# Lecture 13 News Filtering

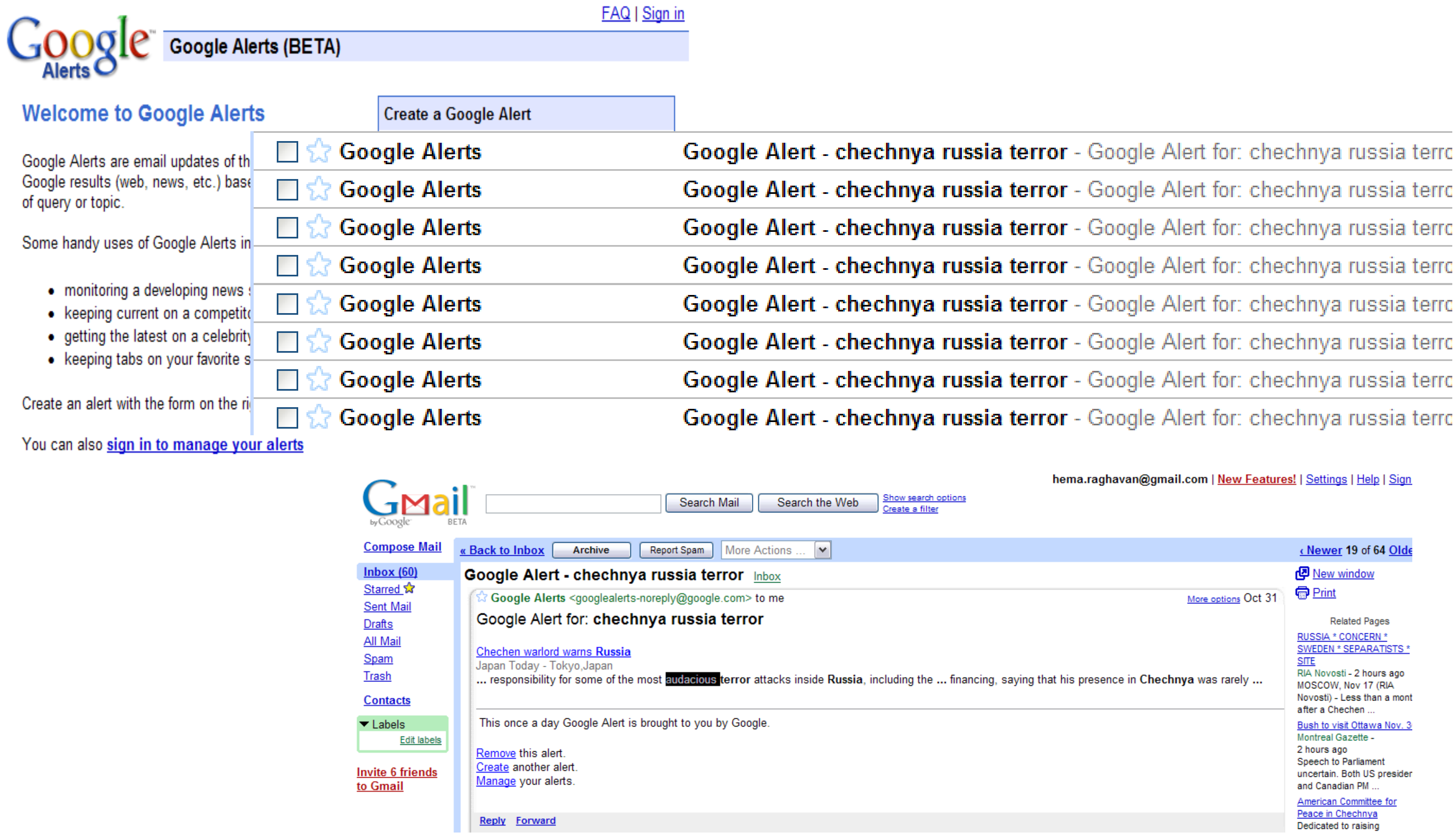
Based on Hema Raghavan

# News – Yesterday and Today

# Outline

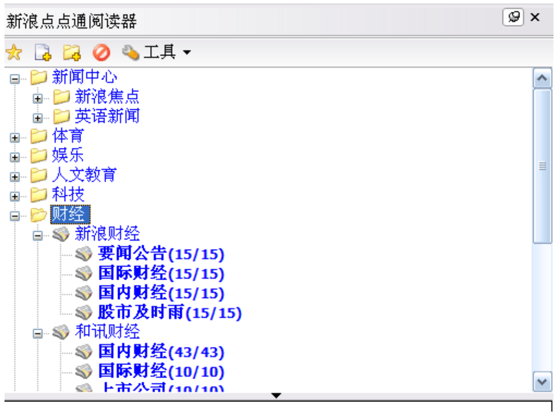
- **News Filtering**
- Topic Detection and Tracking
- Document Clustering

# News Filtering – Google Alerts



# News Filtering – RSS feeds

- RSS (Really Simply Syndication)
- XML feeds
- Lots of News sites provide it now
- Web content providers can easily create and disseminate feeds of data that include news links, headlines, and summaries.




# Outline

- *News Filtering*
- **Topic Detection and Tracking**
- Document Clustering

# Topic Detection and Tracking

- **What is TDT**
- Data
- Approaches to tracking
- Evaluation of TDT
- First story detection (FSD)

# What is TDT?

- Automatic organization of news by events
  - Wire services and broadcast news
  - Organization on the fly--as news arrives
  - No knowledge of events that have not happened
- Topics are event-based topics
  - Unlike subject-based topics in IR (TREC)
- Events such as…

