A Fuzzy, Incremental and Semantic Trending Topic Detection in Social Feeds

Mona A. Abou-Of
Department of Computer Engineering
Pharos University in Alexandria, Egypt
mona.abouof@pua.edu.eg

2020 11th International Conference on Information and Communication Systems (ICICS) | 978-1-7281-6227-0/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICICS49469.2020.239492

***Abstract*—Nowadays, a huge number of people participating in social networks is triggering a fast and wide spectrum of topics. Such trending topics are usually derived from the most frequent searches, the published posts and the daily news. The automated analysis for such data requires topics detection and tracking methods. Many challenges are being faced. It is difficult to discover the semantic relatedness when the same event is presented by different titles and to handle merging semantically identical topics from different channels (aggregation). Other hardships are the vagueness regarding the vast web collection, the scalability to analyze them, and the fact that it is a time consuming task. The framework introduced in this paper aims to solve these issues. Because a web document often consists of several topics, the suggested model employs a fuzzy C-Means (FCM) clustering based trending topics detection. It applies a semantic document similarity algorithm to resolve such ambiguity issues caused by the usage of synonyms, homonyms or different abstraction levels. This algorithm is also used to summarize the long documents. Furthermore, an incremental clustering technique is utilized to preserve high cohesiveness up-to-date top trending topics. The experimental results finally illustrate the effectiveness and the superiority of this model, compared with other trending topics detection algorithms, in terms of entropy and F-score measures.**

Keywords—News Aggregator, Trending Topics Detection, Semantic Similarity, Text and Web Mining, NLP, Incremental FCM Clustering.

I. INTRODUCTION

The increasing number of individuals joining the Internet gives us great information of social feeds with various individual tastes, and human conduct. The web has dependably been quickly developing. At present, social highlights are available on numerous websites with a relevant idea called "Trends". Understanding client's associations is an intricate and extensive theme. The covering of interesting questions and methods of analysis requires volumes of text. The core of these complex interactions is the relationships that people create online, either directly with other people or through the web trending topics. The objective of this research is to introduce a model for analyzing those hidden relationships and building a personalized smart trending of web topics. The suggested model aggregates data from different resources, such as user predefined resources or prepared categories. All these collected data are filtered in order to provide users with the required information. Thus, the main purpose is to reduce the time that the user wastes on "scrolling" by providing trends over appropriate regions or categories. As shown in [1], a few hot trending topics is produced from the large dataset crawled in October 2017 among 3 regions (Egypt, KSA, and world) and under 2 categories (sports and politics).

A trending topic discovery model is highly needed to use the provided lists of web topics to form clusters, also to contextualize these trending topics and to arrange them as top events around the globe. It is very important to track emerging related topics at the occurrence of major events. This paper mainly presents two successive algorithms to achieve the required target and to overcome the faced challenges as shown in the proposed framework. The first algorithm is the "Semantic Document Similarity", developed in [1], that is a refined measure of [2]. It utilizes the overlap concept to find the longest common substring with maximal consecutive words. It is needed twice in the suggested model: in the document summarization algorithm, then in the incremental FCM clustering. Document summarization is highly needed when a long web document is being received in order to reduce the clustering running time. The second is an "Incremental FCM clustering" algorithm in which the fuzzy concept is employed to handle the uncertainty within the aggregation. In addition, It is incremental to preserve high cohesiveness hot trending topics, while new documents are being received. This process assigns the new arriving documents to the current created clusters and reduces the overall clustering running time. In the next section, a related literature on trending topics detection and fuzzy clustering algorithms are presented. In section 3, the applied techniques, and overview of the suggested model are discussed, also the "Semantic Document Similarity" algorithm is analyzed. Then a novel technique is demonstrated in the second algorithm of incremental FCM clustering for updating trending topics reports with daily news in section 4. Finally, we compare the experimental results with other techniques described in [1, 3] in section 5, followed by the conclusion and the future work in section 6.

