No.

remove markup

detection & deletion of duplicate symbols

task, data & algorithm

pre-processing

1. data acquisition & preparation
2. feature extraction
3. evaluation
4. use

Train → fit the parameters.

Development / Validation / Tuning

split criterion: 50 50

model selection / hyperparameters

after experimental condition

parameters tune

evaluation: test / held out

unseen; once instead of looking

✓ unbiased.

different metrics & approaches

Supervised

Manually annotated (expert)

found (star review)

Unsupervised

Semi-Supervised

moderately

gradient descent.

stochastic gradient descent.

rm bias

algo robustness to unexpected data

Consistency across datasets

No. Tick 1

Date Jan 20

sparsity
insufficient data

preannotated →

× human labour
× what to include
× static (language change)

Review 1 ✓ Sentiment lexicons (specialized) fixed, predefined. Tick: 2y317

best. well. satisfying gory &

\+ \+ \+ sentiment strength

Review 2 ×

but bland lacking not like ✓

— — — —

Review 3 ×

he isn't ironic

Review 4 ✓

escapist fun

Capitalise

preprocessing : normalization

✓tokeniser ~ split

tick.py tokenise the review texts.

utils / tokenizer.py.

default : 10

\+ \+

classification. records positive / negative. orientation (writer,

review_data. lexicon Dict[str, int]

Evaluation output Ground Truth Accuracy.

Raise accuracy

predict_sentiment ↓ List[str] Dict[str, int]

seven, period, word tokenization (next ch)

Language modelling LLM (next word)

Spam detection (spam / not-spam) No. Tick 2

Text categorization (sentiment analysis)
Language, authorship, lib. subject category

Naive Bayes Classification : assign a category to each input
→ predefined classes. sentiment (labels)
classes: c_1, c_2 ...
labels
{Positive ; Negative}

Tokenise → features O (Reviews)
$\{w_1, w_2, \dots w_n\}$ | unordered set ; position ×
2. features are independent (observable properties)

Bayes Theorem: Multinomial naive Bayes classifier

$\hat{c}_{NB} = \arg\max_{c \in C} P(c|O) = \frac{P(c) \cdot P(O|c)}{P(O)}$ — prior, likelihood (joint PDF), posterior, evidence ~ constant

dist of $c \in C$ — $P(c|d)$ ← for

WLOG, represent document as a set of features

$= p(w_1)\, p(w_2|w_1) \cdot p(w_3| x_1 = w_1, x_2 = w_2) \dots$

$p(w_1, \dots w_n | c)$ — 2. naive Bayes assumption (conditional independence, mutually)

$= \prod_{i \in \text{positions}} P(w_i|c)$ — marginal PDF

✓ training: collect $P(c)$, $P(w_i|c)$

✓ $P(c) = \frac{N_c}{N_{doc}}$

MLE ✓ $P(w_i|c) = \frac{\#(w_i, c)}{\# \sum_{w \in V}(w, c)}$

testing → CNB — count

☐ avoid underflow & increase speed (+ than ×) (addition)

2. $\arg\max_{c \in C} \log P(c) + \sum_{i=1}^{n} \log P(w_i|c)$ — linear classifier

$\frac{N_c}{N_{doc}}$ — concatenate, lexicon, spam, not

✓ data sparseness (sparsity)

Add-one / Laplace Smoothing — including zero counts — estimating parameters of statistical model — MLE — category "c" test

if (w_i, c) doesn't appear before — 0 × mult → whole class

✓ 0.1 for not related — tuning — $\log$ ×

✓ $P(w_i|c) = \frac{\#(w_i, c) + 1}{\sum_{w \in V} \#(w, c) + |V|}$ — all words in all classes; distinct words

dict [int, Dict [str, float]]

