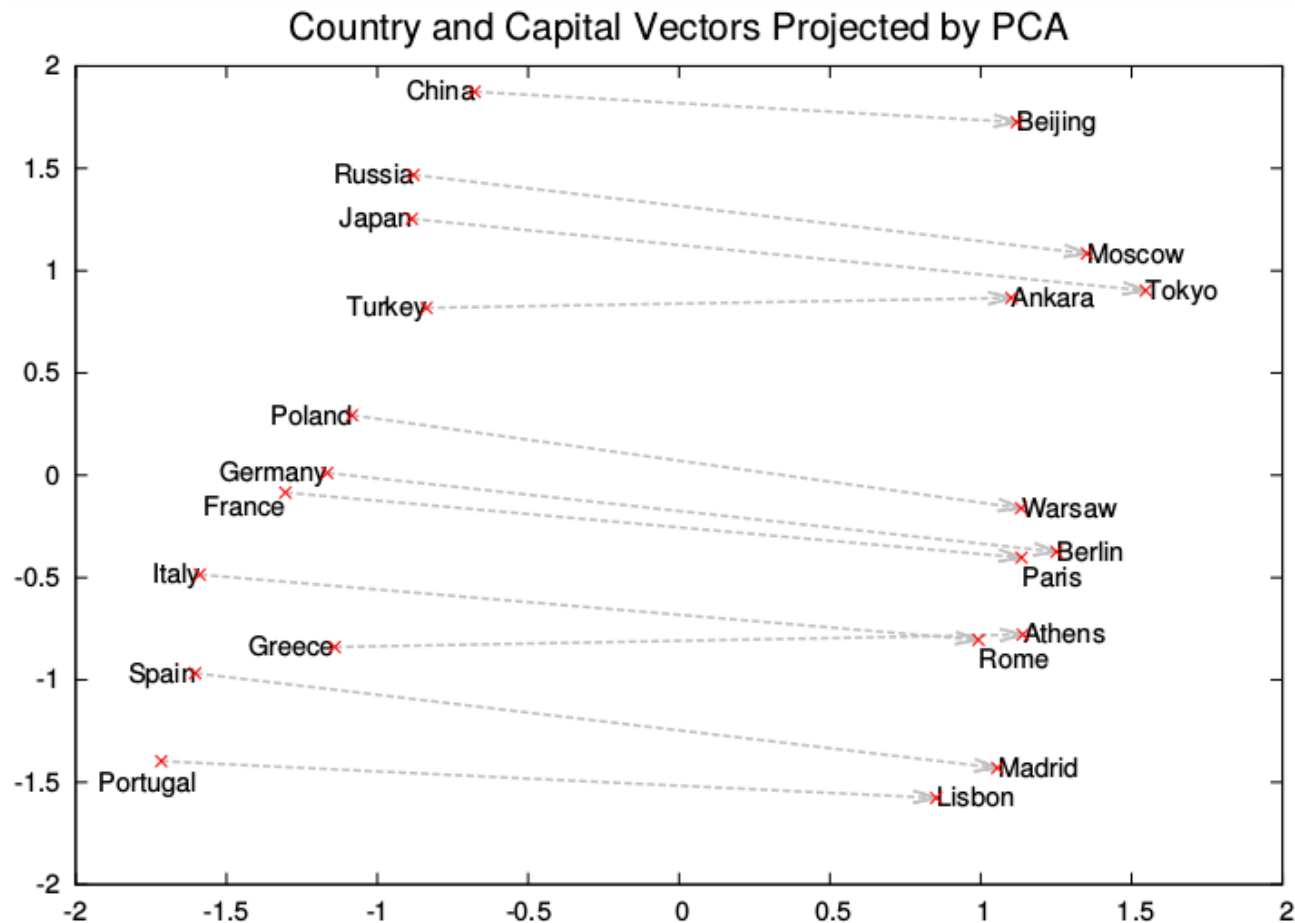
# CBOW (cont.)

