Quick Tour of Text Mining

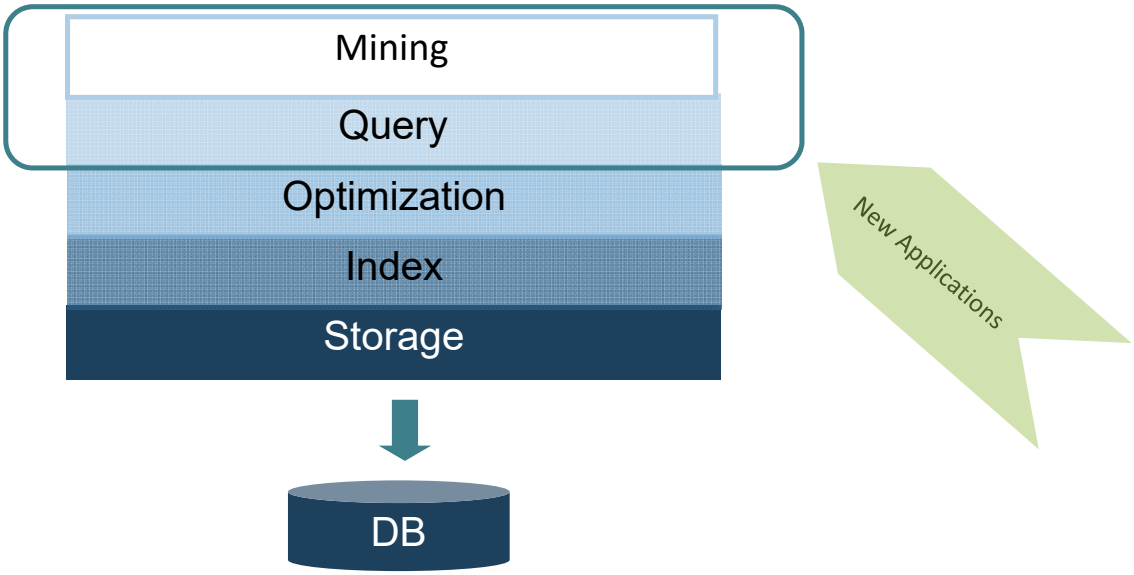
Yi-Shin Chen
Institute of Information Systems and Applications
Department of Computer Science
National Tsing Hua University
yishin@gmail.com

About Speaker

陳宜欣 Yi-Shin Chen

- Currently
  - 清華大學資訊工程系副教授
  - 主持智慧型資料工程與應用實驗室 (IDEA Lab)
- Education
  - Ph.D. in Computer Science, USC, USA
  - M.B.A. in Information Management, NCU, TW
  - B.B.A. in Information Management, NCU, TW
- Courses *(all in English)*
  - Research and Presentation Skills
  - Introduction to Database Systems
  - Advanced Database Systems
  - Data Mining: Concepts, Techniques, and Applications

Research Focus from 2000



Free Resources

▷免費數據

- 不管公網、私網，能合法下載的資料都是好物

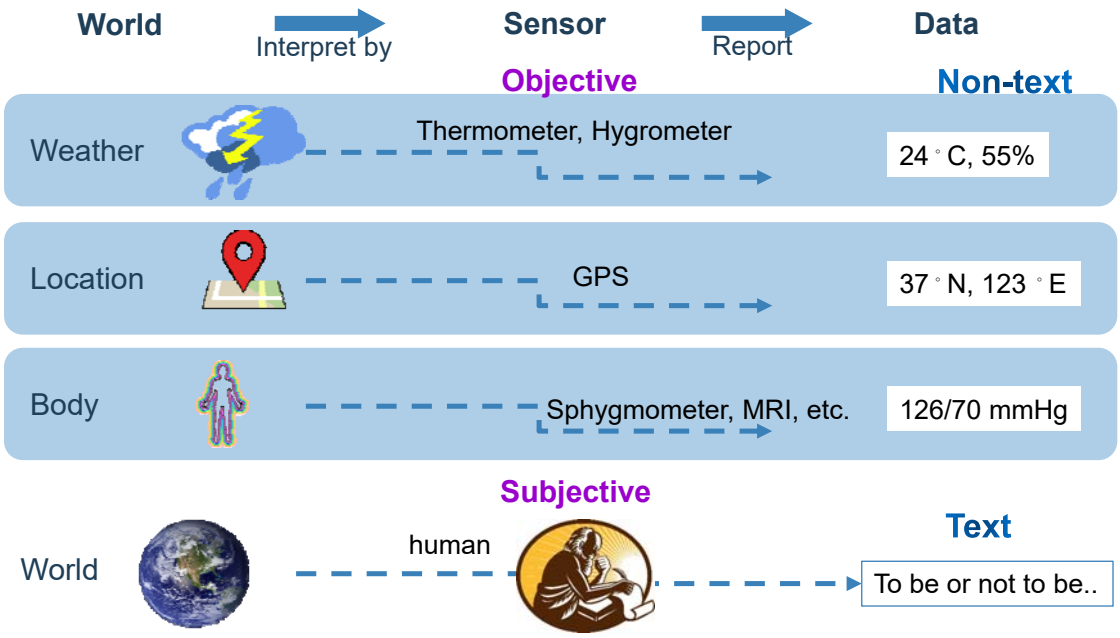
Past and Current Studies

- Location identification
- Interest identification
- Event identification
- Extract semantic relationships
- Unsupervised multilingual sentiment analysis
- Keyword extraction and summary
- Emotion analysis
- Mental illness detection

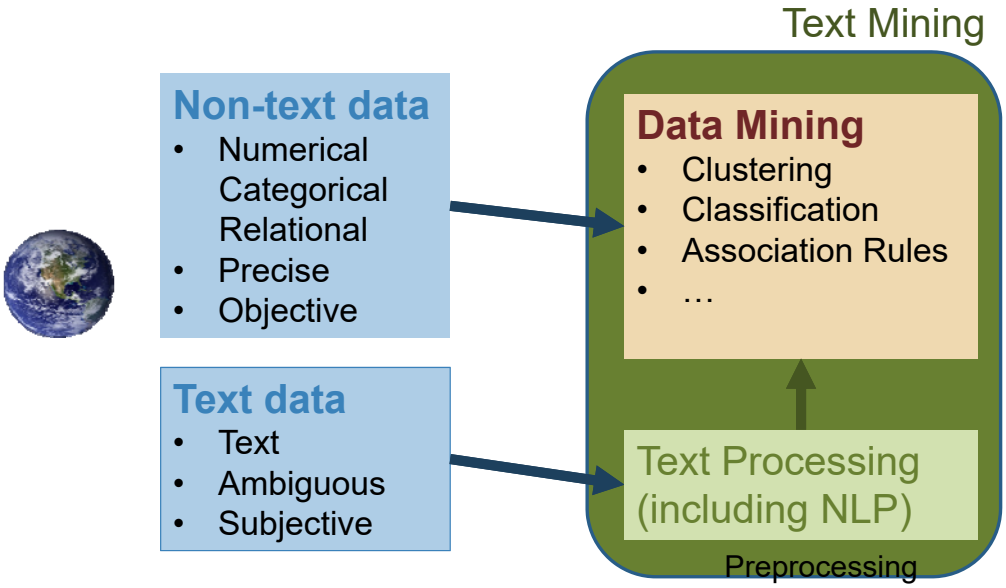
Text Processing

Text Mining Overview

Data (Text vs. Non-Text)



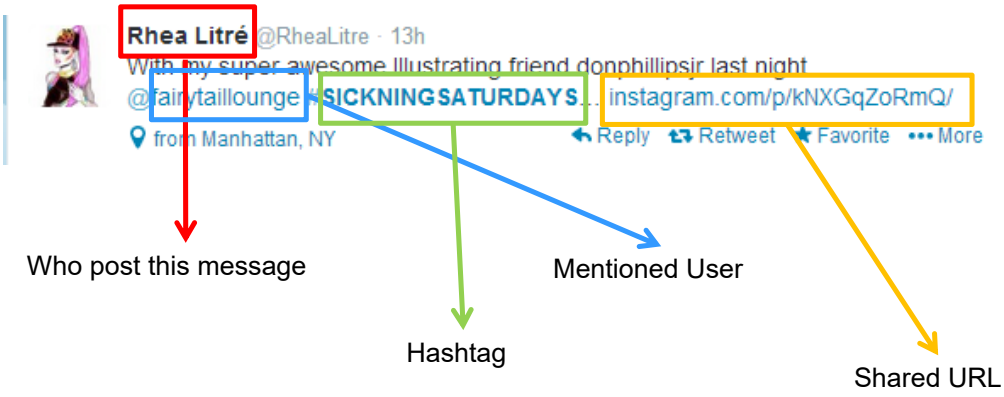
Data Mining vs. Text Mining



Preprocessing in Reality

Data Collection

- Align /Classify the attributes correctly



Language Detection

- To detect an language (possible languages) in which the specified text is written

你好 現在幾點鐘
apa kabar sekarang jam berapa ?

繁體中文 (zh-tw)
印尼文 (id)

- Difficulties
  - Short message
  - Different languages in one statement
  - Noisy

Wrong Detection Examples

Before / after removing noise

@sayidatynet top song #LailaGhofran
shokran ya garh new album #listen

en -> id

中華隊的服裝挺特別的，好藍。。。
#ChineseTaipei #Sochi #2014冬奧

it -> zh-tw

授業前の雪合戦w
http://t.co/d9b5peaq7J

en -> ja

Removing Noise

- Removing noise before detection
  - Html file ->tags
  - Twitter -> hashtag, mention, URL

<meta name=\"twitter:description\" content=\"觸犯法國隱私法〔駐歐洲特派記者胡蕙寧、國際新聞中心／綜合報導〕網路搜尋引擎巨擘Google8日在法文版首頁（www.google.fr）張貼悔過書 ...\"/>

→ 英文 (en)

觸犯法國隱私法〔駐歐洲特派記者胡蕙寧、國際新聞中心／綜合報導〕網路搜尋引擎巨擘Google8日在法文版首頁（www.google.fr）張貼悔過書 ...

→ 繁中 (zh-tw)

Data Cleaning

- Special character

Unicode emotions	☺, ♥...
Symbol icon	☏, ✉ ...
Currency symbol	€, £, $...

- Utilize regular expressions to clean data

Tweet URL

(^|\\s*)http(\\S+)?(\\s*|$)

ube playlist
mie Riepe

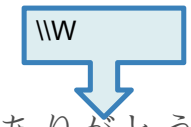
Filter out non-(letters, space, punctuation, digit)

(\\p{L}+)|(\\p{Z}+)|(\\p{Punct}+)|(\\p{Digit}+)

g ♥ ✉

Japanese Examples

- Use regular expression remove all special words



  - うふふふふ(*^^*)楽しむ！ありがとうございます^o^ アイコン、ラブラブ(-_-)♡

  - うふふふふ 楽しむ ありがとうございます アイコン ラブラブ

Part-of-speech (POS) Tagging

- Processing text and assigning parts of speech to each word
- Twitter POS tagging
  - Noun (N), Adjective (A), Verb (V), URL (U)…

Happy Easter! I went to work and came home to an empty house now im going for a quick run http://t.co/Ynp0uFp6oZ

Happy_A Easter_N !_, I_O went_V to_P work_N and_& came_V home_N to_P an_D empty_A house_N now_R im_L going_V for_P a_D quick_A run_N http://t.co/Ynp0uFp6oZ_U

Stemming

love miss be

RT @kt_biv : @caycelynnn loving and missing you! we are still looking for Lucy

look

- @DirtyDTran gotta be caught up for tomorrow nights episode
- @ASVP_Jaykey for some reasons I found this very amusing

- @DirtyDTran gotta be catch up for tomorrow night episode
- @ASVP_Jaykey for some reason I find this very amusing

Hashtag Segmentation

- By using Microsoft Web N-Gram Service (or by using **Viterbi algorithm**)

Wow! explosion at a boston race ... #prayforboston

#pray #for #boston

#citizenscience	→	#citizen #science
#bostonmarathon	→	#boston #marathon
#goodthingsarecoming	→	#good #things #are #coming
#lowbloodpressure	→	#low #blood #pressure

More Preprocesses for Different Web Data

- Extract source code without javascript
- Removing html tags

Extract Source Code Without Javascript

- Javascript code should be considered as an exception
  - it may contain hidden content

```
<html>
 <title>Parakweet - Actionable Signals for Social Media</title>
 <script src="//use.typekit.net/ygl8ojq.js" type="text/javascript"></script>
 <script>
  try
   {
       Typekit.load();
   }
   catch(e){}
 </script>
 <link href="stylesheets/parakweet.css" media="screen" rel="stylesheet" type="text/css" />
 <body><div class='lead'>
       Parakweet is a new platform offering social media analytics, recommendations and metadata to media companies.</div>
   <div class='subtext hide' id='subtext_two'>
       Parakweet's Product Recommendations provide customers with relevant, actionable ideas for brand and product discovery.</div>
 </body>
</html>
```

```
<html>
 <title>Parakweet - Actionable Signals for Social Media</title>
 <link href="stylesheets/parakweet.css" media="screen" rel="stylesheet" type="text/css" />
 <body><div class='lead'>
       Parakweet is a new platform offering social media analytics, recommendations and metadata to media companies.</div>
   <div class='subtext hide' id='subtext_two'>
       Parakweet's Product Recommendations provide customers with relevant, actionable ideas for brand and product discovery.</div>
 </body>
</html>
```

Remove Html Tags

- Removing html tags to extract meaningful content

```
<html>
  <title>Parakweet - Actionable Signals for Social Media</title>
  <link href="stylesheets/parakweet.css" media="screen" rel="stylesheet" type="text/css" />
  <body><div class='lead'>
        Parakweet is a new platform offering social media analytics, recommendations and metadata to media companies.</div>
    <div class='subtext hide' id='subtext_two'>
        Parakweet's Product Recommendations provide customers with relevant, actionable ideas for brand and product discovery.</div>
  </body>
</html>
```

```
Parakweet - Actionable Signals for Social Media
Parakweet is a new platform offering social media analytics, recommendations and metadata to media companies.
Parakweet's Product Recommendations provide customers with relevant, actionable ideas for brand and product discovery.
```


More Preprocesses for Different Languages

- Chinese Simplified/Traditional Conversion
- Word segmentation

Chinese Simplified/Traditional Conversion

- Word conversion
  - 请乘客从后门落车 → 請乘客從後門下車
- One-to-many mapping
  - @shinrei 出去旅游还是崩坏 → @shinrei 出去旅游還是崩壞
    游 (zh-cn) → 游|遊 (zh-tw)
- Wrong segmentation
  - 人体内存在很多微生物 → 內存: 人體 記憶體 在很多微生物
    → 存在: 人體內 存在 很多微生物

內存|存在

Wrong Chinese Word Segmentation

- Wrong segmentation
  - 這(Nep) 地面(Nc) 積(VJ) 還(D) 真(D) 不(D) 小(VH) http://t.co/QlUbiaz2lz

地面|面積

- Wrong word
  - @iamzeke 實驗(Na) 室友(Na) 多(Dfa) 危險(VH) 你(Nh) 不(D) 知道(VK) 嗎(T) ?

實驗室|室友

- Wrong order
  - 人體(Na) 存(VC) 內在(Na) 很多(Neqa) 微生物(Na)

存在|內在

- Unknown word
  - 半夜(Nd) 逛團(Na) 購(VC) 看到(VE) 太(Dfa) 吸引人(VH) !!

未知詞: 團購

Back to Text Mining

Let's come back to Text Mining

Data Mining vs. Text Mining

Non-text data
- Numerical
  Categorical
  Relational
- Precise
- Objective

Text data
- Text
- Ambiguous
- Subjective

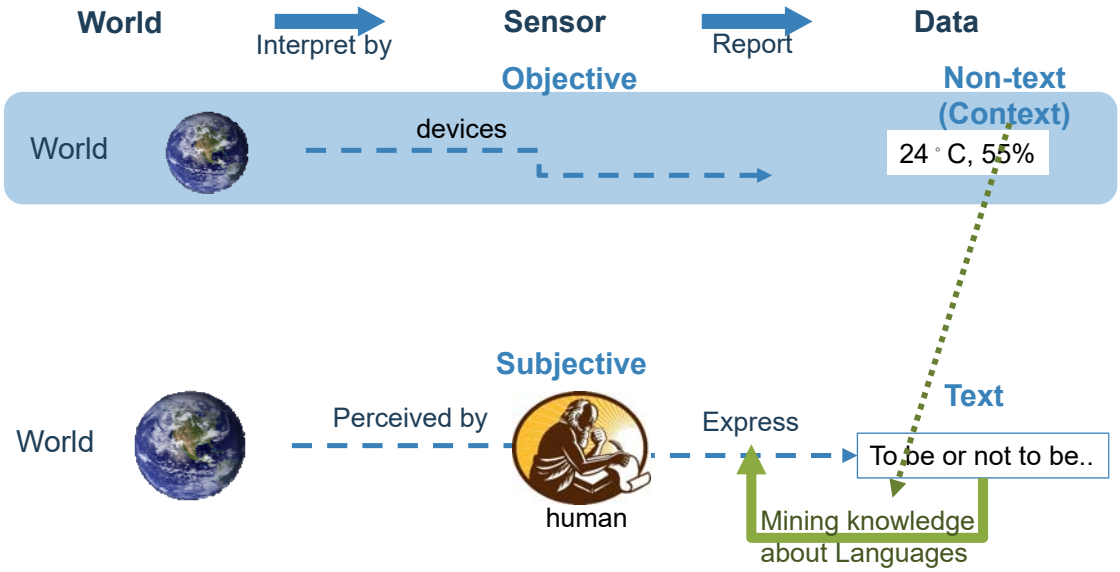
Text Mining

Data Mining
- Clustering
- Classification
- Association Rules
- …

Text Processing (including NLP)

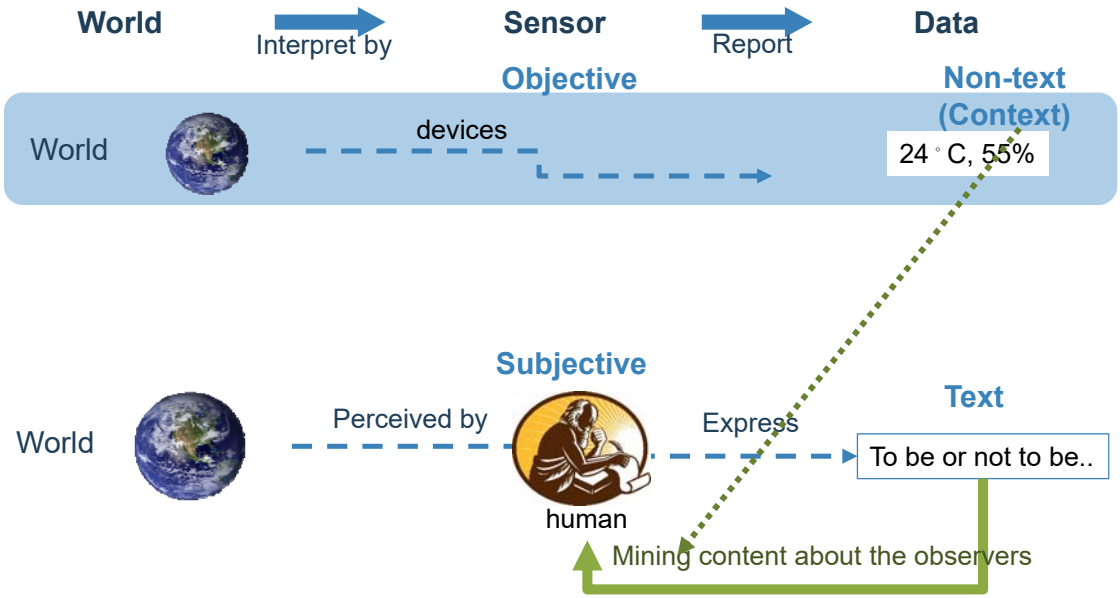
Preprocessing

Landscape of Text Mining



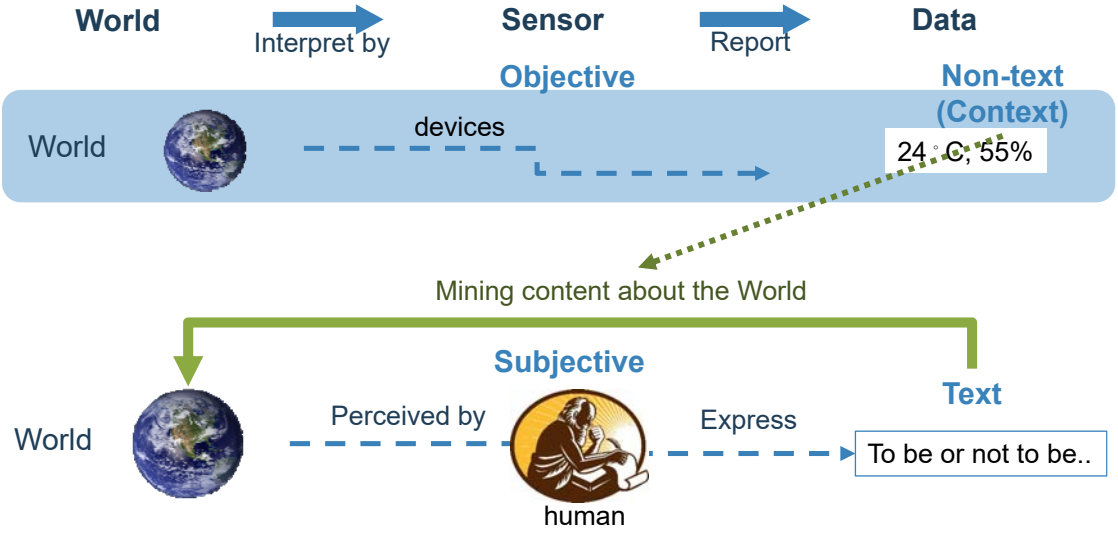
Nature Language Processing & Text Representation; Word Association and Mining

Landscape of Text Mining



Opinion Mining and Sentiment Analysis

Landscape of Text Mining



Topic Mining , Contextual Text Mining

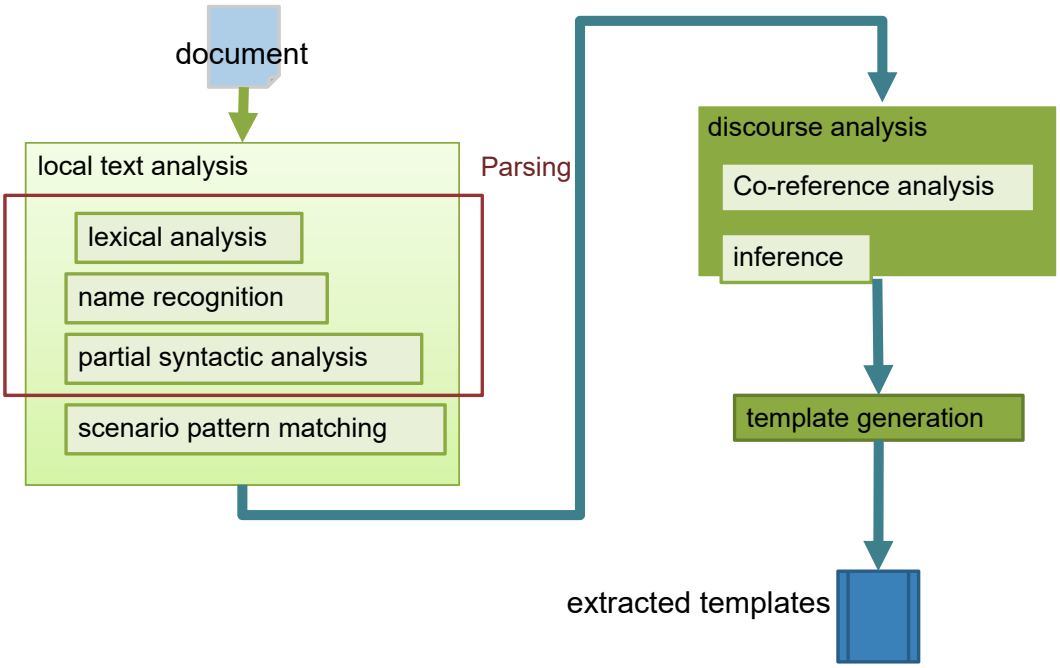
Basic Concepts in NLP

This	is	the	best	thing	happened	in	my	life.
Det.	Verb	Det.	Adj	N	Verb	Pre.	PN	N

辭彙
Lexical analysis (Part-of Speech Tagging)

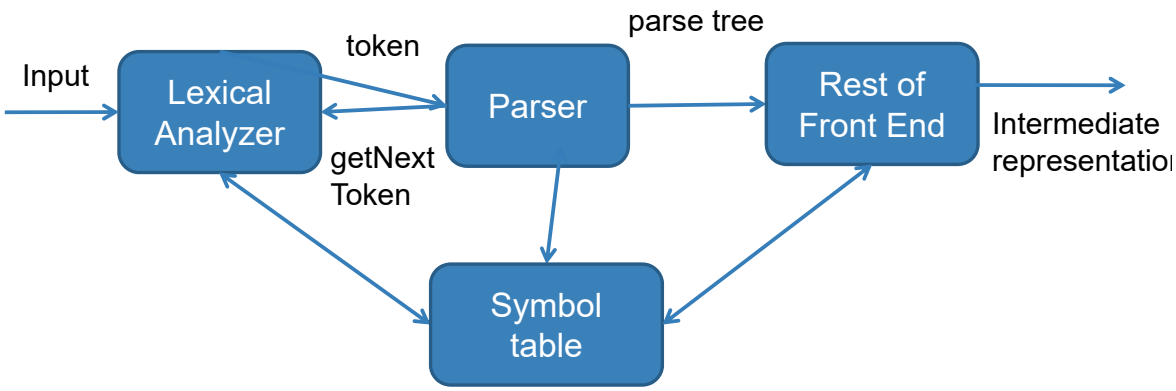
句法
Syntactic analysis (Parsing)

Structure of Information Extraction System



Parsing

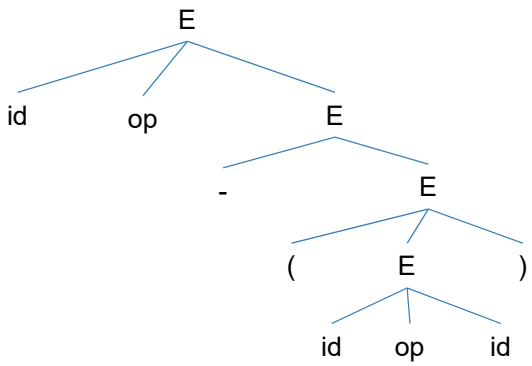
- Parsing is the process of determining whether a string of tokens can be generated by a grammar