if $(p w_i | c_1) = \text{N/A}$

The unseen words in training set are dropped completely → $(p w_i | c_1) = 1$

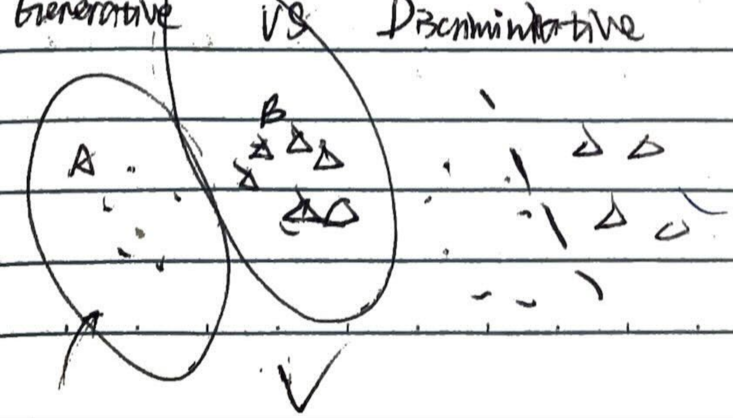
$P(\neg w_i | c) = 1 - 0 = 1$

$c_1 \ c_2$

Merge dictionary

+ weight.

Generative VS Discriminative



No. Title 3

Date Jan. 2

type ✓ ✓ ✓ ✓
token the cat+ likes the dog

(could remove stop word)

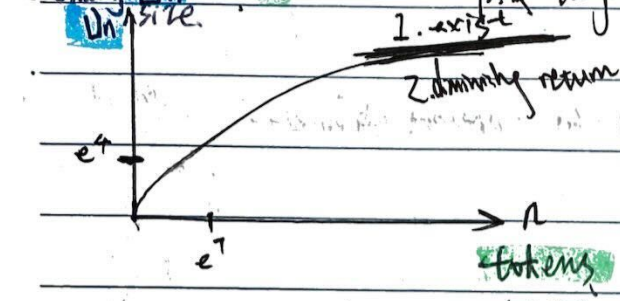
type: any unique word ~ dictionary

token: an instance of a type.

Heap's Law:

type
Vocabulary $|V_n| = kn^{\beta}$

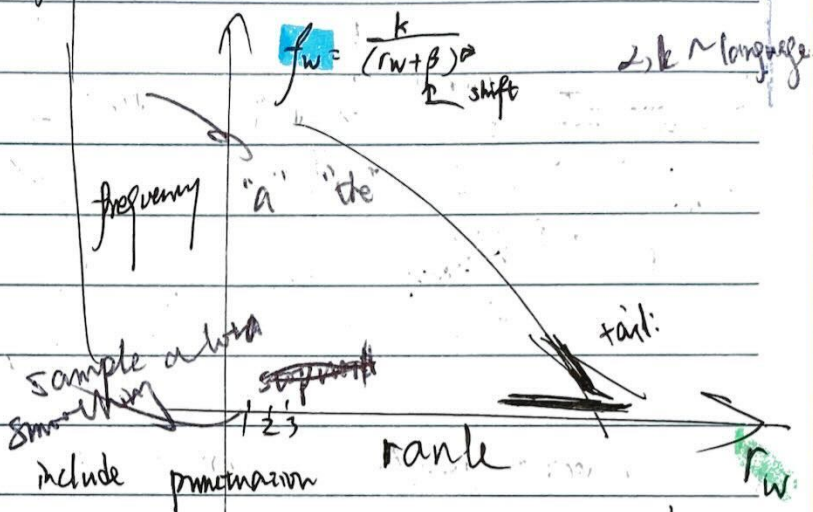
β, k language specific



Zipf's Law:

$f_w = \frac{k}{r_w^{\alpha}}$ ~ frequency rank

nth rank frequency of word $\propto \frac{1}{n}$



unique words vs # total

log $2^0 \sim 2^n$

draw first 10,000 higher ranked words

chart_plot (data, title, xl, yl)

(1). $|V|$ ↑. (2) large

P=0

★ unseen words → smoothing

Estimate. V

Dictionary Compression.

log − log → linear

$\log f_w = -\alpha \log r_w + \log k$

linear least-square

$y = mx + b$

$m = \frac{N\Sigma xy - \Sigma x \Sigma y}{N\Sigma(x^2) - (\Sigma x)^2} = \frac{\text{Covariance}}{\text{Var}(x)}$

$b = \frac{\Sigma y - m\Sigma x}{N} = \bar{y} - m\bar{x}$

→ $y = mx + b$

V↑ variance ↑ mean less

1. exists or not instead of frequency

chps word count → 1

remove duplicate in a file

negation

didn't like this movie → NOT_like NOT_this NOT_movie

punctuation mark

Assumption: Individual tests on data are not linked
documents ~ isolation   i.i.d

ignores Ground Truth and Estimation

effects by chance. (natural variation) Date

Statistical Significance Testing

differences between values

(+ balanced by −)

BST

Null hypothesis: H_0: come from the same distribution baseline sceptical

1°: significance level α. $\alpha = 0.01$ or 0.05.