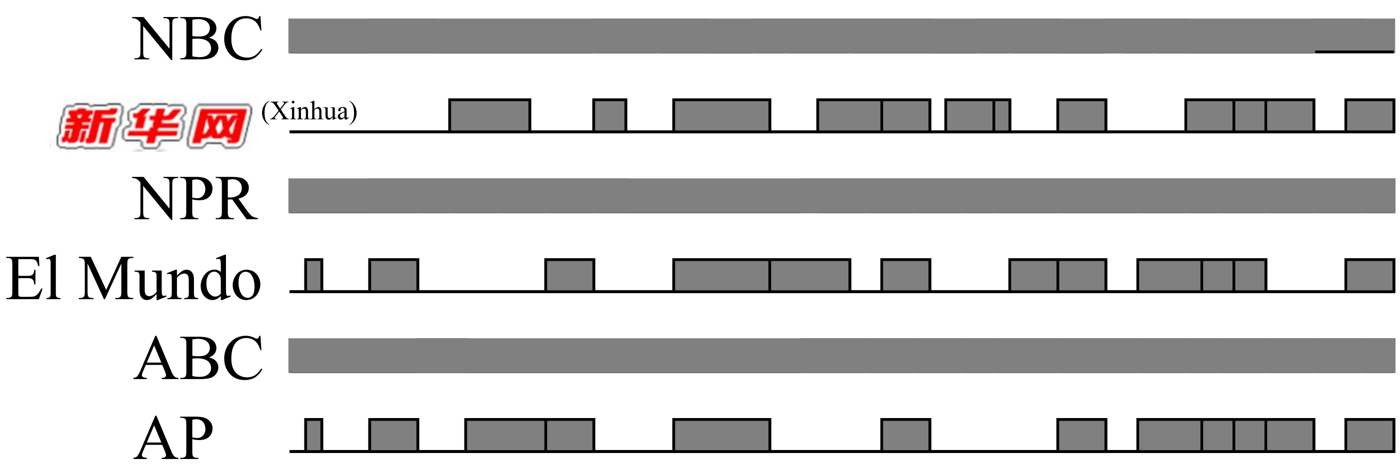

Karla Faye Tucker executed

Nuclear Test in India

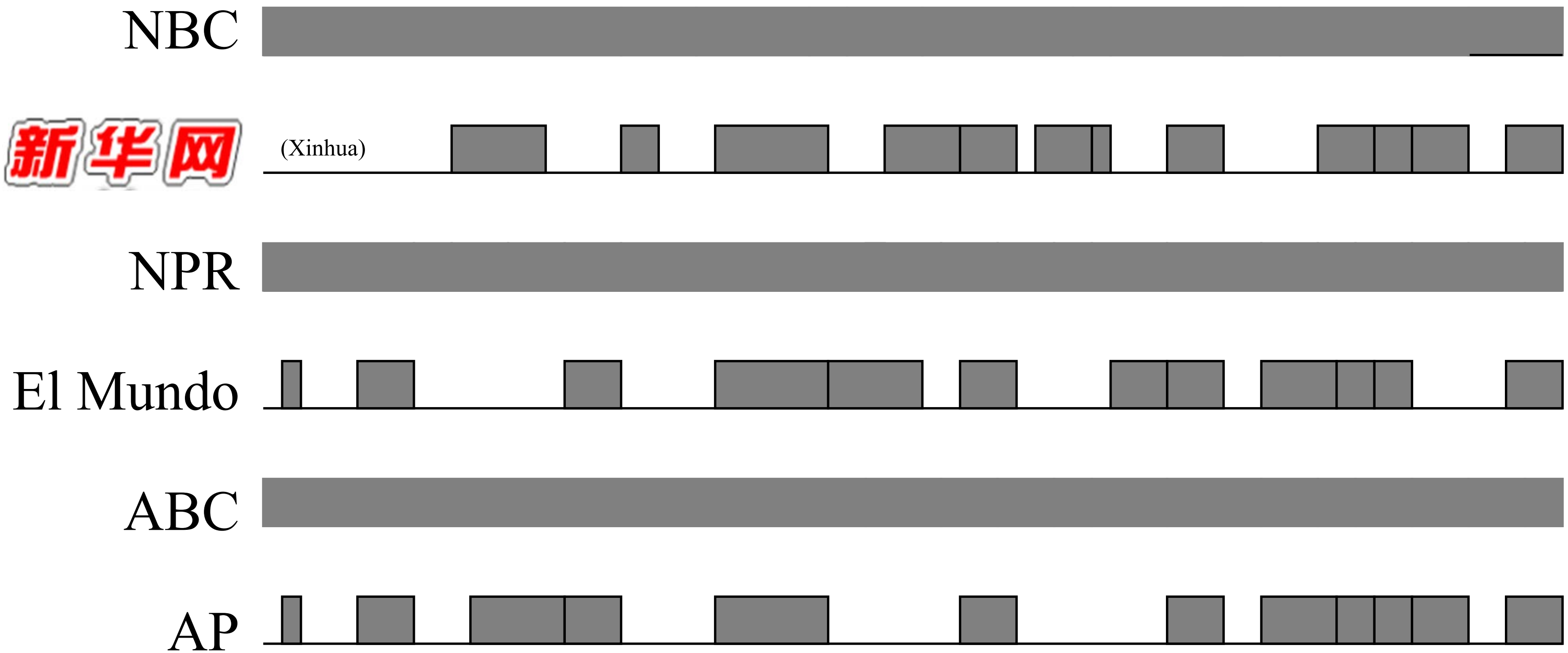
JJ rescued!

# But the reality is...

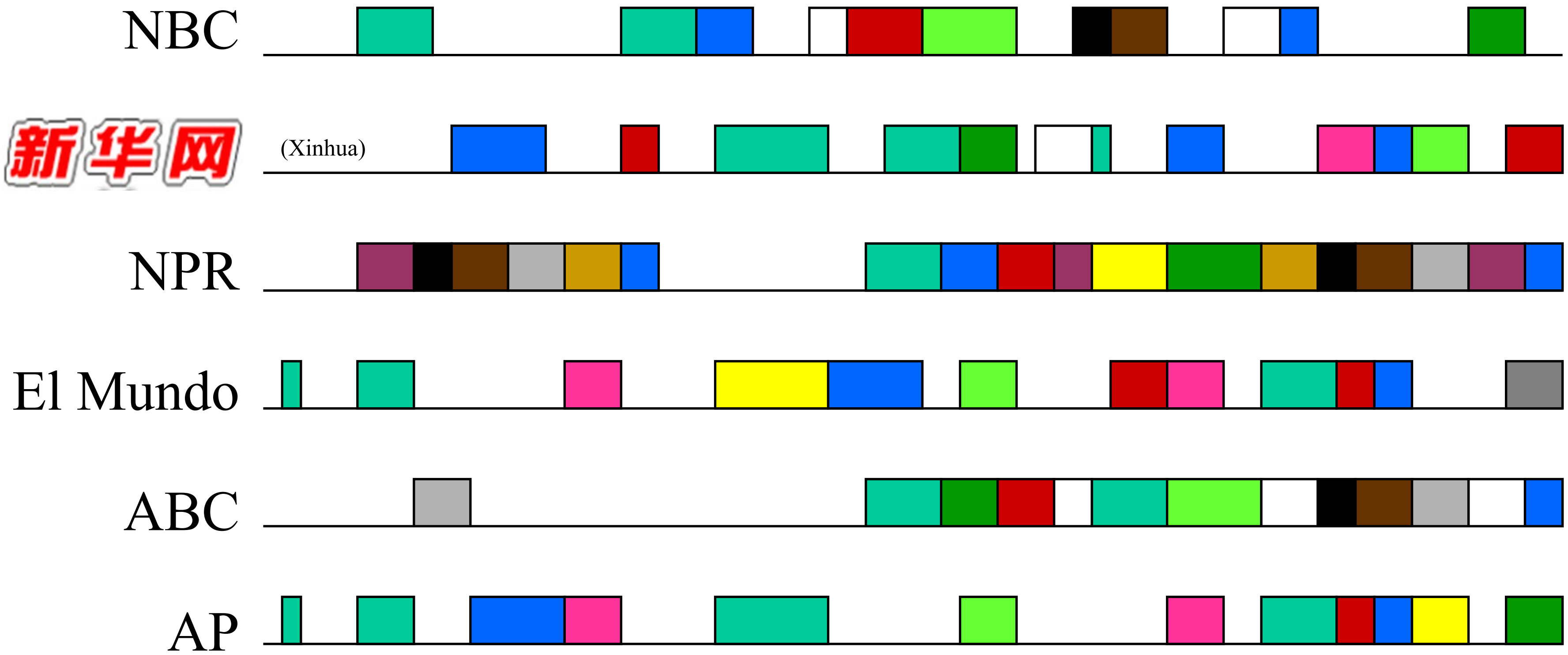
- Events/topics are not given
- Do not know story boundaries for broadcast sources
- Do not know where all of the news is in broadcast sources



# So TDT means going from this...



# …to this



# What TDT is, summary

- Five technology evaluation tasks
  - **Story segmentation** – find story boundaries in broadcast news
  - **Topic tracking** – given sample stories, find rest on same topic
  - **First story detection** – detect onset of new event in the news
  - **Cluster detection** – group stories into events (unsupervised)
  - **Story link detection** – decide if two stories discuss same event
- Tracking and detection on *event*-based topics
  - Though most approaches are the same as those used for subject-based tasks
- All tasks are on-line (not batch) evaluations
  - Cluster detection task has a “retrospective” (回顾性的) variation

# Topic Detection and Tracking

- ***What is TDT***
- **Data**
- Approaches to tracking
- Evaluation of TDT
- First story detection (FSD)

# TDT data

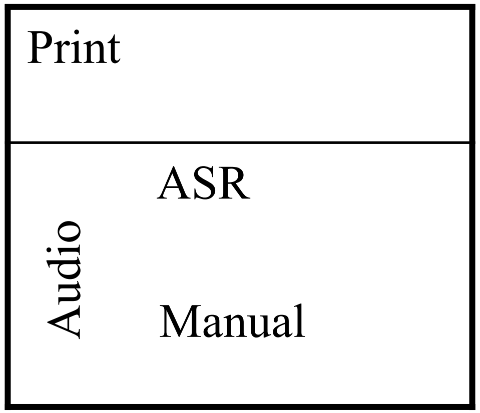
- TDT4 corpus
  - Oct 2000 – Jan 2001
- News in Different Languages

| English | | |
|---|---|---|
| Foreign | Mandarin | MT<br>Nat |
| | Arabic | MT<br>Nat |

Machine Translated
SYSTRAN

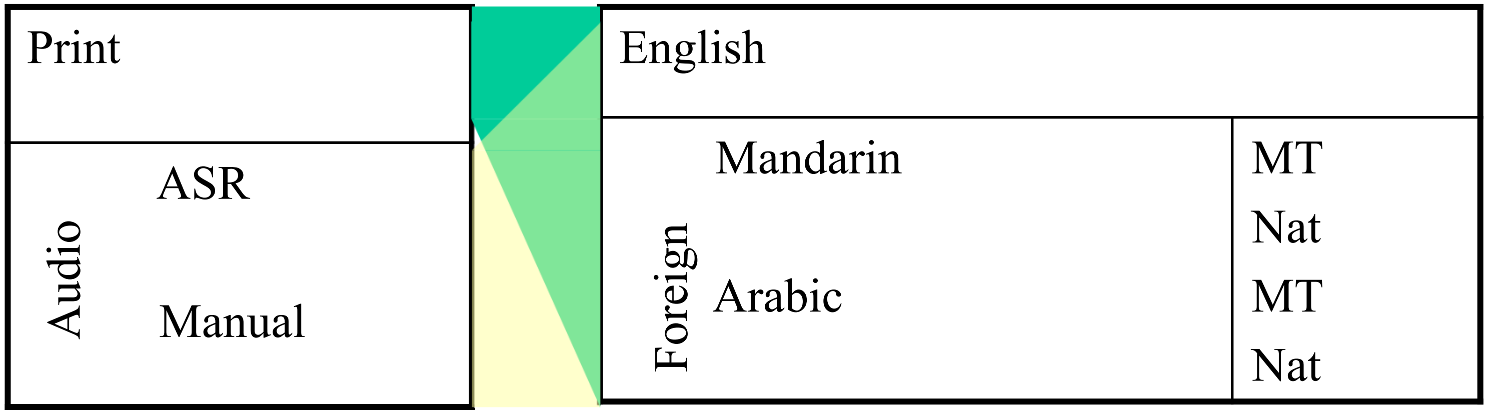
# TDT data

- TDT4 corpus
  - Oct 2000 – Jan 2001
- News from Different Sources



# TDT data

- TDT4 corpus
  - Oct 2000 – Jan 2001
- News from Different Sources



<DOC>
<DOCNO> CNN19981002.1600.0051 </DOCNO>
<DOCTYPE> NEWS STORY </DOCTYPE>
<DATE_TIME> 10/02/1998 16:00:51.26 </DATE_TIME>
<BODY>
<TEXT>
new details are out about president clinton's relationship with monica lewinsky, the house judiciary committee has released the last major batch of evidence collected by ken starr in his investigation. the 4,600 pages made public today include transcripts of linda tripp's secret tape recordings of her conversations with lewinsky. testimony by most of the major witnesses who appeared before the grand jury is also included. while this new material doesn't contain the controversial details of previously released documents, it does add color to the contacts between while this new material doesn't contain the controversial details of previously released documents, it does add color to the contacts between tripp and lewinsky.

