IMPROVING TEXT STREAM CLUSTERING USING TERM BURSTINESS AND CO-BURSTINESS

Argyris Kalogeratos
Joint work with:
Panagiotis Zagorisios and Aristidis Likas

19 May 2016

OUTLINE

- Preliminaries
- Related Work
- The proposed **CBTC** method
- Experiments
- Conclusion

PRELIMINARIES

Text Clustering

- Input: a **static** collection of text documents
- Target: thematic segmentation into *sufficiently different* groups containing *similar* documents
- Representation: usually in the *vector space model* (VSM)

  \- Term-document vectors in Bag-of-Words (TFIDF-BOW) model:

$$d_i = [d_{i1}, \ldots, d_{iV}]^\top = [tf_{i1} \cdot idf_1, \ldots, tf_{iV} \cdot idf_V]^\top$$

- Challenges
  - Curse of dimensionality & high sparsity
  - Language phenomena: polysemy, synonymy, homonymy, complex semantics, etc.

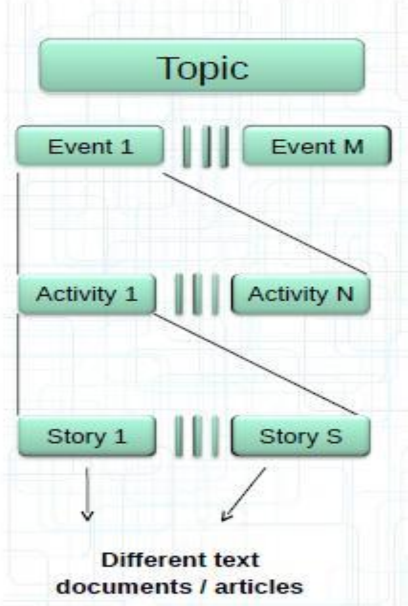
PRELIMINARIES
Text stream clustering

- Input: a stream of documents published over time
- Target: identification of document clusters referring to the same real-life topic (or set of events)
- Representation: using the document vectors + timestamps
  - A stream of T batches:

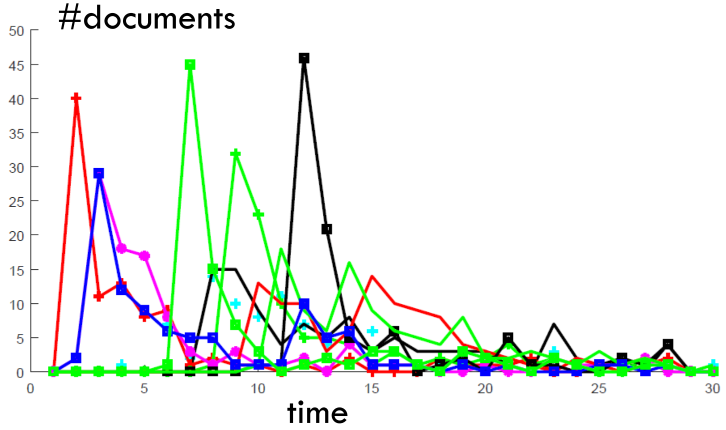
$$S = [s_1, \ldots, s_T]$$

- Challenges
  - Conventional clustering neglects the timestamp information
  - Added complexity - How to combine *content* and *temporal* proximity between documents?
  - Feature-based vs. <u>document-based topic representation</u>
  - Online vs. <u>offline processing</u>

Content hierarchy



Stream example



ENHANCING VSM REPRESENTATION

The standard recipe

A. Make semantically richer VSM

- Temporal information is seek into the distribution of terms over time
  - **Term burst:** a rapid increase in term's occurrence rate
- Re-weight the vectors favoring bursty terms