Reject the null hypothesis with confidence $1-\alpha$ sepcicific

There is less than 5% we fooled ourselves / null hypothesis is false   0.99   0.95.

20 times on different sets of items   (as good as)

random var X over all test sets.   p-value

Sign Test.

count

Pos: $S1 > S2$ +

Neg: $S2 > S1$ −

Tie: $S_1 \sim S_2$ null

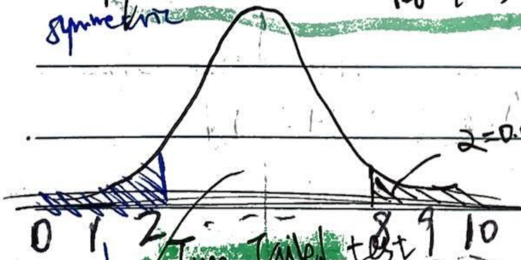
Binomial (N, q)

A negative outcome q

$$P_q(X=k \mid N) = \binom{N}{k} q^k (1-q)^{N-k}$$

$$P_q(X \le k \mid N) = \sum_{i=0}^{k} \binom{N}{k} q^k (1-q)^{N-k}$$

symmetric



balanced   Two Tailed test   Pos : Neg $2P(X \le k) \le 0.05$ harder

$B(N, 0.5)$

$p(+) = p(-)$

too battle data

→ normal approximation to binomial

Type 1 error: declares a difference when it doesn't exist

test α ✓

ties

Type 2 error: declares no difference : β

power: $1-\beta$

- use more data.
- change system (stronger)
- powerful test.   / permutation test

⇒ One-Tailed Test

$$\frac{N!}{i!(N-i)!}$$

Pros P_{Neg}

$$P_q(X \le k \mid N) = \sum_{i=0}^{k} \binom{N}{i} q^k (1-q)^{N-k}$$

0.5

K = less sign $\lceil \frac{null}{2} \rceil$

+   test statistic # min(+/−)   len( )

−   evenly + 0   balanced

0   less. $n = 2 \lceil \frac{null}{2} \rceil$ + plus + minus

common sign   one more odd

representative for the population.

Magnitude classifiers.

Times:	
2	0.65
4	0.663
8	
10	0.665

No. Tick 5.

Date · Feb · 3

Generalise

· Only those characteristics of data general enough.

Ignore specific training data. → test data (not traing set) ✓

Type III Errors

Overtraining testing is based on data

repeated on test data → properties of test data

Same small

"Wayne Rooney" time effects (change meaning)

Confusion Matrix

Truth \ Sys	POS	NEG	TOTAL
POS	V	W	
NEG	X	Y	
Total	V+X	W+Y	

test the model's ability to predict new data that was not used in estimating it.

"Leave-one-out" LOOV

Cross Validation ← Random Stratified

N-Fold

K=1 disjoint subsets

% about percentage

= same +/-

error rate

Ave (score)

train

dev

Test

validation corpus

Test

Significance Testing

mean accuracy

Equally well for each fold.

Folds together. ⇒ one

mini-events

$var = \frac{1}{n}\sum_i (x_i - \mu)^2$

$p(c)$ calculate

$p(x|c)$ probabilities

predict ← validation

error bar

→ fewer significant bugs

Validation Corpus TxTx → parameters

new review set.

2004 → 2016 performance

p-value = 1

No. Tide 6.

Date Feb. 6

Human agreement

Subjective.

· Positive. Negative. Neutral

{ Luke-warm (words)
Pro-con

(threat)

Star rating: Inter-personal differences.

Subjective. Components.

Reader's / writer's perception

Kappa

Majority Class.