<DOC>
<DOCNO> CNN19981002.1600.0051 </DOCNO>
<DOCTYPE> NEWS </DOCTYPE>
<TXTTYPE> ASRTEXT </TXTTYPE>
<TEXT>
YOU'RE DETAILS ABOUT PRESIDENT CLINTON'S RELATIONSHIP WITH MONICA LEWINSKI TODAY THE HOUSE JUDICIARY COMMITTEE HAS RELEASED THE LAST MAJOR BATCH OF EVIDENCE COLLECTED BY KEN STARR IN HIS SEVEN MONTH PROBE FORTY SIX HUNDRED PAGES MADE PUBLIC TODAY INCLUDE TRANSCRIPTS OF LINDA TRIP SECRET TAPE RECORDINGS OF CONVERSATIONS WITH HER TESTIMONY BY MOST OF THE MAJOR WITNESSES TO APPEAR BEFORE A GRAND JURY IS ALSO INCLUDED WHILE THIS NEW MATERIAL DOESN'T CONTAIN THE CONTROVERSIAL DETAILS OF PREVIOUSLY RELEASED DOCUMENTS IT DOES ADD COLOR THE CONTACTS BETWEEN PRINT AND LOWENSTEIN

# Topic Detection and Tracking

- *What is TDT*
- *Data*
- **Approaches to tracking**
- Evaluation of TDT
- First story detection (FSD)

# The Tracking Task

- The system is given one training document $T_j$ per story.
- Stories come in sequence $S_1 \ldots S_n$
- How can we make the decision of on-topic or not for each story?

# The Tracking Task

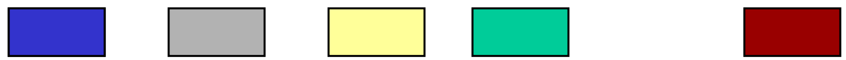
- Stories with similarity above a threshold $thresh_{yes/no}$ to the training story are marked YES

Truth N N Y Y

- How can we measure the performance of the system?

# The Tracking Task

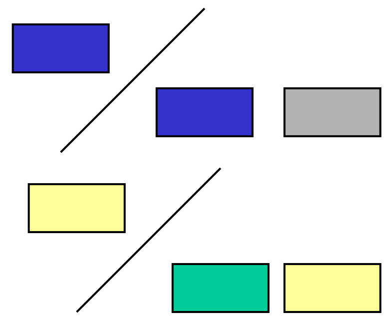
- Misses and False Alarms
- What are the differences of these measures and P/R?



Truth N N Y Y

$$P_{fa} =$$

$$P_{miss} =$$

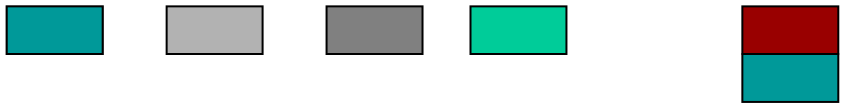


# The Tracking Task-- Adaptation

- Consider that $\mathrm{sim}(T_j, S_4) > \mathrm{thresh}_{\mathrm{adapt}}$

# The Tracking Task-- Adaptation

- add story $S_4$ to topic $T_j$ and recompute model



- Danger of adapting with a false alarm story.

# The Tracking task -- adaptation

- Adaptation
  - If sim($T_j$, $S_i$) > $\text{thresh}_{\text{yes/no}}$ then story $S_i$ is on topic $T_j$
  - If sim($T_j$, $S_i$) > $\text{thresh}_{\text{adapt}}$ add story $S_i$ to topic $T_j$
    and recompute model
  - $\text{thresh}_{\text{adapt}} > \text{thresh}_{\text{yes/no}}$

# Vector Space approach to Tracking

- Treat stories as "bags of words"
- Really as a vector of weighted features
  - Features are word stems (no stopwords)
  - Weights are a variant of tf-idf

IDF is incremental or retrospective

$$S = s_1 \ldots s_{|V|}$$

# Vector Space approach to Tracking

- Compare vectors by cosine of angle between the story and the topic.
  - If use same words in same proportion, stories are the same
  - If have no words in common, are about different topics

$$sim(S,T) = \frac{\sum_w s_w t_w}{\sqrt{\sum_w s_w^2 \sum_w t_w^2}}$$

# Topic Detection and Tracking

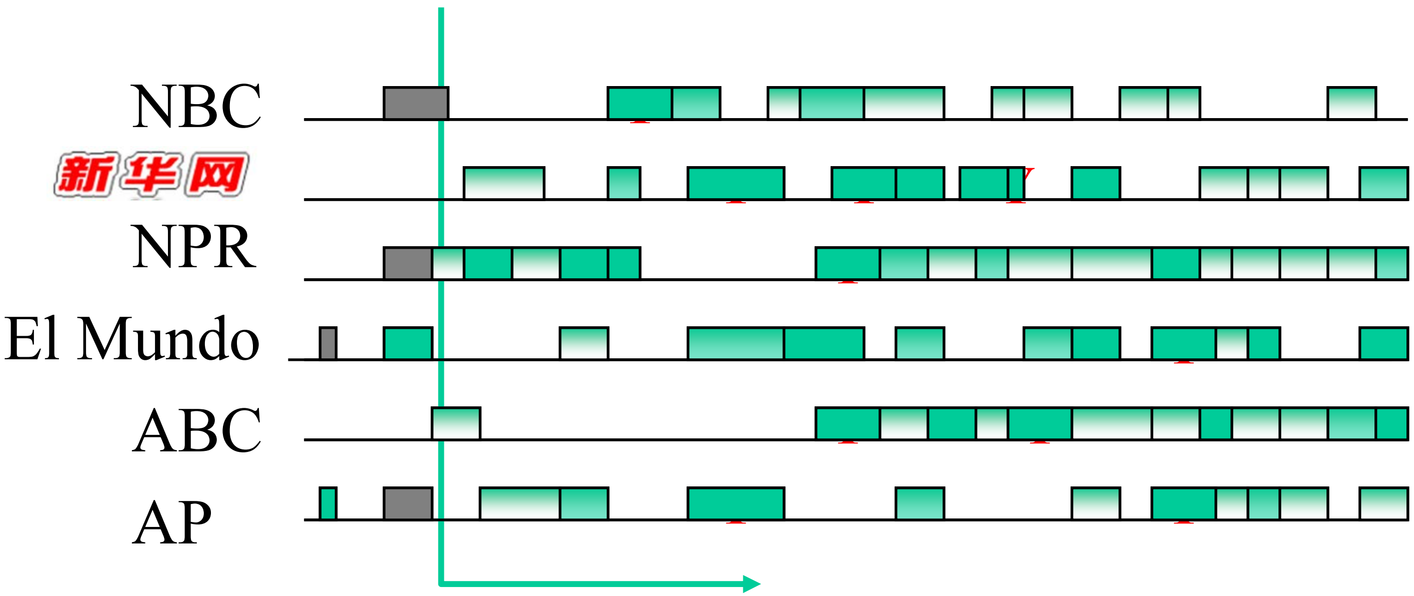
- *What is TDT*
- *Data*
- *Approaches to tracking*
- **Evaluation of TDT**
- First story detection (FSD)

# Measuring progress in TDT

- All tasks viewed as detection tasks (yes/no)
  - Is there a story boundary here?
  - Is this story on the topic being tracked?
  - Are these two stories on the same topic?
- Evaluations based on miss and false alarm
- Use linear combination as cost function