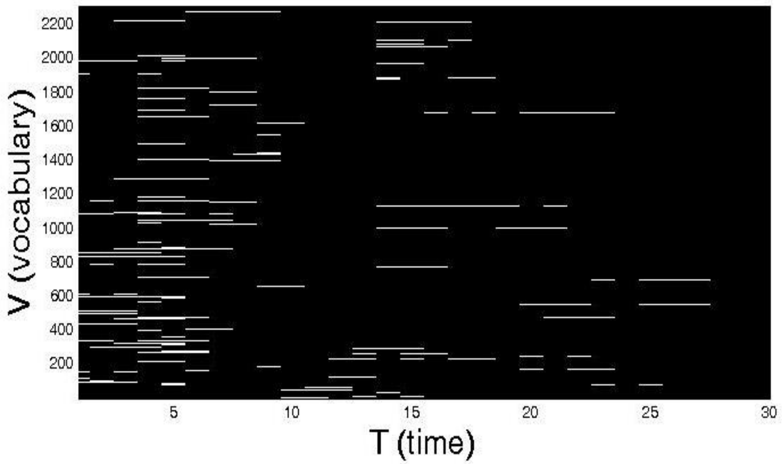
VSM ***bursty VSM***

$$X \rightarrow XB$$

B. Use traditional clustering algorithms

- e.g. hierarchical agglomerative or k-means

Term bursts in stream

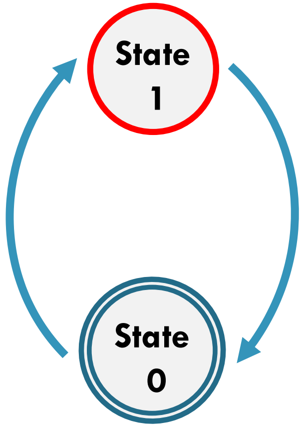


ENHANCING VSM REPRESENTATION

Burst detection: the popular Kleinberg's two-state automaton

$p_1 = \alpha \cdot p_0$

cost= $\gamma \log(T)$



cost=0

$p_0 = |D_j|/T$

Stream: $S = [s_1, \ldots, s_T]$

Find the sequence ($q_1 \ldots q_T$) of states for term j by minimize the cost to be at state i:

$$\sigma(i, |s_{tj}|, |s_t|) = -ln\left[\binom{|s_t|}{|s_{tj}|} p_i^{|s_{tj}|}(1 - p_i)^{|s_t| - |s_{tj}|}\right]$$

Output a **burst weight** for term j:

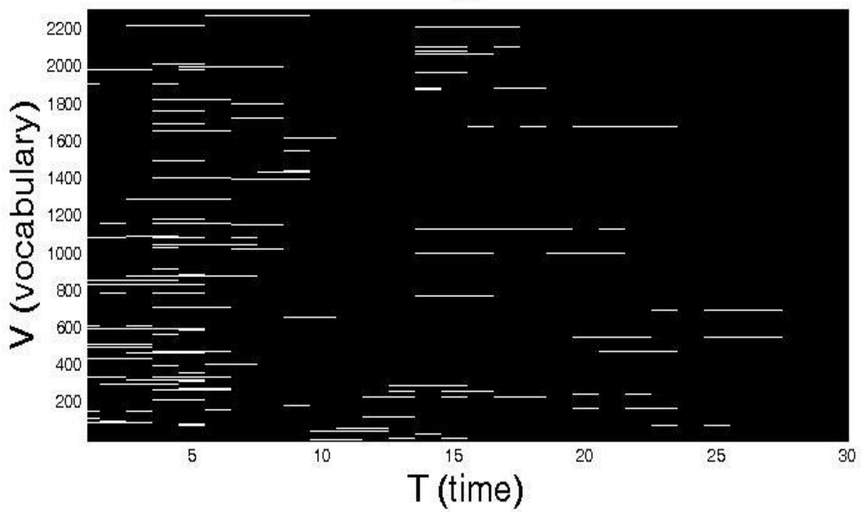
$$w_j^{[t_1, t_2]} = \sum_{t=t_1}^{t_2} (\sigma(0, |s_{tj}|, |s_t|) - \sigma(1, |s_{tj}|, |s_t|))$$

- Statistically simple and popular
- Difficult to tune the parameters: α and γ , but not cheap computationally

ENHANCING VSM REPRESENTATION

Existing burst-based approaches (1)

Term bursts in stream



[He et al. 2007a]

B-VSM: $$d_{ij}^{(t)} = \begin{cases} \mathbb{1}\{tf_{ij} > 0\} + \delta w_j^{(t)}, & \text{if } t \in \tau_j \\ \mathbb{1}\{tf_{ij} > 0\}, & \text{otherwise} \end{cases}$$

[He et al. 2007b]

SAB: $$d_{ij}^{(t)} = \begin{cases} tfidf_{ij} + \overline{w}_j^{(t)}, & \text{if } f_j \in \mathcal{B} \\ tfidf_{ij}, & \text{otherwise} \end{cases}$$