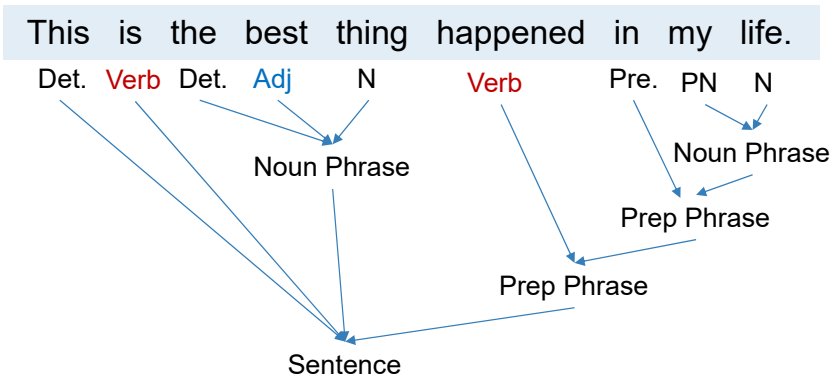
Parsing Example (for Compiler)

- Grammar:
  - E :: = E op E | - E | (E) | id
  - op :: = + | - | * | /

a	*	-	(	b	+	c	)



Basic Concepts in NLP



辭彙
Lexical analysis (Part-of Speech Tagging)

句法
Syntactic analysis (Parsing)

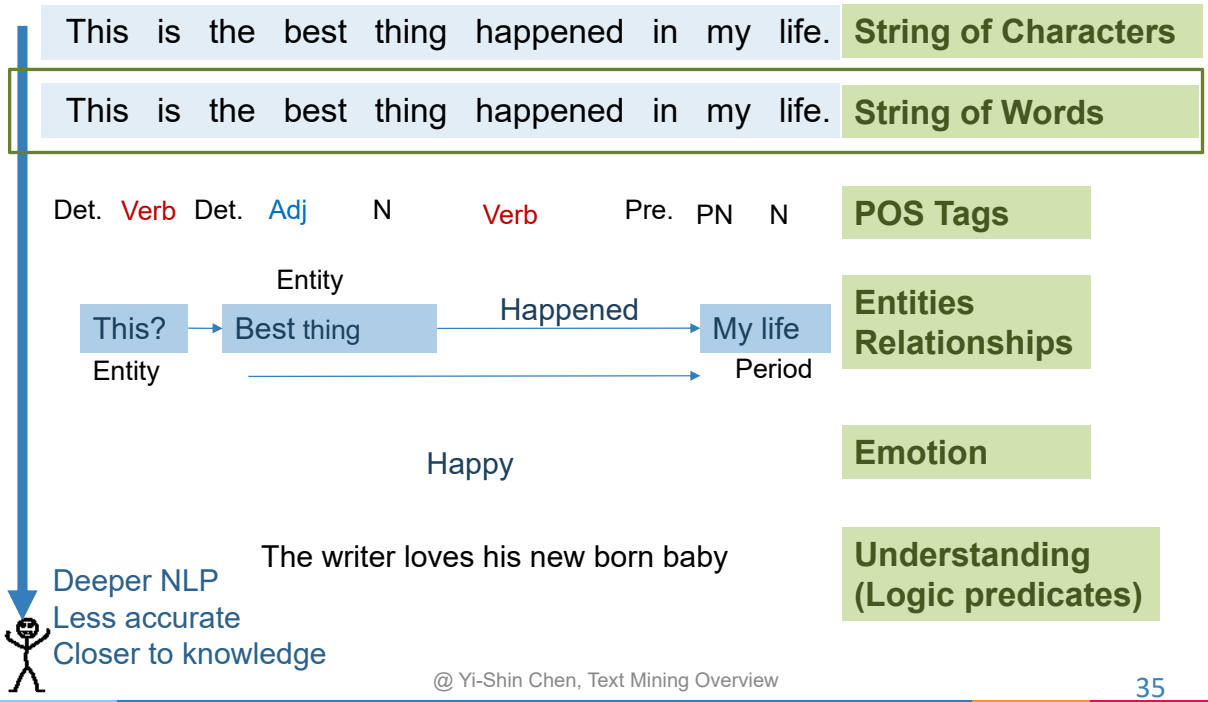
This? (t1)
Best thing (t2)
My (m1)
Happened (t1, t2, m1)

語意
Semantic Analysis

Happy (x) if Happened (t1, 'Best', m1) ⇒ Happy

推理
Inference (Emotion Analysis)

Basic Concepts in NLP



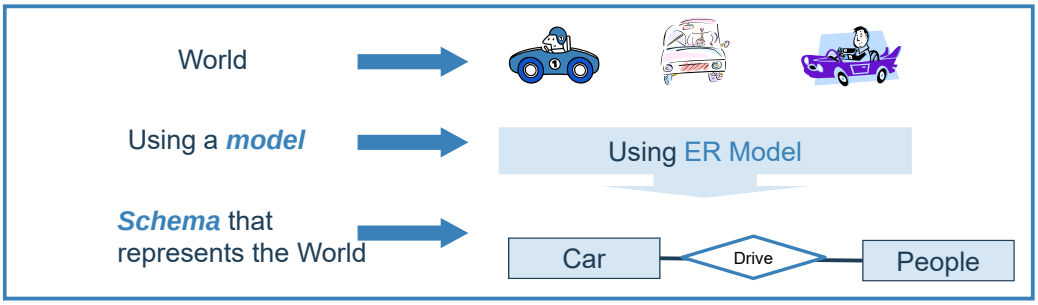
NLP vs. Text Mining

- NLP objectives
  - Understanding
  - Ability to answer
  - Immaculate
- Text Mining objectives
  - Overview
  - Know the trends
  - Accept noise

Basic Data Model Concepts

Let's learn from giants

Data Models



- Data model/Language: a *collection* of concepts for describing data
- Schema/Structured observation: a *description* of a particular collection of data, using the a given data model
- Data instance/Statements

E-R Model

- Introduced by Peter Chen; ACM TODS, March 1976
  - https://www.linkedin.com/in/peter-chen-43bba0/

  - Additional Readings
    - → Peter Chen. "English Sentence Structure and Entity-Relationship Diagram." Information Sciences, Vol. 1, No. 1, Elsevier, May 1983, Pages 127-149
    - → Peter Chen. "A Preliminary Framework for Entity-Relationship Models." Entity-Relationship Approach to Information Modeling and Analysis, North-Holland (Elsevier), 1983, Pages 19 - 28

E-R Model Basics -Entity

- Based on a perception of a real world, which consists
  - A set of basic objects ⇒ Entities
  - Relationships among objects
- Entity: Real-world object distinguishable from other objects
- Entity Set: A collection of similar entities. E.g., all employees.
  - Presented as:

Animals | Time | People | Things

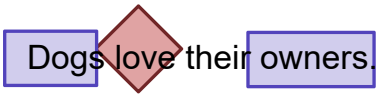
This is the best thing happened in my life.

Dogs love their owners.

E-R: Relationship Sets

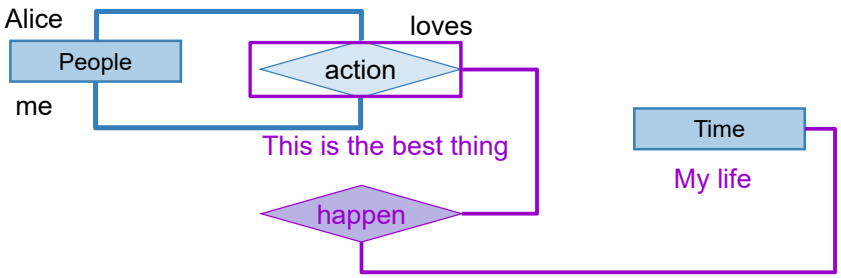
- Relationship: Association among two or more entities
- Relationship Set: Collection of similar relationships.
  - Relationship set are presented as:

  - The relationship cannot exist without having corresponding entities



High-Level Entity

- High-level entity: Abstracted from a group of interconnected low-level entity and relationship types



Alice loves me.
This is the best thing happened in my life.

Word Relations

Back to text

Word Relations

- Paradigmatic: can be substituted for each other (similar)
  - E.g., Cat & dog, run and walk

- Syntagmatic: can be combined with each other (correlated)
  - E.g., Cat and fights, dog and barks

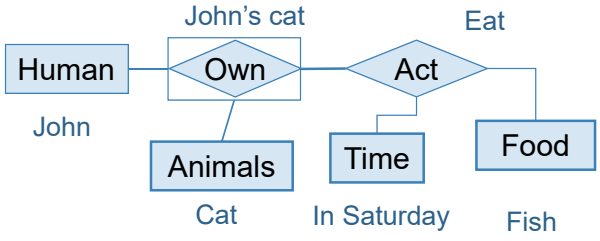
→These two basic and complementary relations can be generalized to describe relations of any times in a language

Mining Word Associations

- Paradigmatic
  - Represent each word by its context
  - Compute context similarity
  - Words with high context similarity
- Syntagmatic
  - Count the number of times two words occur together in a context
  - Compare the co-occurrences with the corresponding individual occurrences
  - Words with high co-occurrences but relatively low individual occurrence

Paradigmatic Word Associations

John's cat eats fish in Saturday
Mary's dog eats meat in Sunday
John's cat drinks milk in Sunday
Mary's dog drinks beer in Tuesday



Similar right content

How similar are context ("cat") and context ("dog")?
How similar are context ("cat") and context ("John")?
→ Expected Overlap of Words in Context (EOWC)

Overlap ("cat", "dog")
Overlap ("cat", "John")

Vector Space Model (Bag of Words)

- Represent the keywords of objects using a term vector
  - Term: basic concept, e.g., keywords to describe an object
  - Each term represents one dimension in a vector
  - N total terms define an n-element terms
  - Values of each term in a vector corresponds to the importance of that term
- Measure similarity by the vector distances

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

Common Approach for EOWC: Cosine Similarity

- If d_1 and d_2 are two document vectors, then
  $\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \, ||d_2||$,
  where $\bullet$ indicates vector dot product and $||d||$ is the length of vector d.
- Example:

d_1 = **3 2 0 5 0 0 0 2 0 0**
d_2 = **1 0 0 0 0 0 0 1 0 2**

$d_1 \bullet d_2$= 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5
$||d_1||$ = (3*3+2*2+0*0+5*5+0*0+0*0+0*0+0*0+2*2+0*0+0*0)$^{0.5}$ = (42) $^{0.5}$ = 6.481
$||d_2||$ = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2) $^{0.5}$ = (6) $^{0.5}$ = 2.245

$\cos(d_1, d_2) = .3150$

→ Overlap (“John”, “Cat”) =.3150

Quality of EOWC?

- The more overlap the two context documents have, the higher the similarity would be
- However:
  - It favor matching one frequent term very well over matching more distinct terms
  - It treats every word equally (overlap on “the” should not be as meaningful as overlap on “eats”)

Term Frequency and Inverse Document Frequency (TFIDF)

- Since not all objects in the vector space are equally important, we can weight each term using its occurrence probability in the object description
  - Term frequency: TF(***d,t***)
    - → number of times ***t*** occurs in the object description ***d***
  - Inverse document frequency: IDF(***t***)
    - → to scale down the terms that occur in many descriptions

Normalizing Term Frequency

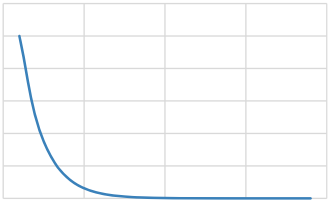
- n_{ij} represents the number of times a term t_i occurs in a description d_j . tf_{ij} can be normalized using the total number of terms in the document
  - $tf_{ij} = \frac{n_{ij}}{NormalizedValue}$
- Normalized value could be:
  - Sum of all frequencies of terms
  - Max frequency value
  - Any other values can make tf_{ij} between 0 to 1
  - BM25*: $tf_{ij} = \frac{n_{ij} \times (k+1)}{n_{ij}+k}$

Inverse Document Frequency

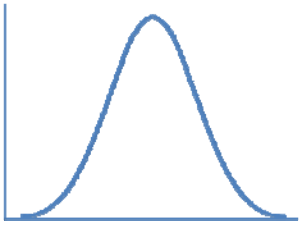
- IDF seeks to scale down the coordinates of terms that occur in many object descriptions
  - For example, some stop words(the, a, of, to, and…) may occur many times in a description. However, they should be considered as non-important in many cases
  - $idf_i = log\left(\frac{N}{df_i} + 1\right)$
    - → where df_i (document frequency of term t_i) is the number of descriptions in which t_i occurs
- IDF can be replaced with ICF (inverse class frequency) and many other concepts based on applications

Reasons of Log

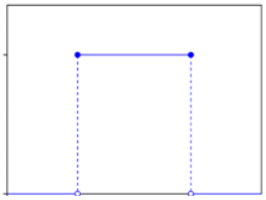
Power-law distribution



Normal distribution



Uniform distribution



- Each distribution can indicate the hidden force
  - Time
  - Independent
  - Control

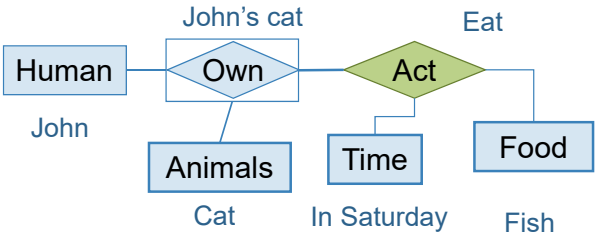
Mining Word Associations

- Paradigmatic
  - Represent each word by its context
  - Compute context similarity
  - Words with high context similarity
- Syntagmatic
  - Count the number of times two words occur together in a context
  - Compare the co-occurrences with the corresponding individual occurrences
  - Words with high co-occurrences but relatively low individual occurrence

Syntagmatic Word Associations

Correlated occurrences

John's cat eats fish in Saturday
Mary's dog eats meat in Sunday
John's cat drinks milk in Sunday
Mary's dog drinks beer in Tuesday



What words to the right?

John's	***	eats	***	in Saturday
Mary's	***	eats	***	in Sunday
John's	---	drinks	---	in Sunday
Mary's	---	drinks	---	in Tuesday

What words tend to occur to the left of "eats"

Whenever "eats" occurs, what other words also tend to occur?

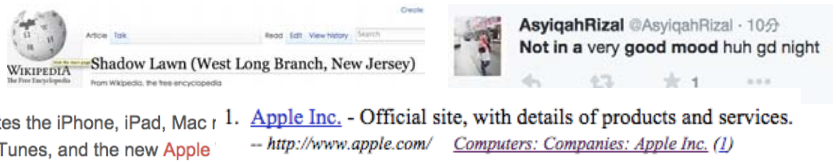
P(dog | eats) = ? ; P(cats | eats) = ?

Word Prediction

Prediction Question: Is word W present (or absent) in this segment?

Text Segment (any unit, e.g., sentence, paragraph, document)

apple companies



Predict the occurrence of word

W1 = 'meat'

W2 = 'a'

W3 = 'unicorn'

Word Prediction: Formal Definition

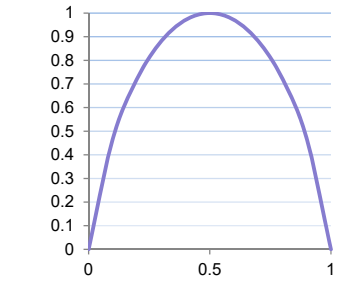
- Binary random variable {0,1}
  - $x_w = \begin{cases} 1 & w \text{ is present} \\ 0 & w \text{ is abset} \end{cases}$
  - $P(x_w = 1) + P(x_w = 0) = 1$
- The more random x_w is, the more difficult the prediction is
- How do we quantitatively measure the ***randomness***?

Entropy

- Entropy measures the amount of randomness or surprise or uncertainty
- Entropy is defined as:

$$H(p_1, \cdots p_n) = \sum_{i=1}^{n} \left(p_i \times \log \frac{1}{p_i} \right) = -\sum_{i=1}^{n} (p_i \times \log p_i)$$

$$where \sum_{i=1}^{n} (p_i) = 1$$



Entropy(p,1-p)

- entropy = 0
  easy
- entropy=1
  - difficult

Conditional Entropy

$$H(Y|X) = \sum_{x \in X} (p(x) H(Y|X = x))$$

Know nothing about the segment

$p(x_{meat} = 1)$
$p(x_{meat} = 0)$

Know "eats" is present (X_{eat}=1)

$p(x_{meat} = 1|x_{eats} = 1)$
$p(x_{meat} = 0|x_{eats} = 1)$

$$H(x_{meat}) = -p(x_{meat} = 0) \times log_2(p(x_{meat} = 0)) - p(x_{meat} = 1) \times log_2(p(x_{meat} = 1))$$

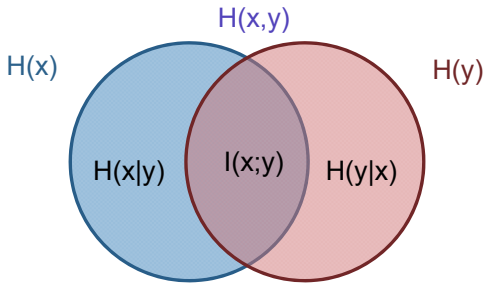
$$H(x_{meat}|x_{eats} = 1) = -p(x_{meat} = 0|x_{eats} = 1) \times log_2(p(x_{meat} = 0|x_{eats} = 1)) - p(x_{meat} = 1|x_{eats} = 1) \times log_2(p(x_{meat} = 1|x_{eats} = 1))$$

Mining Syntagmatic Relations

- For each word W1
  - For every word W2, compute conditional entropy $H(x_{w1}|x_{w2})$
  - Sort all the candidate words in ascending order of $H(x_{w1}|x_{w2})$
  - Take the top-ranked candidate words with some given threshold
- However
  - $H(x_{w1}|x_{w2})$ and $H(x_{w1}|x_{w3})$ are comparable
  - $H(x_{w1}|x_{w2})$ and $H(x_{w3}|x_{w2})$ are not
    → Because the upper bounds are different
- Conditional entropy is not symmetric
  - $H(x_{w1}|x_{w2}) \neq H(x_{w2}|x_{w1})$

Mutual Information

- $I(x;y) = H(x) - H(x|y) = H(y) - H(y|x)$

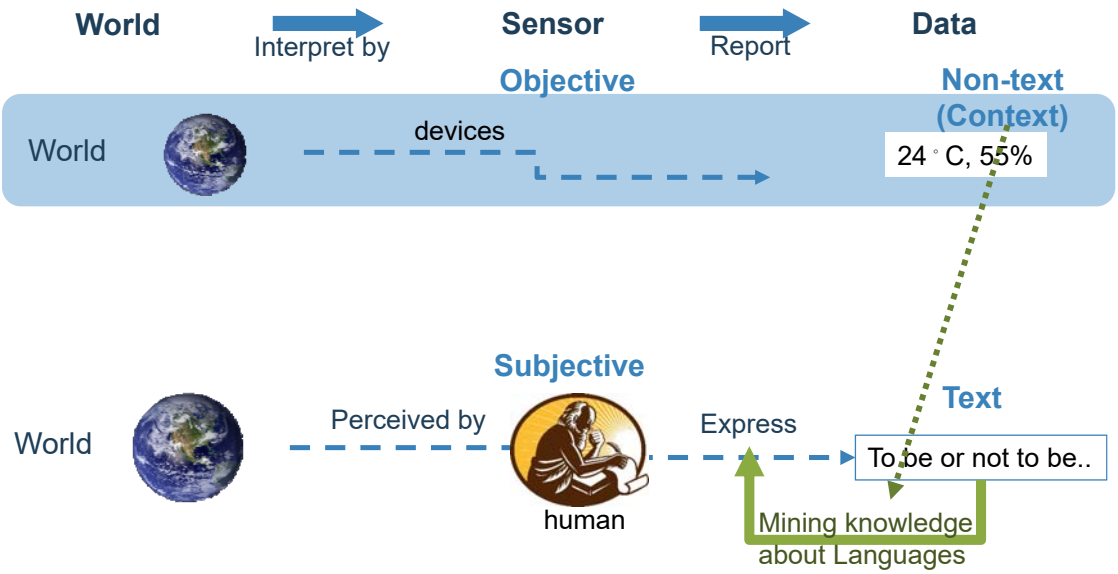


- Properties:
  - Symmetric
  - Non-negative
  - I(x;y)=0 iff x and y are independent
- Allow us to compare different (x,y) pairs

Topic Mining

Assume we already know the word relationships

Landscape of Text Mining

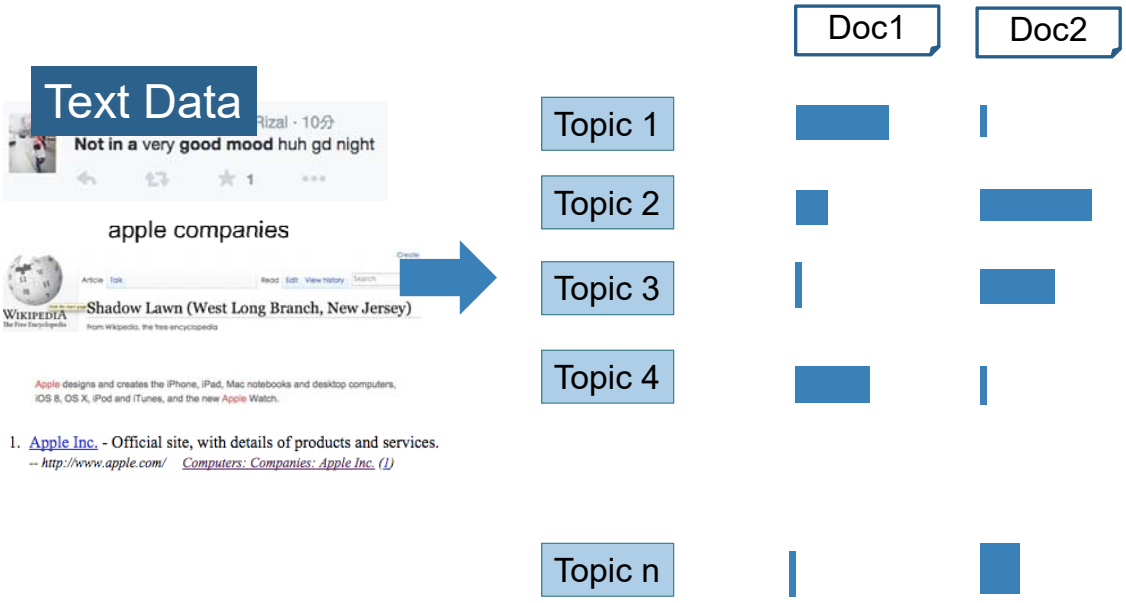


Nature Language Processing & Text Representation; Word Association and Mining

Topic Mining: Motivation

- Topic: key idea in text data
  - Theme/subject
  - Different granularities (e.g., sentence, article)
- Motivated applications, e.g.:
  - Hot topics during the debates in 2016 presidential election
  - What do people like about Windows 10
  - What are Facebook users talking about today?
  - What are the most watched news?

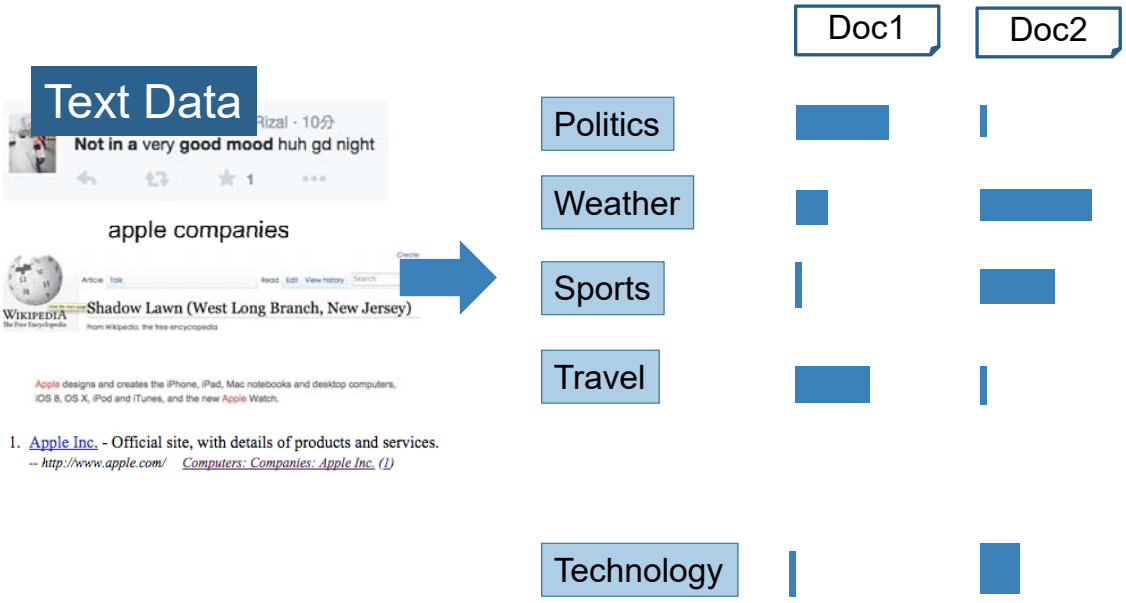
Tasks of Topic Mining



Formal Definition of Topic Mining

- Input
  - A collection of N text documents $S = \{d_1, d_2, d_3, \dots d_n\}$
  - Number of topics: *k*
- Output
  - k topics: $\{\theta_1, \theta_2, \theta_3, \dots \theta_n\}$
  - Coverage of topics in each d_i: $\{\mu_{i1}, \mu_{i2}, \mu_{i3}, \dots \mu_{in}\}$
- How to define topic θ_i?
  - Topic=term (word)?
  - Topic= classes?