# Evaluating tracking

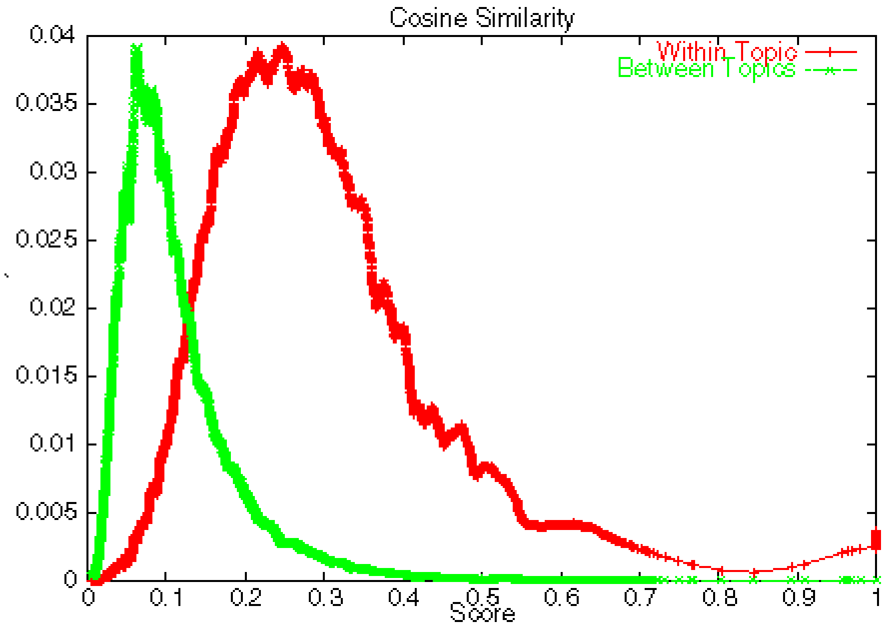
- Perfect tracker says YES to on-topic stories and no to all other stories
- In reality, system emits confidence of topic



# Evaluating tracking (cont.)

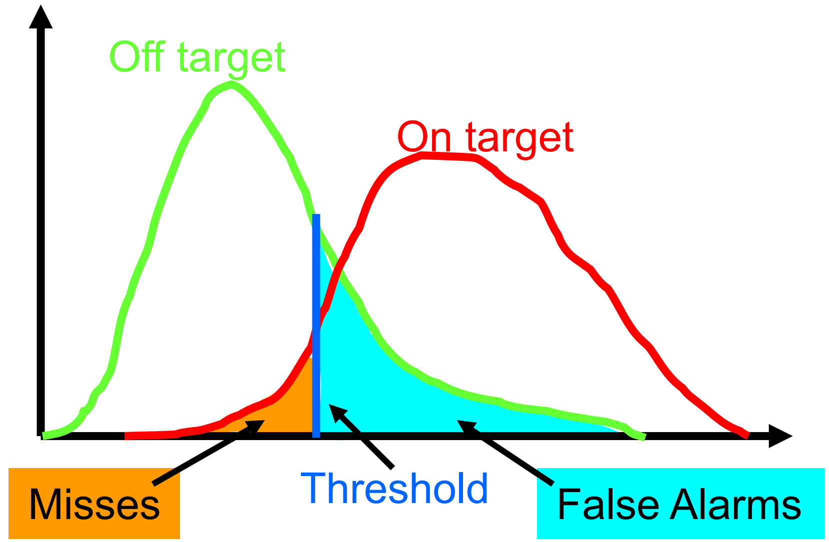
- At every score, there is a miss and false alarm rate
  - Any on-topic stories below score are misses
  - Any off-topic stories above score are false alarms
- Plot (false alarm, miss) pairs for every score
  - Result is a ROC curve (Relative Operating Characteristic)
  - TDT uses a modification called the “DET curve” or “DET plot” (Detection error tradeoff)

# What is a DET plot?

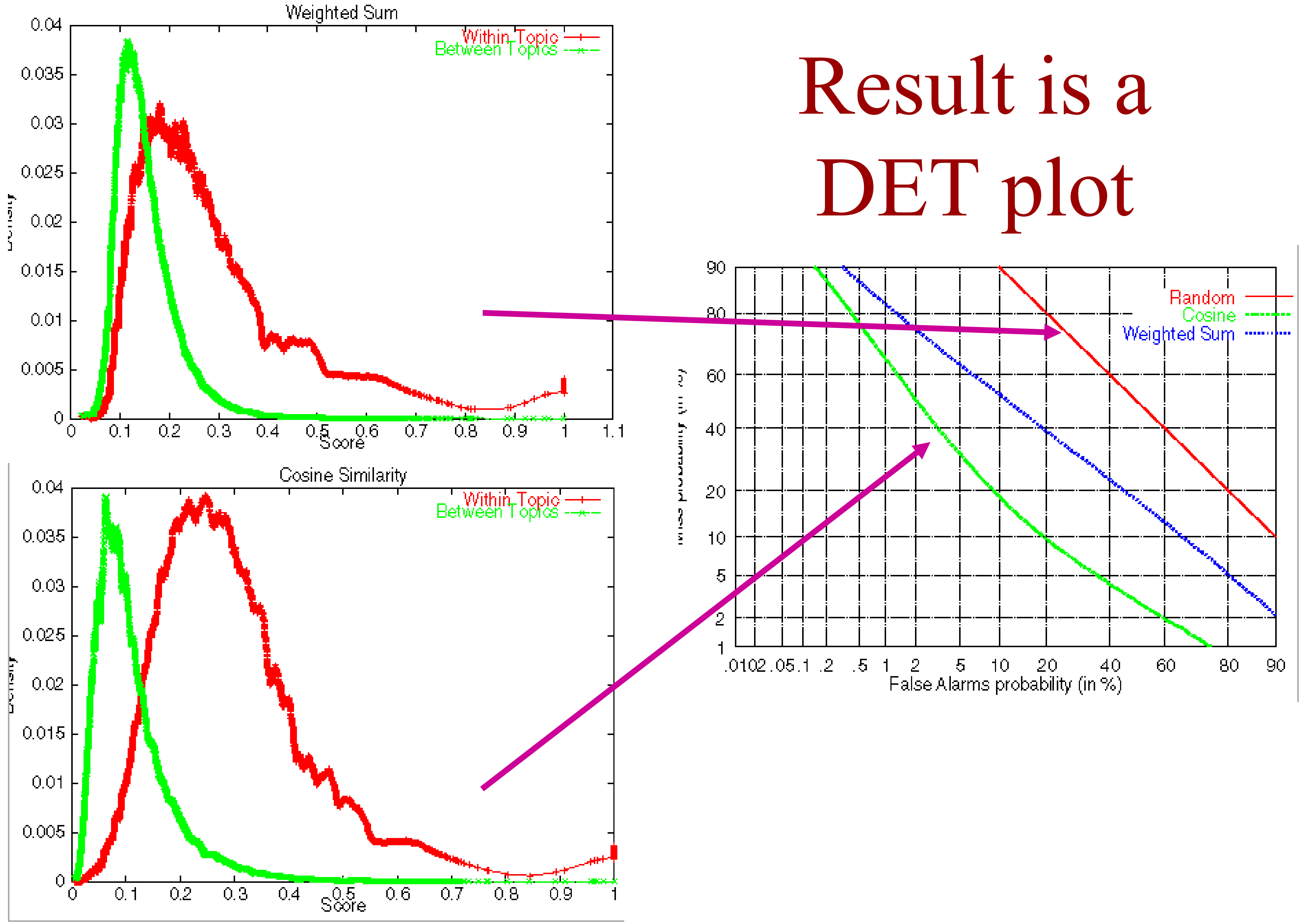



- Sweep through scores
- Note P(miss) and P(fa)
- Plot values at every score
- Plot of distribution of scores