SMB: $$d_{ij}^{(t)} = \begin{cases} tfidf_{ij} \cdot w_j^{(t)}, & \text{if } f_j \in \mathcal{B} \\ tfidf_{ij}, & \text{otherwise} \end{cases}$$

BAB: $$d_{ij}^{(t)} = tfidf_{ij} + \overline{w}_j^{(t)}$$

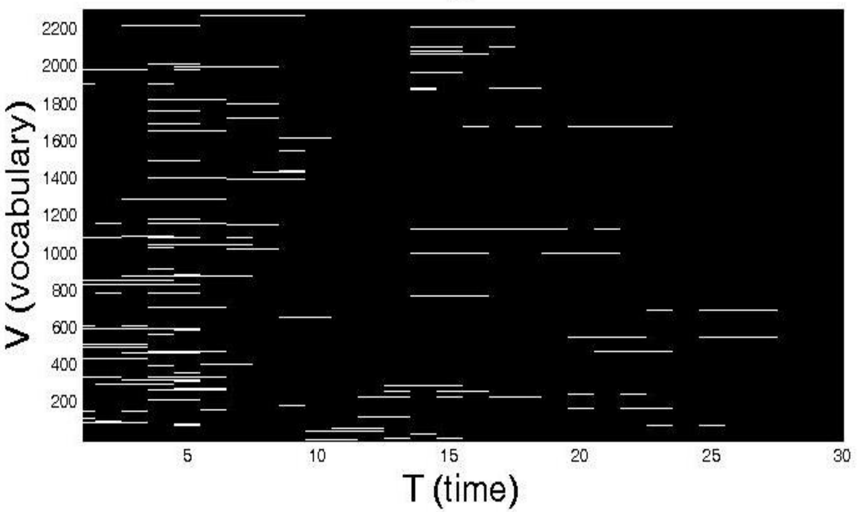
BMB: $$d_{ij}^{(t)} = tfidf_{ij} \cdot w_j^{(t)}$$

BT: $$d_{ij}^{(t)} = tfidf_{ij}$$

ENHANCING VSM REPRESENTATION

Existing burst-based approaches (2)

Term bursts in stream



[Zhao et al. 2012]

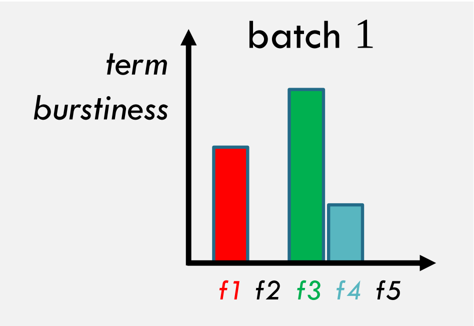
Burst-VSM: $$d_{ij}^{(t)} = \begin{cases} tfidf_{ij}, & \text{if } t \in \tau_j \\ 0, & \text{otherwise} \end{cases}$$