Tasks of Topic Mining (Terms as Topics)



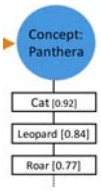
Problems with "Terms as Topics"

- Not generic
  - Can only represent simple/general topic
  - Cannot represent complicated topics
    - → E.g., "uber issue": political or transportation related?
- Incompleteness in coverage
  - Cannot capture variation of vocabulary
- Word sense ambiguity
  - E.g., Hollywood star vs. stars in the sky; apple watch vs. apple recipes

Improved Ideas

- Idea1 (Probabilistic topic models): topic = word distribution
  - E.g.: Sports = {(Sports, 0.2), (Game 0.01), (basketball 0.005), (play, 0.003), (NBA,0.01)…}
  - √: generic, easy to implement

- Idea 2 (Concept topic models): topic = concept
  - Maintain concepts (manually or automatically)
    - → E.g., ConceptNet



Possible Approaches for Probabilistic Topic Models

- Bag-of-words approach:
  - Mixture of unigram language model
  - Expectation-maximization algorithm
  - Probabilistic latent semantic analysis
  - Latent Dirichlet allocation (LDA) model
- Graph-based approach :
  - TextRank (Mihalcea and Tarau, 2004)
  - Reinforcement Approach (Xiaojun et al., 2007)
  - CollabRank (Xiaojun er al., 2008)

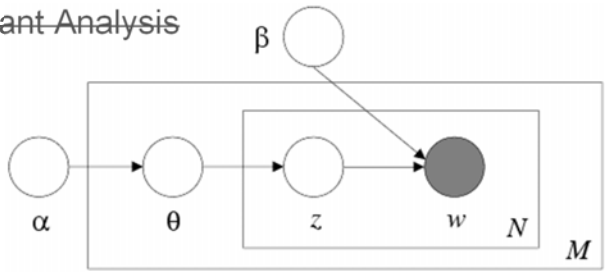
Bag-of-words Assumption

- Word order is *ignored*
- "bag-of-words" – exchangeability
- **Theorem (De Finetti, 1935)** – if $(x_1, x_2, \ldots, x_n)$ are infinitely exchangeable, then the joint probability $p(x_1, x_2, \ldots, x_n)$ has a representation as a mixture:

- $p(x_1, x_2, \ldots, x_n) = \int d\theta p(\theta) \prod_{i=1}^{N} p(x_i|\theta)$ for some random variable θ

Latent Dirichlet Allocation

- Latent Dirichlet Allocation (D. M. Blei, A. Y. Ng, 2003)
  ~~Linear Discriminant Analysis~~



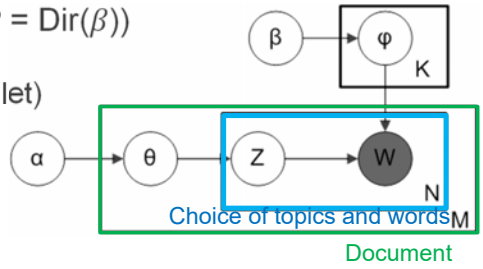
$$(2)\ p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\theta|\alpha)\prod_{n=1}^{N} p(z_n|\theta)p(w_n|z_n, \beta)$$

$$(3)\ p(\mathbf{w}|\alpha, \beta) = \int p(\theta|\alpha)\left(\prod_{n=1}^{N}\sum_{z_n} p(z_n|\theta)p(w_n|z_n, \beta)\right)d^k\theta$$

$$p(D|\alpha, \beta) = \prod_{d=1}^{M}\int p(\theta_d|\alpha)\left(\prod_{n=1}^{N_d}\sum_{z_{dn}} p(z_{dn}|\theta_d)p(w_{dn}|z_{dn}, \beta)\right)d^k\theta_d$$

LDA Assumption

- Assume:
  - When writing a document, you
    1. Decide how many words
    2. Decide distribution(P = Dir(α) P = Dir(β))
    3. Choose topics (Dirichlet)
    4. Choose words for topics (Dirichlet)
    5. Repeat 3



  - Example
    1. 5 words in document
    2. 50% food & 50% cute animals
    3. 1st word - food topic, gives you the word “bread”.
    4. 2nd word - cute animals topic, “adorable”.
    5. 3rd word - cute animals topic, “dog”.
    6. 4th word - food topic, “eating”.
    7. 5th word - food topic, “banana”.

“bread adorable dog eating banana”

LDA Learning (Gibbs)

- How many topics you think there are ?
- Randomly assign words to topics
- Check and update topic assignments (Iterative)
  - p(topic *t* | document *d*)
  - p(word *w* | topic *t*)
  - Reassign *w* a new topic, p(topic *t* | document *d*) * p(word *w* | topic *t*)

I eat fish and vegetables.
Dog and fish are pets.
My kitten eats fish.

#Topic: 2

p(red|1)=0.67; p(purple|1)=0.33; p(red|2)=0.50; p(purple|2)=0.50; p(red|3)=0.67; p(purple|3)=0.33;

p(eat|red)=0.20; p(eat|purple)=0.33; p(fish|red)=0.20; p(fish|purple)=0.33;
p(vegetable|red)=0.20; p(dog|purple)=0.33; p(pet|red)=0.20; p(kitten|red)=0.20;

p(purple|2)*p(fish|purple)=0.5*0.33=0.165; p(red|2)*p(fish|red)=0.5*0.2=0.1;

Related Work – Topic Model (LDA)

- I eat fish and vegetables.
- Dog and fish are pets.
- My kitten eats fish.

Topic 1	
0.268	fish
0.210	pet
0.210	dog
0.147	kitten

Sentence 1: 14.67% Topic 1, 85.33% Topic 2
Sentence 2: 85.44% Topic 1, 14.56% Topic 2
Sentence 3: 19.95% Topic 1, 80.05% Topic 2

Topic 2	
0.296	eat
0.265	fish
0.189	vegetable
0.121	kitten

Possible Approaches for Probabilistic Topic Models

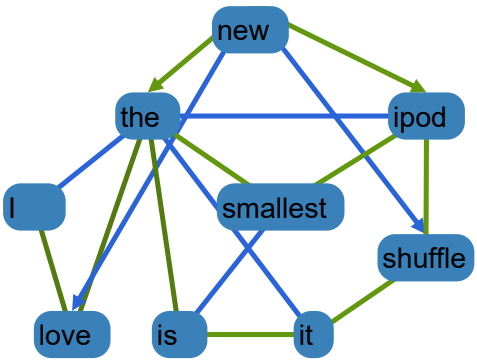
- Bag-of-words approach:
  - Mixture of unigram language model
  - Expectation-maximization algorithm
  - Probabilistic latent semantic analysis
  - Latent Dirichlet allocation (LDA) model
- Graph-based approach :
  - TextRank (Mihalcea and Tarau, 2004)
  - Reinforcement Approach (Xiaojun et al., 2007)
  - CollabRank (Xiaojun er al., 2008)

Construct Graph

- Directed graph
- Elements in the graph
  - → Terms
  - → Phrases
  - → Sentences

Connect Term Nodes

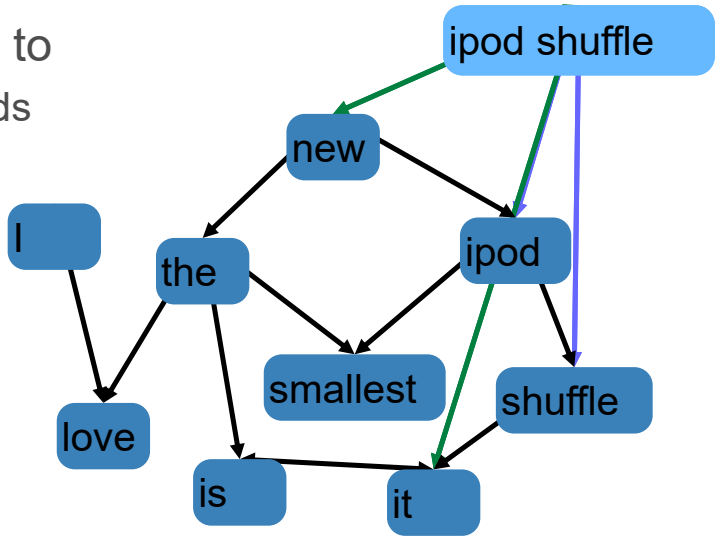
- Connect terms based on its slop.



Connect Phrase Nodes

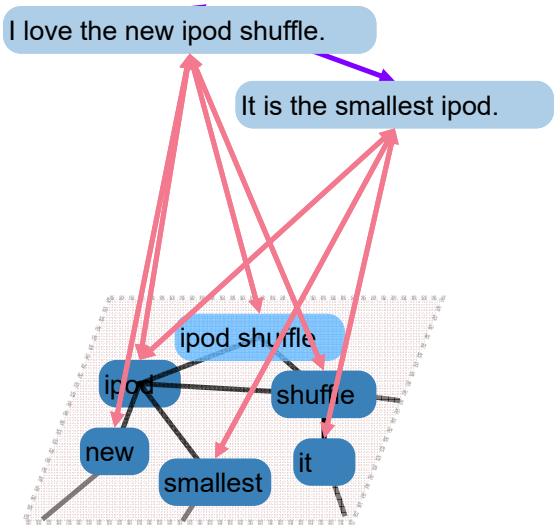
- Connect phrases to
  - Compound words
  - Neighbor words

I love the new ipod shuffle.
It is the smallest ipod.



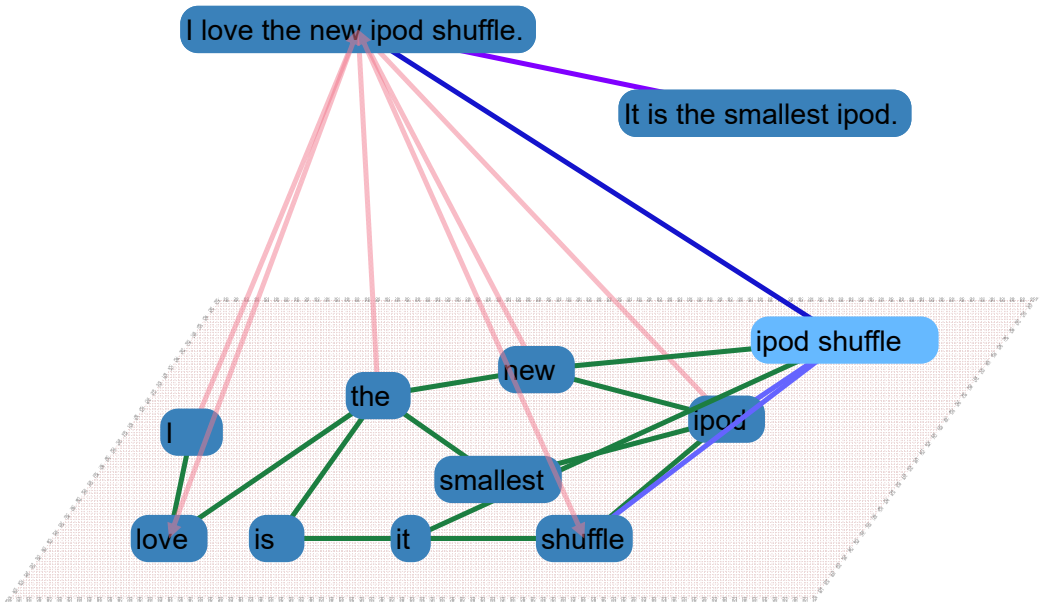
Connect Sentence Nodes

- Connect to
  - Neighbor sentences
  - Compound terms
  - Compound phrase



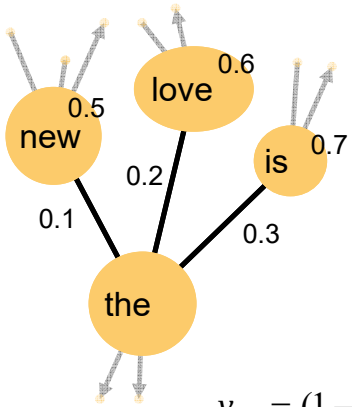
I love the new ipod shuffle.
It is the smallest ipod.

Edge Weight Types



Graph-base Ranking

- Scores for each node (TextRank 2004)



$$WS(V_i) = (1-d) + d \sum_{j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j)$$

damping factor

Score from parent nodes

$$y_{the} = (1-0.85) + 0.85 * (0.1 \times 0.5 + 0.2 \times 0.6 + 0.3 \times 0.7)$$

y_{new} y_{love} y_{is}

Result

the	1.45
ipod	1.21
new	1.02
is	1.00
shuffle	0.99
it	0.98
smallest	0.77
love	0.57

Graph-based Extraction

- Pros
  - → Structure and syntax information
  - → Mutual influence
- Cons
  - → Common words get higher scores

Summary: Probabilistic Topic Models

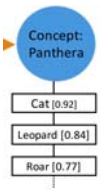
- Probabilistic topic models): topic = word distribution
  - E.g.: Sports = {(Sports, 0.2), (Game 0.01), (basketball 0.005), (play, 0.003), (NBA,0.01)…}
  - √: generic, easy to implement
  - ?: Not easy to understand/communicate
  - ?: Not easy to construct semantic relationship between topics

Topic= {(Crooked, 0.02), (dishonest, 0.001), (News, 0.0008), (totally, 0.0009), (total, 0.000009), (failed, 0.0006), (bad, 0.0015), (failing, 0.00001), (presidential, 0.0000008), (States, 0.0000004), (terrible, 0.0000085),(failed, 0.000021), (lightweight,0.00001),(weak, 0.0000075), ……}

Improved Ideas

- Idea1 (Probabilistic topic models): topic = word distribution
  - E.g.: Sports = {(Sports, 0.2), (Game 0.01), (basketball 0.005), (play, 0.003), (NBA,0.01)…}
  - √: generic, easy to implement

- Idea 2 (Concept topic models): topic = concept
  - Maintain concepts (manually or automatically)
    → E.g., ConceptNet



NLP Related Approach: Named Entity Recognition

- Find and classify all the named entities in a text.
- What's a named entity?
  - A mention of an entity using its name.
    - → *Kansas Jayhawks*
  - This is a subset of the possible mentions...
    - → *Kansas, Jayhawks, the team, it, they*
- Find means identify the exact span of the mention
- Classify means determine the category of the entity being referred to

Named Entity Recognition Approaches

- As with partial parsing and chunking there are two basic approaches (and hybrids)
  - Rule-based (regular expressions)
    - → Lists of names
    - → Patterns to match things that look like names
    - → Patterns to match the environments that classes of names tend to occur in.
  - Machine Learning-based approaches
    - → Get annotated training data
    - → Extract features
    - → Train systems to replicate the annotation

Rule-Based Approaches

- Employ regular expressions to extract data
- Examples:
  - Telephone number: (\d{3}[-.\ ()]){1,2}[\dA-Z]{4}.
    - → 800-865-1125
    - → 800.865.1125
    - → (800)865-CARE
  - Software name extraction: ([A-Z][a-z]*\s*)+
    - → Installation Designer v1.1

Relations

- Once you have captured the entities in a text you might want to ascertain how they relate to one another.
  - Here we're just talking about explicitly stated relations

Relation Types

- As with named entities, the list of relations is application specific. For generic news texts...

Relations		Examples	Types
Affiliations			
	Personal	*married to, mother of*	PER → PER
	Organizational	*spokesman for, president of*	PER → ORG
	Artifactual	*owns, invented, produces*	(PER | ORG) → ART
Geospatial			
	Proximity	*near, on outskirts*	LOC → LOC
	Directional	*southeast of*	LOC → LOC
Part-Of			
	Organizational	*a unit of, parent of*	ORG → ORG
	Political	*annexed, acquired*	GPE → GPE

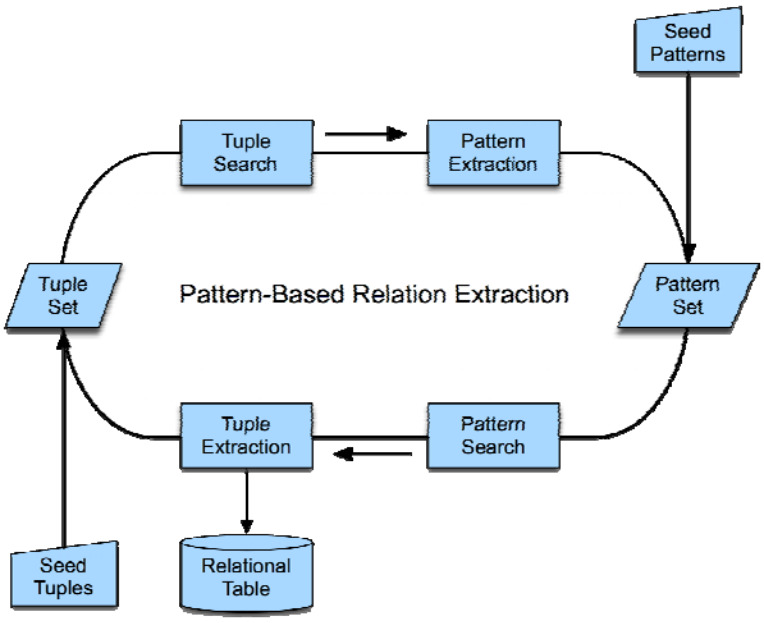
Bootstrapping Approaches

- What if you don't have enough annotated text to train on.
  - But you might have some seed tuples
  - Or you might have some patterns that work pretty well
- Can you use those seeds to do something useful?
  - Co-training and active learning use the seeds to train classifiers to tag more data to train better classifiers...
  - Bootstrapping tries to learn directly (populate a relation) through direct use of the seeds

Bootstrapping Example: Seed Tuple

- <Mark Twain, Elmira> Seed tuple
  - Grep (google)
  - "Mark Twain is buried in Elmira, NY."
    → X is buried in Y
  - "The grave of Mark Twain is in Elmira"
    → The grave of X is in Y
  - "Elmira is Mark Twain's final resting place"
    → Y is X's final resting place.
- Use those patterns to grep for new tuples that you don't already know

Bootstrapping Relations



Wikipedia Infobox

- Infoboxes are kept in a namespace separate from articles
  - Namespce example: Special:SpecialPages; Wikipedia:List of infoboxes
  - Example:
  -

```
{{Infobox person
 |name = Casanova
 |image = Casanova_self_portrait.jpg
 |caption = A self portrait of Casanova
          ...
|website = }}
```


Concept-based Model

- ESA (Egozi, Markovitch, 2011)
  Every Wikipedia article represents a concept

Manually-maintained knowledge base

Yago

- YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia, WWW 2007
- Unification of Wikipedia & WordNet
- Make use of rich structures and information
  - Infoboxes, Category Pages, etc.

Type	Private
Endowment	US$ 16.5 billion (2011)[3]
President	John L. Hennessy
Provost	John Etchemendy
Academic staff	1,910[4]
Students	15,319
Undergraduates	6,878[5]
Postgraduates	8,441[5]
Location	Stanford, California, U.S.
Campus	Suburban, 8,180 acres (3,310 ha)[6]
Colors	Cardinal red and white

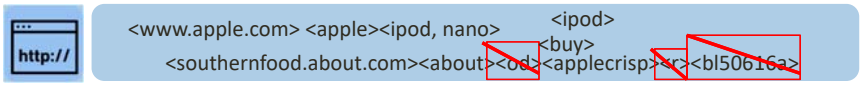
Mining Concepts from User-Generated Web Data

- Concept: in a word sequence, its meaning is known by a group of people and everyone within the group refers to the same meaning

Pei-Ling Hsu, Hsiao-Shan Hsieh, Jheng-He Liang, and Yi-Shin Chen*, Mining Various Semantic Relationships from Unstructured User-Generated Web Data, Journal of Web Semantics, 2015

Concepts in Word Sequences

- Word sequences are likely meaningless and noise



→ A word sequence as a candidate to be a concept

→ About noun
- A noun — e.g., Basketball
- A sequence of noun — e.g., Christmas Eve
- A noun and a number — e.g., 311 earthquake
- An adjective and a noun — e.g., Desperate housewives

→ Special format
- A word sequence contains "of" — e.g., Cover of harry potter

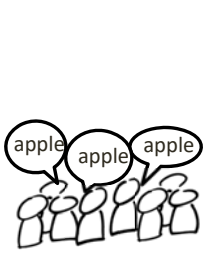
Concept Modeling

- Concept: in a word sequence, its meaning is known by a group of people and everyone within the group refers to the same meaning.

→ Try to detect a word sequence that is known by a group of people.

Concept Modeling

- Frequent Concept:
  - A word sequence mentioned frequently from a single source.



Mentioned number is normalized in each data source

$$b^{ws} = \frac{COUNT(ws)}{max_{ws_i \in \{S \cap CN\}} COUNT(ws_i)}, \forall ws \in \mathbb{C}_S^f, b^{ws} > \rho_S^f$$

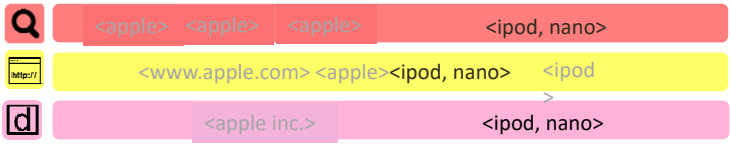
<apple> <apple> <apple> <apple, <color, of, ipod, nano, 1gb> companies> <ipod> <apple> <apple> <ipod, nano> <apple>

Concept Modeling

- Some data sources are public, sharing data with other users.
- These data sources with frequently seen word sequences.
  - → These data sources provide concepts with higher confidence.

- Confident Value: every data source has a confident value
- Confident concept: a word sequence is from data sources with higher confident values.

$$ac_{ws} = \sum_{S_k \in SS,\ ws \in (S_k \cap CN)} (cv_k), \forall ws \in \mathbb{C}^c, ac_{ws} > \rho^c.$$

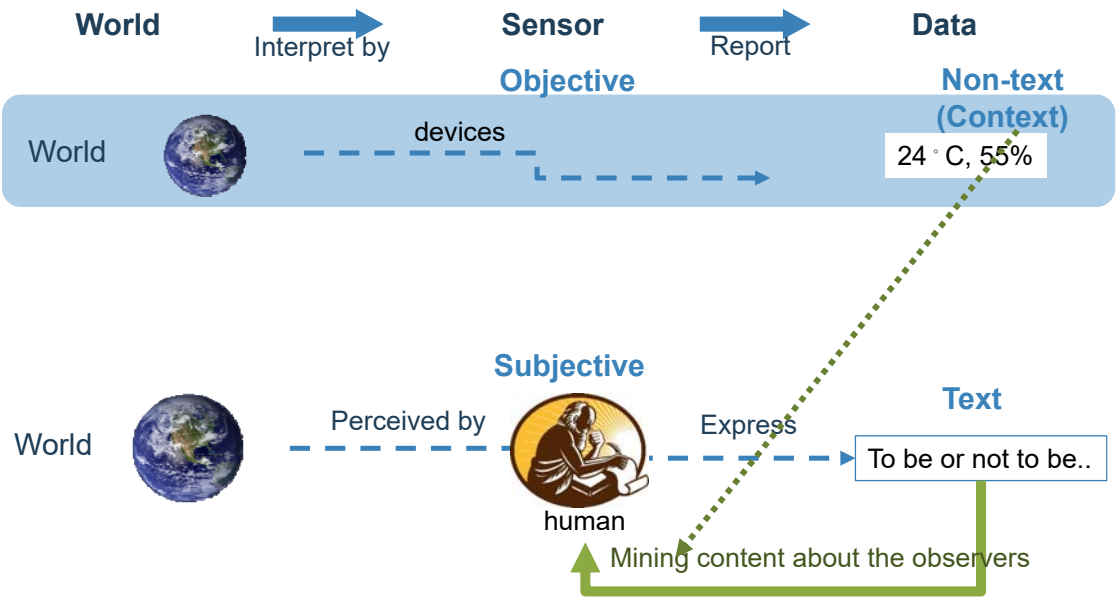


<apple inc.> <ipod, nano>

Opinion Mining

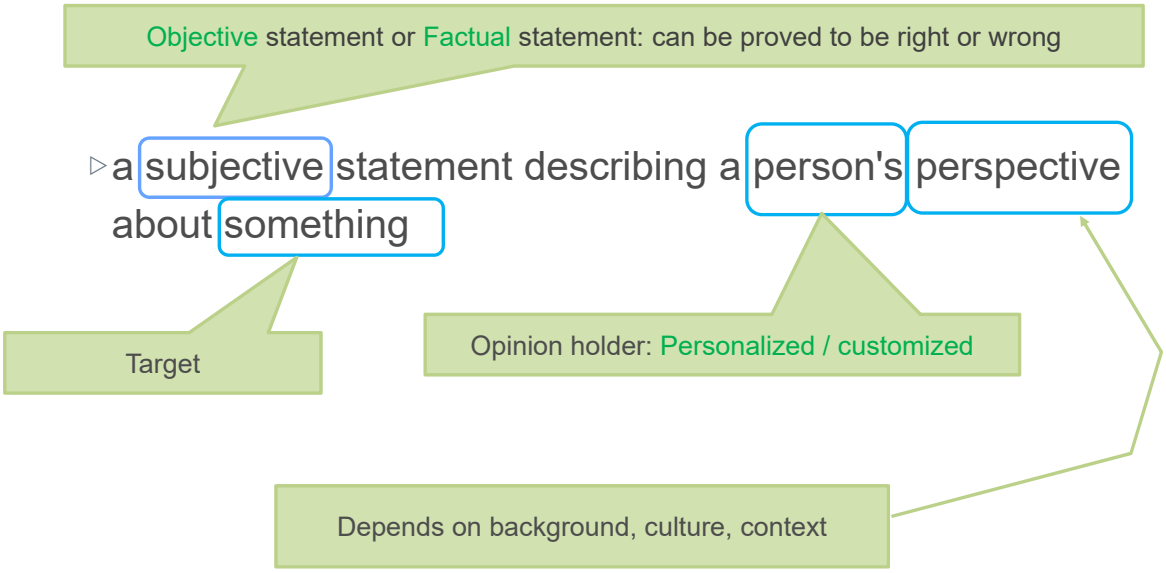
How people feel?