$\frac{C_3^2 + C_2^2}{C_5^2}$

Agreement metrics

$P_a = \frac{1}{N} \sum_{i=0}^{N-1} \frac{\text{observed rater}}{\text{possible rater}} =$

above

By chance: $P_e = p^2(\text{pos}) + p^2(\text{NEG})$

majority vote

min(,)

1 ✓

0.8 ✓

0 x small sample size

$K = \frac{P_a - P_e}{1 - P_e}$ ← achieved. reliability of agreement

← attainable

number of categories

$P_e = \sum_{j=1}^{n} \left(\frac{1}{Nk} \sum_{i=1}^{N} n_{ij} \right)^2$ P for each category

pair

N number of documents / reviews

k number of annotators

n_{ij} item i, class j

$P_a = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{k(k-1)} \sum_{j=1}^{n} n_{ij}(n_{ij} - 1)$

annotator, k

item	categories
1	
2	n_{ij}
3	

REVIEW

1 2 3 4 → j / N

pos

NEG

i

k

N

k

j

· Initial State π_i $\sum \pi_i = 1$
start in certain state.

No. 7

Date · Feb · 9   second order ^ more than twice in a row

special start state   special end state

$\{B \; q_1 \cdots q_N \; Z\}$ ($B = q_0$, $Z = q_f$)

Hidden Markov Models

Weather (Rainy & Cloudy)

Markovianity · Q

✓ Hidden states sequence

1. Markov Assumption (first order):

$P(W_t \mid W_{t-1}, \cdots W_1) = P(W_t \mid W_{t-1})$ ✓ when predicting next future of state, the past doesn't matter, only the present state ⇑ second order

JOINT $P(W_1, \cdots W_n) \propto \prod_{t=1}^{n} P(W_t \mid W_{t-1})$

Markov Chains

A_{ij} $q_i \to q_j$

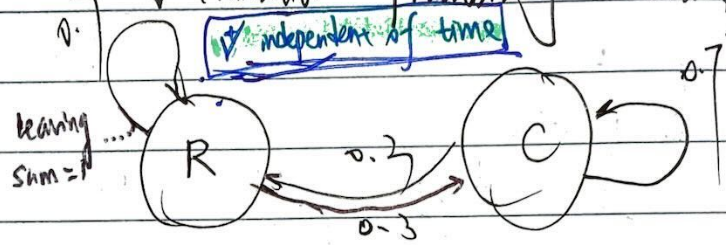
✓ Transition Probability Matrix

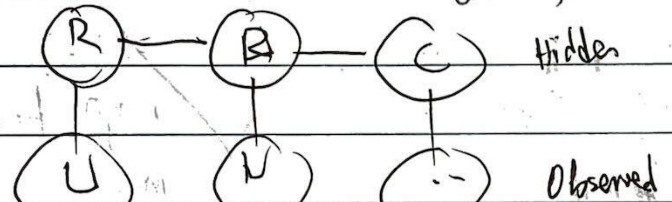
✓ independent of time

Tomorrow

Today	R	C
R	0.7	0.3
C	0.3	0.7

$(N+2) \times (N+2)$





Two State

Finite State Automation

$a_{ij} = P(X_t = S_j \mid X_{t-1} = S_i)$

Sequential

no 1:1 mapping

	U	$\bar{U}$
R:	0.9	0.1
C	0.2	0.8

N emitting hidden states   special S_0, S_f hidden states $X = B X_1 \cdots X_T Z$

$S: S_0 \{S_1, \cdots, S_N\}$ S_f end   ~~hidden states~~ observations $O = O_1 \cdots O_T$