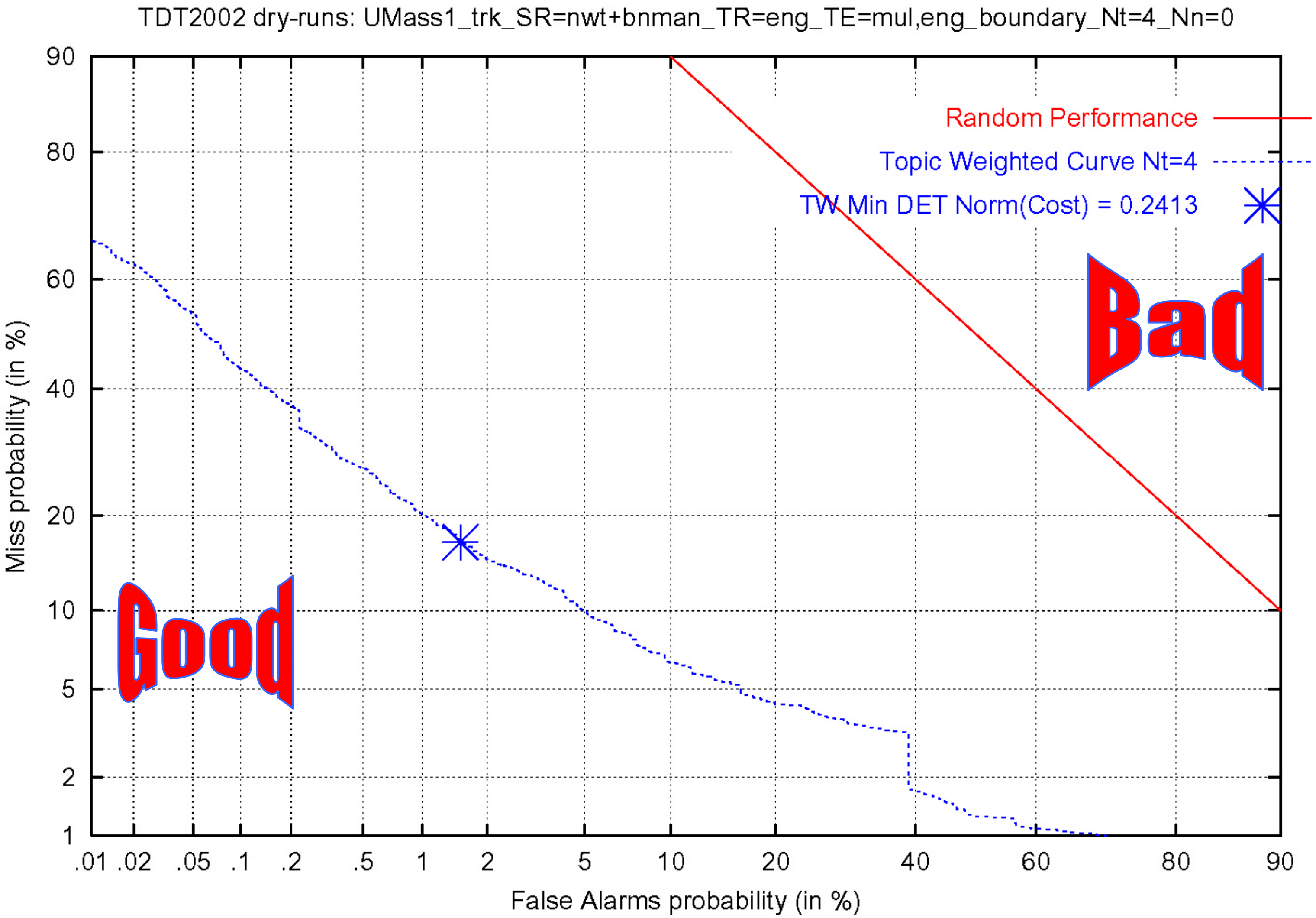
- Green curve on left is “no”
- Red curve on right is “yes”
- X axis represents scores



# Result is a DET plot



# Tracking DET curve (UMass, 2002)

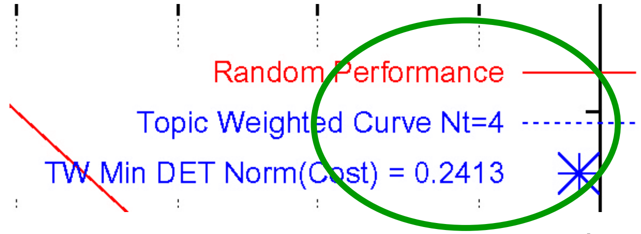


# Evaluation with cost function

- Systems must choose "hard" decision point
  - Score that optimizes system performance
  - Determines a miss and false alarm pair
- Measure by cost (e.g., "tracking cost")

$$\begin{aligned} C_{miss} &= 1.0 \\ C_{fa} &= 0.1 \\ P_{target} &= 0.02 \end{aligned}$$

$$\begin{aligned} C_{track} &= C_{miss} \cdot P_{miss} \cdot P_{target} \\ &+ C_{fa} \cdot P_{fa} \cdot (1 - P_{target}) \end{aligned}$$




$$(C_{track})_{norm} = C_{track} \div min \left\{ \begin{array}{l} C_{track, P_{miss}=1, P_{fa}=0} \\ C_{track, P_{miss}=0, P_{fa}=1}) \end{array} \right\}$$

- Topic Weighted

# Topic Detection and Tracking

- *What is TDT*
- *Data*
- *Approaches to tracking*
- *Evaluation of TDT*
- **First story detection (FSD)**

# First story detection (FSD)

(Some slides are based on slides of J. Allan)

# First Story Detection

- Automatically identify the first story on a new event from a stream of text
- Applications
  - Intelligence services
  - Finance: Be the first to trade a stock

# Examples

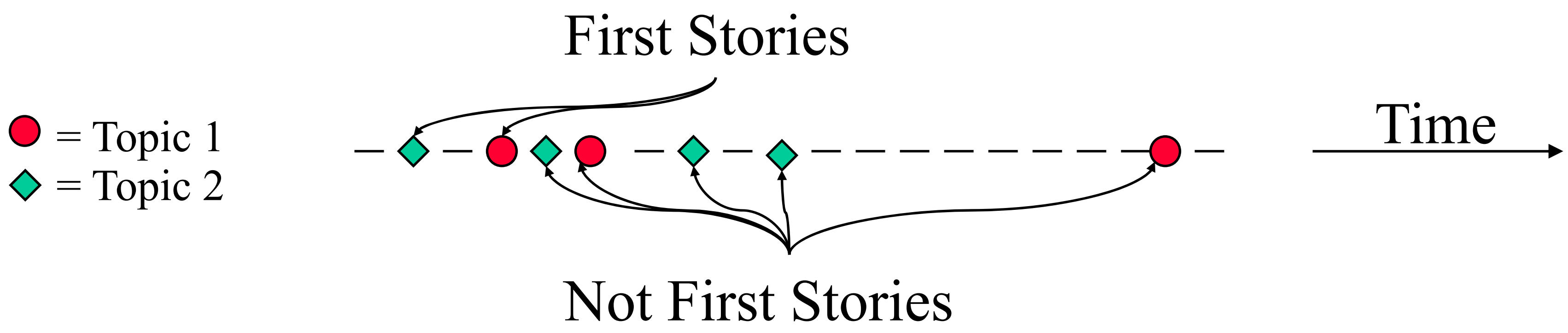
- 2002 Presidential Elections
- Thai Airbus Crash (11.12.98)
  - **On topic:** stories reporting details of the crash, injuries and deaths; reports on the investigation following the crash; policy changes due to the crash (new runway lights were installed at airports).
- Euro Introduced (1.1.1999)
  - **On topic:** stories about the preparation for the common currency (negotiations about exchange rates and financial standards to be shared among the member nations); official introduction of the Euro; economic details of the shared currency; reactions within the EU and around the world.

# First Story Detection

- Other technologies don't work for this
  - Information retrieval
  - Text classification
  - Why?

# The First-Story Detection Task

***To detect the first story that discusses a topic, for all topics.***




- There is no supervised topic training

  (like Topic Detection)

# Definitions

- **Event:** A reported occurrence at a specific time and place, and the unavoidable consequences.
  - Specific elections, accidents, crimes, natural disasters, etc.
- **Activity:** A connected set of actions that have a common focus or purpose
  - campaigns, investigations, disaster relief efforts, etc.
- **Topic:** a seed event or activity, along with all directly related events and activities
- **Story:** a topically cohesive segment of news that includes two or more DECLARATIVE(陈述性的) independent clauses（分句） about a single topic

# Definitions

- **Event** … c time and place,
- … lisasters, etc.
- … have a common
- … ts, etc.
- … with all directly
- … ews that includes
- … nt clauses about

沪深股市今暴跌 受上调印花税传闻影响



事件趋势图

受市场传闻上调印花税等因素拖累，沪深股市12日突现暴跌。上证综指失守3000点重要心理关口，出现5%以上的巨大跌幅，深证成指跌幅则高达近7%，双双创下一年多来的最大单日跌幅。 当日沪深股市双双低开。上证综指开盘报点，最初一个小时窄幅盘整，并一度出现红盘。但上摸点的全天高点后，沪指突然快速下挫，相继跌破3100点和3000点两大整数位，尾盘下探点后，以点报收，较前一交易日收盘大跌点，跌幅达到5.16%。 深证成指失守13000点整数位，收盘报点，跌点，跌幅高达7%。 伴随股指暴跌，沪深两市个股普跌，仅有107只交易品种上涨 全文>>

**最新报道**

- 股市强劲反弹的概率有多大？ 新浪 11-14 08:39
- 官方否认调印花税 央行回应加息传闻 新浪 11-13 04:16
- A股创14个月最大单日跌幅 惨状只能排到历史第17 搜狐 11-14 09:34
- 股市周五暴跌，虽然上周已经清仓观望，可是跌势如此之 天涯 11-13 01:00
- 为什么传言印花税上调 新浪 11-13 15:28
- 高盛报告引发周五暴跌 大盘疑似假摔 A股周五重挫 财政部辟谣上调印花税 央行副行长马德伦释疑周小川“池子论” 和讯 11-13 06:59
- A股再次演绎黑色星期五 一份高盛报告引发暴跌 天涯 11-13 01:00
- 短期调整基本确立 网易 11-13 13:38
- 周评：暴跌原因解析及应对策略 网易 11-13 09:39