Landscape of Text Mining



Opinion Mining and Sentiment Analysis

Opinion



Opinion Representation

- Opinion holder: user
- Opinion target: object
- Opinion content: keywords?
- Opinion context: time, location, others?
- Opinion sentiment (emotion): positive/negative, happy or sad

Sentiment Analysis

- Input: An opinionated text object
- Output: Sentiment tag/Emotion label
  - Polarity analysis: {positive, negative, neutral}
  - Emotion analysis: happy, sad, anger
- Naive approach:
  - Apply classification, clustering for extracted text features

Text Features

- Character *n*-grams
  - Usually for spelling/recognition proof
  - Less meaningful
- Word *n*-grams
  - n should be bigger than 1 for sentiment analysis
- POS tag *n*-grams
  - Can mixed with words and POS tags
    → E.g., “adj noun”, “sad noun”

More Text Features

- Word classes
  - Thesaurus: LIWC
  - Ontology: WordNet, Yago, DBPedia
  - Recognized entities: DBPedia, Yago
- Frequent patterns in text
  - Could utilize pattern discovery algorithms
  - Optimizing the tradeoff between *coverage* and *specificity* is essential

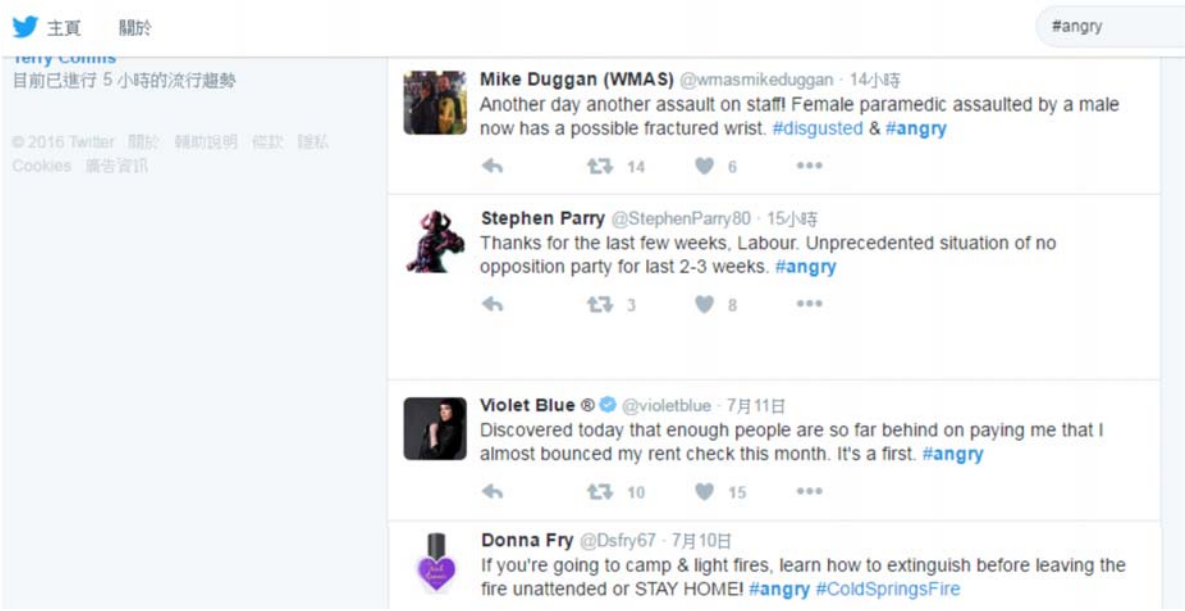
LIWC

- Linguistic Inquiry and word count
  - LIWC2015
- Home page: http://liwc.wpengine.com/
- >70 classes
- Developed by researchers with interests in social, clinical, health, and cognitive psychology
- Cost: US$89.95

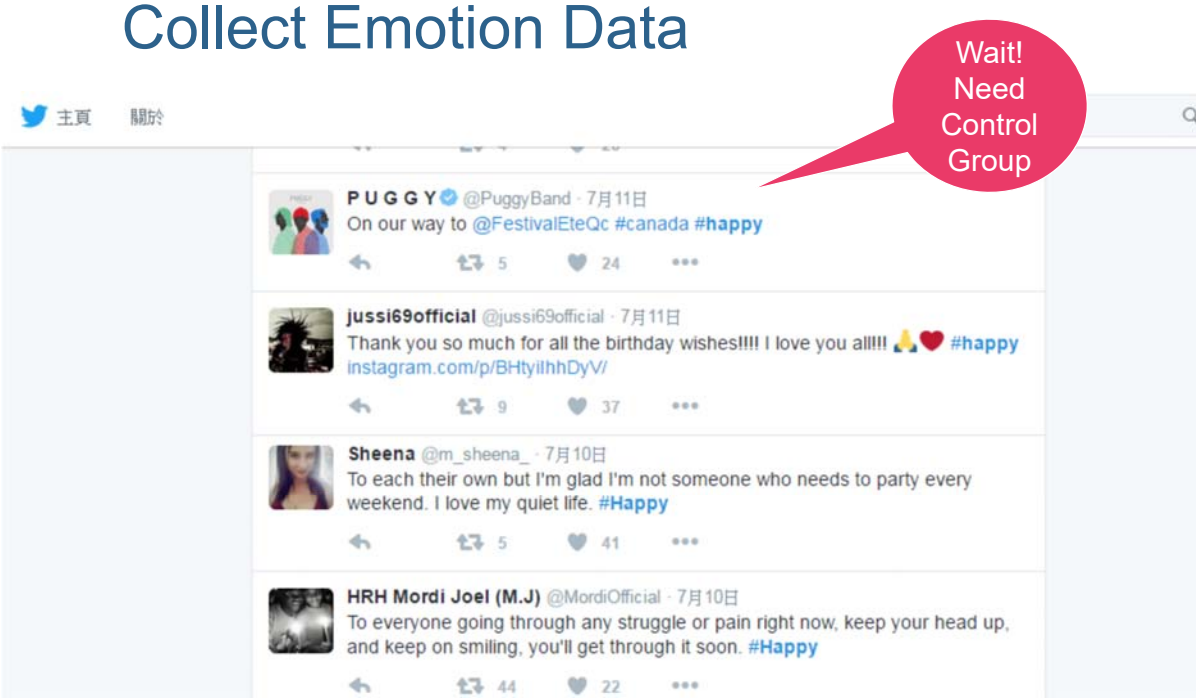
Emotion Analysis: Pattern Approach

- Carlos Argueta, Fernando Calderon, and Yi-Shin Chen, Multilingual Emotion Classifier using Unsupervised Pattern Extraction from Microblog Data, Intelligent Data Analysis - An International Journal, 2016

Collect Emotion Data



Collect Emotion Data



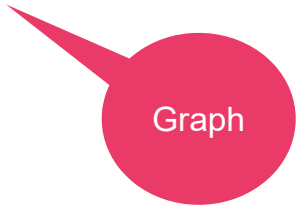
Not-Emotion Data

Preprocessing Steps

- Hints: Remove troublesome ones
  - Too short
    - → Too short to get important features
  - Contain too many hashtags
    - → Too much information to process
  - Are retweets
    - → Increase the complexity
  - Have URLs
    - → Too trouble to collect the page data
  - Convert user mentions to <usermention> and hashtags to <hashtag>
    - → Remove the identification. We should not peek answers!

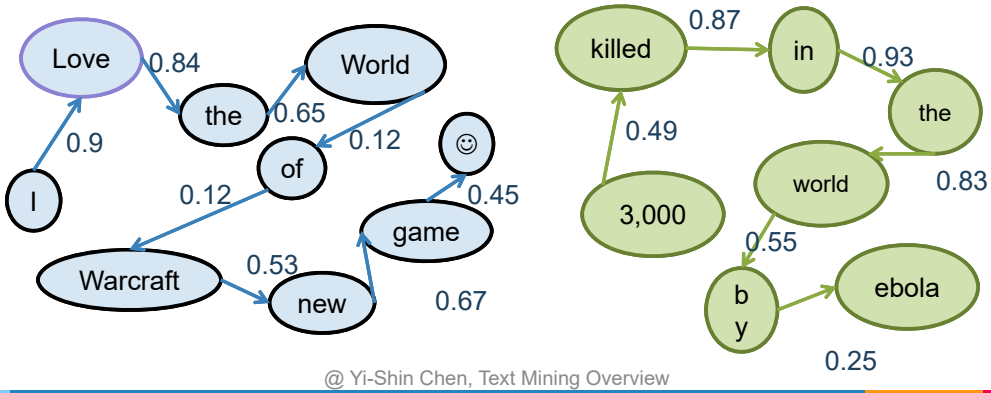
Basic Guidelines

- Identify the common and differences between the experimental and control groups
  - Analyze the frequency of words
    - → TF•IDF (Term frequency, inverse document frequency)
  - Analyze the co-occurrence between words/patterns
    - → Co-occurrence
  - Analyze the importance between words
    - → Centrality



Graph Construction

- Construct two graphs
  - E.g.
    - → Emotion one: I love the World of Warcraft new game ☺
    - → Not-emotion one: 3,000 killed in the world by ebola



Graph Processes

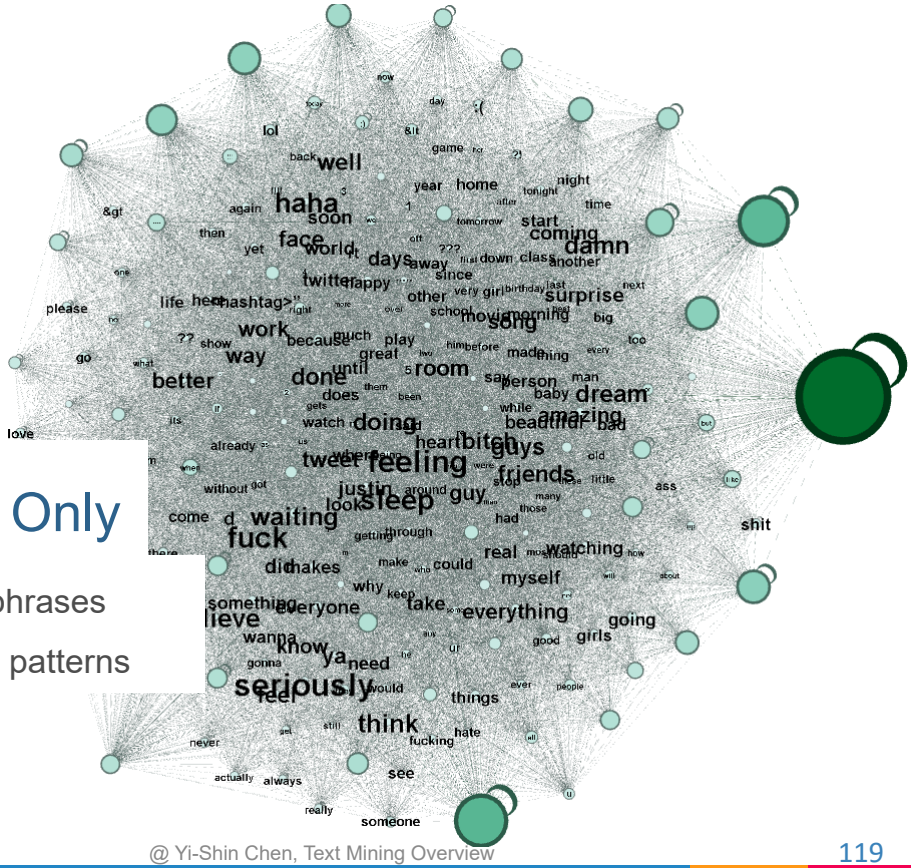
- Remove the common ones between two graphs
  - Leave the significant ones only appear in the emotion graph
- Analyze the centrality of words
  - Betweenness, Closeness, Eigenvector, Degree, Katz
    - → Can use the free/open software, e.g, Gaphi, GraphDB
- Analyze the cluster degrees
  - Clustering Coefficient

Key patterns

Essence Only

Only key phrases

→emotion patterns



Emotion Patterns Extraction

- The goal:
  - Language independent extraction – not based on grammar or manual templates
  - More representative set of features - balance between generality and specificity
  - High recall/coverage – adapt to unseen words
  - Requiring only a relatively small number – high reliability
  - Efficient— fast extraction and utilization
  - Meaningful - even if there are no recognizable emotion words in it

Patterns Definition

- Constructed from two types of elements:
  - Surface tokens: hello, ☺, lol, house, …
  - Wildcards: * (matches every word)
- Contains at least 2 elements
- Contains at least one of each type of element

Examples:

Pattern	Matches
* this *	“Hate this weather”, “love this drawing”
* * ☺	“so happy ☺”, “to me ☺”
luv my *	“luv my gift”, “luv my price”
* that	“want that”, “love that”, “hate that”

Patterns Construction

- Constructed from instances
- An instance is a sequence of 2 or more words from CW and SW
- Contains at least one CW and one SW

Examples

Connector Words
this
luv
my
☺
…

SubjectWords
love
hate
gift
weather
…

Instances
“hate this weather”
“so happy ☺”
“luv my gift”
“love this drawing”
“luv my price”
“to me ☺
“kill this idiot”
“finish this task”

Patterns Construction (2)

Connector Words
this
luv
my
☺
…

- Find all instances in a corpus with their freq
- Aggregate counts by grouping them based on length and position of matching CW

Instances	Count
“hate this weather”	5
“so happy ☺”	4
“luv my gift”	7
“love this drawing”	2
“luv my price”	1
“to me ☺”	3
“kill this idiot”	1
“finish this task”	4

Groups	Count
“Hate this weather”, “love this drawing”, “kill this idiot”, “finish this task”	12
“so happy ☺”, “to me ☺”	7
“luv my gift”, “luv my price”	8
…	…

Patterns Construction (3)

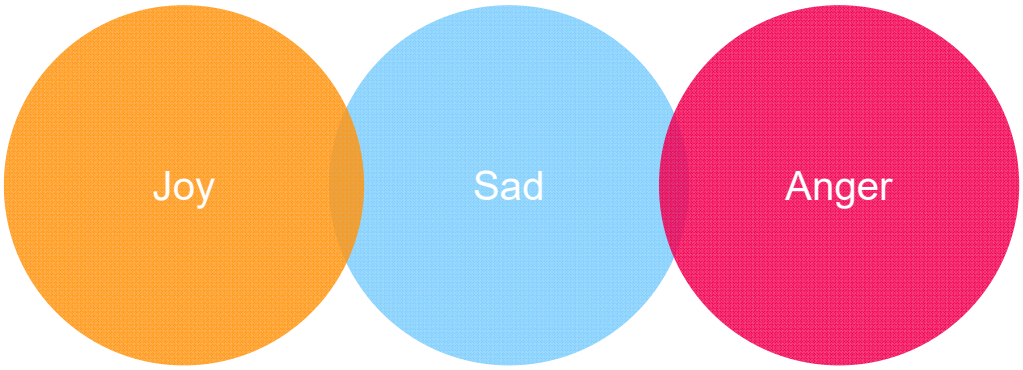
Connector Words
this
got
my
pain
…

- Replace all the SWs by a wildcard * and keep the CWs to convert all instances into the representing pattern

Pattern	Groups	Count
* this *	“Hate this weather”, “love this drawing”, “kill this idiot”, “finish this task”	12
* * ☺	“so happy ☺”, “to me ☺”	7
luv my *	“luv my gift”, “luv my price”	8
…	…	…

- The wildcard matches any word and is used for term generalization
- Infrequent patterns are filtered out

Ranking Emotion Patterns



- Ranking the emotion patterns for each emotion
  - Frequency, exclusiveness, diversity
  - One ranked list for each emotion

Contextual Text Mining

Basic Concepts

Context

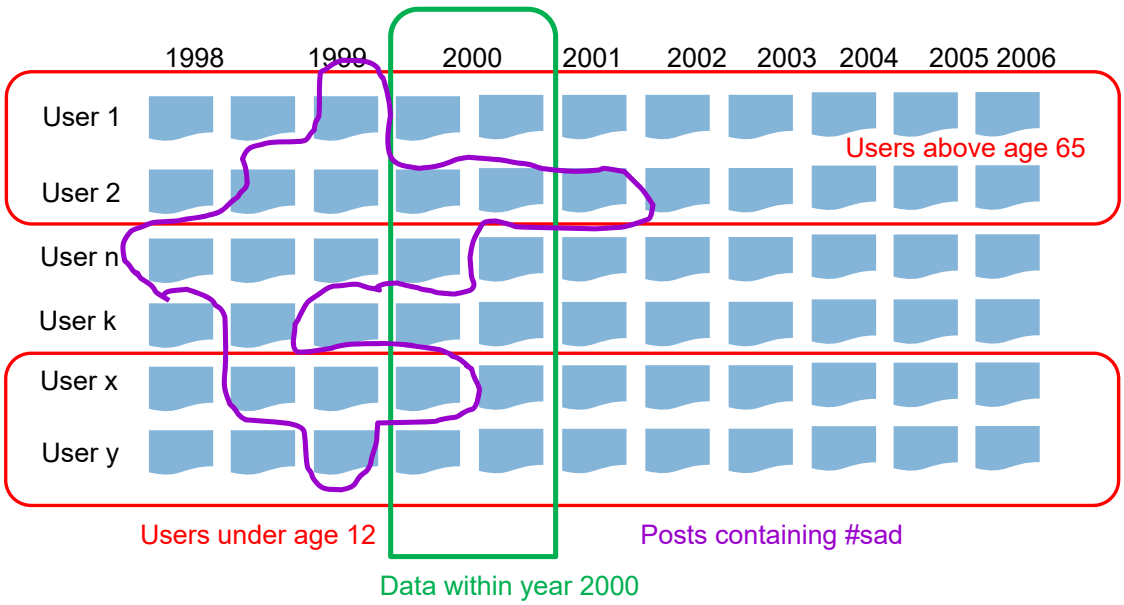
- Text usually has rich context information
  - Direct context (meta-data): time, location, author
  - Indirect context: social networks of authors, other text related to the same source
  - Any other related text
- Context could be used for:
  - Partition the data
  - Provide extra features

Contextual Text Mining

- Query log + User = Personalized search
- Tweet + Time = Event identification
- Tweet + Location-related patterns = Location identification
- Tweet + Sentiment = Opinion mining

- Text Mining +Context → Contextual Text Mining

Partition Text



Generative Model of Text

$P(word \mid Topic)$

$P(Topic \mid Document)$

Generation

Dog eat and fish kitten pets I are My vegetables

I eat fish and vegetables. Dog and fish are pets. My kitten eats fish.

Analyze Model

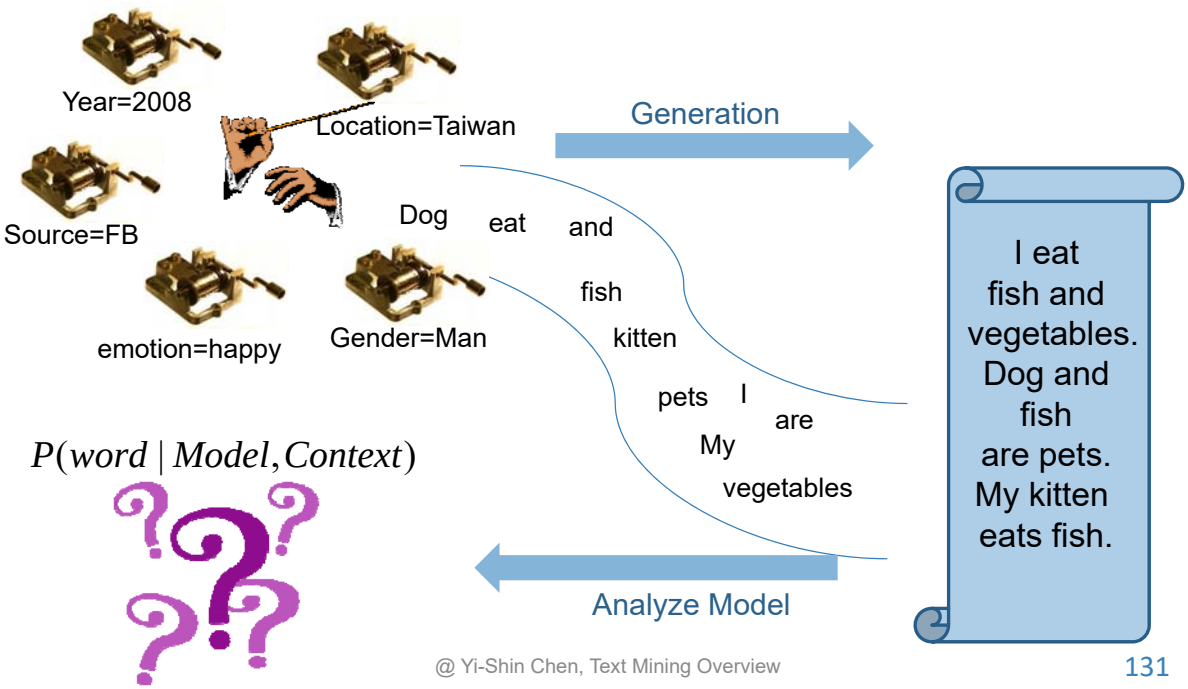
Topic 1

0.268	f
0.210	p
0.210	c
0.147	k

Topic 2

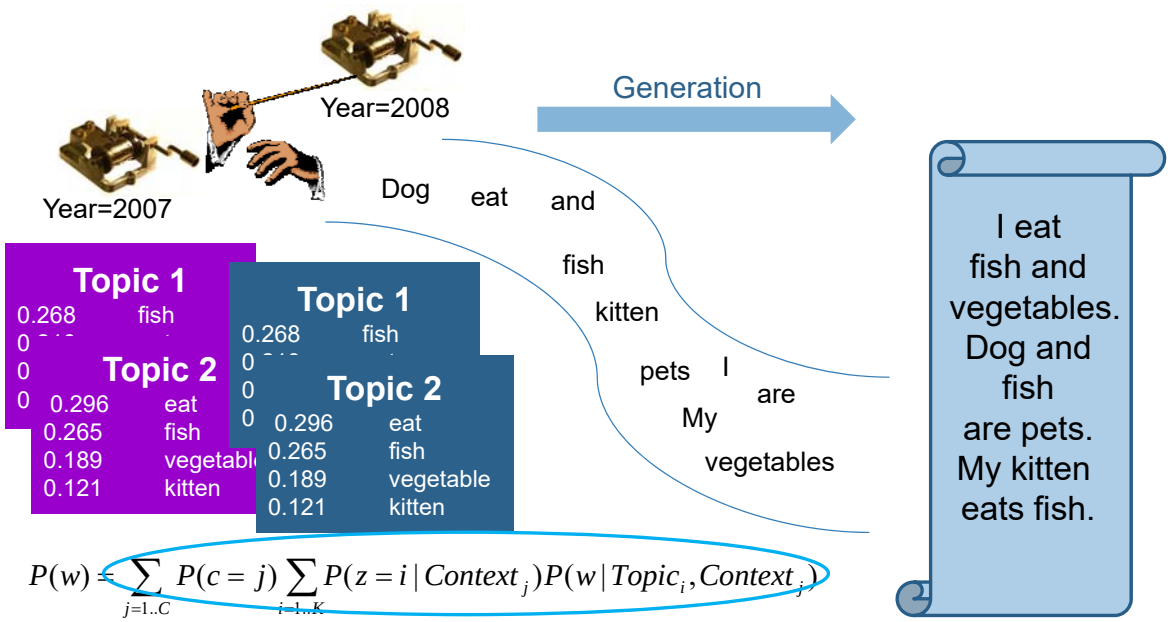
0.296	eat
0.265	fish
0.189	vegetable
0.121	kitten

Contextualized Models of Text



$$P(word \mid Model, Context)$$

Naïve Contextual Topic Model



$$P(w) = \sum_{j=1..C} P(c=j) \sum_{i=1..K} P(z=i \mid Context_j) P(w \mid Topic_i, Context_j)$$

How do we estimate it? → Different approaches for different contextual data and problems

Contextual Probabilistic Latent Semantic Analysis (CPLAS) (Mei, Zhai, KDD2006)

- An extension of PLSA model ([Hofmann 99]) by
  - Introducing context variables
  - Modeling views of topics
  - Modeling coverage variations of topics
- Process of contextual text mining
  - Instantiation of CPLSA (*context*, *views*, *coverage*)
  - Fit the model to text data (EM algorithm)
  - Compare a topic from different views
  - Compute strength dynamics of topics from coverages
  - Compute other probabilistic topic patterns

The Probabilistic Model

- A probabilistic model explaining the generation of a document D and its context features C: if an author wants to write such a document, he will
  - Choose a view v_i according to the view distribution $p(v_i|D,C)$
  - Choose a coverage κ_j according to the coverage distribution $p(k_j|D,C)$
  - Choose a theme θ_{il} according to the coverage κ_j
  - Generate a word using θ_{il}
  - The likelihood of the document collection is:

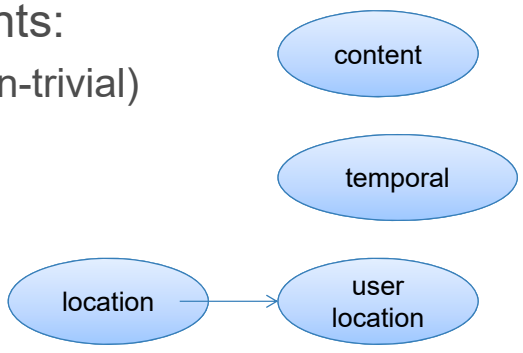
$$\log p(\mathbb{D}) = \sum_{(D,C)\in\mathbb{D}} \sum_{w\in V} c(w,D) \log\left(\sum_{i=1}^{n} p(v_i \mid D,C) \sum_{j=1}^{m} p(\kappa_j \mid D,C) \sum_{l=1}^{k} p(l \mid \kappa_j)\, p(w \mid \theta_{il})\right)$$

Contextual Text Mining Example 1

Event Identification

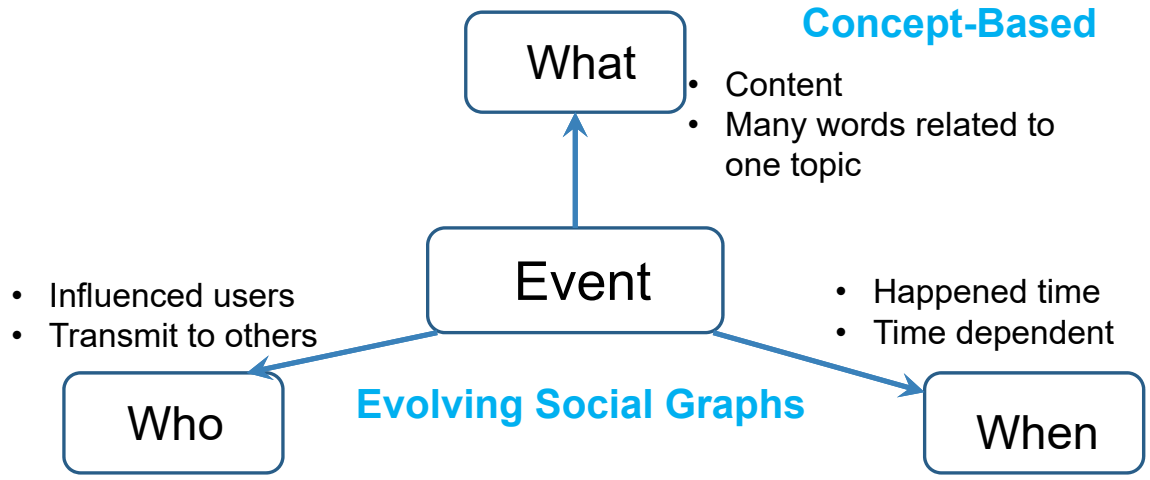
Introduction

- Event definition:
  - Something (non-trivial) happening in a certain place at a certain time (Yang et al. 1998)
- Features of events:
  - Something (non-trivial)
  - Certain time
  - Certain place