start

M output alphabet

$K: \{k_1, \cdots k_n\}$

K_0 alphabet. Vocabulary k_f observations

1. Output Independence: only current state, not independent of other states or any other observations

$P(O_t \mid X_1, \cdots X_T, O_1, \cdots O_T) = P(O_t \mid X_t)$

· sequence   label

only option state: unlabel

State Transition Probability

State $i \to j$ $\quad a_{ij} = P(X_t = S_j \mid X_{t-1} = S_i)$

$\forall i \quad \sum_{j=0}^{N+1} a_{ij} = 1$

a_{0i}; a_{if} ✓ ; a_{i0} ✗ ; a_{fi} ✗

Σ — a_{01} ... a_{0N} of a_{ij}

a_{N1} ... a_{NN} a_{Nf}

$f \to f$ — ✗

B

✓ Emission Probability

	1	2	3	4	5	6	B	Z
F								
L								
state								

$B = \begin{bmatrix} b_0(k_0) & - & - & - & - & - \\ - & b_1(k_1) & \cdots & \cdots & \cdots & b_N(k_1) \\ - & \vdots & & & & \vdots \\ - & b_i(k_m) & \cdots & \cdots & \cdots & b_N(k_m) \\ - & - & - & - & - & - \end{bmatrix}$

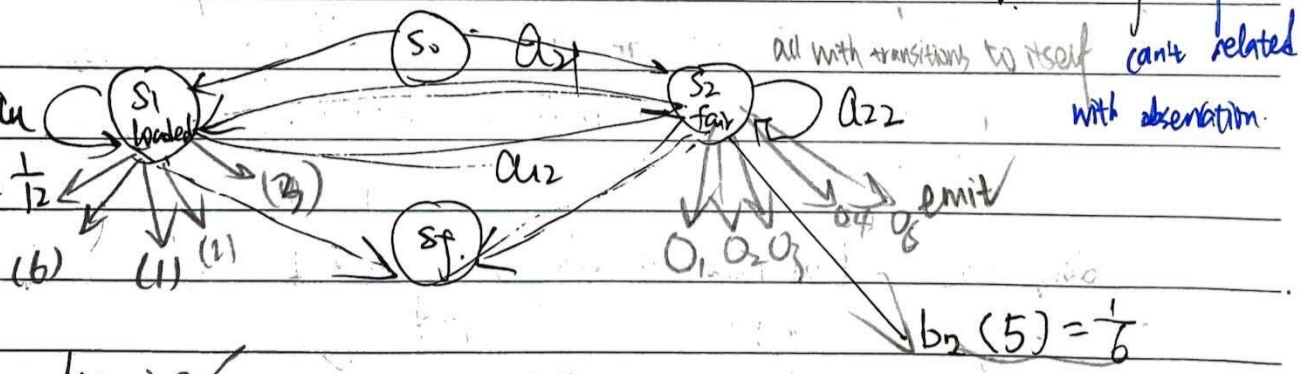
an output alphabet of $K = \{k_1, \ldots k_m\}$ $\quad (N+2) \times (M+2)$

k_0 start symbol. k_f special end.

observation likelihoods

$b_i(k_j) = P(O_t = k_j \mid X_t = S_i)$ *

v. item from ↑ can't related with observation.



Labelled learning

Dataset:

observation. 诊断

hidden states 天气

1. $P(F \to L)$ $\quad \frac{\#(F \to L)}{\#(F \to *)}$

2. $\frac{\#(O_i, S_i)}{\#S_i}$ —— zip

ons B Z 4×4

B→Z : Z→B ?0

Z → 4 -2

→B 4

No. 8

Date

Viterbi Algorithm: hidden state sequence X explains observation O

Decoder $\hat{x} = \underset{x}{\operatorname{argmax}} P(x, O \mid \mu)$

HMM parameter: $\mu = (A, B)$ Conditional

Likelihood Bayes & ∧ drop denominator

$$= \underset{x}{\operatorname{argmax}} P(O \mid X, B)\ P(X \mid A)$$

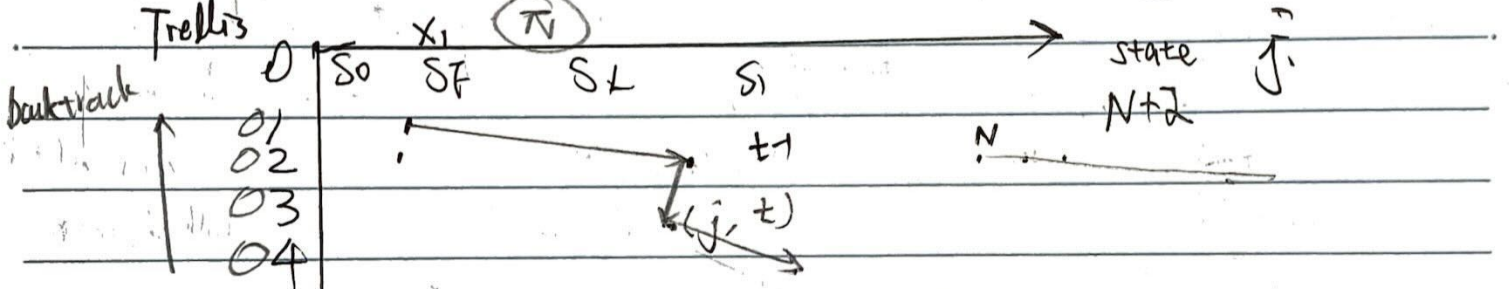
$$= \underset{X_1 \cdots X_T}{\operatorname{argmax}} \prod_{t=1} P(O_t \mid X_t, B)\ P(X_t \mid X_{t-1}, A)$$

$O(N^T)$

DP: Optimal substructure ✓ Overlapping subsolutions ✓ Forward Algo: Greedy

$O(N^2 T)$ first order memorize the pr / only cal once

Trellis



Best route path ending in s_j | $O_1 \cdots O_t$ $O(N^2 T)$ $N(N \cdot T)$

$$\delta_j(t) = \max_{1 \le i \le N} [\delta_i(t-1)\ a_{ij}\ b_j(O_t)]$$

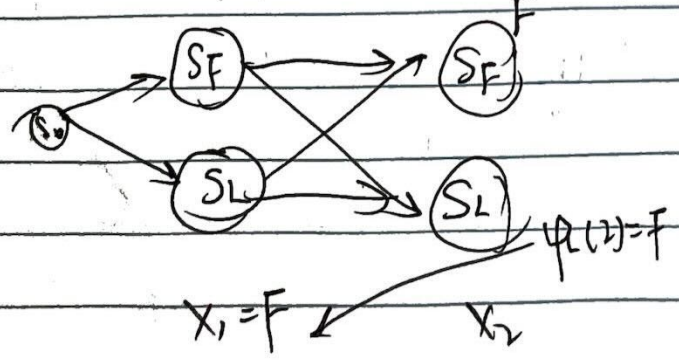
$t-1$ π_j $s_i \to s_j$ O_t $|S|$

$\psi_j(t)$ helper

$t-1$ state index i best previous state. $|S|$

backpointer $[s,t] \leftarrow \underset{}{\operatorname{argmax}}$ $O(|S|^2 \times L)$

$O(T \times |S| + |E|)$ state chg



Q: in a state : sequence to count backtrack

no most likely sequence / state individually

✓ For each class / category!

T:
To detect spam

TP ↑ FP — TN ↑ FN ✓

more weight ↑× ↑×

FP ↓ (unspam→spam) medical

		Sys / Pred		
Gold standard (T)		L	F	Total
	L	TP	FN	Recall
	F	FP (precision)	TN	

difference

system → in fact: Precision of L: $P_L = \frac{TP}{TP+FP}$ — L_{sys} * precisely correct I

true/Gold → in fact: Recall of L: $R_L = \frac{TP}{TP+FN}$ — L_{truth} — sys coverage II

F-measure (spam→unspam) (β^2+1) β factor $F_L = \frac{2 P_L R_L}{\beta P_L + R_L}$ harmonic mean — conservative metric; lower; i: equal importance to FP, FN

Fn

correct / total: accuracy $\frac{TP+TN}{FP+FN+TP+TN}$ 2: unbalanced; baseline: majority class prob (classify all classes to it)

sys v.s. truth

using logarithm — symplify calculation — monotonically increase ✓

when t = 0

table I. $\delta_{sj}(t) = \log(b_{sj}(O_t))$ — a_{os_j}

Iterations: semi-supervised

self-training

0: HMM training data.

1. predict all
2. (Merge)
3. Train on merged

hidden: B Z i o M
O: B Z p ... M

unlabeled data
{ observed: o. for ( ) in Z]
hidden: prediction
}

observed sequences as unlabelled data: larger dataset: ~~at~~ performance ↓✓

~~training~~ dev data: ✓ meet before × ~~no base of~~

seed = 1 ↓ different result.

