Online Topic discovery in Microblogs through Online Clustering

Debjyoti Paul, deb@cs.utah.edu
Vinitha Yaski,vinitha@cs.utah.edu

Keywords—Online K Means, Sequential K Means, hard clustering, topic detection.

1 Introduction

THIS work is done with an objective to explore the effectiveness of clustering in determining topic of microblogs document. Conceptually topic modeling and clustering are different, however topic modeling is a viable way of deciding how similar the documents are. This implies that topic modeling is a way to do document clustering. In microblogs setting where the texts are constrained to only limited set of 100-200 characters, the assumption of a document belonging to a single topic is widely accepted in the research domain. In a way, it assumes that multivariate distribution problem can be reduced to a single objective clustering problem for microblogs [1].

The ultimate aim of this work is to produce an online topic discovery framework to cluster same topic documents. Our work started with the exploration of clustering techniques in online setting. We wanted to explore if the microblog topics can be modeled using hard clustering approach. Clustering is an unsupervised learning technique. The basic Llyod's algorithm for clustering is computationally expensive for online settings. A very recent paper [5] by Edo Liberty *et al.* explored an approximate version K-means in Online setting. An early work called "sequential K-means" is also related to same.

Further we explored the work on "Clustering Data Streams" [3] and [2] to make the approach viable in online stream of documents.

We implemented new algorithms on Online K-means [2016], Sequential K-means and applied different settings to explore its relevancy in microblog topic discovery. We are now in the process of implementing clustering data streaming model [3] with Online K-means. If this technique works, this can open a new direction to microblog topic discovery problem.

Since Online K-means itself is a randomized approximation to K-Means algorithm we believe it's weak learnability model can be leveraged to build a strong learnability model by ensemble and boosting method. We propose this direction of work for future research.

The rest of this paper is organized as follows. Section 2 presents the formal problem definition. In section 3, we explain the solution of the problem with Online K-means and Sqequential K-means. We also present a model of incorporating them with *"Clustering Data Streams"* [3] to make it computationally viable and improve performance of the framework. Section 4 presents the experiments, datasets and facts we have explored. We conclude with the direction of future work.

2 Problem Definition and Related Works

The problem of "Microblog Topic Discovery" can be defined as a combination of three sub-problems.

- Representation techniques of documents
- Online clustering

- Small space clustering for large scale document streaming.

2.1 Document Representation

Document representation can be broadly classified into two categories *(a)* Content-based: based on endogenous information of the given corpus. (b) Context-based: exploiting additional, external information in order to acquire more contextual information.

Context-based representation is application dependent and for this project we keep it out of scope for exploration. We represent our documents with content-aware representation techniques.
Future work on document representation technique can be related to content-context aware document representation technique. One of the recent papers on this topic is written by Rode *et al.*[7].

Let D be set of documents, D=$\{d_1, d_2, d_3, d_4, \ldots, d_n\}$. Each document $d_i \in D$ is then represented as a vector $v_{d_i} = (v_1, v_2, \ldots, v_m)$, where m is the size of the vector. m varies with different representation techniques.

1. Term Vector Model: Each document $d_i \in D$ is then represented as a vector $v_{d_i} = (v_1, v_2, \ldots, v_{|W|})$ of size $|W|$ with its j-th dimension v_j quantifying the information that the j-th term $w_j \in W$ conveys for d_i.

Term Frequency vector model takes number of occurances of a term in a document.

Term Frequency- Inverse Document Frequency (TF-IDF) takes into account both the number of occurrences of a term in a document and its overall frequency in the entire corpus, in order to reduce the impact of particularly common words.

2. n-gram model: There are two forms of this model- *character n-grams model*, which relies on sequences of distinctive, frequent letters and the *word n-grams model* which takes words into consideration. Typical values for n are 2 (bigrams), 3 (trigrams) and 4 (four-grams).

A document d_i is, thus, represented by a vector whose jth dimension encapsulates the information conveyed by the jth n-gram for d_i.