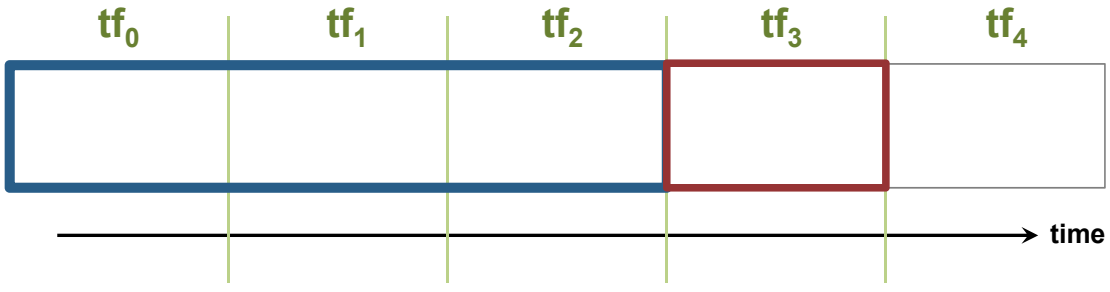
Goal

- Identify events from social streams
- Events contain these characteristics



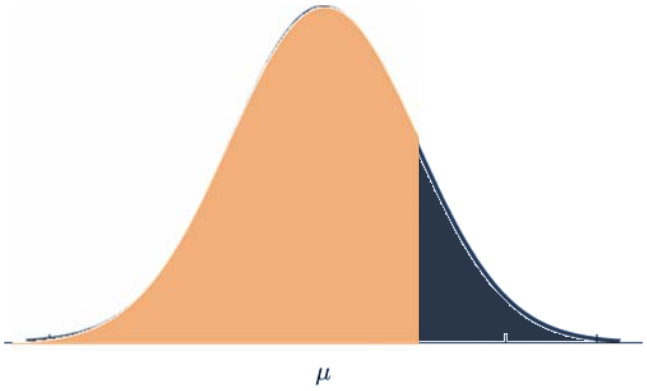
Keyword Selection

- Well-noticed criterion
  - Compared to the past, if a word suddenly be mentioned by many users, it is well-noticed
  - Time Frame – a unit of time period
  - Sliding Window – a certain number of past time frames



Keyword Selection

- Compare the probability proportion of this word count with the past sliding window

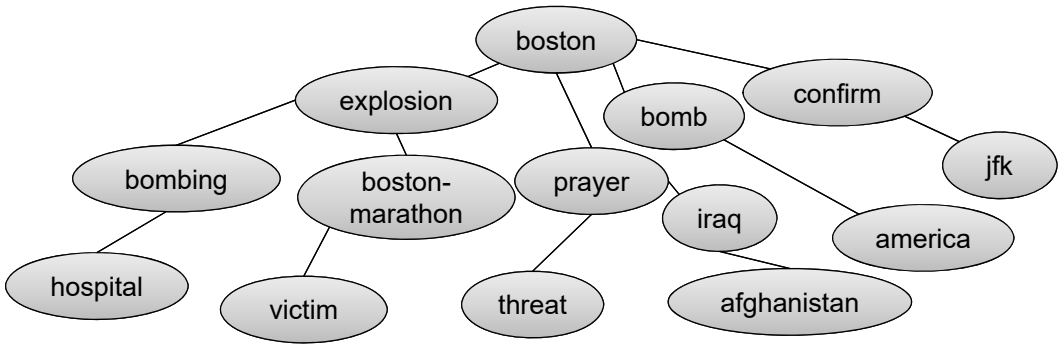


$$Z_{tf_j,s}(w_i) = \frac{\kappa_{tf_j}(w_i) - AVG_{tf_j,s}(w_i)}{SD_{tf_j,s}(w_i)}$$

$$Prob(Z \leq Z_{tf_j,s}(w_i)) > \delta_{category}$$

Event Candidate Recognition

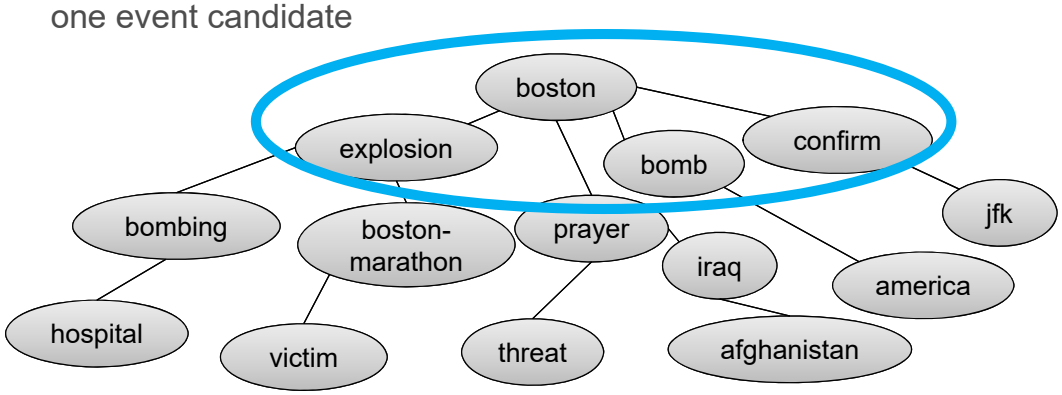
- Concept-based event: all keywords about the same event
  - Use the co-occurrence of words in tweets to connect keywords



Huge amount of keywords connected as one event

Event Candidate Recognition

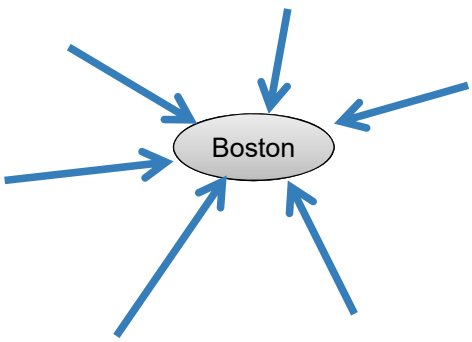
- Idea: group one keyword with its most relevant keywords into one event candidate



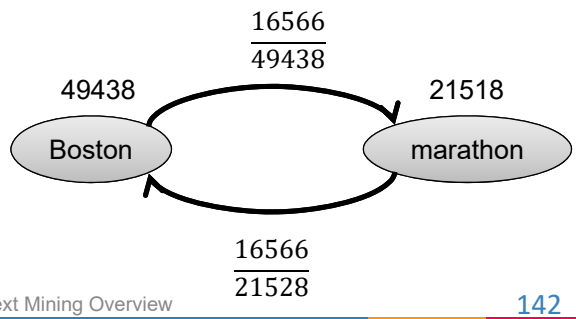
- How do we decide which keyword to start grouping?

Event Candidate Recognition

- TextRank: rank importance of each keyword
  - Number of edges
  - Edge weights (word count and word co-occurrence)



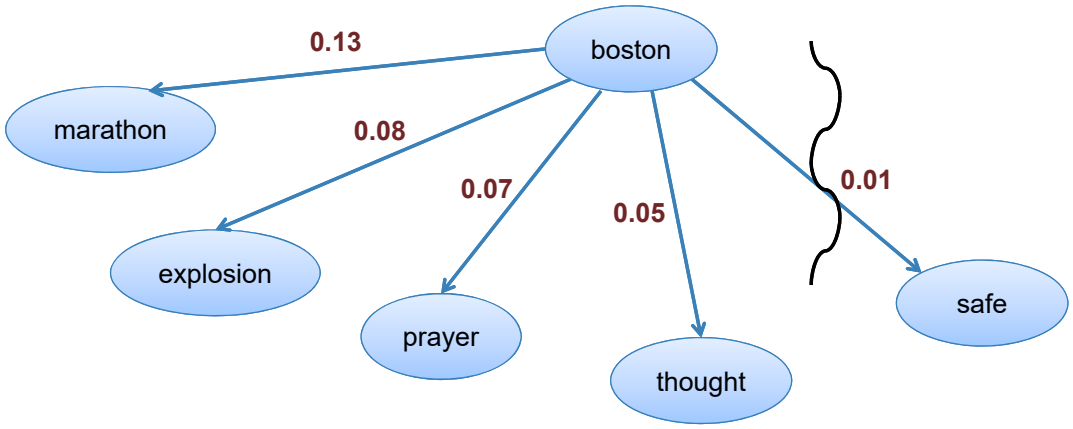
$$weight = \frac{cooccur(k1, k2)}{count(k1)}$$



Event Candidate Recognition

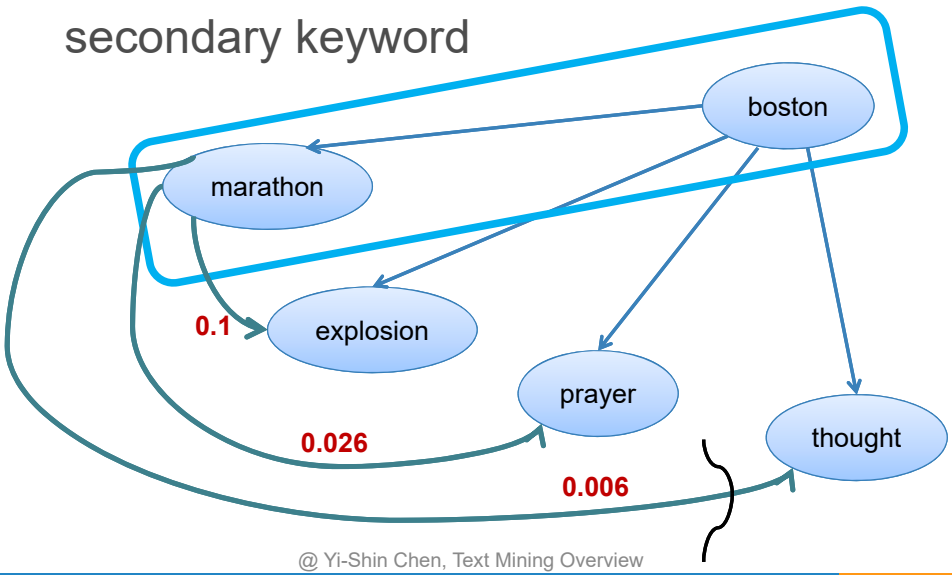
1. Pick the most relevant neighbor keywords

Normalized weight



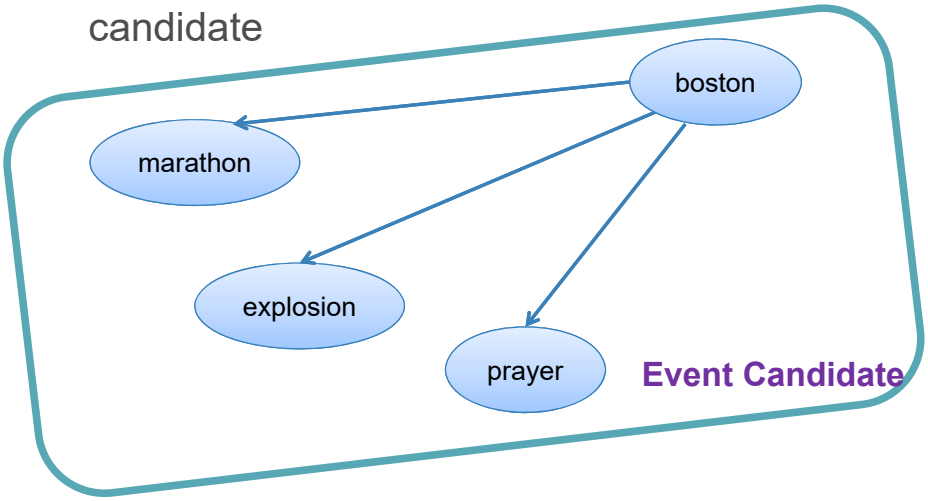
Event Candidate Recognition

2. Check neighbor keywords with the secondary keyword



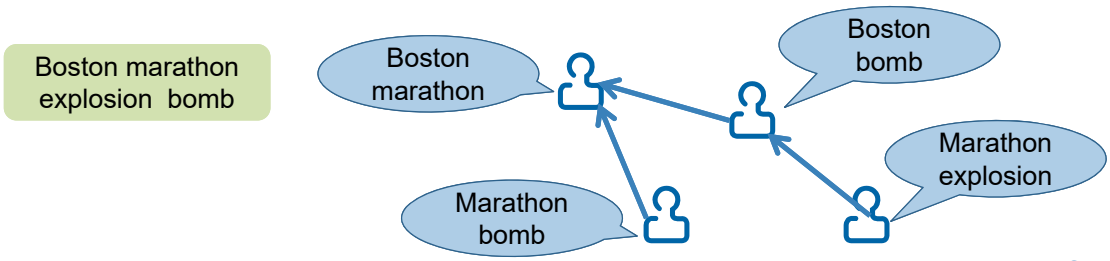
Event Candidate Recognition

3. Group the remaining keywords as one event candidate



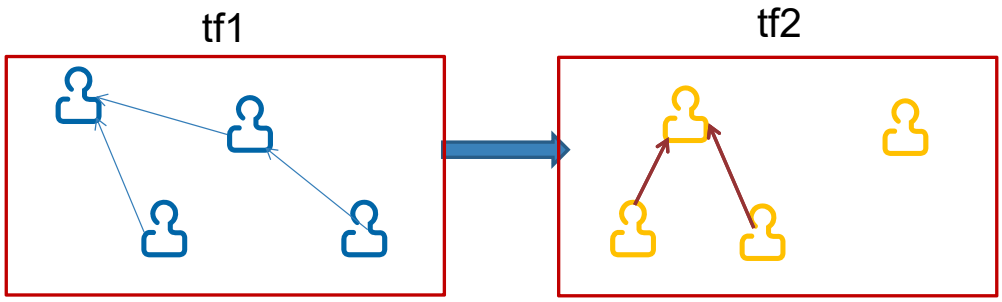
Evolving Social Graph Analysis

- Bring in social relationships to estimate information propagation
- Social Relation Graph
  - Vertex: users that mentioned one or more keywords in the candidate event
  - Edge: the following relationship between users



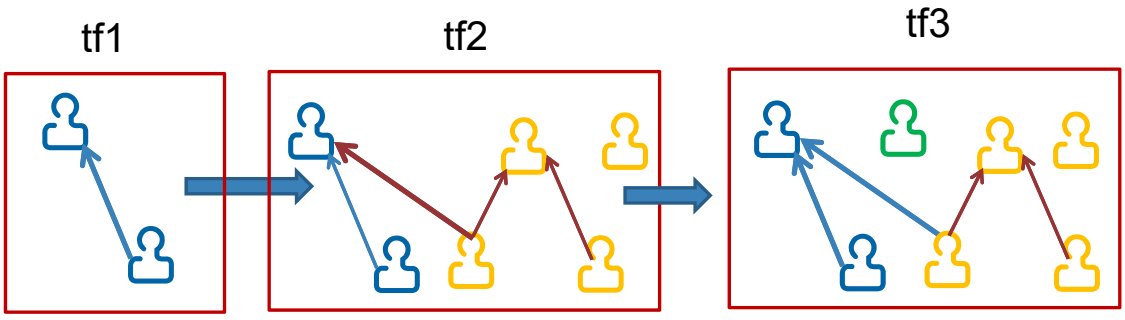
Evolving Social Graph Analysis

- Add in evolving idea (time frame increment): graph sequences



Evolving Social Graph Analysis

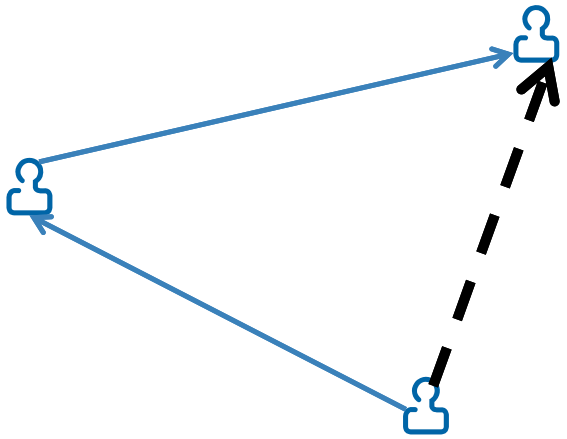
- Information decay:
  - Vertex weight, edge weight
  - Decay mechanism



- Concept-Based Evolving Graph Sequences (cEGS): a sequence of directed graphs that demonstrate information propagation

Methodology – Evolving Social Graph Analysis

- Hidden link – construct hidden relationship
  - To model better interaction between users
  - Sample data



Evolving Social Graph Analysis

- Analysis of cEGS
  - Number of vertices(nV)
    - → The number of users mentioned this event candidate
  - Number of edges(nE):
    - → The number of following relationship in this cEGS
  - Number of connected components(nC)
    - → The number of communities in this cEGS
  - Reciprocity(R)
    - → The degree of mutual connections is in this cEGS

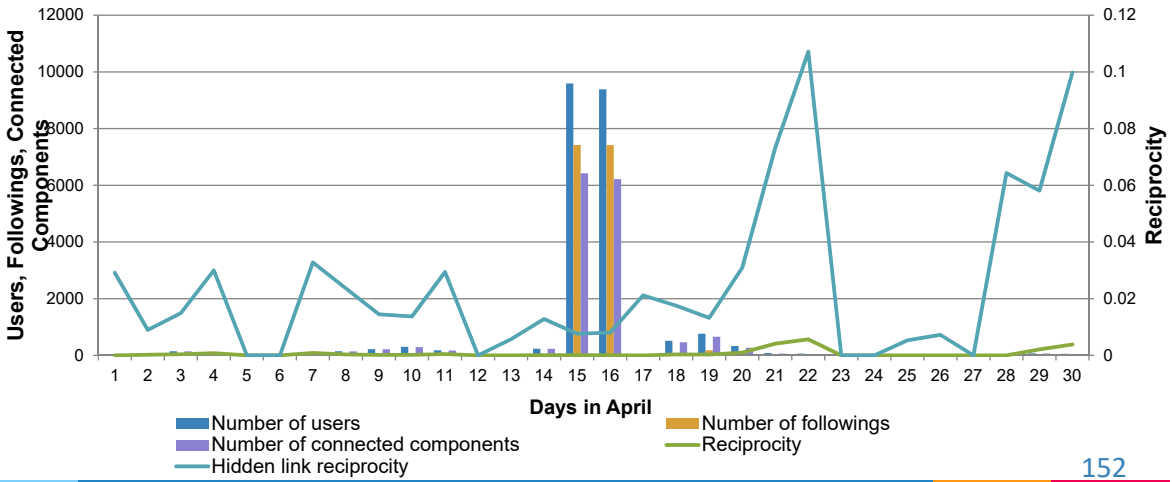
$$\text{Reciprocity} = \frac{number\ of\ edges}{number\ of\ possible\ edges}$$

Event Identification

- Type1 Event: One-shot event
  - An event that receives attention in a short period of time
    - → Number of users, number of followings, number of connected components suddenly increase
- Type2 Event: Long-run event
  - An event that attracts many discussion for a period of time
    - → Reciprocity
- Non-event

Experimental Results

- April 15, *"library jfk blast bostonmarathon prayforboston pd possible area boston police social explosion bomb bombing marathon confirm incident"*



Contextual Text Mining Example 2

Location Identification

Twitter and Geo-Tagging

- Geospatial features in Twitter have been available
  - User's Profile location
  - Per tweet geo-tagging
- Users have been slow to adopt these features
  - On a 1-million-record sample, 0.30% of tweets were geo-tagged
- Locations are not specific

Our Goal

- Identify the location of a particular Twitter user at a given time
  - Using exclusively the content of his/her tweets

Data Sources

- Twitter Dataset
  - 13 million tweets, 1.53 million profiles
  - From Nov. '11 to Apr. '12
- Local Cache
  - Geospatial Resources
  - GeoNames Spatial DB
  - Wikiplaces DB
  - Lexical Resources
  - WordNet
- Web Resources

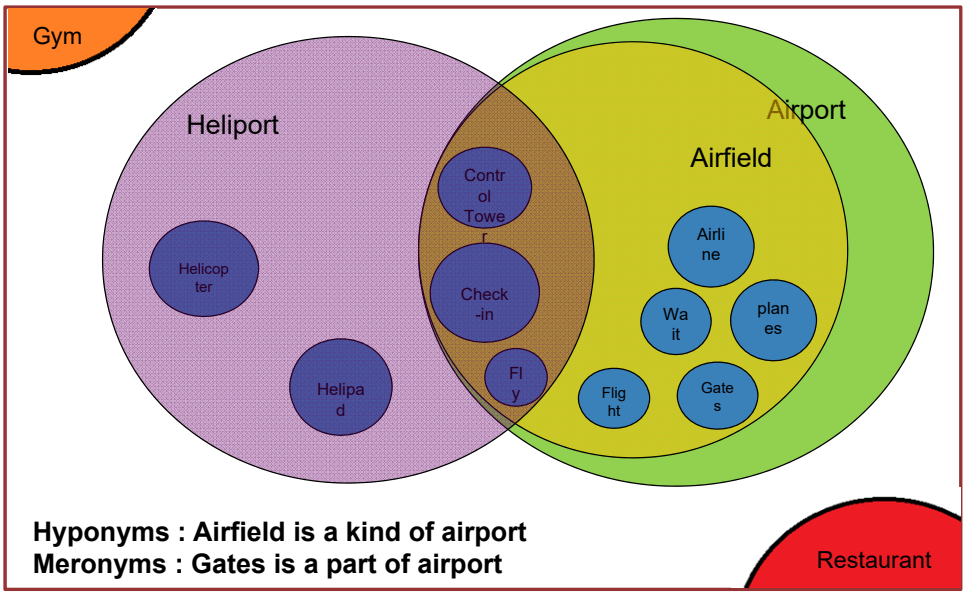
Baseline Classification

- We are interested only in tweets that might suggest a location.
- Tweet Classification
  - Direct Subject
  - Has a first person personal pronoun – "I love New York"
    → (I, me, myself,…)
  - Anonymous Subject
    → Starts with Verbs – "Rocking the bar tonight!"
    → Composed of only nouns and adjectives – "Perfect day at the beach"
  - Others

Rule Generation

- By identifying certain association rules, we can identify certain locations
- Combinations of certain verbs and nouns (bigrams) imply some locations
  - "Wait" + "Flight" → Airport
  - "Go" + "Funeral" → Funeral Home
- Create combinations of all synonyms, meronyms and hyponyms related to a location
- Number of occurrences must be greater than K
- Each rule does not overlap with other categories

Examples



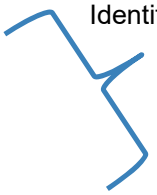
N-gram Combinations

- Identify likely location candidates
- Contiguous sequences of n-items
- Only nouns and adjectives
- Extracted from Direct Subject and Anonymous Subject categories.