- The input layer consists of  $V$  neurons that represent words in the context in the form: one-hot
- The hidden layer consists of  $n$  neurons
- The output layer consists of  $V$  neurons used to predict center word
- The weight between the input layer and the hidden layer after learning is used as a lookup table of word representation

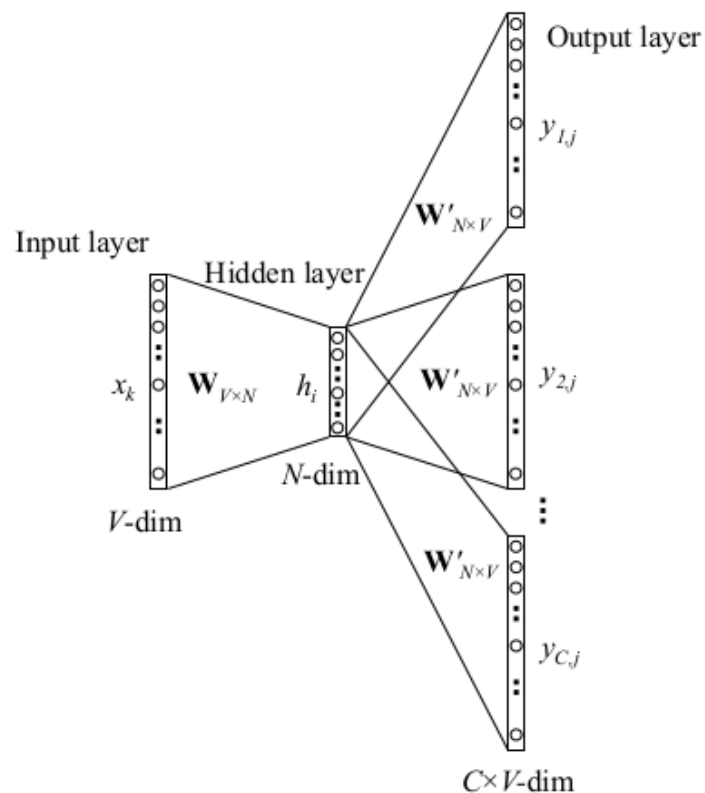
*king – queen = man - ...*

- Vector representation ‘king’:  $\mathbf{a}$
- Vector representation ‘queen’:  $\mathbf{b}$
- Vector representation ‘man’:  $\mathbf{x}$
- Calculate vector  $\mathbf{d} = \mathbf{a} - \mathbf{b} + \mathbf{c}$
- search word  $\mathbf{d}'$  whose distance (Euclidean, cosine) to  $\mathbf{d}$  is closest:  $\mathbf{d}' \sim$  'woman'