II. RELATED WORK

Nowadays, numerous organizations institutionalize their activities through Business Process (BP). They are put away in storehouses and reused when new functionalities are required [4]. One task in this field is to detect topics of general interest at the moment they occur. To summarize the events in real world, a significant trending topic detection method was introduced in [5, 6]. A deep understanding of social media discussions and their involved demographics became critical for many applications like political or business analysis [7]. Clustering is the most important step in topics aggregators.

Recently, a wide survey of clustering algorithms has been given in [8].The center-based partitioned clustering FCM algorithm relies on minimizing an objective function that compromises a weighted sum of all distances between each data point and its cluster representatives. Fuzzy clustering can start by handling the presences of outliers and noisy data as in [9]. The Gustafson and Kessel (GK) clustering algorithm is the first important extension to the FCM algorithm. Some of the recent proposed fuzzy algorithms are GK based like [10, 11]. They can be classified either multi-attributes approach in a weighting scheme as in [12], a maximization of the Shannon's entropy of membership functions to the GK objective function [10], or optimization of multi-objective generic algorithms [13-15]. As regards to these applications areas, incremental clustering ideas have been acquired from [16, 17]. In this paper, these approaches were combined together to enhance the clustering performance within a day or within multiple days of received topics over time.

III. PROPOSED TRENDING TOPICS DETECTION

The proposed incremental semantic trending topics discovery model integrates the FCM clustering technique and the incremental semantic measures to improve the accuracy of this trending topics aggregation.

A. Problem Formulation

In the broadest meaning, the main goal is to map between two sets of correlated trends in order to discover patterns in one of the trends sets. For fast access to the trends, they are stored in clusters.

Definition 1(Trend, Domain): A trend e is a tuple t = (start, end, type, confidence) with start, end $\in$ TIME, type$\in$ C, where C is a set of trend types and confidence $\in [0, 1]$. T denotes the set of all trends. A domain D is a pair (length, E) consisting of its duration length and a set of trends E $\subseteq$T.

Definition 2 (Automated and Manual Trends): The set of trends in a domain D is partitioned into two disjoint sets A and B: the automated trends A$\subseteq$E and the manual trends B $\subseteq$E where E = A $\cup$B. Furthermore, for all t = (start, end, type, confidence) $\in$E we impose $1 \leq$ start<end $\leq$ length; and for all t = (start, end, type, confidence) $\in$ A, and t' = (start', end', type', confidence') $\in$B we impose confidence' = 1 and type$\neq$ type'.

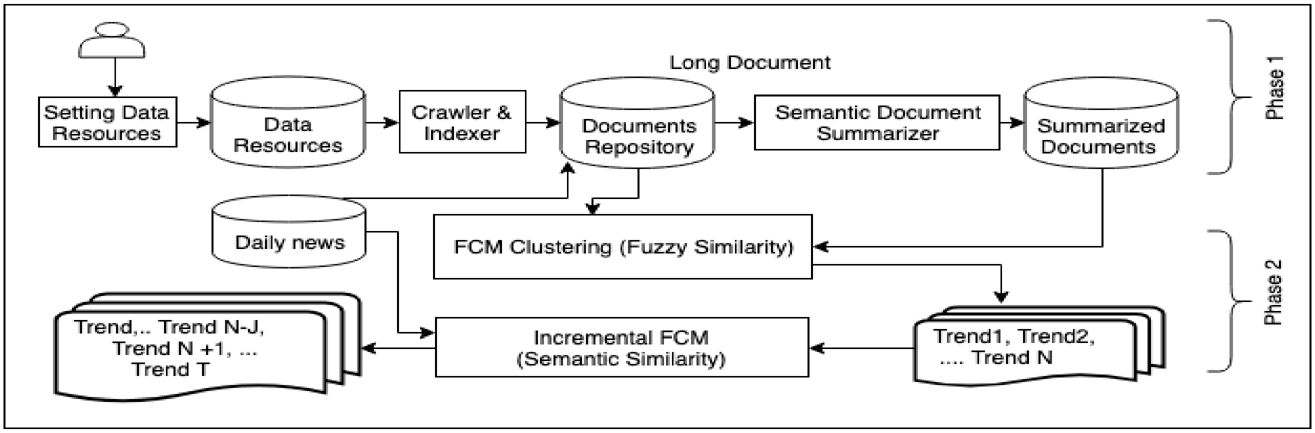
Long Document
Setting Data Resources
Data Resources
Crawler & Indexer
Documents Repository
Semantic Document Summarizer
Summarized Documents
Phase 1
Daily news
FCM Clustering (Fuzzy Similarity)
Trend,.. Trend N-J, Trend N +1, ... Trend T
Incremental FCM (Semantic Similarity)
Trend1, Trend2, ... Trend N
Phase 2