Field	Original Tweet	After Processing	1-gram Seq	2-gram Seq	3-gram Seq
Vernacular name			Unigram	Bigram	Trigram
Order of resulting Markov model			0	1	2
Example	… I love the beautiful city of Hsinchu …	…beautiful city Hsinchu…	…,beautiful, city, Hsinchu,…	…,beautiful city city Hsinchu,…	…,beautiful city Hsinchu,…

Tweet Types

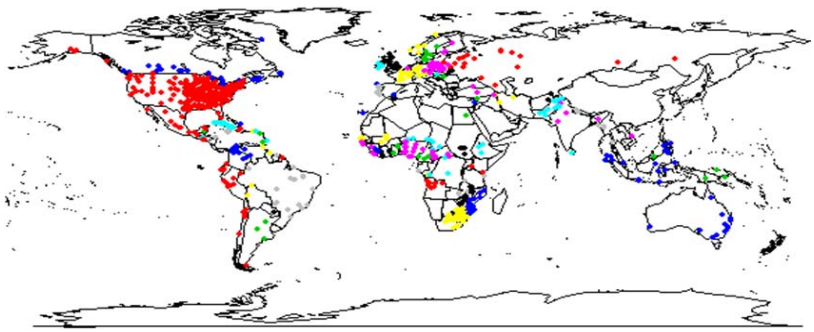
- Coordinates
  - Tweet has geographical coordinates
- Explicit Specific
  - "I Love **Hsinchu**"
    - → Toponyms
- Explicit General
  - "Going to the **Gym**"
    - → Through Hypernyms
- Implicit
  - "Buying a new scarf" -> Department Store
  - Emphasize on actions



Identified through a Web Search

Country Discovery

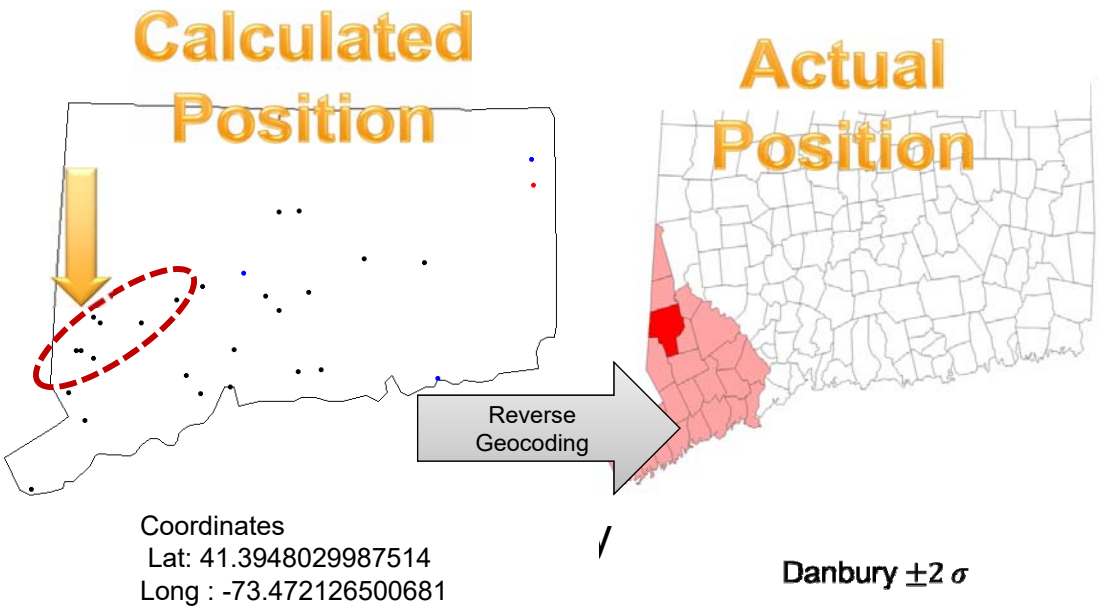
- Identify the user's country
- Identified with OPTICS algorithm
- Cluster all previously marked n-grams
- Most significant cluster is retrieved
  - User's country



Inner Region Discovery

- Identify user's Hometown
- Remove toponyms
- Clustered with OPTICS
- Only locations within ± 2 σ

Inner Region Discovery

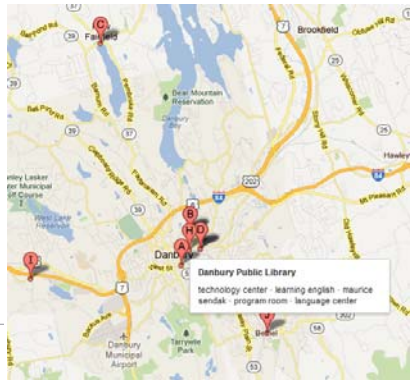


Web Search

- Use web services
- Example :
  - "What's the weather like outside? I haven't left the library in three hours"
  - Search: Library near Danbury, Connecticut , US

Google places

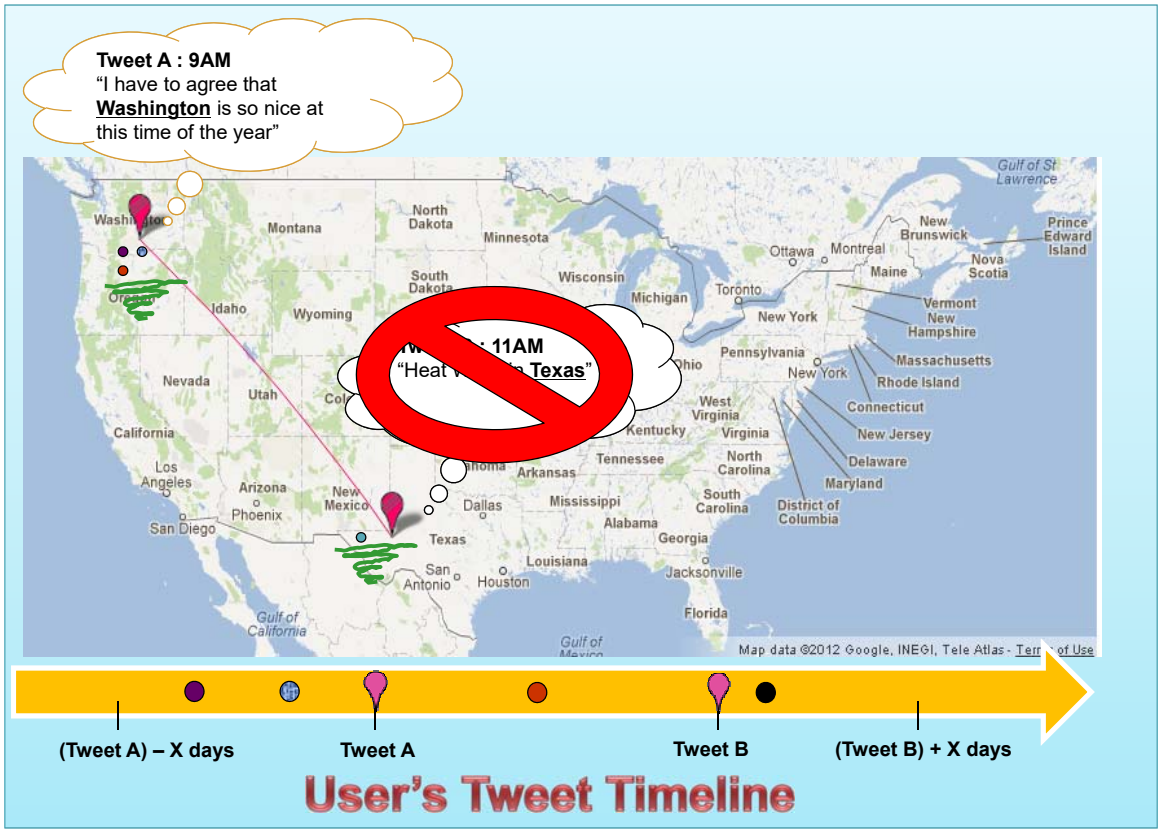
Danbury Public Library
Danbury Health Sciences Library
New Fairfield Free Public Library
Ruth A Haas Library
Danbury Hospital Library
Robert S Young Business Library
Long Ridge Library
Superior Court Law Library



Timeline Sorting

9AM
"I have to agree that **Washington** is so nice at this time of the year"

11AM
"Heat Wave in **Texas**"



Location Inferred

- Users are classified according to the particular findings :
  - No Country (No Information)
  - Just Country
  - Timeline
    - → Current and past locations
  - Timeline with Hometown
    - → Current and past locations
    - → User's Hometown
    - → General Locations

General Statistics

Country	Profiles	Factual	Empty	Fictional	MAE Factual	MAE Empty	Fictional
US	26	16	10		6.26	21.31	
GB	22	19	3		6.18	6.45	
CA	29	25	4		11.18	12.63	
AU	8	8	0		6.47	-	
IN	15	13	2		7.76	18.63	
Others	5	4	1		2.06	14.13	
Total	105	85	20		6.65	14.62	

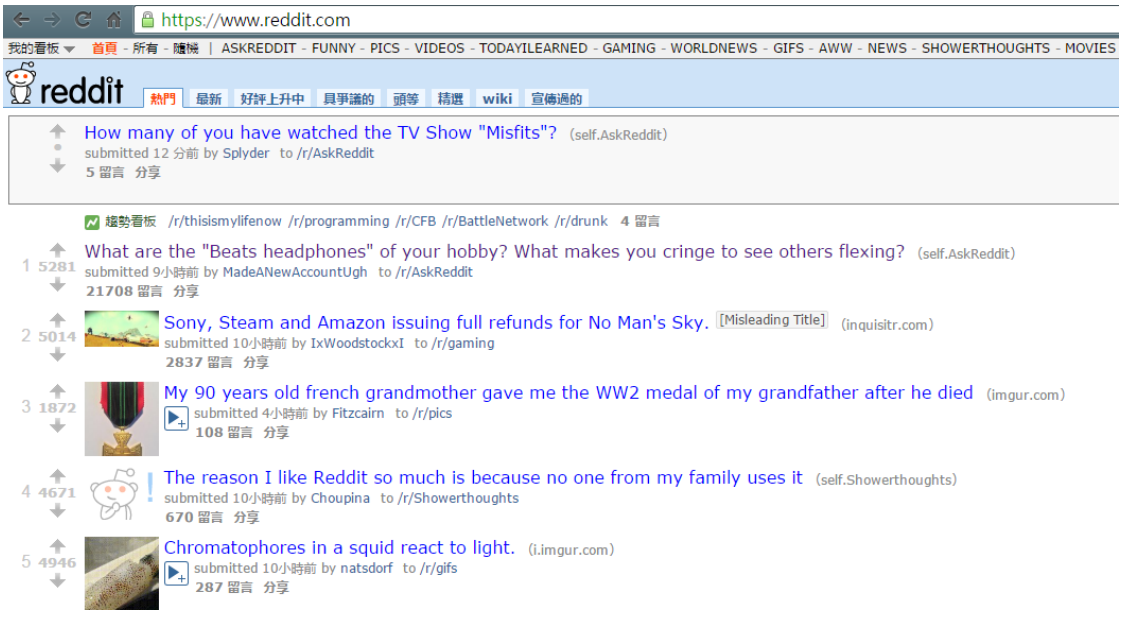
Country	ACC100	WMAE	Tw MAE
US	96%	12.05	9.51
GB	100%	6.22	2.02
CA	100%	11.38	3.41
AU	100%	6.47	2.42
IN	93%	9.21	1.89
Others	100%	4.47	27.54

Case Studies

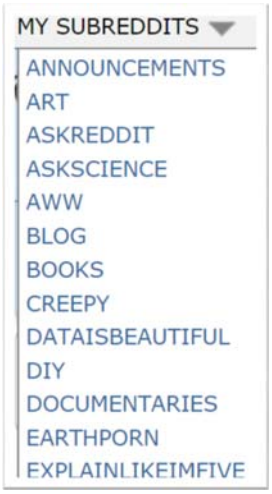
Reddit Data

Reddit Data

https://www.kaggle.com/c/reddit-comments-may-2015/data



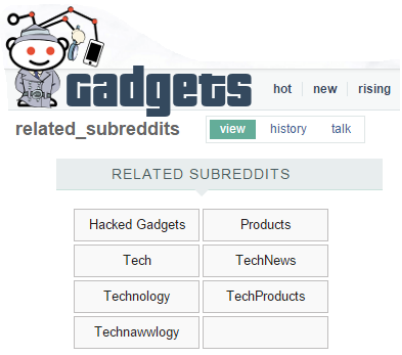
Reddit: The Front Page of the Internet

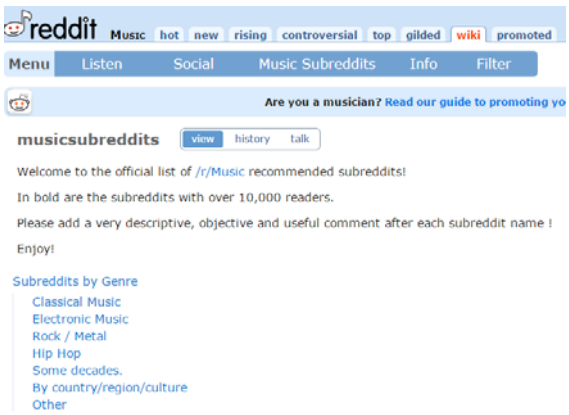


50k+ on this set

Subreddit Categories

- Reddit's structure may already provide a baseline similarity





Provided Data

- created_utc
- ups
- subreddit_id
- link_id
- name
- score_hidden
- author_flair_css_class
- author_flair_text
- subreddit
- id
- removal_reason
- gilded
- downs
- archived
- author
- score
- retrieved_on
- body
- distinguished
- edited
- controversiality
- parent_id

Recover Structure

DjSoundBlast	2015-05-01 10:10:30	2 points
Will the stems be available, or will we have to work with the accepalla on Zedd's Sound cloud?		

JustTheNorm	2015-05-01 10:14:50	6 points
Stems provided.		

DjSoundBlast	2015-05-01 10:23:13	6 points
Do you know if zedd is looking for people to completely transform the song like he did for his first remix contests, or a remix that still stays true to the original?		

rycar	2015-05-01 14:09:35	2 points
Watch the video on the contest page.		

_Nawabi	2015-05-01 20:30:02	1 points
the video only explains how the contest works, not what hes looking for. Id say transform.		

Facebook Fanpage

Fan Page?

Fan Page Post

Fan Page Feedback

Twitter

Tweet

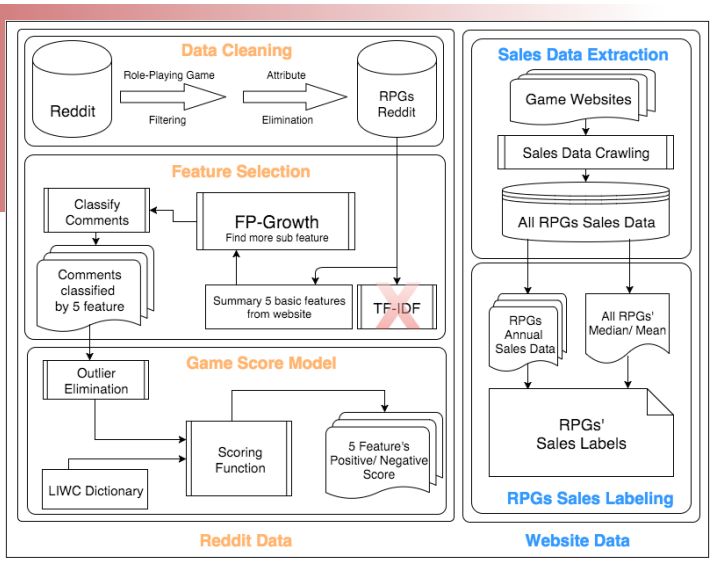
Reddit DM Examples

Dataset

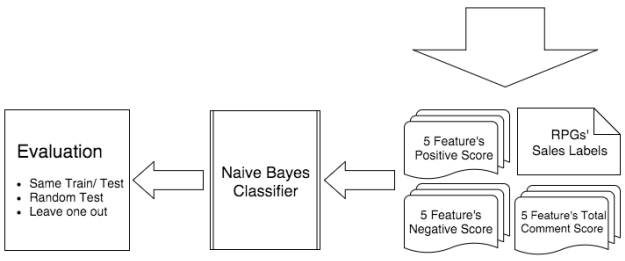
- Use Reddit comment dataset
- Data Selection
  - Choose 30 Role-playing games from the subreddits.

mountandblade	witcher	Diablo	assassinscreed	metalgearsolid	HalfLife
blackops3	battlefield_4	bloodborne	Borderlands2	FF14	Warhammer
GrandTheftAutoV	thelastofusfactions	halo	DarkSouls2	KingdomHearts	StreetFighter
CodAW	BF_Hardline	dragonage	masseffect	darksouls	Shadowrun
StarWarsBattlefront	skyrim	batman	Fallout	FinalFantasy	arma

  - Choose useful attributes form their comment
    - → Body
    - → Score
    - → Subreddit



Pre-processing



Role-Playing Games (RPGs) Sales Prediction Model

Preprocessing

▷Common Game Features

- Drop trivial words

  → We manually build the trivial words dictionary by ourselves.

- Ex : “is” “the” “I” “you” “we” “a”.

- TF-IDF

  → We use Tf-IDF to calculate the importance of each words.

Preprocessing

- TF-IDF result

  → But the TF-IDF value are too small to find the significant importance.

Words	TFIDF
souls	0.005845
bloodborne	0.00536
boss	0.00531
blood	0.005161
damage	0.004916
weapon	0.004386
beast	0.004062

Words	TFIDF
ancient	0.003847
solo	0.003843
group	0.003536
blizzard	0.003419
weapon	0.003242
speed	0.003196
character	0.003131

Words	TFIDF
city	0.00209
guns	0.001887
rifle	0.001883
nuclear	0.001785
sounds	0.00158
shot	0.001375
building	0.001351

- Define Comment Feature

  → We manually define the common gaming words in five categories by referring several game website.

Game-Play | Criticize | Graphic | Music | Story

Preprocessing

▷Filtering useful comments

- Common keywords of 5 categories:

GamePlay	music	story	graphic	criticize
Balance	orchestral	main story	graphic	Downgrade
Combat	BGM	character story	graphical	engine
Atmosphere	soundtrack	story	texture	performance
Gameplay	voice act	storyline	Framerate	load%time

- Filtering useful comments

→ Using the frequent keywords to find out the useful comments in each category.

Preprocessing

- Using FP-Growth to find other feature

→ To Find other frequent keywords

{time, play, people, gt, limited}

{time, up, good, now, damage, gear, way, need, build, better, d3, something, right, being, gt, limited}

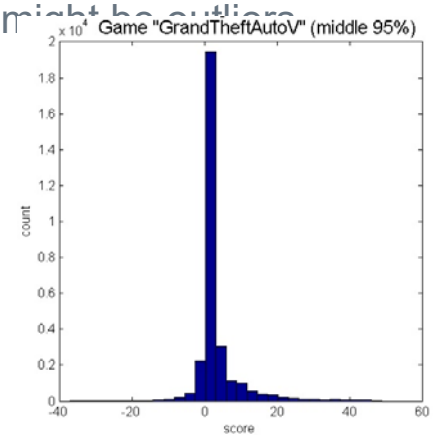
- Then, manually choose some frequent keywords

Preprocessing

▷Comment Emotion

- Filtering outliers

→ First filter out those comment whose “score” separate in top and bottom 2.5% which indicate that they might be outliers



Preprocessing

- Emotion detection

→ We use LIWC dictionary to calculate each comment’s positive and negative emotion percentage.

Ex: I like this game. -> positive

EX: I hate this character. -> negative

→ Then find each category’s positive and negative emotion score.

- $Score_{each\_category\_P} = \sum_{j=0}^{n} score\_each\_comment_j$ * *(positive words)*
- $Score_{each\_category\_N} = \sum_{j=0}^{n} score\_each\_comment_j$ * *(negative words)*
- $TotalScore_{each\_category} = \sum_{i=0}^{m} score\_each\_comment_m$
- $FinalScore_{each\_category\_P} = Score_{each\_category\_P} / TotalScore_{each\_category}$
- $FinalScore_{each\_category\_N} = Score_{each\_category\_N} / TotalScore_{each\_category}$

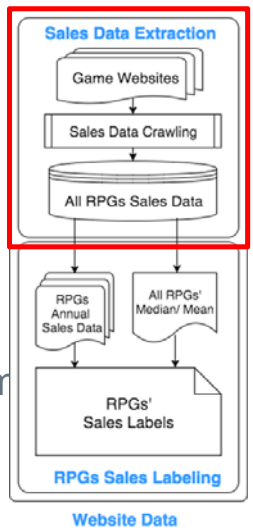
Preprocessing

- Emotion detection
  - → We use LIWC dictionary to calculate each comment's positive and negative emotion percentage.
    - Ex: I like this game. -> positive
    - EX: I hate this character. -> negative
  - → Then find each category's positive and negative emotion score.

The meaning is :

- Finding each category's emotion score for each game
- Calculating TotalScore of each category's comments
- Having the average emotion score FinalScore of each category

Preprocessing

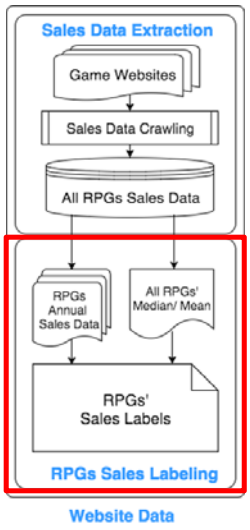


▷Sales Data Extraction

- Crawling website's data
- Find the games' sales on each platform

1		Call of Duty: Black Ops 3 (PS4) Activision, Shooter	6	8,303,244	8,303,244
2		FIFA 16 (PS4) Electronic Arts, Sports	12	6,451,242	6,451,242
3		Star Wars: Battlefront (2015) (PS4) Electronic Arts, Shooter	4	4,832,058	4,832,058
4		Call of Duty: Black Ops 3 (XOne) Activision, Shooter	6	4,423,401	4,423,401
5		Fallout 4 (PS4) Bethesda Softworks, Role-Playing	5	4,418,849	4,418,849
6		Grand Theft Auto V (PS4) Take-Two Interactive, Action	56	3,684,126	9,026,560
7		Splatoon (WiiU) Nintendo, Shooter	29	3,335,728	3,335,728
8		Batman: Arkham Knight (PS4) Warner Bros. Interactive Entertainment, Action	25	3,044,664	3,044,664
9		The Witcher 3: Wild Hunt (PS4) Warner Bros. Interactive Entertainment, Role-Playing	30	2,980,543	2,980,543

Preprocessing



- Annually sales for each game
- Median of each platform
- Mean of all platform

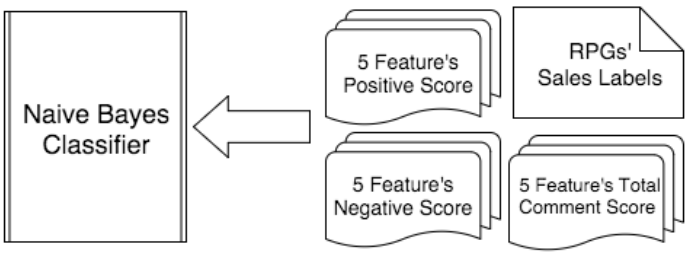
Median : 0.17 -> 30 games (26H 4L)
Mean : 0.533 -> 30 games (18H 12L)

- Try both of Median/ Mean to do the prediction.

Sales Prediction

▷Model Construction

- We use the 4 outputs from pre-processing step to be the input of the Naïve Bayes classifier



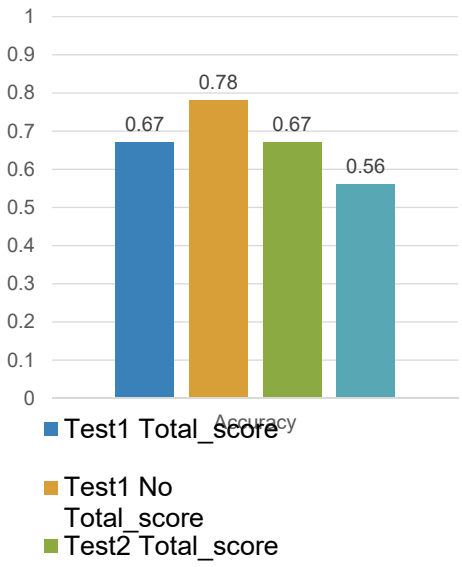
Evaluation

We use naïve Bayes to evaluate our result

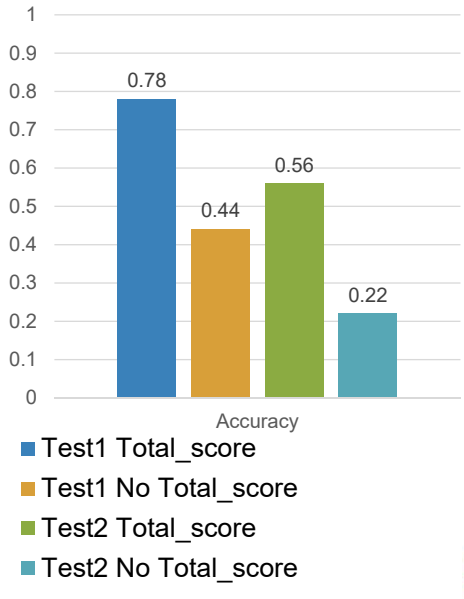
1. training 70% & test30%
2. Leave-one-out

Evaluation (train70% & test30%)

Median 0.17



Mean 0.533

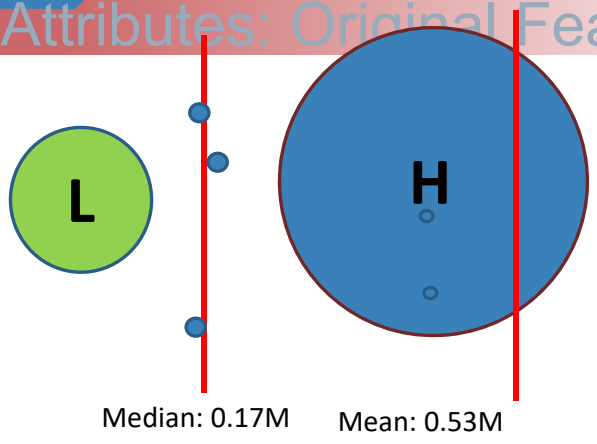


Evaluation(Leave-one-out)

Choose 29 games as training data and the other one as test data, and totally run 30times to get the accuracy.

	Total 30 times	Accuracy
Median Total_score	7 times wrong	77%
Median NO Total_score	5 times wrong	83%
Mean Total_score	6 times wrong	80%
Mean No Total_score	29 times wrong	3%

Attributes: Original Features Scores



Evaluation: Leave-one-Out
Total Sample size: 30

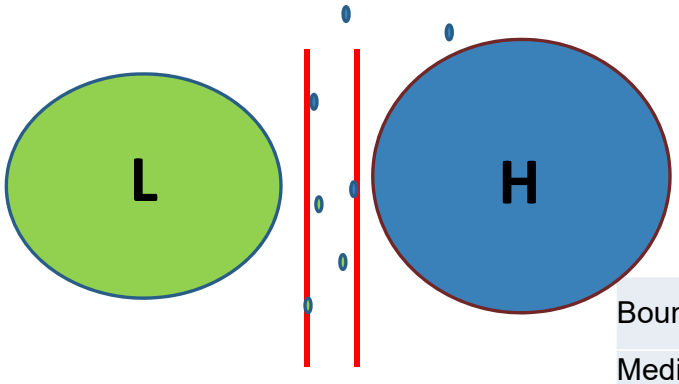
	Error Times	Accuracy
Median	5	83%
Mean	29	3%

Attribute distribution to target(sales) domain

H, L — Sales boundary for H and L class

Sales class

Attributes: Transformed Features Scores



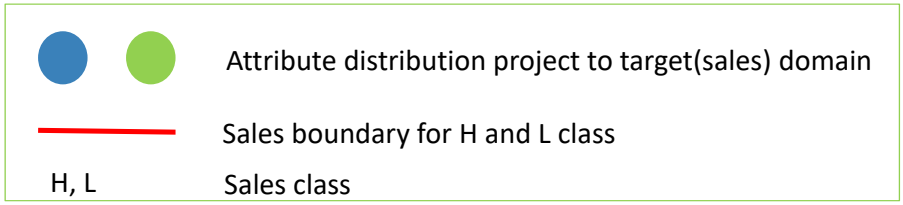
Better

Evaluation: Leave-one-Out
Total Sample size: 30

Boundary	Error Times	Accuracy
Median	7	77%
Mean	6	80%

Median: 0.17M

Mean: 0.53M



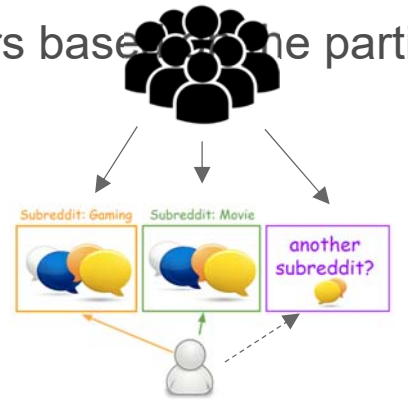
Finding related subreddits based on the gamers' participation

Data Mining Final Project

Group#4

Goal & Hypothesis

- To suggest new subreddits to the Gaming subreddit users based on the participation of other users



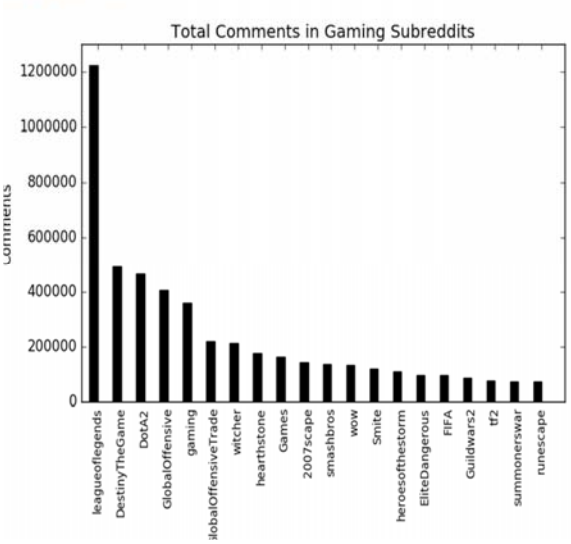
Data Exploration

- We extracted the Top 20 gaming related subreddits to be focused on.