**已有0条评论** 评论此

还没有网友对此事件

**发帖区**

昵称： 匿名 登录

# Definitions

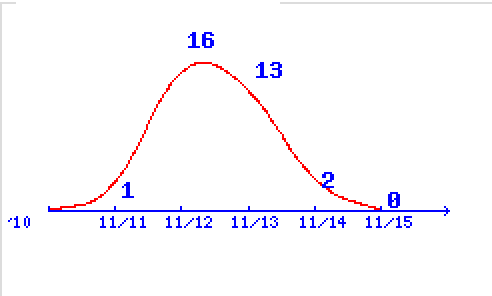

- ... e and place,
  ... s, etc.
- ... a common
- ... ll directly
- ... at includes
  ... ises about

# First Story Detection (FSD)

- First story detection is an unsupervised learning task.
- On-line vs. Retrospective
  - On-line: Flag onset of new events from live news feeds as stories come in
  - Retrospective: Detection consists of identifying first story looking back over longer period
- Lack of advance knowledge of new events, but have access to unlabeled historical data as a contrast set
- FSD input: stream of stories in chronological order simulating real-time incoming document stream
- FSD output: YES/NO decision per document

# Patterns in Event Distributions

**Event** 沪深股市今暴跌 受上调印花税传闻影响



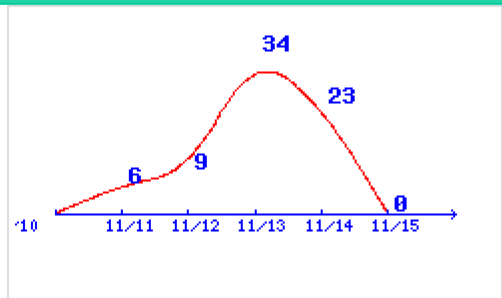
事件趋势图

受市场传闻上调印花税等因素拖累，沪
跌。上证综指失守3000点重要心理关口，出
跌幅，深证成指跌幅则高达近7%，双双创下
日跌幅。 当日沪深股市双双低开。上证
初一个小时窄幅盘整，并一度出现红盘。但
后，沪指突然快速下挫，相继跌破3100点和
位，尾盘下探点后，以点报收，较前一交易
幅达到5.16%。 深证成指失守13000点
跌点，跌幅高达7%。 伴随股指暴跌，沪
仅有107只交易品种上涨 全文>>

- … tend to be
- … milar stories is
- A signi… in term frequen…
  - typical of stories re… unseen proper nou…
- Events are typicall… window of 1- 4 we…

What's the difference of distributions for Event and Activity?

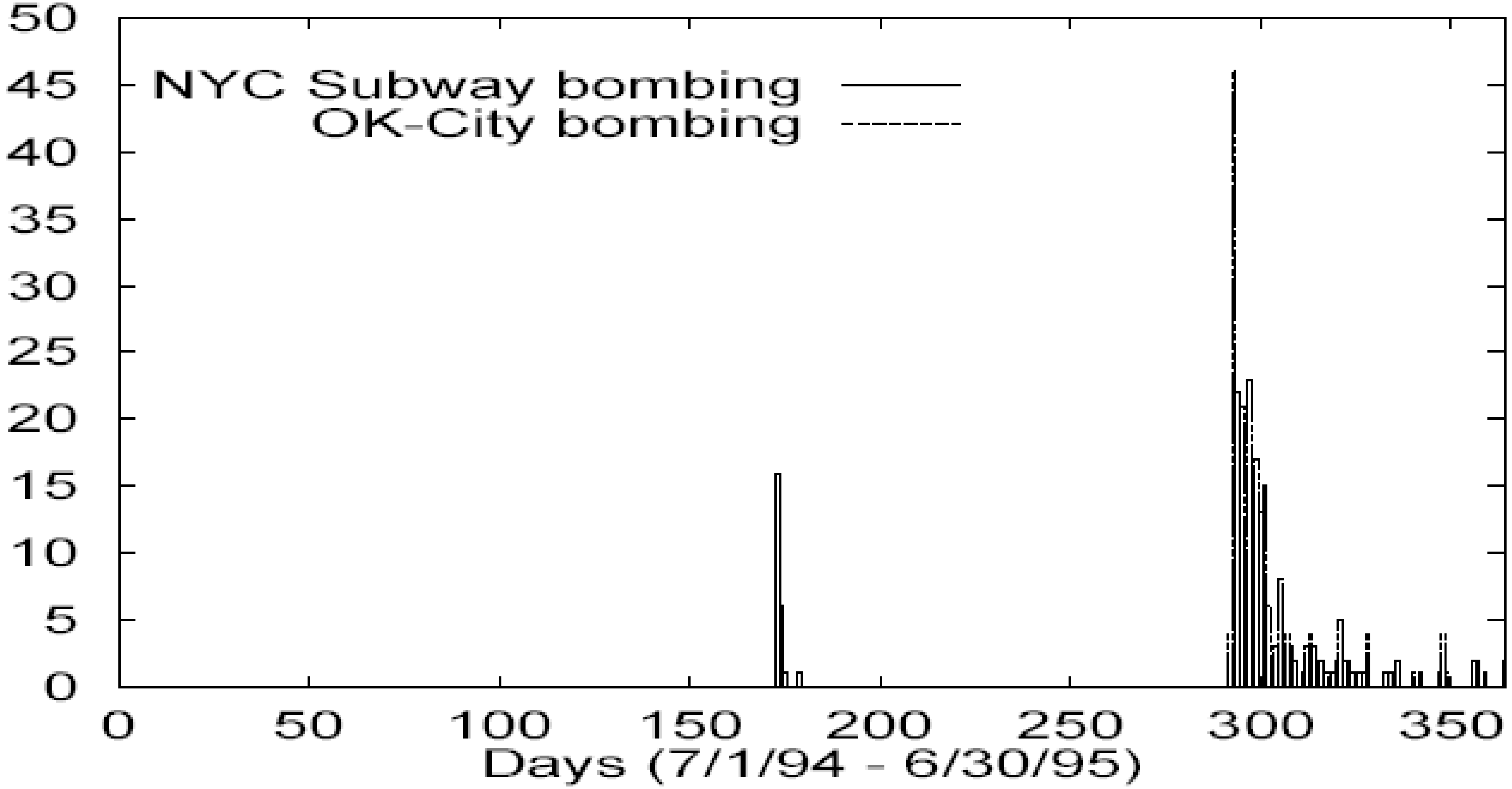
**Activity：** 胡锦涛出席APEC会议



事件趋势图

11月13日，国家主席胡
第十八次领导人非正式会议期间
新华社记者 李学仁摄 11月
出席亚太经合组织第十八次领导
首相菅直人会晤。 新华社记者
主席胡锦涛在横滨出席亚太经合
议期间应约同日本首相菅直人会
新华网日本横滨11月13日电 国
经合组织第十八次领导人非正式
人会晤，进行了交 全文>>

# Similar Events over Time



# Ideas?

# Approach 1: KNN

- On-line processing of each incoming story
- Compute similarity to all previous stories
  - Cosine similarity
  - Language model
  - Prominent terms
  - Extracted entities
- If similarity is below threshold:
  - new story
- If similarity is above threshold for previous document d:
  - assign to topic of d
- Optimal threshold can be chosen based on historical data
  - Threshold is not topic specific!

# Variant: Single Pass Clustering

- Assign each incoming document to one of a set of topic clusters
- A topic cluster is represented by its centroid (vector average of members)
- For incoming story compute similarity s with centroid
- As before:
  - $s>\theta$: add document to corresponding cluster
  - $s<\theta$: first story!

# Approach 2: KNN + Time

- Only consider documents in a (short) time window
- Compute similarity in a time weighted fashion:

$$score(x) = 1 - \max_{d_i \in window} \{ \frac{i}{m} sim(\vec{x}, \vec{d_i}) \}$$

- m: number of documents in window,
- $d_i$: $i^{th}$ document in window
- Time weighting significantly increases performance.