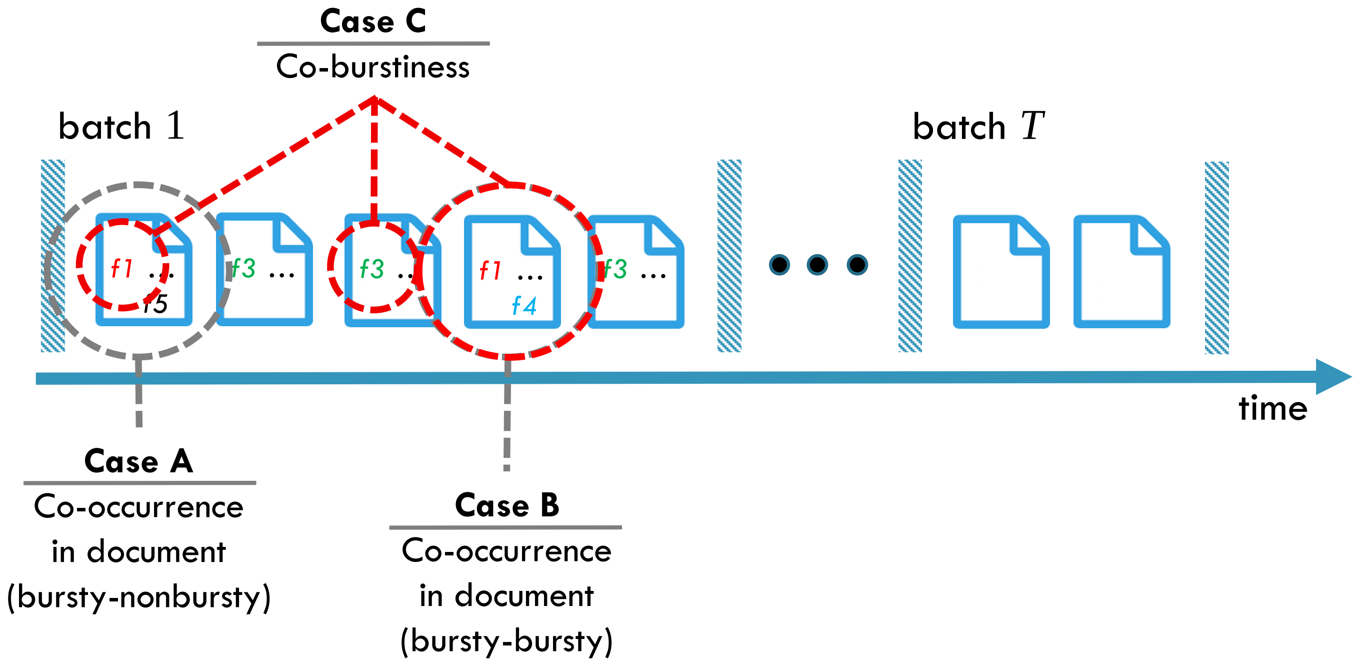
Employed B-VSM: $$d_{ij}^{(t)} = \begin{cases} tfidf_{ij} \cdot w_j^{(t)}, & \text{if } t \in \tau_j \\ tfidf_{ij}, & \text{otherwise} \end{cases}$$

OUR CONTRIBUTION

A. Exploiting term burstiness and… co-burstiness

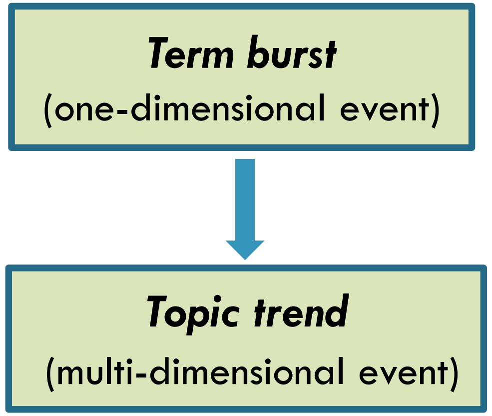
- If documents containing the same term during one of its burst periods, this is an indication that they are part of the same event/topic
- But there is more happening in a stream…





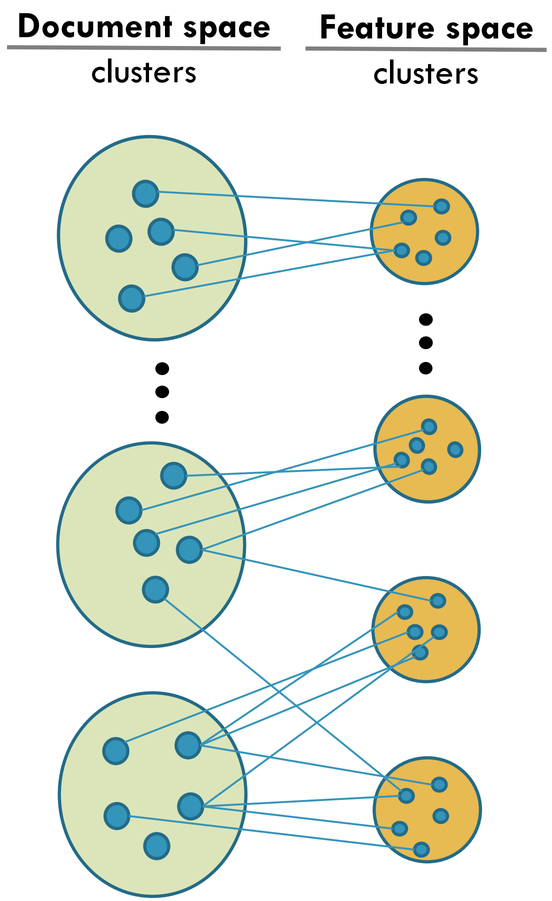
OUR CONTRIBUTION

B. Exploiting space duality



Our direction of work

- Capitalizing on the duality between feature and document space
- Bursty terms could indicate the most representative documents for their topic