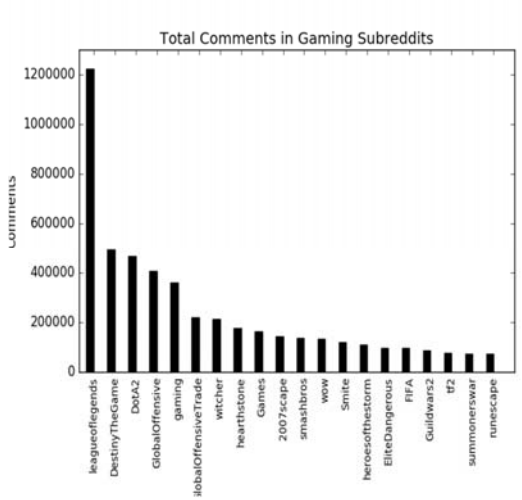
leagueoflegends
DestinyTheGame
DotA2
GlobalOffensive
Gaming
GlobalOffensiveTrade
witcher
hearthstone
Games
2007scape
smashbros
wow
Smite
heroesofthestorm
EliteDangerous
FIFA
Guildwars2
tf2
summonerswar
runescape

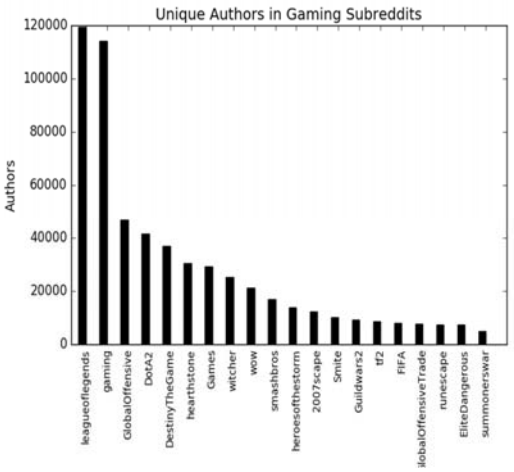
Data Exploration (Cont.)

- We sorted the subreddits by users' comment activities.
- We found out that the most active subreddit is Leagueoflegends.



Data Exploration (Cont.)





Data Pre-processing

- Removed comments from moderators and bots (>2,000 comments)
- Removed comments from default subreddits
- Extracted all the users that commented on at

```
subreddit
3ch,4chan,AdviceAnimals,AndroidQuestions,AutoModerator,Boxing,Ca
ndidFashionPolice,CoonTown,CuteFemaleCorpses,DotA2,DumpsterDivin
g,Ellenpaoinaction,GasTheKikes,JusticePorn,KotakuInAction,Public
Freakout,RedditFacelift,SRSsucks,SexWithDogs,SubredditDrama,Tumb
lrInAction,WhiteSmite,WhitesWinFights,beta,circlejerk,cssnews,da
taisbeautiful,modhelp,pussypassdenied,redditdev,siliconvalley,sn
ew,subredditcancer,techsupport,userstyles
4chan,Agario,Drugs,GlobalOffensive,TheWeeknd,fatpeoplehate,killi
ngfloor,leagueoflegends,millionairemakers
Animesuggest,BlackPeopleTwitter,CringeAnarchy,PussyPass,Robocraf
t,TumblrInAction,WTF,awwnverts,comics,ftlgame,homeopathy,mantids
,pokemon,tf2
```

ket Basket"

```
DestinyTheGame,Fireteams,theflash,FIU
```

S

Processing

- Sorted the frequent items (sort.js)
  - Eg. DotA2, Neverwinter, atheism → DotA2, atheism, Neverwinter

Original Transactions

```
107452  Dirtybomb,Games,Warframe
107453  4chan,BlackPeopleTwitter,FlashTV,Games,Guildwars2,Hig
107454  KotakuInAction,NotTimAndEric,PublicFreakout,thatHappe
107455  AdviceAnimals,AskCulinary,Boise,CampingandHiking,DotA
107456  DunderMifflin,GlobalOffensive,HotlineMiami,paydaytheh
107457  Cynicalbrit,DotA2,DrinkingGames,Pathfinder,Pathfinder
107458  DotA2,Neverwinter,atheism
107459  Brazil,Fallout,GlobalOffensive,Minecraft,Steam,brasil
107460  Games,bloodborne,firstworldproblems,pcgaming,witcher
107461  Berserk,Eyebleach,Games,OnePiece,OutOfTheLoop,TokyoGh
107462  Berserk,DnD,EDH,Guildwars2,GundamExVs,Gunpla,fatpeopl
107463  AdviceAnimals,battlestations,gameofthrones,hardwaresw
107464  Starwarscommander,TumblrInAction,fireemblemcasual,fri
107465  Games,MechanicalKeyboards,churning,gamecollecting,wit
107466  EliteDangerous,TalesFromRetail,WTF,brum,patientgamers
107467  DestinyTheGame,Fireteams,Frugal,nfl
```

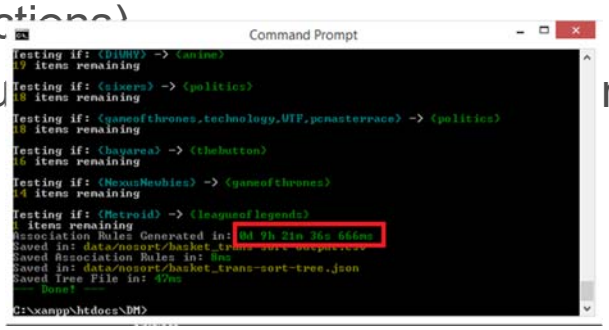
Sorted Transactions

```
107452  Games,Warframe,Dirtybomb
107453  leagueoflegends,Games,WTF,thebutton,TumblrInAction,PS
107454  witcher,KotakuInAction,thatHappened,PublicFreakout,No
107455  WTF,AdviceAnimals,DotA2,gameofthrones,buildapc,pcgami
107456  pcmasterrace,GlobalOffensive,paydaytheheist,DunderMif
107457  DotA2,Cynicalbrit,Pathfinder_RPG,Pathfinder,DrinkingG
107458  DotA2,atheism,Neverwinter
107459  pcmasterrace,GlobalOffensive,science,woahdude,Steam,M
107460  Games,witcher,pcgaming,bloodborne,firstworldproblems
107461  leagueoflegends,Games,reactiongifs,OutOfTheLoop,OnePi
107462  thebutton,fatpeoplehate,Guildwars2,magicTCG,DnD,EDH,v
107463  pcmasterrace,AdviceAnimals,wow,gameofthrones,pcgaming
107464  hearthstone,TumblrInAction,fireemblemcasual,friendsaf
107465  Games,witcher,MechanicalKeyboards,gamecollecting,chur
107466  WTF,EliteDangerous,patientgamers,TalesFromRetail,brum
107467  DestinyTheGame,nfl,Fireteams,Frugal
```

Processing

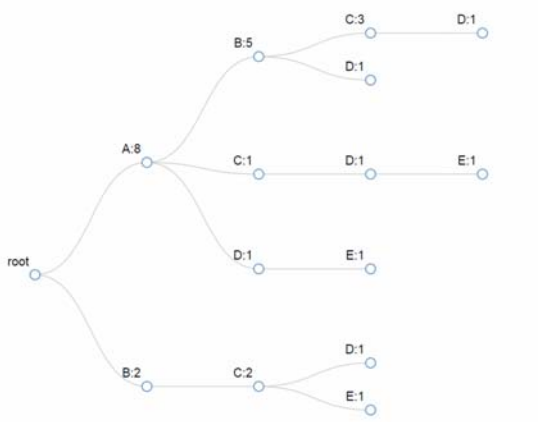
- Choosing the minimum support from 159,475 Transactions
- Max possible support 25.71% (at least in 41,004 Transactions)
- Min possible su n 1 Transaction)

Min_Support as 0.05 % = 79.73 transactions



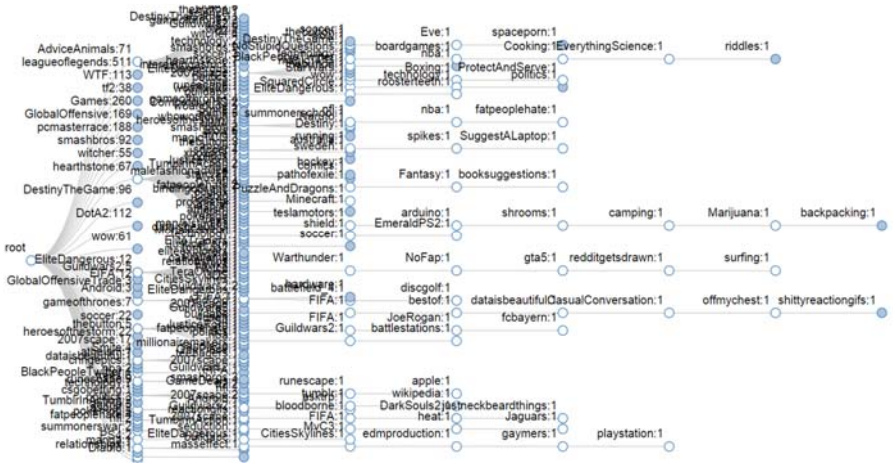
Processing

- How a FP-Growth tree looks li



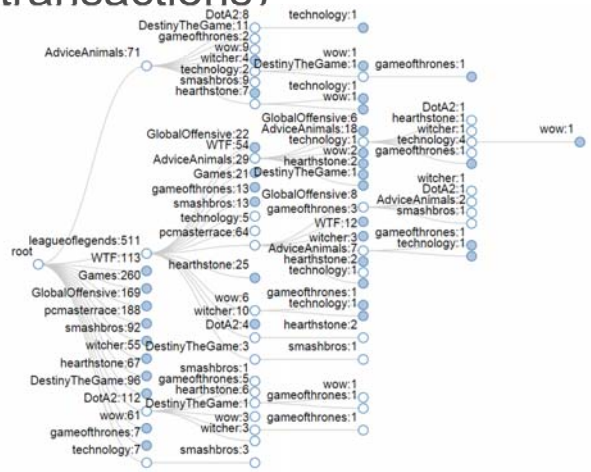
Processing

- Our FP-Growth (minimum support = 1% at least

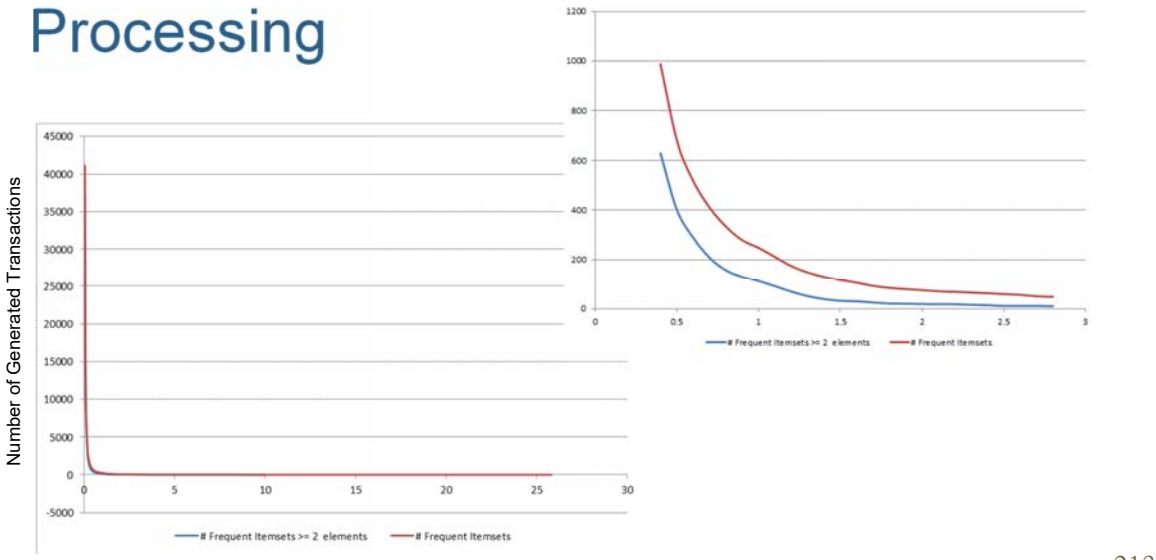


Processing

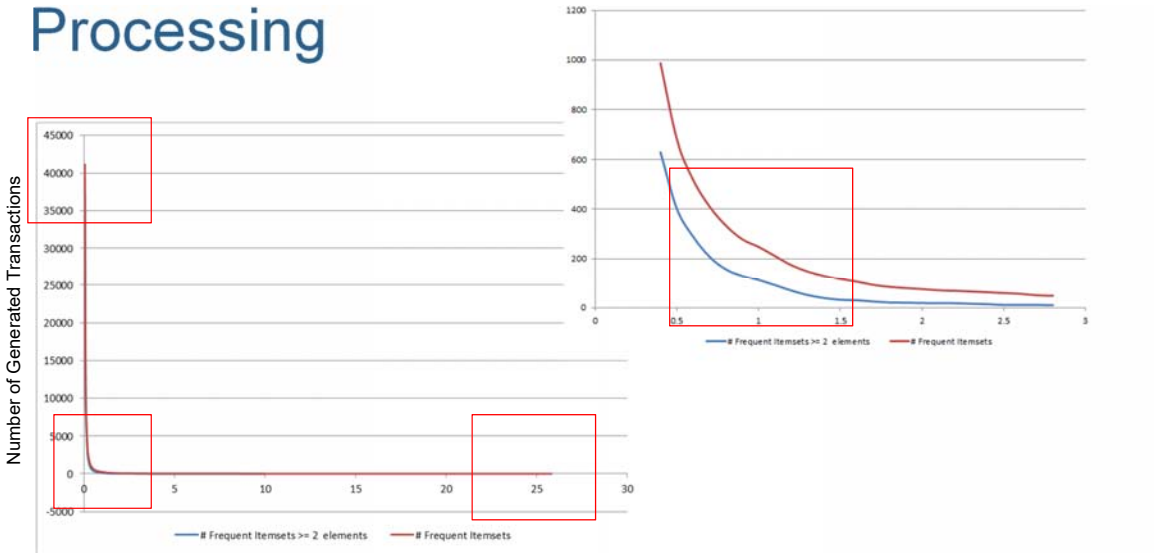
- Our FP-Growth (minimum support = 5% at least in 7,974 transactions)



Processing



Processing



Processing

- Our minimum support = 0.8% (at least in 1,276 trans

Association Rules , min-support = 79.7375 items (0.05%)

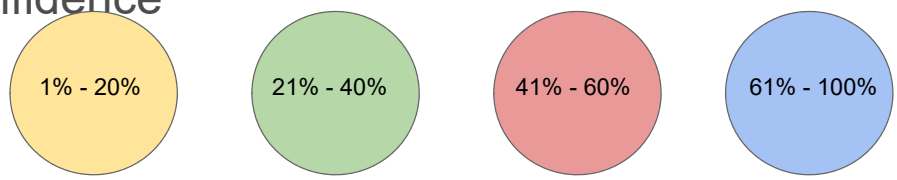
Limit to: 100	Search By Item: item name...	Min Support 0.8	Max Support 100	Min
Confidence 0	Max Confidence 100	Min Items 1	Max Items 10	Showing 100 of 41169 Itemsets	Save Rules

1 [1.51% , 47.70% , 5034] { pcmasterrace , AdviceAnimals } -> WTF
2 [1.51% , 46.68% , 5144] { pcmasterrace , WTF } -> AdviceAnimals
3 [1.51% , 30.70% , 7821] { WTF , AdviceAnimals } -> pcmasterrace
4 [1.46% , 38.85% , 5984] { leagueoflegends , WTF } -> AdviceAnimals
5 [1.46% , 39.17% , 5935] { leagueoflegends , AdviceAnimals } -> WTF
6 [1.46% , 29.73% , 7821] { WTF , AdviceAnimals } -> leagueoflegends
7 [1.28% , 26.02% , 7821] { WTF , AdviceAnimals } -> technology

- Games -> politics Support : 1.30% (2,073 users)

Post-processing

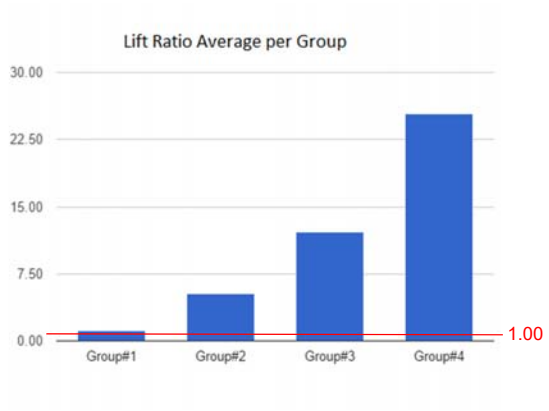
- We classified the rules in 4 groups based on their confidence



Group #1 Group #2 Group #3 Group #4

Post-processing

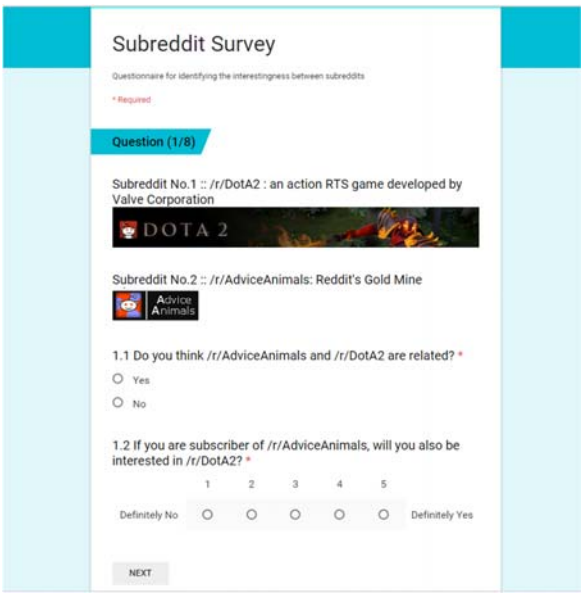
- {Antecedent} → {Con
  - eg. {FIFA, Soccer} –



- **Lift ratio** = confidence / Benchmark confidence
- **Benchmark confidence** = # consequent element

Post-processing

We created **8 surveys** with **64 rules** (2 rules per group) and **2 questions** per rules.



Post-processing

- Q1: Do you think **subreddit A** and **subreddit B** are related? → Dependency
  - **[yes | no]**
- Q2: If you are subscriber of **subreddit A**, will you also be interested in **subreddit B**?
  - Definitely No **[1 , 2 , 3 , 4 , 5]** Definitely Yes

Expected Confidence

Post-processing

- We got response from 52 persons
- Data Confidence vs Expected Confidence → Lift

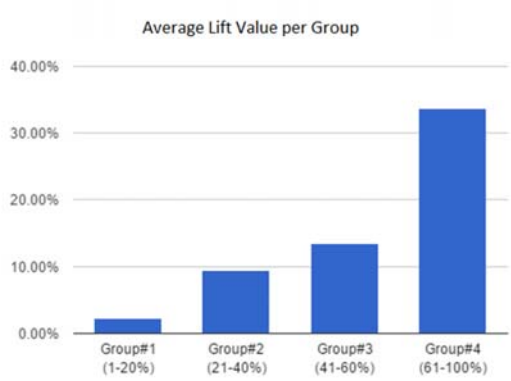
Subreddit No.1	Form#	# of response	Related Yes	Related No	0	25	50	75	100	% Not related	Expected Con (1=0)	Unrelatedness	Lift (1=0)
AdviceAnimals	1	4	0	4	1		3			100.00%	37.50%	68.75%	0
pcmasterrace	1	4	2	2	1	2		1		50.00%	31.25%	40.63%	0
AdviceAnimals	2	7	0	7		5	2			100.00%	32.14%	66.07%	0
WTF	3	7	1	6	5	2				85.71%	7.14%	46.43%	2.36%
AdviceAnimals	3	7	2	5	2	1	3		1	71.43%	39.29%	55.36%	0
Games	4	6	4	2		1	3	1	1	33.33%	58.33%	45.83%	0
WTF	4	6	3	3			4	2		50.00%	58.33%	54.17%	0
leagueoflegends	5	6	4	2		1	2	3		33.33%	58.33%	45.83%	0
Games	5	6	1	5		3	2	1		83.33%	41.67%	62.50%	0
leagueoflegends	6	7	1	6		4	3			85.71%	35.71%	60.71%	0
Games	4	7	6.5	0.5	0	0	1.5	4.5	1	7.14%	73.21%	40.18%	0
wow	7	4	1	3	2		2			75.00%	25.00%	50.00%	0
Games	7	4	3	1			2	2		25.00%	62.50%	43.75%	0
pcmasterrace	8	11	0	11	1	2	5	3		100.00%	47.73%	73.86%	0
WTF	8	11	0	11	2	6	3			100.00%	27.27%	63.64%	0
politics	1	4	0	4	4					100.00%	0.00%	50.00%	28.70%
pcgaming	1	4	4	0	1		3			0.00%	37.50%	18.75%	0
technology	2	7	7	0			1	6		0.00%	71.43%	35.71%	0

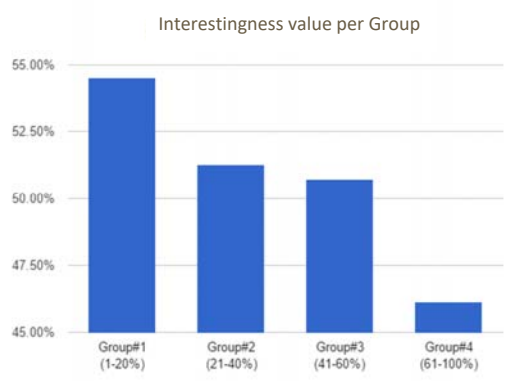
Post-processing

- % Non-related was proportion of the “No” answer to entire response in first question.
- Interestingness = Average of Expected confidenc* % N[illegible]

# of response	Related Yes	Related No	% Not related	Expected Con (1=0)	Interestingness
4	0	4	100.00%	0.375	0.6875
4	2	2	50.00%	0.3125	0.40625
7	0	7	100.00%	0.3214285714	0.6607142857
7	1	6	85.71%	0.07142857143	0.4642857143
7	2	5	71.43%	0.3928571429	0.5535714286
6	4	2	33.33%	0.5833333333	0.4583333333
6	3	3	50.00%	0.5833333333	0.5416666667
6	4	2	33.33%	0.5833333333	0.4583333333
6	1	5	83.33%	0.4166666667	0.625
7	1	6	85.71%	0.3571428571	0.6071428571
7	6.5	0.5	7.14%	0.7321428571	0.4017857143
4	1	3	75.00%	0.25	0.5
4	3	1	25.00%	0.625	0.4375
11	0	11	100.00%	0.4772727273	0.7386363636
11	0	11	100.00%	0.2727272727	0.6363636364
4	0	4	100.00%	0	0.5
4	4	0	0.00%	0.375	0.1875
7	7	0	0.00%	0.7142857143	0.3571428571
7	4	3	42.86%	0.5357142857	0.4821428571
7	6	1	14.29%	0.75	0.4464285714

Post-processing

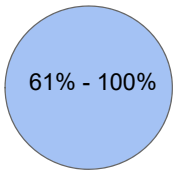




Results

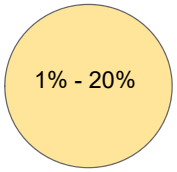
- How we can suggest new subreddits then?

High Confidence
Less Interestingness



Group #4

Less Confidence
High Interestingness

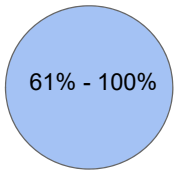


Group #1

Results

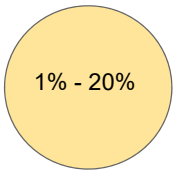
- Suggest them As:

Other People also
talks about….



Group #4

Maybe you could be
Interested in...