3. Word2vec model: Word2vec is a group of related models that are used to produce word embeddings. The simplest version of the word2vec is continuous bag-of-word model (CBOW) introduced in Mikolov et al. [6] or skip-gram model which is build by learning the content of the dataset. Advanced optimization technique like softmax and negative sampling are used to embed the word in a d-dimensional vector. This is made possible by two-layer hidden neural network.

3. Doc2vec model: Similar to word2vec the way to represent sentences or documents as a vector is doc2vec model [4]. A document d_i is, thus, represented by a vector of size m where m is input to the doc2vec model and can be considered as a hyperparameter.
(Details of word2vec and doc2vec model is beyond the scope of this project report)

2.2 Online Clustering

Given D be stream of documents, D=$\{d_1, d_2, d_3, d_4, \ldots, d_n, \ldots\}$. and each document d_i is represented as a vector v_{d_i}. Create a mutually exclusive exhaustive disjoint subsets $S = \{S_1, S_2, S_3, \ldots, S_k\}$ of documents such that similar documents are in the same subset.

One of the most well-studied optimization models in clustering is the k-means clustering. Given the set V of n vectors in Euclidian space. The goal is to partition V into k sets called clusters $S_1, \ldots, S_k$ and choose one cluster center c_i for each cluster S_i to minimize

$$\sum_{i=1}^{k} \sum_{v \in S_i} ||v - c_i||^2$$

We define the problem in online streaming setting where we optimize the above expression by randomized approximation algorithm and output the means of the clusters c_i to predict the future data points.

To our knowledge there are two algorithms for online k-means clustering.

- Sequential k-means
- Online k-means

2.3 Small space clustering in streaming

A framework has been developed by Guha *el al.* [3] for large scale online clustering is relevant to our aim. The need for small space and distributed framework [2] will significantly enhance the computing performance. The summary of the problem is to create $O(k)$ clusters without much loss of accuracy with less main memory.

Details of the framework algorithm is presented in section 3.

3 Our approach

We present our work in this section. Firstly, we present the details of online clusterings such as *sequential k-means* and *online k-means* repectively. Then we present the modified small space algorithm with online clustering methods. Lastly the document representation methods we have explored.

3.1 Sequential k-means

Algorithm 1 Sequential k-means

function SEQUENTIALKMEANS(V, k)
  where $V = \{v_1, v_2, \ldots, v_\infty\}$;
  v_i vector represent of document d_i
  Randomly select cluster means
  $c_1, c_2, \ldots, c_k$
  Set the counts n_1, n_2, ..., n_k to zero.
  for each $v \in V$ **do**
    Acquire the next example, v
    if c_i is closest to v **then**
      Increment n_i
      Replace c_i by $c_i + \frac{1}{n_i} * (v - c_i)$
    end if
  end for
end function

Sequential k-means updates the mean of the cluster one at a time on seeing a example at a time rather than all at once. The algorithm learns the means over a period of time. It is notable if the concept class of the data is learnable then the clustering algorithm will find a hypothesis to obtain a near optimal cluster. Sequential k-means starts clustering even before we have seen all of the examples. The sequential algorithm is presented in Algorithm 1.

3.2 Online k-means

Online k-means produces $O(k)$ cluster means.[5] This randomized approximation algorithm also gives a bound on the cost of the cluster. Let W be the cost of the online assignments of Algorithm 2 and W^* the optimal k-means clustering cost then

$$\mathbb{E}[W] = O(W^* \log n)$$

where n is the number of vectors it has seen till now.

Algorithm 2 Online k-means

function ONLINEKMEANS(V, k)
  where $V = \{v_1, v_2, \ldots, v_\infty\}$;
  v_i vector represent of document d_i
  $C \leftarrow$ the first $k+1$ distinct vectors in V;
  $n = k + 1$
  $w' \leftarrow \min_{v,v' \in C} ||v - v'||^2/2$
  $r \leftarrow 1; q_1 \leftarrow 0; f_1 = w'/k$
  for $v \in$ the remainder of V **do**
    $n \leftarrow n + 1$
    $p =$min$(D^2(v, C)/fr, 1)$
    $r =$random_uniform$(0, 1)$
    if $r < p$ **then**
      $C \leftarrow C \cup \{v\}; q_r \leftarrow q_r + 1$
    end if
    if $q_r \geq 3k(1 + \log n)$ **then**
      $r \leftarrow r + 1; q_r \leftarrow 0; f_r \leftarrow 2 \cdot f_{r-1}$
    end if
    yield $c = \arg\min_{c \in C} ||v - c||^2$
  end for
end function

Note: Due to space constraint we are restricting the description of Online-k-means.