CORRELATED BURSTY TERM CLUSTERING

Proposed CBTC method

- **Step 1:** Create $k' > k$ groups of bursty terms
- **Step 2:** Construct the k' synthetic cluster prototypes [Kalogeratos et al. 2011]
- **Step 3:** Apply *agglomerative k-sp* $k' \rightarrow k$ clusters
- **Step 4:** <u>*Deterministic initialization*</u> of *spherical k-means* with the k produced prototypes

CORRELATED BURSTY TERM CLUSTERING

Proposed method (1)

- **Step 1:** Create $k' > k$ groups of bursty terms
  - **a)** Construct the novel ***bursty term correlation graph*** *(B nodes)*

Co-occurrence in documents during burst periods **Co-burstiness**

$$a_{ij} = \begin{cases} \frac{1}{2}\left(\frac{|D_i \cap D_j|}{|D_i|} + \frac{|D_i \cap D_j|}{|D_j|}\right), & \text{if } \underbrace{h(D_i \cap D_j)}_{\text{Co-occurrence in documents during burst periods}} \cap \underbrace{(\tau_i \cap \tau_j)}_{\text{Co-burstiness}} \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

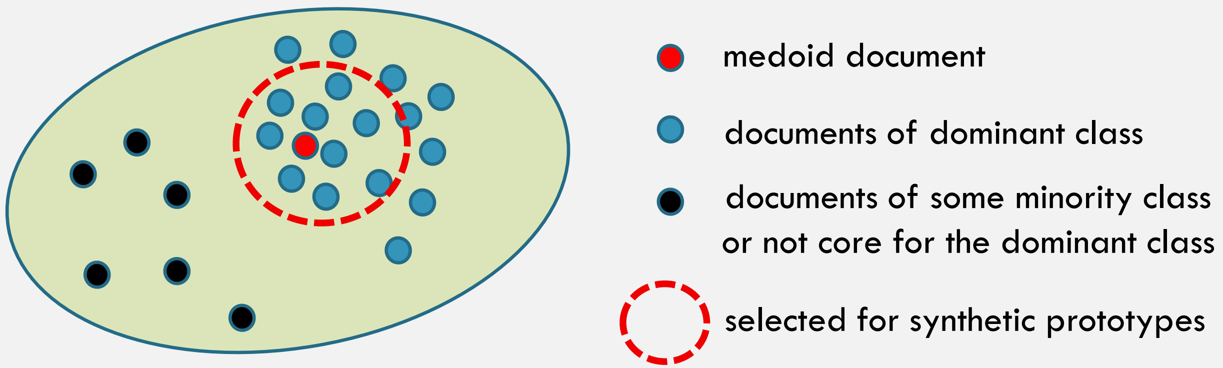
  - **b)** Segment the graph with *spectral clustering* [Ng et al. 2002]

CORRELATED BURSTY TERM CLUSTERING

Proposed method (2)

- **Step 1:** Create $k' > k$ groups of bursty terms
- **Step 2:** Construct the k' **synthetic cluster prototypes** [Kalogeratos et al. 2011]
  - For each term group, select the documents that contain at least one bursty term
  - Then, robust representatives are built with a subset of objects around the **medoid**
  - They favor the dominant class in a cluster
  - Two parameters: the percentage of cluster members to use, and an L_1 filter

Inhomogeneous cluster example



CORRELATED BURSTY TERM CLUSTERING

Proposed method (3)

- **Step 1:** Create $k' > k$ groups of bursty terms
- **Step 2:** Construct the k' synthetic cluster prototypes [Kalogeratos et al. 2011]
- **Step 3:** Apply *agglomerative k-sp* $k' \rightarrow k$ clusters
  - Merge the pair of nearest document clusters (recall: they correspond to term clusters)
  - Recompute the synthetic prototypes… repeat
  - Finally, produce k cluster prototypes

CORRELATED BURSTY TERM CLUSTERING

Proposed method (4)