Group #1

Results

- Results from Group# 4 (9 Rules)

1 [0.94% , 63.05% , 2387] { AdviceAnimals , BlackPeopleTwitter } -> WTF

2 [0.94% , 61.71% , 2439] { WTF , BlackPeopleTwitter } -> AdviceAnimals

3 [2.80% , 96.36% , 4640] { Fireteams } -> DestinyTheGame

4 [2.77% , 96.43% , 4589] { summonerschool } -> leagueoflegends

5 [1.64% , 84.54% , 3086] { csgobetting } -> GlobalOffensive

6 [1.03% , 73.12% , 2251] { GlobalOffensiveTrade } -> GlobalOffensive

7 [1.00% , 89.48% , 1787] { CompetitiveHS } -> hearthstone

8 [0.91% , 63.76% , 2285] { truegaming } -> Games

9 [0.81% , 96.19% , 1338] { CLG } -> leagueoflegends

Results

- Results from Group# 1 (173 Rules)

13 [2.31% , 14.68% , 25063] { Games } -> AdviceAnimals

14 [2.31% , 17.85% , 20605] { AdviceAnimals } -> Games

15 [2.31% , 17.85% , 20605] { AdviceAnimals } -> technology

16 [2.29% , 14.56% , 25063] { Games } -> technology

17 [2.28% , 16.52% , 22022] { GlobalOffensive } -> leagueoflegends

18 [2.28% , 8.87% , 41004] { leagueoflegends } -> GlobalOffensive

19 [2.01% , 14.26% , 22501] { WTF } -> GlobalOffensive

20 [2.01% , 14.57% , 22022] { GlobalOffensive } -> WTF

21 [1.96% , 12.49% , 25063] { Games } -> witcher

22 [1.90% , 12.32% , 24560] { pcmasterrace } -> pcgaming

23 [1.79% , 11.62% , 24560] { pcmasterrace } -> DotA2

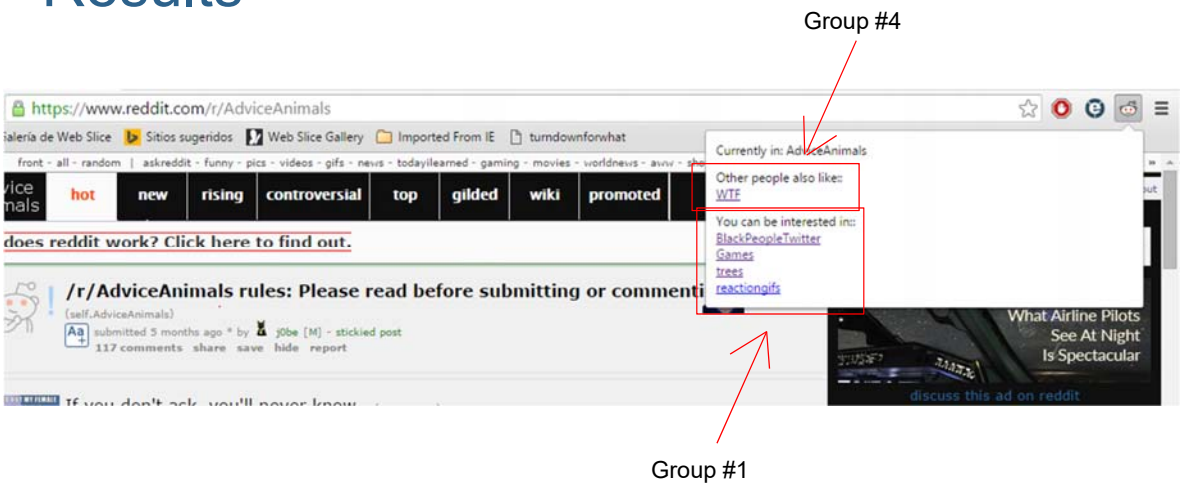
24 [1.79% , 17.19% , 16598] { DotA2 } -> pcmasterrace

25 [1.78% , 13.81% , 20605] { AdviceAnimals } -> politics

26 [1.77% , 12.51% , 22501] { WTF } -> politics

```
13  GlobalOffensive,->,leagueoflegends,2.28%,16.52%,22022
14  leagueoflegends,->,GlobalOffensive,2.28%,8.87%,41004
15  WTF,->,GlobalOffensive,2.01%,14.26%,22501
16  GlobalOffensive,->,WTF,2.01%,14.57%,22022
17  Games,->,witcher,1.96%,12.49%,25063
18  pcmasterrace,->,pcgaming,1.90%,12.32%,24560
19  pcmasterrace,->,DotA2,1.79%,11.62%,24560
20  DotA2,->,pcmasterrace,1.79%,17.19%,16598
21  AdviceAnimals,->,politics,1.78%,13.81%,20605
22  WTF,->,politics,1.77%,12.51%,22501
23  AdviceAnimals,->,GlobalOffensive,1.73%,13.42%,20605
24  GlobalOffensive,->,AdviceAnimals,1.73%,12.56%,22022
25  pcmasterrace,->,technology,1.70%,11.02%,24560
26  Games,->,pcgaming,1.64%,10.41%,25063
```

Results



Challenges

- Reducing scope of data for decreasing computational time
- Defining and calculating the interestingness value
- How to suggest the rules that we got to reddit users

Conclusion

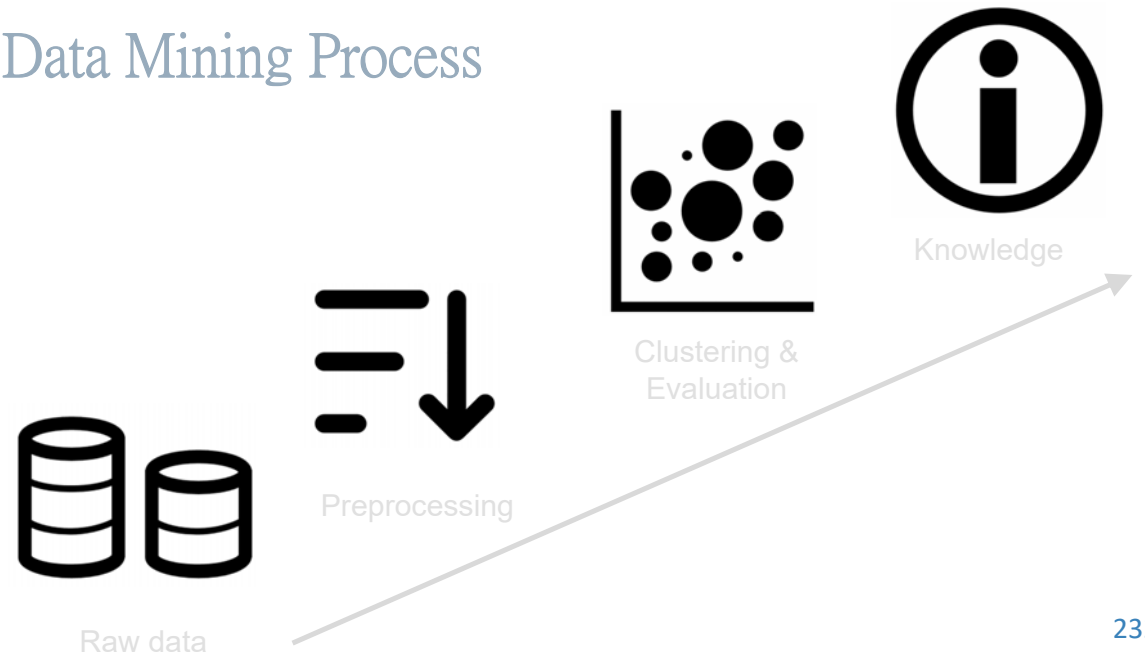
- We cannot be sure that our recommendation system will be 100% useful to user since the interestingness can vary depending on the purpose of the experiment
- To getting more accurate result, we need to ask all generated association rules, more than 300 rules in survey

References

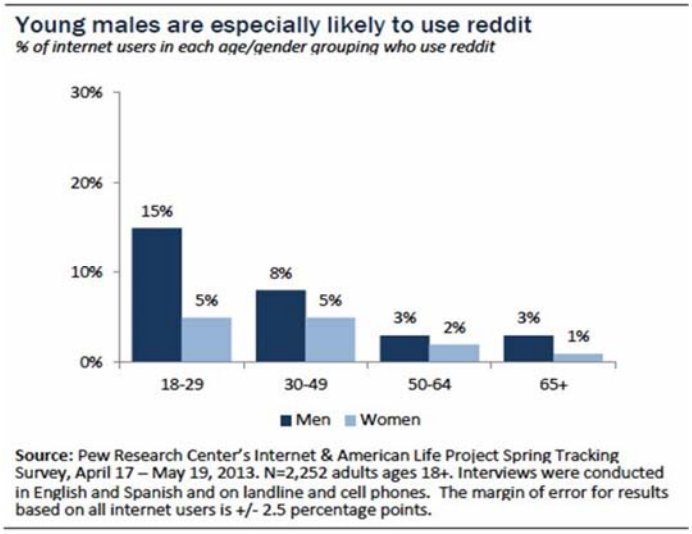
- Liu, B. (2000). Analyzing the subjective interestingness of association rules. 15(5), 47-55. doi:10.1109/5254.889106
- Calculating Lift, How We Make Smart Online Product Recommendations (https://www.youtube.com/watch?v=DeZFe1LerAQ)

Application of Data Mining Techniques on Active Users in Reddit

Data Mining Process

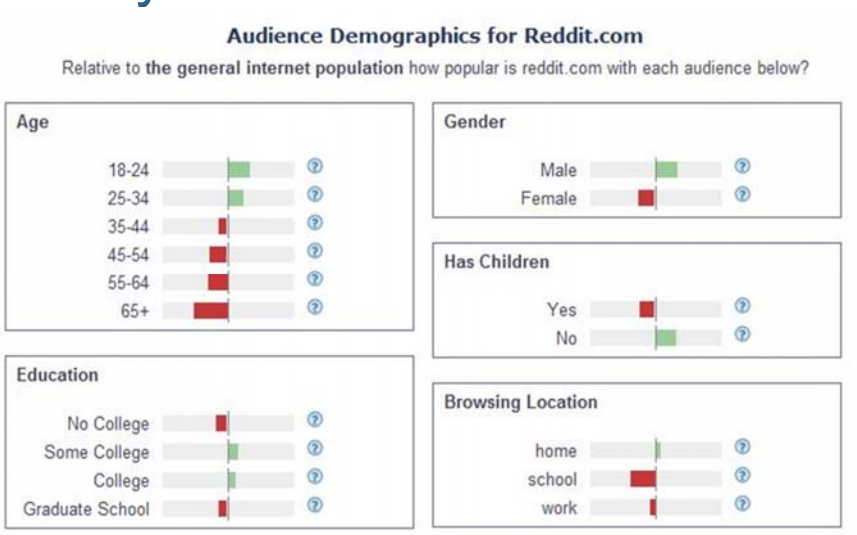


Facts - Why Active Users?



** http://www.pewinternet.org/2013/07/03/6-of-online-adults-are-reddit-users/

Facts - Why Active Users?



** http://www.snoosecret.com/statistics-about-reddit.html

Active Users Definitions

- We define active users as people who :
  1. who had posted or commented in at least 5 subreddits
  2. who has at least 5 Posts or Comments in each of the subreddits
  3. # of Users' comments above Q3
  4. Average Score of the User, who satisfies three criteria above > Q3

Preprocessing

- Total posts in May2015: 54,504,410
- Total distinct authors in May2015: 2,611,449

- After deleting BOTs, [deleted], non-English, only-URL posts, and length < 3 posts, we got 46,936,791 rows and 2,514,642 distinct authors.

- *Finally, we extracted **25,298 "active users"** (0.97%)*
- *and **5,007,845** posts **(9.18%)** by our active users definitions*

Clustering:

K-means, K-prototype

- # of clusters (K) = 10 , $k = \sqrt{(n/2)}$, $k = \sqrt{(\sqrt{(n/2)})}$
- using python sklearn module - *KMeans(), open source -*

Attributes	Type	Description
subreddit	nominal	The title of the subreddit.
time	nominal	The median time that the author posted comments. It ranges from 0 to 23.
activity	ratio	(number of user's comment in the subreddit they commented most) / (number of user's comment in total) It indicates how much the user devoted their time in the subreddit they commented most.
assurance	ratio	(user's median score within the subreddit) / (median score of the subreddit) It indicates their similarity with others in the subreddit they commented most.

237

Attributes

author	subreddit	frequency	median score
1	A	11	5
1	B	6	2
1	C	27	3
1	D	17	0
1	E	3	1
2	C	19	1
2	D	5	1

subreddit	A	B	C	D	E	F
median score	2	3	1	1	0	4

author = 1

subreddit = C (frequency: 27>others)

activity = 27/(11+6+27+17+3) = 0.42

assurance = 3/1 = 3

238

Clustering: K-means, K-prototype

Attributes	Type	Description
subreddit	nominal	The title of the subreddit.
time	ratio	The median time that the author posted comments. It ranges from 0 to 23.
activity	ratio	(number of user's comment in the subreddit they commented most) / (number of user's comment in total) It indicates how much the user devoted their time in the subreddit they commented most.
assurance	ratio	(user's median score within the subreddit) / (median score of the subreddit) It indicates their similarity with others in the subreddit they commented most.

Clustering: K-means, K-prototype

Attributes	Type	Description
subreddit	nominal	The title of the subreddit.
time	nominal	The median time that the author posted comments. It ranges from 0 to 23.
activity	ratio	(number of user's comment in the subreddit they commented most) / (number of user's comment in total) It indicates how much the user devoted their time in the subreddit they commented most.
assurance	ratio	(user's median score within the subreddit) / (median score of the subreddit) It indicates their similarity with others in the subreddit they commented most.

Evaluation

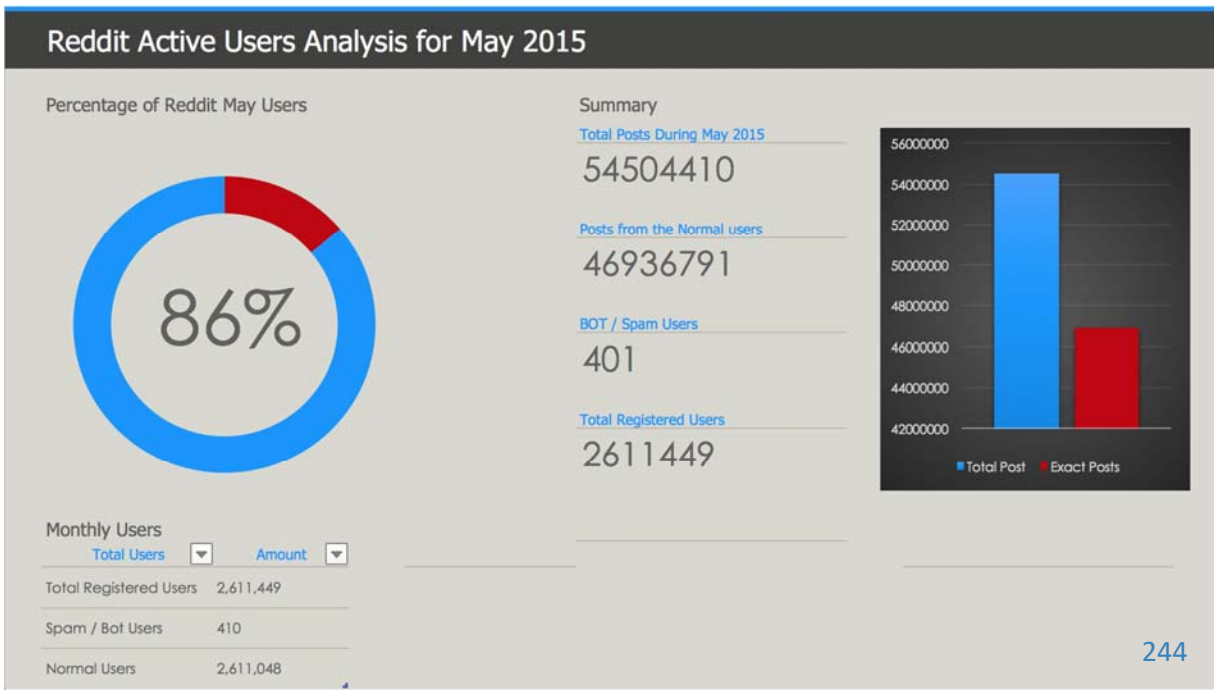
data	A	B	C
data	A	B	C

- Separate data into 3 parts,
- K-means && K-prototype for AB, AC
- Compare labels of A which was from clustering result of AB, AC.
- Measurement:
  - **Adjusted Rand index:** measure similarity between two list. (Ex. AB-A & AC-A)
  - **Homogeneity**: each cluster contains only members of a single class
  - **Completeness**: all members of a given class are assigned to the same cluster.
  - **V-measure**: harmonic mean of homogeneity & completeness

Evaluation: K-means v.s. K-prototype

	K-means	K-prototype
Adjusted Rand index	0.510904	0.193399
Homogeneity	0.803911	0.326116
Completeness	0.681669	0.298049
V-measure	0.737761	0.311451

VISUALIZATION



The Active Users in Reddit

Summary

TOTAL USERS

2,611,449

ACTIVE USERS

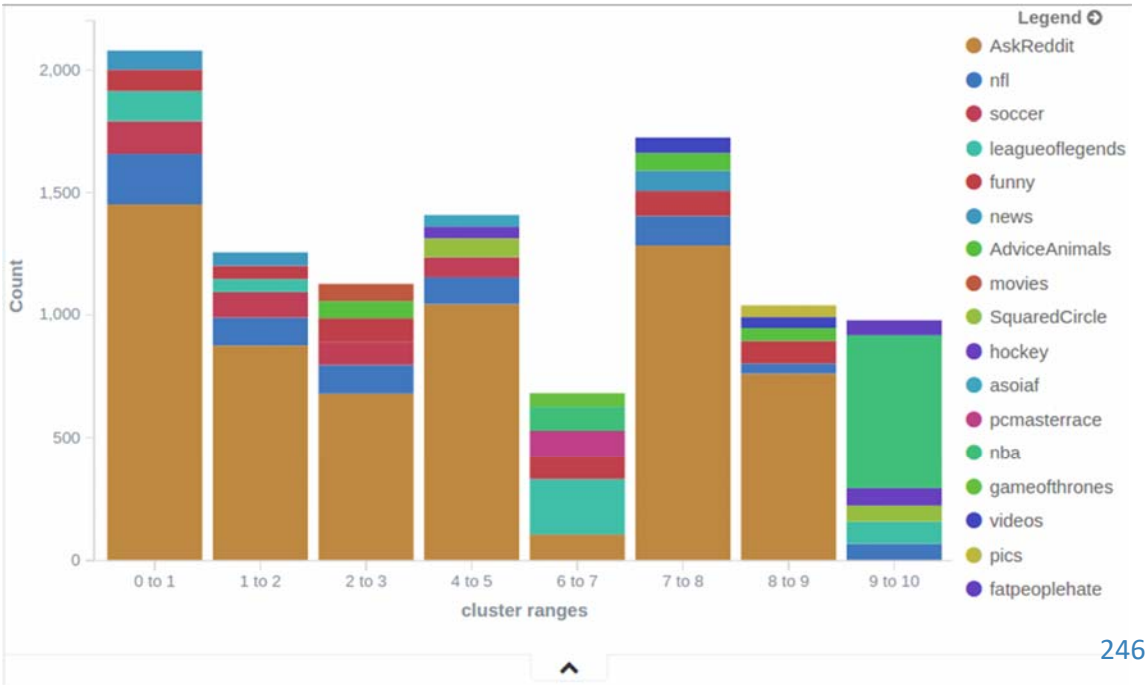
25,298

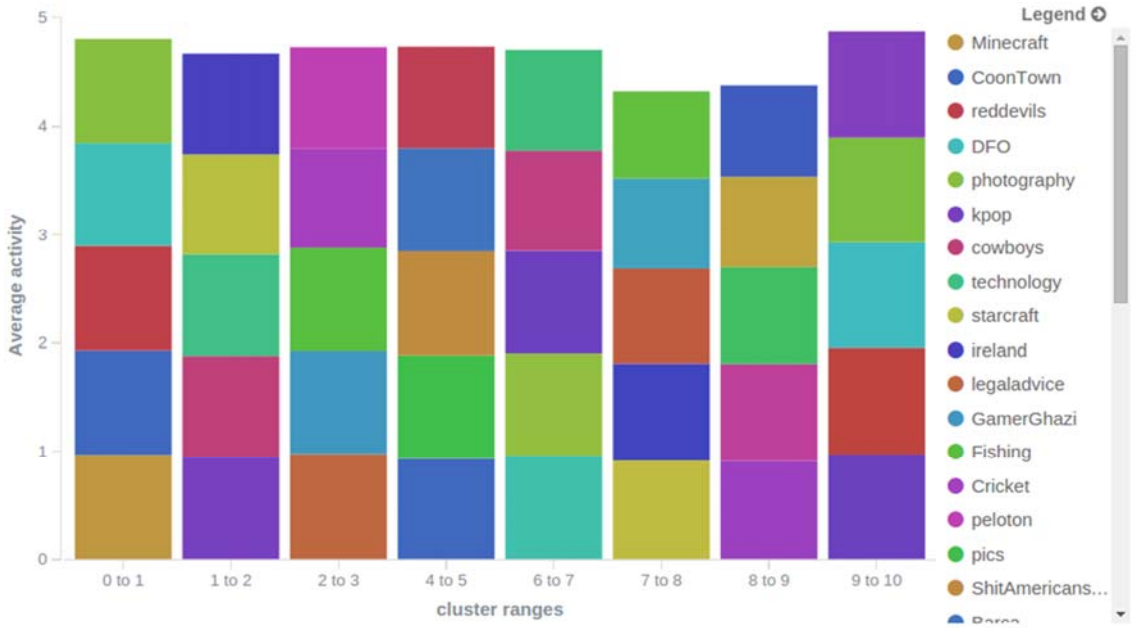
ACTIVE USERS RATIO

1%

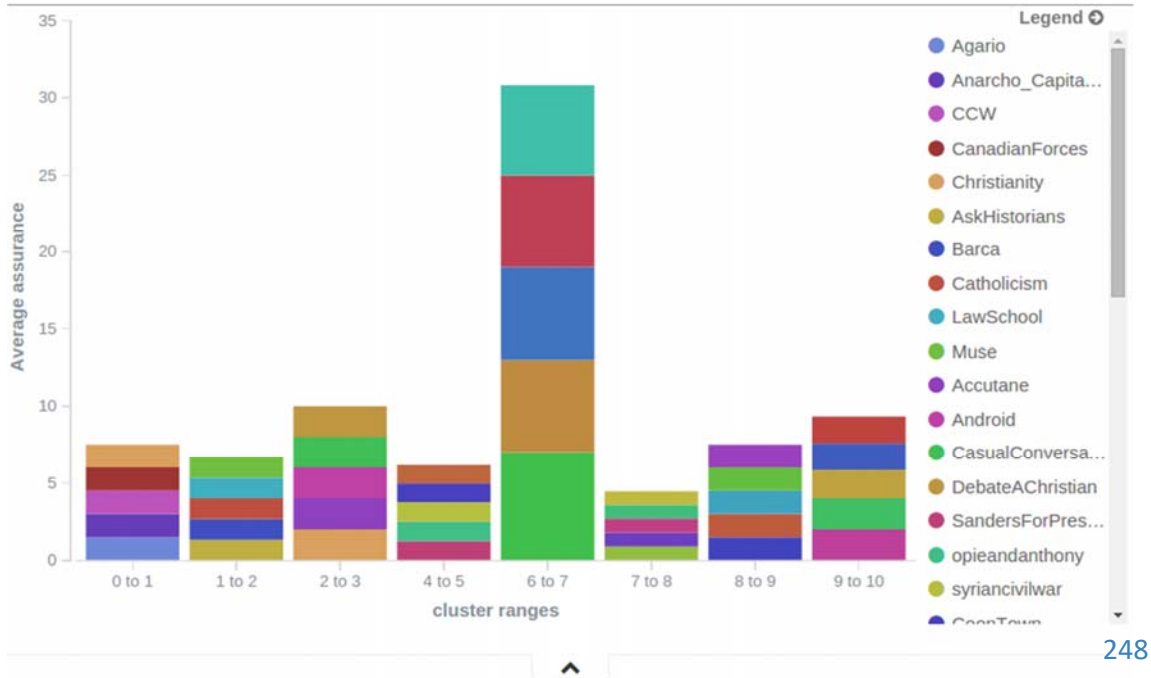
USERS WHO SHOWN UP DURING MAY 2015

99%




Legend
Minecraft
CoonTown
reddevils
DFO
photography
kpop
cowboys
technology
starcraft
ireland
legaladvice
GamerGhazi
Fishing
Cricket
peloton
pics
ShitAmericans...
Average activity
0
1
2
3
4
5
0 to 1
1 to 2
2 to 3
4 to 5
6 to 7
7 to 8
8 to 9
9 to 10
cluster ranges

247

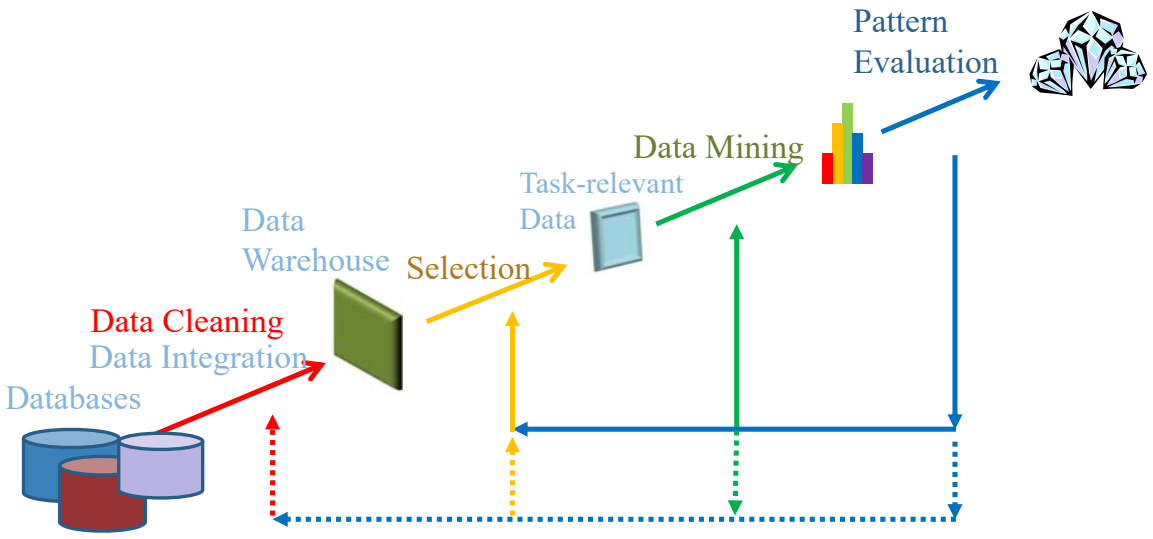

Legend
Agario
Anarcho_Capita...
CCW
CanadianForces
Christianity
AskHistorians
Barca
Catholicism
LawSchool
Muse
Accutane
Android
CasualConversa...
DebateAChristian
SandersForPres...
opieandanthony
syriancivilwar
Average assurance
0
5
10
15
20
25
30
35
0 to 1
1 to 2
2 to 3
4 to 5
6 to 7
7 to 8
8 to 9
9 to 10
cluster ranges

248

Some Remarks

Some but not all

Knowledge Discovery (KDD) Process



Always Remember

- Have a good and solid objective
  - No goal no gold
  - Know the relationships between them