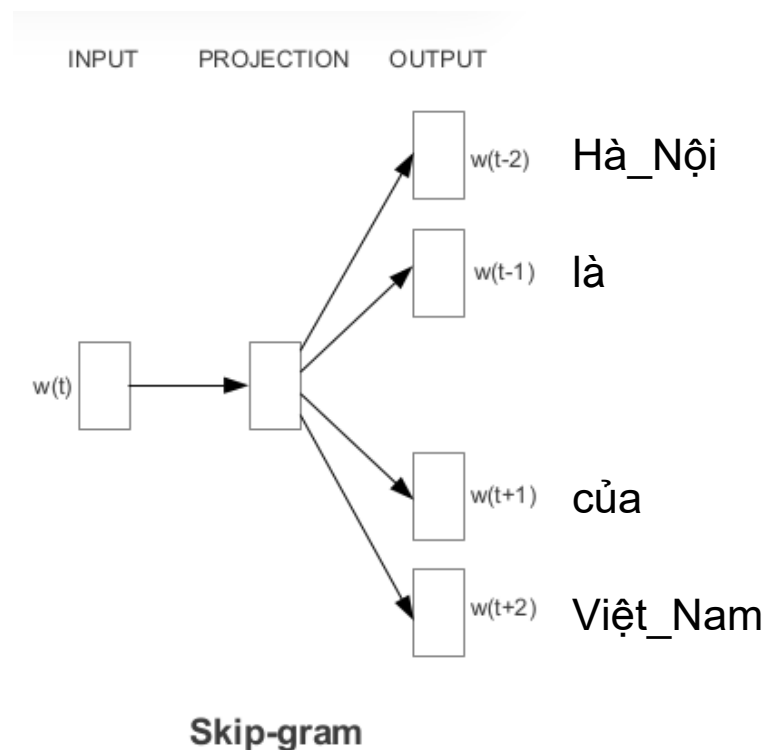
# Visualization word representation



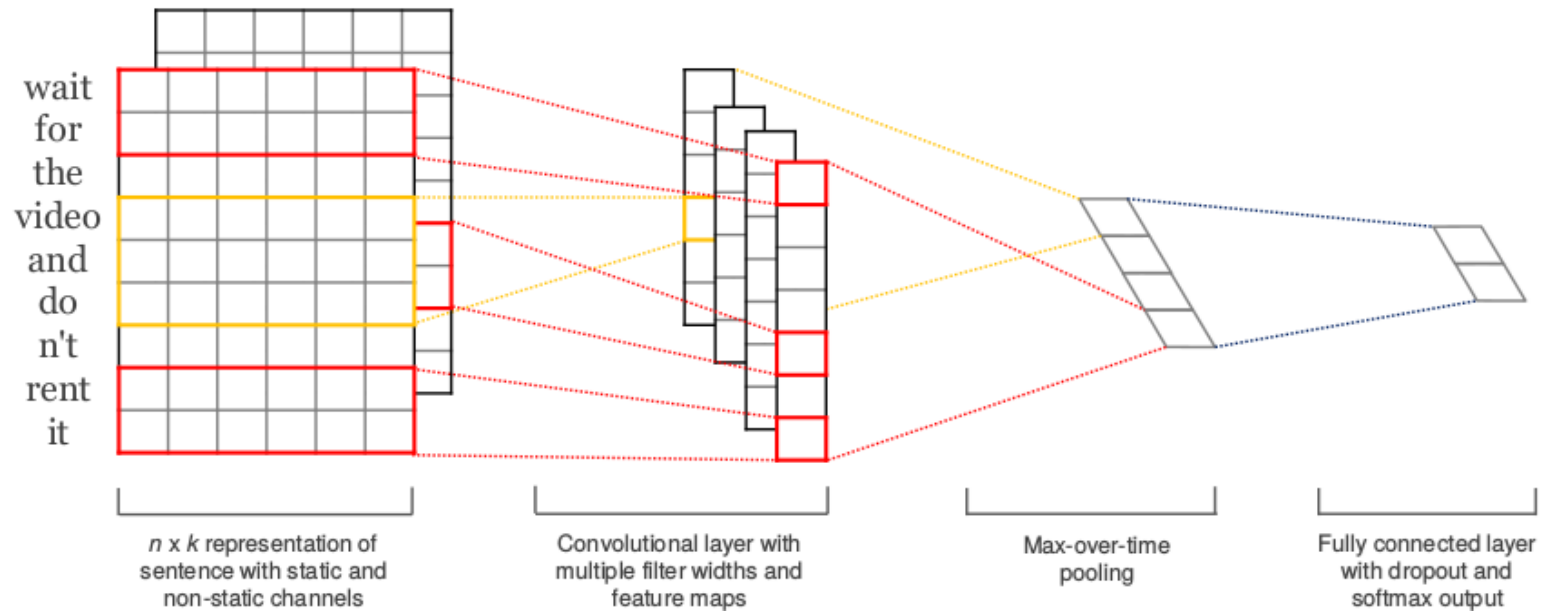
# Skip-gram



thủ\_đô



# Model architecture



# Input layer

- $\mathbf{x}_i \in \mathbb{R}^k$  is continuous representation of word  $i$ 
  - Randomly initialized and weights updated during learning
  - Initialized based on a pre-trained representation of a large corpus
    - Updated during training
    - “Freeze” in training
- Input consists words  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$  in order
- Document representation is a concatenation of word representations in the order appear in document

$$\mathbf{x}_{1:n} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \dots \oplus \mathbf{x}_n,$$

# Convolutional layer

- Each filter  $\mathbf{w} \in \mathbb{R}^{hk}$  scans a window of  $h$  consecutive words
- $\mathbf{x}_{i:i+h}$  to generate a feature  $c_i$ 
  - Window width:  $h$
  - Window height = word embedding dimension
- Each filter generates a feature map  $\mathbf{c} \in \mathbb{R}^{n-h+1}$ ,  $\mathbf{c} = [c_1, c_2, \dots, c_{n-h+1}]$

$$c_i = f(\mathbf{w} \cdot \mathbf{x}_{i:i+h-1} + b).$$



# Pooling layer

- For each feature map  $c$ , apply max pooling to get the maximum value
- Apply windows  $h \in [3, 4, 5]$
- For each value of  $h$  there are 100 filters
- Total number neurons in the pooling layer:  $100 \times 3 = 300$