- **Step 1:** Create $k' > k$ groups of bursty terms
- **Step 2:** Construct the k' synthetic cluster prototypes [Kalogeratos et al. 2011]
- **Step 3:** Apply *agglomerative k-sp* $k' \rightarrow k$ clusters
- **Step 4:** *Deterministic initialization* of *spherical k-means* with the k produced prototypes
  - This algorithm uses *cosine similarity* and maximizes the *clustering cohesion* [Dhillon et al. 2001]

$$Cohesion(C) = \sum_{j=1}^{k} \sum_{d_i \in c_j} r_j^\top d_i$$

  - VSM or B-VSM could be used for this final clustering

EXPERIMENTS

Datasets and setup (1)

- 5 datasets of moderate and small size
- Standard preprocessing with TMG toolkit [Zeimpekis et al. 2006]

		Text characteristics					Stream characteristics			
	Name	Classes	N	$Balance$	V	$\overline{V}_i$	T	B	$\overline{\lvert s_i \rvert}$	H_s
20Newsgroups	D1	10	1000	1	2352	45.89	30	354	33.3	3.030 ± 0.918
20Newsgroups	D2	10	1000	1	2310	44.54	30	381	33.3	3.030 ± 0.918
Reuters-21578	D3	10	993	0.93	1566	44.16	30	350	33.1	3.028 ± 0.831
TDT5	D4	30	4972	0.06	4717	21.54	183	4020	23.8	2.053 ± 0.581
GoogleNews	D5	11	268	0.43	1298	59.07	31	400	8.6	0.237 ± 0.543

– N denotes the number of documents, $Balance$ the ratio of the smallest to the largest class, V the size of the vocabulary, and $\overline{V}_i$ the average document vocabulary size.
– T is the number of time windows, B the number of bursty terms, $\overline{|s_i|}$ the average number of documents per window, and H_S the temporal topic entropy.

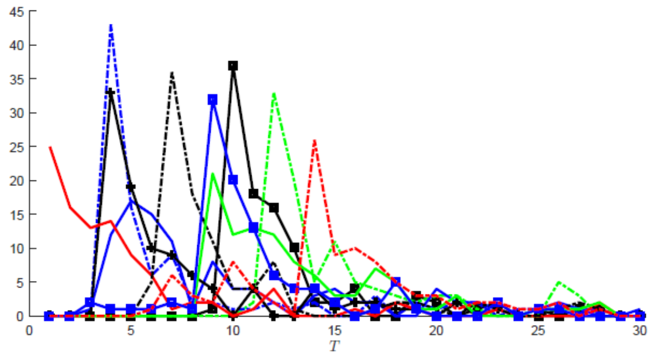
EXPERIMENTS

Datasets and setup (2)

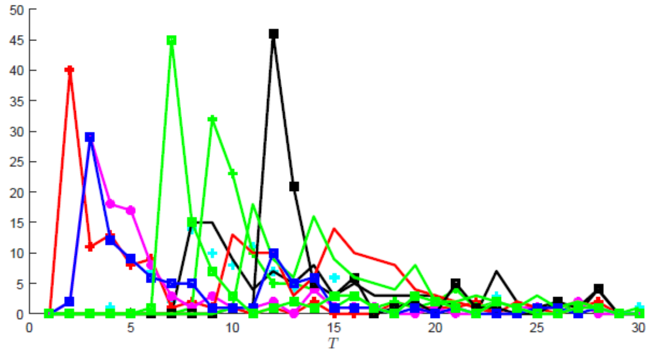
- We used the original timelines for D4 and D5
- Artificially generated timelines for (D1-D2) and D3
  - Though respecting the original document ordering provided
  - This way we can adjust “*stream complexity*”

Parameters for stream generation (timestamps)

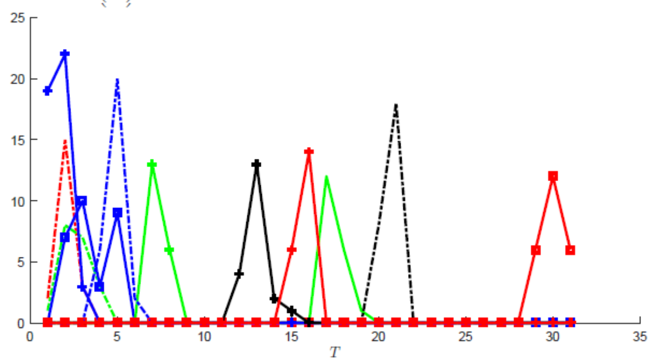
Parameter	Value / Selection range
T	30
λ	$[0.2, 0.9]$
#bursts per topic	$\{1, 2\}$
%docs in bursts	$[0.7, 0.9]$



