Bursty and Hierarchical Structure in Streams*

Jon Kleinberg

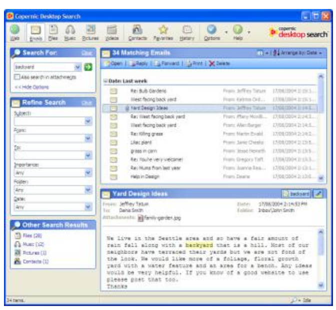
Presented By: Amir Sadovnik

*Or "What happens when Prof. Jon Kleinberg wants to organize his email"

Motivation

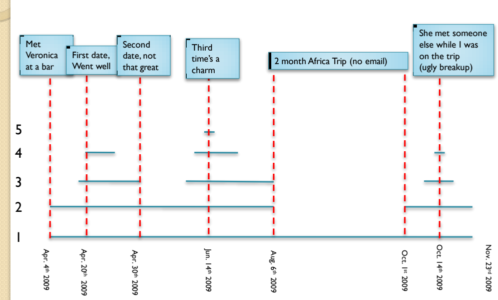
- Many documents can be viewed as streams that arrive continuously over time. (e.g. email, news articles, conference papers).
- An appearance of a topic in a document stream is signaled by a burst of activity.
- The goal of this paper is to model such bursts in a formal way which will provide a framework for analyzing the underlying content.

Motivation



Motivation

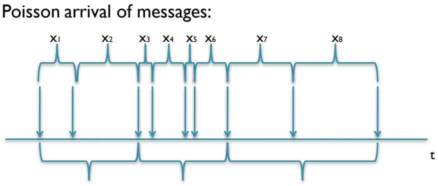
Search: Veronica



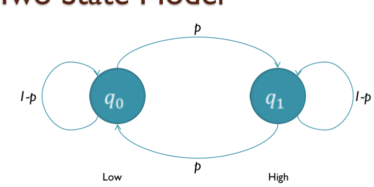
Modeling Bursty Scemes

- Bursts correspond to points at which the intensity of message arrival increases
- Rate of arrival does not rise smoothly and then fall, but exhibits frequents alterations
- Analyzing gaps in a too simplistic way can lead to wrong results

Poisson arrival of messages:



Two State Model



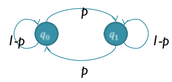
Gaps PDF: $f_0(x) = \propto_0 e^{-\propto_0 x}$ $\quad \propto_0 < \propto_1 \quad$ $f_1(x) = \propto_1 e^{-\propto_1 x}$

Sequence of Gaps: $\vec{x} = (x_1, x_2, \dots, x_n)$

Sequence of States: $\vec{q} = (q_{i_1}, \dots, q_{i_n})$

Probability of gaps given states: $f_q(x_1, x_2, \dots, x_n) = \prod_{t=1}^{n} f_{i_t}(x_t)$

Two State Model



Want to Maximize: $\Pr[\mathbf{q}\mid\mathbf{x}] = \frac{\Pr[\mathbf{q}]\, f_{\mathbf{q}}(\mathbf{x})}{\sum_{\mathbf{q}'}\Pr[\mathbf{q}']\, f_{\mathbf{q}'}(\mathbf{x})}$

Pr[*q*]: $\left(\prod_{i_t\neq i_{t+1}} p\right)\left(\prod_{i_t=i_{t+1}} 1-p\right) = p^b(1-p)^{n-b} = \left(\frac{p}{1-p}\right)^b (1-p)^n.$

Which gives us:

$$-\ln\Pr[\mathbf{q}\mid\mathbf{x}] = b\ln\left(\frac{1-p}{p}\right) + \left(\sum_{t=1}^{n} -\ln f_{i_t}(x_t)\right) - n\ln(1-p) + \ln Z.$$

Cost Function: $c(\mathbf{q}\mid\mathbf{x}) = b\ln\left(\frac{1-p}{p}\right) + \left(\sum_{t=1}^{n} -\ln f_{i_t}(x_t)\right)$

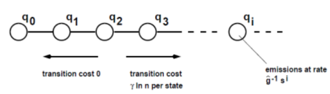
Two State Model

Cost Function: $c(\mathbf{q}\mid\mathbf{x}) = b\ln\left(\frac{1-p}{p}\right) + \left(\sum_{t=1}^{n} -\ln f_{i_t}(x_t)\right)$

- First term tries to minimize state transitions
- Second term tries to maximize probability of ***x***.
- Can be solved using dynamic programming.

This gives us an optimum which tracks the global structure of bursts in the gap sequence while holding to a single state through non-uniformity.

An Infinite-State Model



Define: $\alpha_0 = \hat{g}^{-1} = n/T$

For Each State: $PDF{:}\ f_i(x) = \alpha_i\, e^{-\alpha_i x}$

Rate: $\alpha_i = \hat{g}^{-1} s^i$

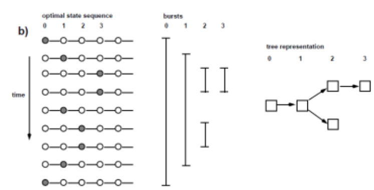
Scaling: $s > 1$

Transition Cost: $\tau(i,j) = \begin{cases} (j-i)\gamma\ln n & if\ j > i \\ 0\ if & if\ j < i \end{cases}$

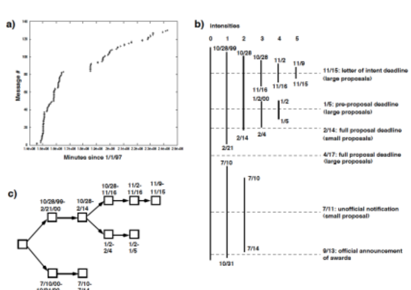
Cost Function: $c(\mathbf{q}\mid\mathbf{x}) = \left(\sum_{t=0}^{n-1} \tau(i_t, i_{t+1})\right) + \left(\sum_{t=1}^{n} -\ln f_{i_t}(x_t)\right).$

Hierarchical Structure

- A burst of intensity is a maximal interval over which q is in a state of index j or higher.



