Background
- text clustering
- text stream

Online Spherical K-means (OSKM)
- k-means
- Spherical K-means
- Online Spherical K-means

Another framework for clustering massive text streams

Other methods
Text clustering (document clustering)

automatically group text/documents into clusters so that documents within a cluster have high similarity to each other, but are dissimilar to documents in other clusters.

Examples

- group similar news/tweets/blog articles
- analysis of customer/employee feedback

- usually involve high-dimensional, sparse features (e.g. bag-of-words vectors)
**BACKGROUND**

bag of words

- **Procedure**
  - tokenize
  - stop words removal
  - lemmatization

  grouping together the inflected forms of a word so they can be analyzed as a single item, identified by the word's lemma, or dictionary form
Text stream:
continuous, often rapid, ordered sequence of texts
(Text information arriving continuously over time in the form of a data stream)

- news, online traffic reports, web searches
- social media
  - twitter, facebook posts, chat, emails
- **objective function:** Residual Sum of Squares (RSS)

\[ E = \frac{1}{N} \sum_{x} \| x - \mu_k(x) \|^2 \]

- \(k(x) = \arg \min_{k \in \{1,2,\ldots,k\}} \| x - \mu_k(x) \|^2\)

minimize RSS!
text mining:

high-dimensional data

- curse of dimensionality
- direction more important than magnitude
- cos similarity better than Euclidean distance
SPHERICAL K-MEANS

standard k-means (Euclidean distance)

\[ E = \frac{1}{N} \sum_x \| x - \mu_k(x) \|^2 \]

\[ k(x) = \arg \min_{k \in \{1, 2, \ldots, k\}} \| x - \mu_k(x) \|^2 \]

spherical k-means (cos similarity)

\[ E = \frac{1}{N} \sum_x x^T \mu_k(x) \]

\[ k(x) = \arg \max_{k \in \{1, 2, \ldots, k\}} x^T \mu_k(x) \]

\[ (X, \mu_\cdot: \text{normalized unit-vectors}) \]

Note:

re-estimated \( \mu \) needed to be normalized (computational bottleneck)
ONLINE SPHERICAL K-MEANS

- divides up the incoming stream into small segments
  - hold in main memory
  - user-defined iteration times (at most)

- a competitive learning technique
  - Winner-Take-All
ONLINE SPHERICAL K-MEANS

- centroid update formula
  
  \[ \mu'_k(x) = \frac{\mu_k(x) + \eta x}{\| \mu_k(x) + \eta x \|} \] (for each \( x \))

- \( \eta \): learning rate
  
  - flat rate (like 0.05)
    
    - exponentially decreasing rate (annealing)

- a decay factor
  
  - to age out the old documents (back later)
ONLINE STREAM MAINTENANCE

Basic procedure:
- **Beginning:** empty set
- \(X_1, \ldots, X_k \rightarrow k\) unit clusters (one point each)
- **new data point** \(\bar{X}\) arrive:
  - cos similarity with each cluster: \(S(\bar{X}, C_j)\)
  - cluster with max \(S(\bar{X}, C_j)\) => possible relevant cluster \(C_r\)
  - outlier detection: threshold \(T\)
    - \(S(\bar{X}, C_r) > T\) assign \(\bar{X}\) to \(C_r\)
    - \(S(\bar{X}, C_r) < T\) potential true outlier, or the beginning of a new trend
ONLINE STREAM MAINTENANCE: DETAILS

- recent data more important than old data
  - fading function $f(t)$ (time-dependent weight of data points)
    non-monotonic decreasing function which decays uniformly with time $t$
  - half life $t_0$
    quantify the rate of decay of the importance of each data point in the stream clustering process
    Definition: $f(t_0) = \frac{1}{2} f(0)$
  - decay rate $\lambda$
    $\lambda = \frac{1}{t_0}$
    $f(t) = 2^{-\lambda t}$
ONLINE STREAM MAINTENANCE

look back:

\[ S(\bar{X}, C_r) < T \Rightarrow \bar{X} \text{ can be a potential true outlier, or the beginning of a new trend} \]

- if there have been k clusters \&\& no inactive cluster
  \[ \Rightarrow \bar{X} \rightarrow C_r \]
- else
  \[ \Rightarrow \text{create a new (inactive) cluster for } \bar{X} \] [outlier or new trend?]
  (replace the former inactive cluster)
ONLINE STREAM MAINTENANCE

how to handle inactive cluster (outlier or new trend)

- create a inactive cluster for $\bar{X}$
- allow it to remain for half life
- During its half life:
  - more points in $\Rightarrow$ cluster active and mature ($\bar{X}$ is an new trend-setting point)
  - no more points $\Rightarrow$ cluster death ($\bar{X}$ is an outlier)
ONLINE STREAM MAINTENANCE

- **Same** criterion for the death of mature clusters:
  - cluster weight: sum of points weights
  - when inactivity period exceeds the half life $1/\lambda$

- **Note:**

  The greater the number of points in the cluster, the greater the level by which the inactivity period would need to exceed its half life in order to meet the criterion.
OTHER WORK

- bursty features for new topic occurrence recognition

- Phrase extraction for feature selection
REFERENCES