3.3 Small Space algorithm

The initial algorithm for small space is presented in Algorithm 3. This algorithm produces constant approximation of the optimal KMeans algorithm. (The proof is beyond the scope of this write up)

We propose the modification of the algorithm with slight variation and apply the online-k-means algorithm for finding the clusters. This

Algorithm 3 Small space

function SMALLSPACE(V, k, l)
where $V = \{v_1, v_2, \ldots, v_n\}$;
v_i vector represent of document d_i
l is the number of parallel operation possible on the system
Divide V into l disjoint pieces $\chi_1, \chi_2, \ldots \chi_l$
For each χ_i, find $O(k)$ centers in i. Assign each point in χ_i to its closest center.
Let χ' be the $O(lk)$ centers obtained in by above step,
where each center c is weighted by the number of points assigned to it
Cluster χ' to find k centers.

method uses the ensemble techniques to predict the assignments for future vectors.

We are in the process of implementing Algorithm 4. We are done with all the other parts. We present the comparative performance of Sequential K-Means and Online K-means in the experiment section.

3.4 Document to vector

For the experiment we are using doc2vec model to generate the vectors from documents. In future we would like to test it with different documents representation technique.

4 EXPERIMENT

The datasets we used are from UCI dataset, Social Sensor dataset, Indian news dataset (University of Utah).

Our implementation is present in github. A demo of cluster is shown here We present different analysis of the clustering algorithm such as the effect of dimension of vectors, effect of k values.

Algorithm 4 Modified Ensemble Online K-means

function ENSEMBLE ONLINE K-MEANS($V_{train}, V_{test}, , k, m, l$)
Input:
$V_{train} = \{v_1, v_2, \ldots, v_n\}$;
v_i vector represent of document d_i
l is the number of parallel operation possible on the system
m is the ensemble parameter, $m > l$

Training Part:
Create m number of instances of Online k-means $KM_1, KM_2, \ldots KM_m$.
for $v \in V_{train}$ **do**
with probability $\frac{l}{m}$
input v to instances of KM_i
end for
For each $_i$, get $O(k)$ centers in i.
Let χ' be the $O(mk)$ centers with cluster sizes obtained in by above step
Cluster χ' to find k centers with the assignments of $O(mk)$ points.
Note: Each point now represents a cluster.

Predict part:
for for a vector $v \in$V$_{test}$ **do**
Find 3-NN cluster from k centers
From 3 clusters find the nearest point and add to it.
Increase cluster size for that point

Update part: new cluster discovery
if cluster size exceeds average size of clusters. **then**
Make this *k+1*th center
$k \leftarrow k + 1$
end if
end for

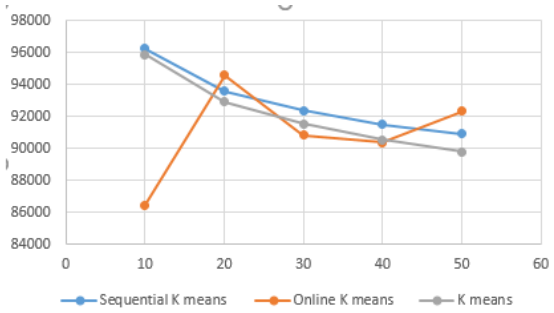

Figure 1: Social Sensor FA cup dataset

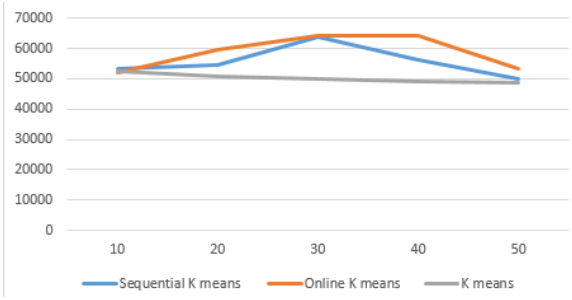

Figure 2: News dataset

In figure 1 the x-axis and y-axis represents k-value and cluster cost respectively. The online-k-means can't be restricted to exact k value it optimizes k based on the data however it is $O(k)$. Hence in figure 1 for k=10 the online-k-means cost is significantly lower because k_{actual} is greater than 10.