Fig. 1. Architecture of the FCM web trending topics model

Definition 3 (Types, Trends, Evaluation): The Function types: T → C maps trends to their according types, the function trends: C × D →T project domains to one of the selected types. The evaluation function $evalA,b: N \rightarrow B$ is a mapping indexed by a set of trends $(A, b)\in 2A\times B$ used for binary classification.

Definition 4 (Trend Detection Task): Given a domain set $\rho = \{D1, . . . , Dn\}$ inducing sets of automated trends $\Lambda = \{A1, . . . , An\}$ and manual trends $\beta = \{B1, . . . , Bn\}$, the trend learning task is to retrieve a binary classifier evalA,b that for set Ai $\in\Lambda$ determines the existence of b $\in$ types (Bi) , Bi $\in\beta$, in a given step, $\forall$ i$\in\{1, . . . , n\}$.

B. Proposed Model Architecture

As shown in Fig. 1, the architecture of the suggested framework consists of two consecutive phases. In the first phase, the main document repository is built by collecting several data resources and user preferences. Then the summarized documents are obtained using the crawler and the summarizer. In the second phase, as described in section 4, the similarity between documents is measured in order to classify the web trending topics with similar patterns using a soft FCM clustering technique. Then this model tracks all daily news in an incremental manner based on user preferences. The following subsections illustrate the different parts of this model.

B.1 Build Documents Repository

The main objective of phase 1, shown in Fig. 1, is to build documents repository that contains the summarized version of all long documents (Web pages) suitable for trending topics' clustering process. This phase includes the following three steps:

Step 1: Setting Data Resources

The user starts to follow the Web portals over some pre-defined regions and categories. In addition, he is updated with hundreds of trends dynamically aggregated. All the data resources, the pre-defined categories, the attached sites containing news, and the current hot topics sources are added to the repository. Furthermore, the user is allowed to increase the number his own favorite sites by adding all URLs on the mobile applications or the Web portal.

Step 2: Crawler & Indexer

A Web crawler starts with an inventory of URLs. The crawler visits these URLs by the multi-threaded downloader to extract all pages contained in a fast way; it identifies all the hyperlinks within the page and adds them to the list of URLs provided to the scheduler. All the current URLs and the recursively visited URLs according to a set of policies (e.g. a number of nested URLs) are added to the queue. In the indexer step, pages content are extracted based on linguistic features. The indexer starts with an article (the output of crawler) and fetches the corresponding appropriate keywords with a high-frequency rate. All these keywords and its corresponding URLs are stored into the documents repository.

Step 3: Semantic Document Similarity

An Automatic summarization of a text document became an essential requirement to resolve the problem of long articles. Text summarization can be mainly classified as automatic **extractive** (which extracts key phrases from the web document and combines them to produce a summary), and **abstractive** (which involves paraphrasing and shortening parts of the web document). Herein, the suggested model uses extractive text summarization for its simplicity and for keeping the original content. This extractive text summarization is described as follows in algorithm 1:

Algorithm 1: Semantic Document similarity

1. Execute Natural Language Processing (**NLP**) steps based on Wordnet library [18]: tokenization, stemming, and then part of speech (POS) [19-21].
2. Find the most suitable senses for each subject (e.g. "Interest" from a bank vs. "Interest" in a subject) by Word Sense Disambiguation (**WSD**) technique [22]. A word is appointed to the sense whose gloss shares the most important variety of words in common [18].
3. Reduce the subject's **overlap**. As stated in [23], the length of words is inversely proportional to their use. Measuring the overlap between two strings is reduced in order to resolve the way of finding the longest substring with maximal consecutive words. Every overlap that contains N consecutive words leads to N^2 in the score of the gloss sense combination.
4. Given a list of sentences (S_1, S_2,. .. , S_n) without disambiguation or overlapping, the **semantic similarity** is computed for each pair of sentences to remove similar sentences. The higher score indicates more meaning similarity in the two sentences. The semantic similarity between two sentences t_1 and t_2 is measured by:

   $$sim(t_1,t_2) = \frac{1}{2}.(\frac{\#(s(t_1)\cap s(t_2))}{\#s(t_1)+1} + \frac{\#(s(t_1)\cap s(t_2))}{\#s(t_2)+1}) \qquad (1)$$

   where *#S(t)* represents the number of subjects in the sentence *t*.
5. Build the semantic **similarity relative matrix** $R_{m\ x\ n}$ for each pair of word senses (synset) of the given sentences t1 and t2, where $m=|t_1|$, $n=|t_2|$. $R[i, j]$ is the semantic similarity measure between the most suitable senses of word at position *i* in sentence t_1 and the most suitable senses of word at position j in sentence t_2. Thus, $R\ [i,j]$ is the weight of the edge connecting *i* to *j*. When a word does not exist in the dictionary, the edit-distance similarity method is applied and outputs a lower weight. The edit distance method quantifies how dissimilar two strings (e.g., words) are to one another by counting the minimum number of operations required to transform one string into another string. If t1 and t2 are two sets of disjoint nodes (words and its senses), the Hungarian method is used to compute the total matching weight of bipartite graph [24].
6. Combine the match results from the previous step by matching average with the Dice coefficient [25] in order to produce a single similarity value for the two sentences. Dice's coefficient measures how similar a set and another set are. It can be used to measure how similar two strings are in terms of the number of common bigrams (a bigram is a pair of adjacent letters in the string). Sentences that are connected to many other sentences are likely to be the center of the paragraphs and will be included in the summary.

This summarization step is performed on long articles and the output is stored into the summarized documents repository. The short articles go directly from documents repository to phase 2, described in the next section.

IV. BUILDING TRENDS USING INCREMENTAL FCM

The main objective of phase 2 of the suggested trending topic detection model (shown in Fig. 1) is to build trends from both the documents repository (for short articles only) and the summarized documents repository (for a summarized version of long articles) by utilizing clustering algorithms in an incremental manner. This phase includes the following steps:

A. Step 1: Initial Documents Clustering Using FCM

At first, each document is represented as a vector using the algebraic vector space model (VSM). By definition, if a term occurs in the document, its value in the vector is non-zero. Several different ways of computing these values, also known as (term) weights, have been developed. One of the best-known schemes is Term Frequency-Inverse Document Frequency (TF-IDF) weighting. The definition of the term depends on the application. Typically terms are single words, keywords, or longer phrases. If words are chosen to be the terms, the dimensionality of the vector is the number of distinct words in the vocabulary. Given the feature vector associated with each document, the FCM clustering algorithm is used to build the initial group of trends.

As known, K-Means is a hard clustering technique where each point is belonging to only one centroid. In contrast, in the soft clustering fuzzy C-Means, each point can be associated to more centroids but with different quality. This is useful when a trend is related to many categories rather than only one category, e.g. trend belongs to a political sport, political science, etc.. The traditional K-

Means handles the trends according to their belonging to one cluster; whereas the fuzzy version handles them according to their belonging to multiple clusters, which is more realistic. This method is frequently used in pattern recognition. It is based on the minimization of the following objective function:

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{c} u_{ij}^{m}\left\|x_i - c_j\right\|^2 \qquad 1 \leq m \leq \infty \qquad (2)$$

Where the fuzzyfier m is any real number greater than 1, uij is the degree of membership of xi in the cluster j, xi is the ith of d-dimensional measured data element, cj is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center. Fuzzy partitioning is achieved by an iterative optimization of the objective function shown above, and the membership u_{ij} and the cluster centers c_j are updated by:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{\left\|x_i - c_j\right\|}{\left\|x_i - c_k\right\|}\right)^{\frac{2}{m-1}}} \quad , \quad c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \qquad (3)$$

The iteration is stopped when

$$max_{ji} = \{|\ u_{ji}\,(k+1)\ -\ u_{jij}(k)\ \ |\}\ <\ \ \varepsilon \qquad (4)$$

where ε is a termination criterion between 0 and 1, whereas *k* is the current iteration. This process converges to a local minimum or a saddle point of J_m.