# Single Pass (R.Papka, J. Allan,1998)

- Use feature selection to build a query *q* for the content of each document
- Compute the relevance of a new document *d* with all existing queries in memory
- If there is no query that gets the relevance value above a given threshold, then *d* talk about a new event
- Time is put into consideration in the determination of FSD threshold
- Threshold model:

$$\theta(q^i, d^j) = 0.4 + p * (eval(q^i, d^j) - 0.4) + tp * (j - i)$$

$$eval(q, d) = \frac{\sum_{i=1}^{N} w_i \cdot d_i}{\sum_{i=1}^{N} w_i}$$

$$d_i = belief(q_i, d, c) = 0.4 + 0.6 * tf * idf$$

- $w_i$ is the relative weight of a query feature $q_i$, its value is depended on the feature selection method
- idf is computed on a stand-alone document collection rather than the on-line document collection
- p, tp are weights optimized by experiments, c is a given collection of documents

# FSD - Results

Umass , CMU: Single-Pass Clustering

| System | Miss Rate | F/A Rate | Recall | Precision | F1 |
|---|---|---|---|---|---|
| UMASS | 50% | 1.34% | 50% | 45% | 0.45 |
| CMU | 59% | 1.43% | 41% | 38% | 0.39 |
| DRAGON | 58% | 3.47% | 42% | 21% | 0.28 |

# Discussion

- Hard problem
- Becomes harder the more topics need to be tracked. Why?
- Second Story Detection much easier than First Story Detection
- Example: retrospective detection of first 9/11 story easy, on-line detection hard

# Hierarchical topic detection

- a new task in the TDT 2004 evaluation
- aims to organize a collection of unstructured news data in a directed acyclic graph (DAG) structure
- allow stories to be assigned to multiple cluster

# TDT 5 Corpus

**Table 1. TDT 5 corpus statistics**

| | *TDT3* | *TDT5* |
|---|---|---|
| Arabic stories | 0 | 72,910 |
| English stories | 34,600 | 278,109 |
| Mandarin stories | n.a. | 56,486 |
| Total stories | n.a. | 407,505 |
| Annotated topics | 160 | 250 |

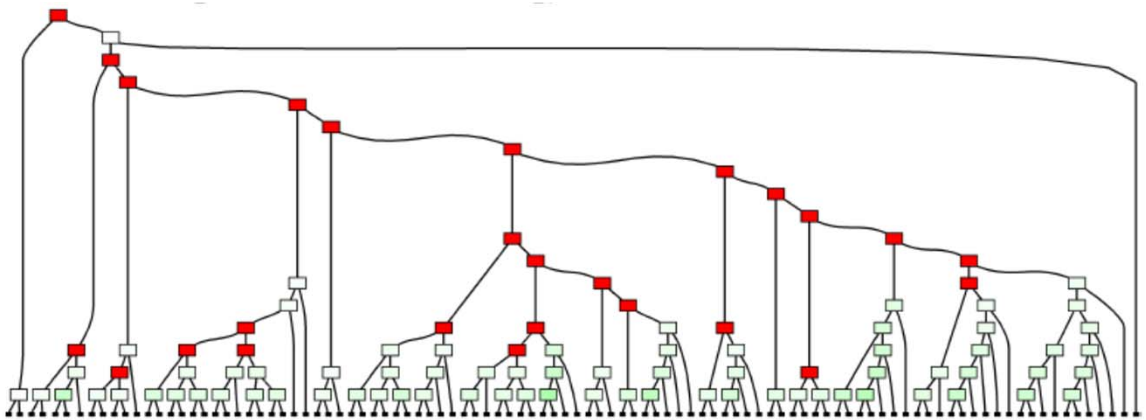
- It's a too large corpus for traditional clustering algorithms that require $O(n^2\log(n))$ in time and $O(n^2)$ in space

# Solution

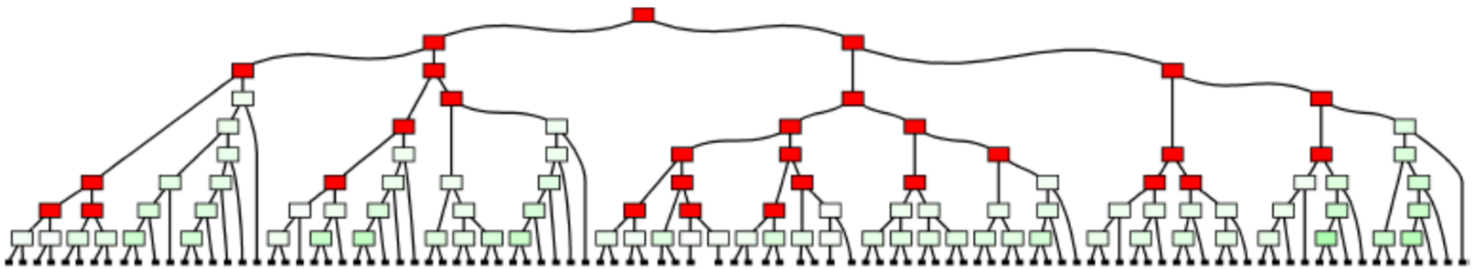
1. sample the full corpus
   - e.g. random select 20,000 stories from 400,000 stories
2. execute clustering algorithms on the sample collection
3. optimize cluster results, build index for each optimized cluster
   - rebalancing of cluster tree
4. assign clusters for complement of sampled story set by document-likelihood match
5. those documents without any matched cluster are assigned into a new cluster

# Cluster Optimization

Before rebranching, marked clusters will be removed



After rebranching, marked clusters are new



# Similarity metric for clustering

- Based on cross-entropy reduction (CER) of unigram

$$sim(D_1, D_2) = \frac{CER(D_1; C, D_2) + CER(D_2; C, D_1)}{2}$$

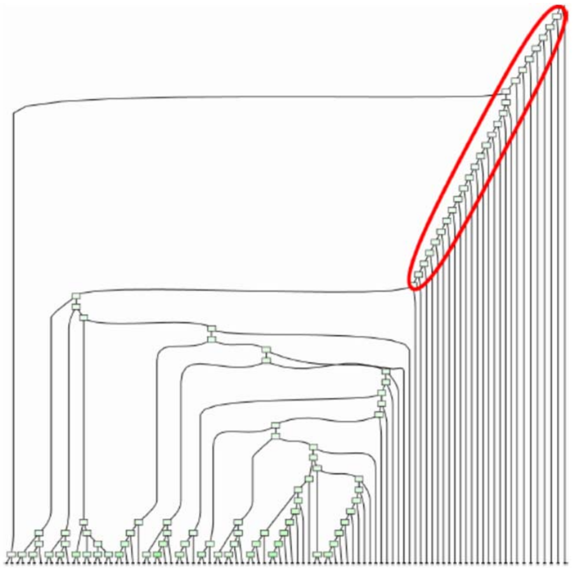
$$\begin{aligned} CER(D_1; C, D_2) &= H(D_1, C) - H(D_1, D_2) \\ &= \sum_{i=1}^{n} P(\tau_i | M_{D_1}) \log \frac{P(\tau_i | M_{D_2})}{P(\tau_i | M_C)} \end{aligned}$$

- $D_1$, $D_2$: two documents to be compared
- $\tau_i$: the i$^{th}$ term, $M_{D_i}$ : the unigram model of $D_i$
- C: a reference document collection

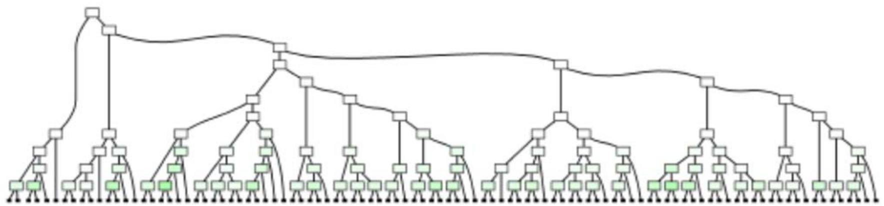
# Clustering Algorithms

- Agglomerative: bottom-up approach
  - Single linkage, complete linkage, minimum-variance etc.
- Divisive clustering: top-down approach

Single link clustering suffers from chaining



Complete link clustering



# Outline

- *News Filtering*
- *Topic Detection and Tracking*
- **Document Clustering**

# Document Clustering

Ref.cs.wellesley.edu, CS 315 – Web Search and Data Mining

# K-means

- Assumes documents are real-valued vectors.
- Clusters based on *centroids* of points in a cluster, *c* (= the *center of gravity* or mean) :

$$\vec{\mu}(c) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

- Reassignment of instances to clusters is based on distance to the current cluster centroids.

# K-Means Algorithm

Let $d$ be the distance measure between instances.

Select $k$ random instances $\{s_1, s_2, \ldots s_k\}$ as seeds.

Until clustering converges or other stopping criterion:

For each instance $x_i$:

Assign $x_i$ to the cluster $c_j$ such that $d(x_i, s_j)$ is minimal.

*(Update the seeds to the centroid of each cluster)*

For each cluster $c_j$

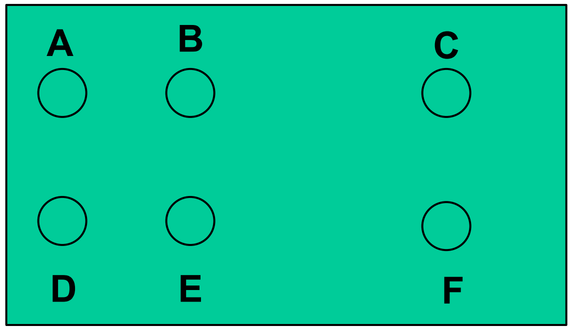
$s_j = \mu(c_j)$

# K-means: Different Issues

- When to stop?
  - When a fixed number of iterations is reached
  - When centroid positions do not change
- Seed Choice
  - Results can vary based on random seed selection.
  - Try out multiple starting points

**Example showing sensitivity to seeds**

**If you start with centroids: B and E you converge to (A, B, C) and (D, E, F)**
**If you start with centroids D and F you converge to: (A, B, D, E) and (C, F)**