(a) Generated stream for D1 dataset



(b) Generated stream for D3 dataset



(c) Original stream for D5 dataset

RESULTS

- *RandInit vs. CBTC*
  (100 restarts)
- *VSM vs. B-VSM*

Results with initializations of spherical k-means

Dataset		VSM representation (X) Purity↑	F1↑	NMI↑		B-VSM representation (XB) Purity↑	F1↑	NMI↑
D1	X (avg.)	0.419	0.423	0.365	XB (avg.)	0.444	0.479	0.410
	(best)	0.510	0.524	0.457	(best)	0.562	0.573	0.490
	X-3k	0.580	0.596	0.578	XB-3k	0.602	0.603	0.558
	X-2k	**0.628**	**0.658**	**0.594**	XB-2k	**0.626**	**0.653**	**0.576**
D2	X (avg.)	0.503	0.515	0.439	XB (avg.)	0.508	0.546	0.451
	(best)	0.571	0.580	0.491	(best)	0.611	0.622	0.535
	X-3k	0.684	0.712	**0.633**	XB-3k	0.684	0.700	0.618
	X-2k	**0.714**	**0.714**	0.619	XB-2k	**0.711**	**0.730**	**0.628**
D3	X (avg.)	0.661	0.649	0.645	XB (avg.)	0.710	0.710	0.686
	(best)	0.771	0.774	0.745	(best)	**0.796**	**0.805**	**0.768**
	X-3k	0.719	0.744	0.703	XB-3k	0.751	0.759	0.745
	X-2k	**0.774**	**0.787**	**0.765**	XB-2k	0.774	0.792	0.766
D4	X (avg.)	0.500	0.457	0.545	XB (avg.)	0.518	0.473	0.584
	(best)	0.564	0.511	0.587	(best)	0.614	0.556	0.641
	X-3k	**0.689**	**0.635**	0.704	XB-3k	**0.701**	**0.638**	0.718
	X-2k	0.678	0.622	**0.712**	XB-2k	0.688	0.625	**0.722**
D5	X (avg.)	0.444	0.441	0.369	XB (avg.)	0.720	0.713	0.710
	(best)	0.557	0.566	0.474	(best)	0.794	0.793	0.772
	X-3k	**0.716**	**0.742**	**0.650**	XB-3k	**0.828**	**0.837**	**0.791**
	X-2k	0.522	0.531	0.504	XB-2k	0.623	0.647	0.658

CONCLUSION

- Discussed the text stream clustering problem
- Pointed out certain limitations in related work
- Developed the CBTC method
  - Uses efficiently the term ***burstiness*** and ***co-burstiness*** information
  - Capitalizes on the duality of feature and document spaces
  - Provides good quality deterministic initialization for standard clustering methods
- Presented experiments on real data (+ artificial timelines)
- Future work
  - experimentation in larger datasets
  - parameter tuning

QUESTIONS

Thank you!

APPENDIX 1/6

$$H_S = \frac{1}{T}\sum_{t=1}^{T}\left[-\sum_{i}\frac{n(C_i^{*t})}{N^t}\cdot log_2\frac{n(C_i^{*t})}{N^t}\right]$$

APPENDIX 2/6

Algorithm 1 Initialization of sp$k$-means with the CBTC.

function `CBTC` $(\hat{X},\ p_{docs},\ p_{terms},\ k,\ k',\ A)$

input : $\hat{X}$ is the document matrix with row vectors, p_{docs}, p_{terms} are parameters for the synthetic prototype construction, k and k' the starting and desired number of clusters ($k' \geq k$), and A the bursty term correlation matrix

output : $R = \{r_1, ..., r_k\}$ the set of final cluster prototypes, $C = \{c_1, ..., c_k\}$ the sets of documents assigned to each cluster

$C^{(f)} \leftarrow$ `SegmentTermGraph` $(A,\ k')$ // see Alg. 2

$\{SP,\ C^{(b)}\} \leftarrow$ `ConstructBurstySP` $(C^{(f)},\ \hat{X},\ p_{docs},\ p_{terms})$ // see Alg. 3

$\{SP\} \leftarrow$ `MergeClusters` $(C^{(b)},\ SP,\ k,\ p_{docs},\ p_{terms})$ // see Alg. 4