# Fully connected layer

- Adjustment technique: Apply dropout at pooling layer with dropout ratio  $p = 0.5$
- Number neurons in output layer:
  - 2: only positive and negative labels
  - 3: positive, neutral, negative

# Dataset

- **MR:** Movie commentary with each comment being a sentence. Label positive/negative
- **SST-1:** Extension MR set with 5 labels (very positive, positive, neutral, negative and very negative)
- **SST-2:** Similar to SST-1 but removes neutral label and has only two positive and negative labels
- **CR:** Product reviews. Label positive/negative.

# Models

- **CNN-rand**: Embedded words are randomly initialized and updated during training
- **CNN-static**: Using pretrained word2vec embedding, word representations (including randomly initialized OOV words) are kept weighted
- **CNN-non-static**: The initial word representation in word2vec is fine-tuned during the training
- **CNN-multichannel**: hybrid static and non-static

# Experimental results

Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-rand	76.1	45.0	82.7	89.6	91.2	79.8	83.4
CNN-static	81.0	45.5	86.8	93.0	92.8	84.7	<b>89.6</b>
CNN-non-static	<b>81.5</b>	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	<b>88.1</b>	93.2	92.2	<b>85.0</b>	89.4
RAE (Socher et al., 2011)	77.7	43.2	82.4	—	—	—	86.4
MV-RNN (Socher et al., 2012)	79.0	44.4	82.9	—	—	—	—
RNTN (Socher et al., 2013)	—	45.7	85.4	—	—	—	—
DCNN (Kalchbrenner et al., 2014)	—	48.5	86.8	—	93.0	—	—
Paragraph-Vec (Le and Mikolov, 2014)	—	<b>48.7</b>	87.8	—	—	—	—
CCAIE (Hermann and Blunsom, 2013)	77.8	—	—	—	—	—	87.2
Sent-Parser (Dong et al., 2014)	79.5	—	—	—	—	—	86.3
NBSVM (Wang and Manning, 2012)	79.4	—	—	93.2	—	81.8	86.3
MNB (Wang and Manning, 2012)	79.0	—	—	<b>93.6</b>	—	80.0	86.3
G-Dropout (Wang and Manning, 2013)	79.0	—	—	93.4	—	82.1	86.1
F-Dropout (Wang and Manning, 2013)	79.1	—	—	<b>93.6</b>	—	81.9	86.3
Tree-CRF (Nakagawa et al., 2010)	77.3	—	—	—	—	81.4	86.1
CRF-PR (Yang and Cardie, 2014)	—	—	—	—	—	82.7	—
SVM <sub>S</sub> (Silva et al., 2011)	—	—	—	—	<b>95.0</b>	—	—

# Word embedding fine tuning

	Most Similar Words for	
	Static Channel	Non-static Channel
<b>bad</b>	<i>good</i> <i>terrible</i> <i>horrible</i> <i>lousy</i>	<i>terrible</i> <i>horrible</i> <i>lousy</i> <i>stupid</i>
<b>good</b>	<i>great</i> <i>bad</i> <i>terrific</i> <i>decent</i>	<i>nice</i> <i>decent</i> <i>solid</i> <i>terrific</i>
<b>n't</b>	<i>os</i> <i>ca</i> <i>ireland</i> <i>wo</i>	<i>not</i> <i>never</i> <i>nothing</i> <i>neither</i>

<b>!</b>	<i>2,500</i> <i>entire</i> <i>jez</i> <i>changer</i>	<i>2,500</i> <i>lush</i> <i>beautiful</i> <i>terrific</i>
<b>,</b>	<i>decasia</i> <i>abysmally</i> <i>demise</i> <i>valiant</i>	<i>but</i> <i>dragon</i> <i>a</i> <i>and</i>



VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Thank you  
for your  
attentions!