No. 10

Date

$G = \langle V, E \rangle$

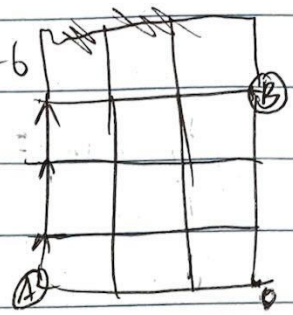
Social Network.

Facebook. people — friend / messages — clustering

Twitter. people — follows → entity; people — retweets → entity

diameter = 6

$\frac{6!}{3!3!}$



Undirected, unweighted graphs.

Distance: Length of (shortest) path between two nodes. — the path that connects two nodes with the fewest edges

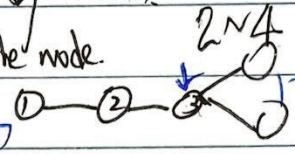
Diameter: Maximum (Longest) distance between any pair of nodes. $\max_{u,v} d(u,v)$

bfs. top

2^4: Degree of a node: # neighbours / links of the node

central & highly connected — periphery

2^4 — diameter = 3 — subclasses



Small World Property

most nodes are not neighbour of each other.

but nodes are likely to be neighbours of each other

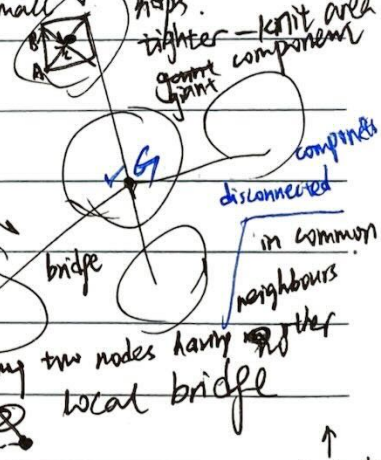
small diameter & high clustering coefficient

closely clustered regions, few links

· can be reached n small hops.

'Chain-links'.

tighter — knit area



giant component: connected; most of nodes in the graph

weak ties: socially distance ties

infrequent interaction.

VS

Strong links: close friends and family

Cut → increase len(path)

Triadic Closure: ✓ A → B strong ties, B → C ⇒ C

↑ Global Clustering coefficient

closed triads / # (closed + open) triads

number of edges between pairs of neighbours of v / # pairs of neighbours of node V.

Random links: $1/d^2$ nodes

Clustering coefficient

edges between pairs of neighbours / # pairs of neighbours of v

k / C_n^2

✓ Gatekeeper

"hub"; center of tightly-knit network

Y → Z

$C_B(v) = \sum_{s \in V} \delta(s|v)$

$\delta(s|v) = \sum_{t \in V} \delta(s,t|v)$ recursively.

No. 11.

Date

"connectors" { remove → disrupt / only one

bridge node: functionality important in information flow; tighter-knit area

Betweenness centrality $\sum$ relative # shortest path (s,t) rely on the node

gate-keeping

~~$\delta(s,t|v)$~~ $C_B(v) = \sum_{s,t \in V, s \neq v \neq t} \frac{\sigma(s,t|v)}{\sigma(s,t)}$ $\frac{\sigma_{s,t}(v)}{\sigma \text{ st.}}$

- total amount of flow it carries.

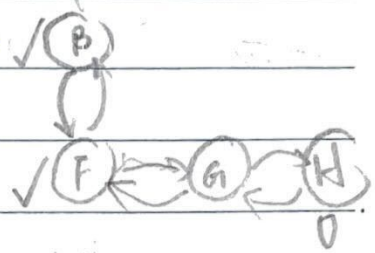
path starting & ending from the node itself don't contribute

$\forall v \in V, C_B(v) = 0$

- storage efficiency.
- and count

$\forall s \in V$. ~~BFS~~ $\rightarrow \sigma(s,t)$

count # v. $\Rightarrow \sigma(s,t|v)$



- Initialization: for each node w: 1
  1. Mark w as unvisited by setting dist [w]. (the dis from s & w) to ∞
  2. Set Pred [w] precede w on a shortest path to [].
  3. Set Paths [w] (the list of all shortest path from s to w) to [].
- While Q (Q [s]; dist[s]=0) not empty:

  v = Q. ~~pop~~ dequeue.