$\{R,\ C\} \leftarrow$ `spkmeans` $(SP,\ \hat{X},\ k)$ // see Sec. 2.1

return $(R,\ C)$

APPENDIX 3/6

Algorithm 2 Segmentation procedure on the bursty terms.

function `SegmentTermGraph` $(A,\ k')$

input : A is the bursty term correlation matrix, k' the desired number of groups

output : $C^{(f)} = \{c_1^{(f)}, ..., c_{k'}^{(f)}\}$ the segmentation solution with $k' \geq k$ groups of bursty terms

1: $C^{(f)} \leftarrow$ `SpectralClustering` $(A,\ k')$

2: $C^{(f)} \leftarrow C^{(f)} \setminus \{\bigcup c_i^{(f)},\ \forall i \in [1,\ k']$ s.t. $|c_i^{(f)}| < 2\}$

3: **return** $(C^{(f)})$

APPENDIX 4/6

Algorithm 3 Construction of bursty synthetic prototypes.

function `ConstructBurstySP` $(C^{(f)}, \hat{X}, p_{docs}, p_{terms})$

input : $C^{(f)}$ is the segmentation of `SegmentTermGraph()`,
$\hat{X}$ the document matrix with row vectors,
p_{docs}, p_{terms} are the parameters for the synthetic prototype construction

output : $SP = \{sp_1, ..., sp_{k'}\}$ the set of synthetic prototypes,
$C^{(b)} = \{c_1^{(b)}, ..., c_{k'}^{(b)}\}$ the documents clusters corresponding to the groups of bursty terms $C^{(f)}$

let : f_j the j-th term (here $f_j \in \mathcal{B}$),
$k' = |C^{(b)}|$ the number of clusters,
D_j the set of documents containing the term f_j,
$\hat{X}_{Docs}$ the submatrix of $\hat{X}$ with the rows that correspond to the documents in the set $Docs$,
`ConstructSP()` constructs a synthetic prototype,
`AssignToClosest()` assigns the documents of a set to the closest of the prototypes provided

$Docs_B \leftarrow \emptyset$
for $i = 1...k'$
  $Docs \leftarrow \emptyset$
  for each $f_j \in c_i^{(f)}$
    $Docs \leftarrow Docs \cup D_j$
  end for
  $Docs_B \leftarrow Docs_B \cup Docs$
  $sp_i \leftarrow$ `ConstructSP` $(\hat{X}_{Docs}, p_{docs}, p_{terms})$
end for
$C^{(b)} \leftarrow$ `AssignToClosest` $(\hat{X}_{Docs}, SP)$
return $(SP, C^{(b)})$

APPENDIX 5/6

Algorithm 4 Agglomerative cluster merging step.

function `MergeClusters` $(C^{(b)}, SP, k, p_{docs}, p_{terms})$

input : $C^{(b)}$, SP are the output of `ConstructBurstySP()`,
k is the final number of clusters to reduce set $C^{(b)}$,
p_{docs}, p_{terms} are for the SP construction

output : SP the synthetic cluster prototypes

let : `ClosestPrototypes()` that returns the indexes of the two most similar prototypes in a given set

$k' \leftarrow |C^{(b)}|$
repeat
    $\{s, u\} \leftarrow$ `ClosestPrototypes` (SP)
    $c_{su}^{(b)} \leftarrow c_s^{(b)} \cup c_u^{(b)}$
    $(C^{(b)} \leftarrow C^{(b)} \setminus \{c_s^{(b)}, c_u^{(b)}\}) \cup c_{su}^{(b)}$
    $sp_{su} \leftarrow$ `ConstructSP` $(c_{su}, p_{docs}, p_{terms})$
    $SP \leftarrow (SP \setminus \{sp_s, sp_u\}) \cup sp_{su}$
    $k' \leftarrow k' - 1$
until $k' == k$
return (SP)

APPENDIX 6/6

Clustering evaluation meatrics:

$$NMI = \frac{\sum \frac{n_{ji}}{N} log_2 \frac{\frac{n_{ij}}{N}}{\frac{n_i^*}{N} \cdot \frac{n_j}{N}}}{\max\{H(C), H(C^*)\}}$$

$$F1 = 2\frac{P \cdot R}{P + R}$$

$$Purity = \frac{1}{N} \sum_{j=1}^{k} \max\{n_{ij}\}$$