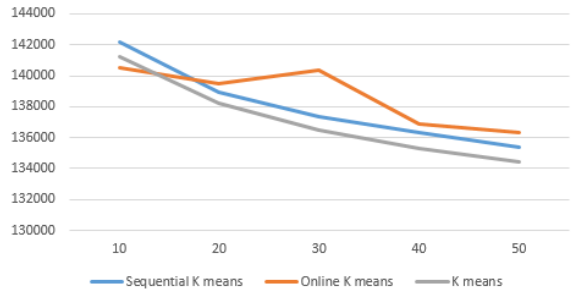

Figure 3: UCI news dataset

It is notable that online-k-means is comparable to K-means.

Here we present the effect of vector dimension on clustering. With increase of vector size the result is consistent with other methods. In figure 4 the x-axis and y-axis represents dimension of vector and cluster cost respectively. We have taken k=20 while varying vector size.

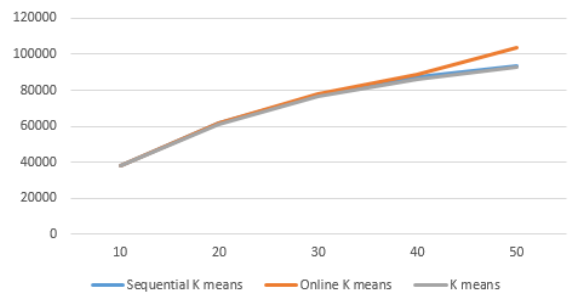

Figure 4: Effect of vector dimension on FA cup dataset

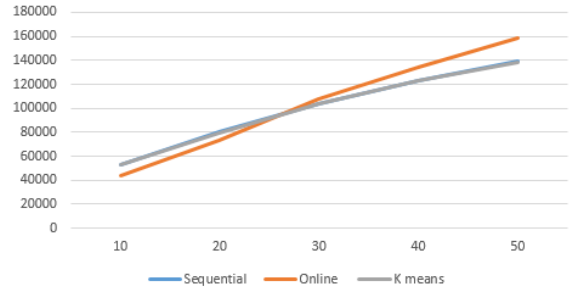

Figure 5: Effect of vector dimension on News dataset

From the experiment we can conclude that online k-means, sequential k-means and standard k-means are comparable. We are yet to test them on large streaming dataset. Sequential k-means and Online k-means can be used wisely on large scale online data and will be effective because of its minimal computation cost.

Note: Due to lack of ground truth cluster and available time we could not report the F-measure of the clusters.

5 Conclusion and Future Work

We propose an algorithm which we believe can be used for hard clustering and in a way will help in discovering topics from microblogs. We present experimental proof of how online k-mean clustering is comparable with than sequential k-means and K-means in practical setting.

Our future work is to complete the implementation of Ensemble Online K-Means, and apply different document representation methods to find the best solution.

REFERENCES

[1] David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM, 2006.

[2] Alina Ene, Sungjin Im, and Benjamin Moseley. Fast clustering using mapreduce. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 681–689. ACM, 2011.

[3] Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. Clustering data streams: Theory and practice. *IEEE transactions on knowledge and data engineering*, 15(3):515–528, 2003.

[4] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196, 2014.

[5] Edo Liberty, Ram Sriharsha, and Maxim Sviridenko. An algorithm for online k-means clustering. *arXiv preprint arXiv:1412.5721*, 2014.

[6] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[7] Henning Rode and Djoerd Hiemstra. Conceptual language models for context-aware text retrieval. 2005.