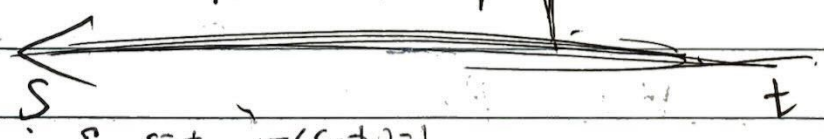
  for w in v's neighbour:

  - if dist [w] = ∞: 1

    dist [w] = dist [v] + 1

    enqueue w.
  - if dist [w] = dist [v] + 1:

    append v to Pred [w]



$\forall s \in V$: $\begin{cases} s = t & \sigma(s,t) = 1 \\ v \in s,t & \sigma(s,t|v) = 0 \end{cases}$

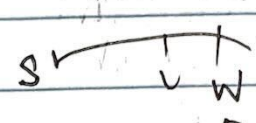
1. set $S(v,t) = 0$
2. BFS. $s \rightarrow t$
3. backward, increment $S(v,t)$ ↑
4. /

No.

Date

1. $\delta(t)=0$ if t is a terminal node
2. Increment $\delta(v)$. based on $\delta(w)$ $v \leftarrow w$.
3. $C_B(v)$.

unique. $\delta(s,t|v)=\frac{1}{0}$



$$\delta(v)=\sum_w (1+\delta(w))$$

$$\delta(v)=\sum_w \frac{\sigma_{s,v}}{\sigma_{s,w}}(1+\delta(w))$$

1. $\forall v$ in V, $C_B(w)=0$
2. $\forall s$ in V:
   $\forall v$ in V:
   1. set $\delta(v)=0$.
   2. While Q is not empty:
      1. $v=Q.pop()$
         $S.push(v)$
      2. w in V. neighbour:
         if $dist[w]=\infty$:
            $dist[w]=dist[v]+1$
            $Q.push(w)$
         if $dist[w]=dist[v]+1$:
            $\sigma(s,w)=\sigma(s,w)+\sigma(s,v)$
            append v to $Pred[w]$.
   3. while S not empty:
      $w=S.pop()$ — correct order.
      $\forall v$ in $Pred(w)$
         $\delta(v)+=\frac{\sigma_{sv}}{\sigma_{sw}}(1+\delta(w))$
         $C[(v,w)]+=c$
      if $w!=s$:
         $C_B(w)+=\delta(w)$

/2

∀ pair of nodes, there is a path between them.

strongly
Connected Component

dfs.(u):
not visited:
visited [V] = true
dfs(.)

for each node:
if u is not visited:
visited [U] = true
connected_Component += 1
dfs(U)

union set

Edge betweeness

$$\sum_{s,t \in V} \frac{\sigma_{s,t|e}}{\sigma_{s,t}}$$

how many
ratio of the # shortest path from s to t between all nodes crossing the edge and the total # shortest paths

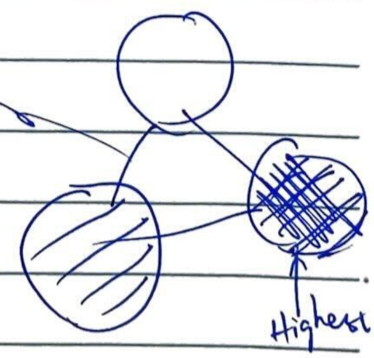
Girvan–Newman : partition of network
While number of connected component < clusters & # edges > 0:
1. Calculate. edge betweenness
2. Rm highest (with ties) e^{-6}
3. Connected Components

closed
triadic closure
<A, B, C> Ratio

= 1 fully connected ✓

std. ( ) ↓
variance (. betweenness centrality)

nodes → similar.



cluster criteria
✓ connectivity is uniform

No.

Date

General Case of S: directed T: undirected.

X_S can't compare. X_T.

Bridge: an edge that connects two nodes that would otherwise (remove) be in disconnected components of the graph.

A local bridge: an edge joining two nodes that have other neighbours in common.