Hierarchical Structure

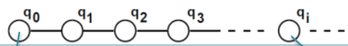


Analogous Model

- For modeling papers gap time cannot be used (appear in batches once a year).
- Instead we can use the portion of documents that are relevant in a batch (e.g. contain a specific word)
- Parameters: r_t - # of relevant doc. In batch *t*.
  d_t - total # of doc. In batch *t*.

$$R = \sum_{t=1}^{n} r_t \qquad D = \sum_{t=1}^{n} d_t$$

Analogous Model



$p_0 = R/D$ $\qquad$ $p_i = p_0 s^{-1}$

Number of states: $p_i \leq 1$

Transition Cost: $\tau(i,j) = \begin{cases} (j-i)\gamma \ln n & if\ j > i \\ 0\ if & if\ j < i \end{cases}$

State Sequence Cost: $\sigma(i, r_t, d_t) = -\ln\left[\binom{d_t}{r_t} p_i^{r_t}(1-p_i)^{d_t - r_t}\right]$

Cost Function:
$$c(\mathbf{q} \mid r_t, d_t) = \left(\sum_{t=0}^{n-1} \tau(i_t, i_{t+1})\right) + \left(\sum_{t=1}^{n} -\ln\left[\binom{d_t}{r_t} p_i^{r_t}(1-p_i)^{d_t - r_t}\right]\right)$$

Weight of Burst

- If we consider just 2 states in the automaton we can define the weight of a burst as: $\sum_{t=t_1}^{t_2}(\sigma(0, r_t, d_t) - \sigma(1, r_t, d_t)).$
- Using this the following experiment was conducted:
  - Analysis to the titles STOC and FOCS papers 1969-2001
  - All words were tracked in experiment

Results

Technical

Language Use

Word	Interval of burst
grammars	1969 STOC — 1973 FOCS
automata	1969 STOC — 1974 STOC
languages	1969 STOC — 1977 STOC
machines	1969 STOC — 1978 STOC
recursive	1969 STOC — 1979 FOCS
classes	1969 STOC — 1981 FOCS
some	1969 STOC — 1980 FOCS
sequential	1969 FOCS — 1972 FOCS
equivalence	1969 FOCS — 1981 FOCS
programs	1969 FOCS — 1986 FOCS
program	1970 FOCS — 1978 STOC
on	1973 FOCS — 1976 STOC
complexity	1974 STOC — 1975 FOCS
problems	1975 FOCS — 1976 FOCS
relational	1975 FOCS — 1982 FOCS
logic	1976 FOCS — 1984 STOC
vlsi	1980 FOCS — 1986 STOC
probabilistic	1981 FOCS — 1986 FOCS
how	1982 STOC — 1988 STOC
parallel	1984 STOC — 1987 FOCS
algorithm	1984 FOCS — 1987 FOCS
graphs	1987 STOC — 1989 STOC
learning	1987 FOCS — 1997 FOCS
competitive	1990 FOCS — 1994 FOCS
randomized	1992 STOC — 1995 STOC
approximation	1993 STOC —
improved	1994 STOC — 2000 STOC
codes	1994 FOCS —
approximating	1995 FOCS —
quantum	1996 FOCS —

Results

Change in word use from: “data base” to “database

Word	Interval of burst
data	1975 SIGMOD — 1979 SIGMOD
base	1975 SIGMOD — 1981 VLDB
application	1975 SIGMOD — 1982 SIGMOD
bases	1975 SIGMOD — 1982 VLDB
design	1975 SIGMOD — 1985 VLDB
relational	1975 SIGMOD — 1989 VLDB
model	1975 SIGMOD — 1992 VLDB
large	1975 VLDB — 1977 VLDB
schema	1975 VLDB — 1980 VLDB
theory	1977 VLDB — 1984 SIGMOD
distributed	1977 VLDB — 1985 SIGMOD
data	1980 VLDB — 1981 VLDB
statistical	1981 VLDB — 1984 VLDB
database	1982 SIGMOD — 1987 VLDB
nested	1984 VLDB — 1991 VLDB
deductive	1985 VLDB — 1994 VLDB
transaction	1987 SIGMOD — 1992 SIGMOD
objects	1987 VLDB — 1992 SIGMOD
object-oriented	1987 SIGMOD — 1994 VLDB
parallel	1989 VLDB — 1996 VLDB
object	1990 SIGMOD — 1996 VLDB
mining	1995 VLDB —
server	1996 SIGMOD — 2000 VLDB
sql	1996 VLDB — 2000 VLDB
warehouse	1996 VLDB —
similarity	1997 SIGMOD —
approximate	1997 VLDB —
web	1998 SIGMOD —
indexing	1999 SIGMOD —
xml	1999 VLDB —

Results

Civil War

Great Depression

WWII

Words	Interval of Bursts
gentlemen	1790 - 1800
militia	1801 - 1816
whilst	1857 - 1860
slaves	1859 - 1863
rebellion	1861 - 1871
depression	1930 - 1937
recovery	1930 - 1937
banks	1931 - 1934
democracy	1937 - 1941
wartime	1941 - 1947
that's	1982 -
we're	1982 -
we've	1982 -
schools	1996 -
teachers	1996 -
21st	1997 -
century	1997 -

Presidential State of the Union Addresses, 1790-2002

Using 2 state model with s=16

Discussion

- The interplay between time and content is crucial.
- This model can be applied in other areas (e.g. web usage data)
- Bursts have sharp boundaries, therefore can be mapped to specific documents/events.