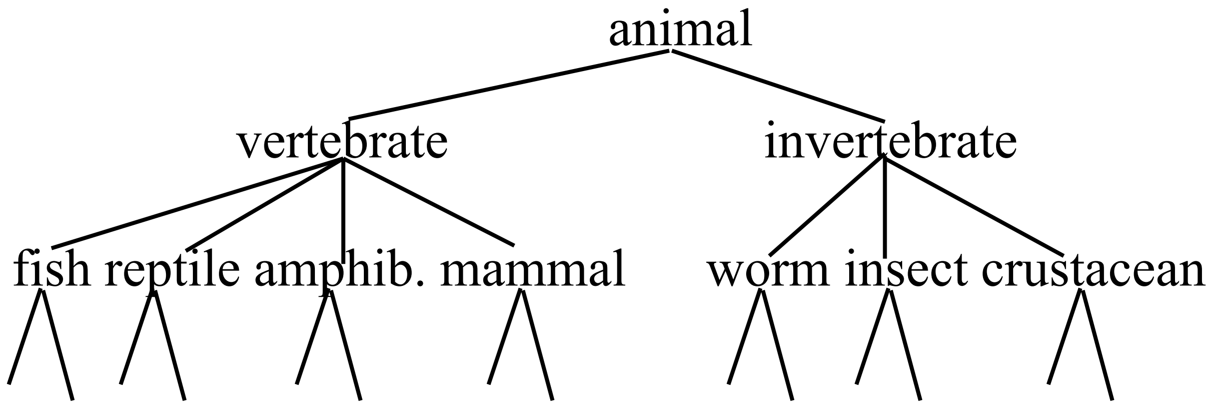


# Time Complexity

- Computing distance between two docs is O$(M)$ where $M$ is the dimensionality of the vectors.
- Reassigning clusters: O$(KN)$ distance computations, or O$(KNM)$.
- Computing centroids: Each doc gets added once to some centroid: O$(NM)$.
- Assume these two steps are each done once for $I$ iterations: O$(IKNM)$.

# Hierarchical clustering

- Build a tree-based hierarchical taxonomy (*dendrogram树状图*) from a set of unlabeled examples.



# Hierarchical Agglomerative Clustering

- We assume there is a similarity function that determines the similarity of two instances.

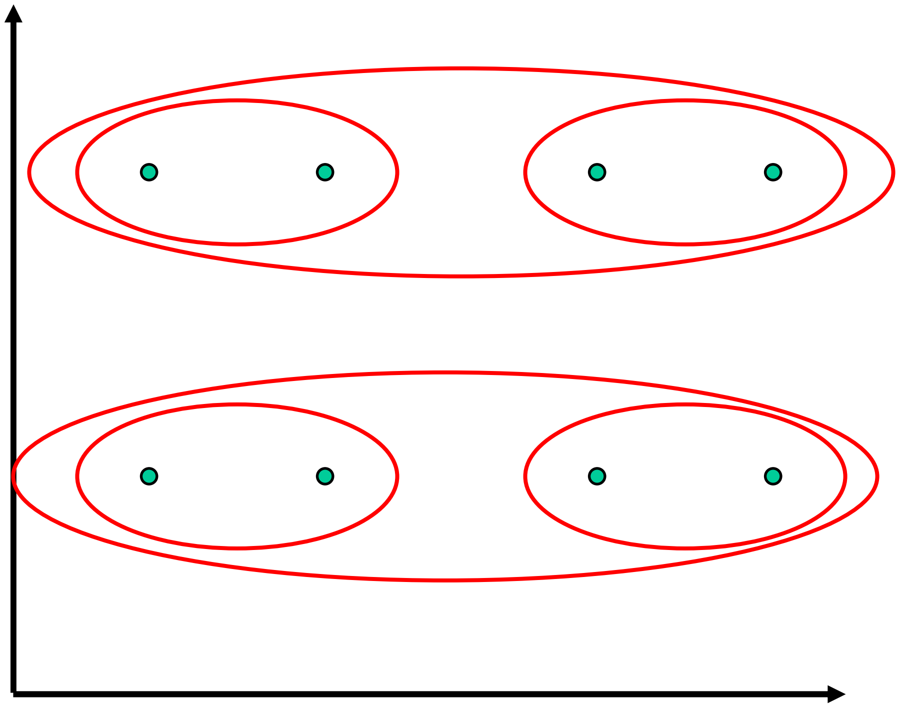
Algorithm:

Start with all instances in their own cluster.
Until there is only one cluster:
    Among the current clusters, determine the two clusters, $c_i$ and $c_j$, that are most similar.
    Replace $c_i$ and $c_j$ with a single cluster $c_i \cup c_j$

# What is the most similar cluster?

- Single-link
  - Similarity of the most cosine-similar (single-link)
- Complete-link
  - Similarity of the “furthest” points, the least cosine-similar
- Group-average agglomerative clustering
  - Average cosine between pairs of elements
- Centroid clustering
  - Similarity of clusters’ centroids

# Single link clustering



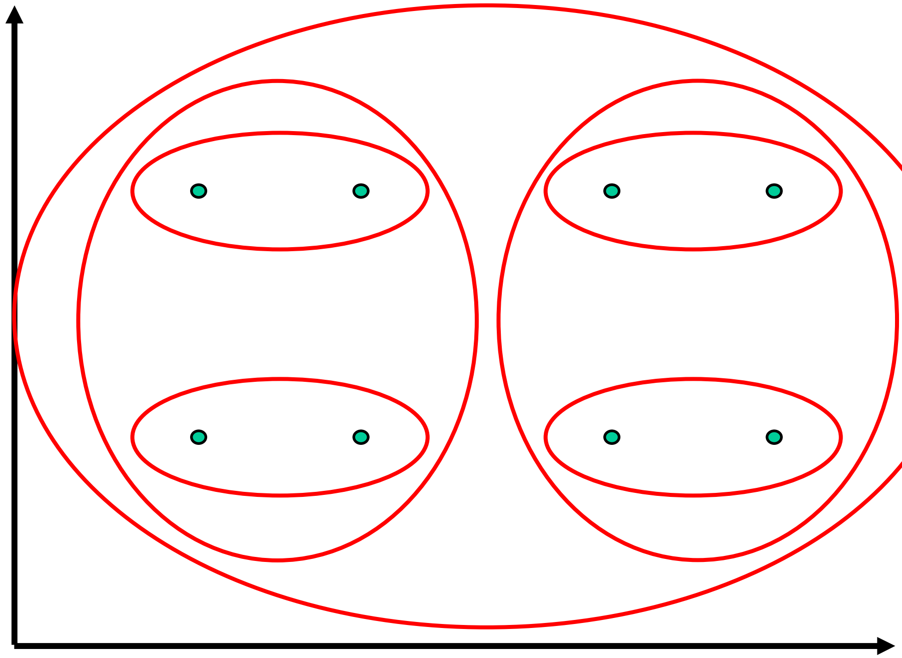
1) Use maximum similarity of pairs:

$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y)$$

2) After merging $c_i$ and $c_j$, the similarity of the resulting cluster to another cluster, $c_k$, is:

$$sim((c_i \cup c_j), c_k) = \max(sim(c_i, c_k), sim(c_j, c_k))$$

# Complete link clustering



1) Use minimum similarity of pairs:

$$sim(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y)$$

2) After merging $c_i$ and $c_j$, the similarity of the resulting cluster to another cluster, $c_k$, is:

$$sim((c_i \cup c_j), c_k) = \min(sim(c_i, c_k), sim(c_j, c_k))$$

# Group Average

- Similarity of two clusters = average similarity of all pairs within merged cluster.

$$sim(c_i, c_j) = \frac{1}{\left|c_i \cup c_j\right|\left(\left|c_i \cup c_j\right| - 1\right)} \sum_{\vec{x} \in (c_i \cup c_j)} \; \sum_{\vec{y} \in (c_i \cup c_j) : \vec{y} \neq \vec{x}} sim(\vec{x}, \vec{y})$$

- Compromise between single and complete link.
- Two options:
  - Averaged across all ordered pairs in the merged cluster
  - Averaged over all pairs *between* the two original clusters
- No clear difference in efficacy

# Computing Group Average Similarity

- Always maintain sum of vectors in each cluster.

$$\vec{s}(c_j) = \sum_{\vec{x} \in c_j} \vec{x}$$

- Compute similarity of clusters in constant time:

$$sim(c_i, c_j) = \frac{(\vec{s}(c_i) + \vec{s}(c_j)) \bullet (\vec{s}(c_i) + \vec{s}(c_j)) - (|c_i| + |c_j|)}{(|c_i| + |c_j|)(|c_i| + |c_j| - 1)}$$

# Further issues

- Complexity:
  - Clustering is computationally expensive. Implementations need careful balancing of needs.
- How to decide how many clusters are best?
- Evaluating the “goodness” of clustering
  - There are many techniques, some focus on implementation issues (complexity/time), some on the quality of

# Further reading


- D. Trieschnigg, W. Kraaij, TNO Hierarchical topic detection report at TDT 2004, in The Task Definition and Evaluation Plan of TDT 2004
- Gabriel Pui Cheong Fung, et al. Time-Dependent Event Hierarchy Construction, KDD’07, 2007: 300-309