The incremental FCM algorithm "Algorithm 2" will be similar to FCM, except that number of cluster classes may increase to a certain limit. Although the FCM algorithm may take more time if the data set is large, the proposed system will be fast, adaptive and dynamic.

Algorithm 2: Incremental FCM with Semantic Similarity

```
Input:
Main_List: list of all N documents
New_Doc: new document to be clustered
c: number of classes, T:time interval
m: the fuzzyfier, ε: the termination threshold
Output:
Main_Clusters; i.e the centers vectors $C_c$
1. Initialize c between 2 and N and the partition matrix $U_{Nxc}(0)$
2. Run Fuzzy CMeans on all documents Main_List (once per period T).
    2.1. At step k: calculate the centers vectors $C(k)=[c_{j,},\forall j=1,..,c]$ with U(k-1)
    2.2. Update the membership matrixU(k).
    2.3. If || U(k) - U(k-1)|| ε < then STOP; otherwise return to step 2.1
3. Incremental Run (for any arriving New_Doc)
    3.1. Compute all semantic similarity (sim) between New_Doc and all clusterscenters (Algorithm1) and increment N.
    3.2. If New_Doc is belonging to cluster with Sim > 0.8, assign it to this cluster and return Main_Clusters
    3.3. If New_Doc is belonging to class with Sim > 0.5, split this cluster, increment c, and return Main_Clusters
    3.4. Else increment c, build a new cluster center with New_Doc, run Fuzzy CMeans on all documents Main_List once, i.e update U(k)then C(k)and return Main_Clusters
4. Archive inactive clusters
5. Repeat Step 3 for any arriving New_Doc within T period, otherwise repeat step2.2 starting with last generated clusters centers
```

B. Step 2: Incremental Documents Clustering

The aim of this step is to initially do a full clustering for all the summarized documents and then perform an incremental one for any newcomer document. This is achieved by the usage of a coarse-grained technique (fuzzy similarity) for all the summarized documents. In contrast, for the newcomer document, it requires a fine-grained technique (Semantic Similarity) to fit it in one of the already made c clusters. Such process will reduce the overall time consumed. The suggested model schedules the full clustering every morning and then the incremental clustering is run for all new items within that day through the end of the day. The whole process is then repeated every single day. The semantic similarity (described in algorithm 1 within phase 1) is used to measure the similarity between newcomer articles of daily news and the current clusters' centers. Algorithm 2 "Incremental FCM" (IFcm) summarizes the main steps to build these up-to-date trends. It preserves high cohesiveness hot trending topics while new documents are being received. This process allows new documents to be assigned to clusters that have already been built.

The main contribution of this suggested model is to investigate the effect of semantic fuzzy outputs by adding a semantic similarity to the fuzzy clustering in an incremental manner. In this regard, the work is going to extend a commonly used clustering-based aggregation trending topic technique to an incremental fuzzy clustering technique. The suggested model towards trending topic aggregation, based on news context modeling and text mining, shows the benefits of using such semantic similarity algorithm in the incremental clustering. It has refined the trend detection and addresses the identification of the semantically related topics from user preferences.

V. EXPERIMENTAL RESULTS

In this section, the efficiency of the suggested model is analyzed. Experiments were conducted to demonstrate the robustness of this model. The clustering results of this presented model are evaluated by two well-known measures named the Entropy and the F-measure. Generally, lower values of entropy and higher values of F-measure represent better clustering results.

A. Dataset

The benchmark dataset used to evaluate different document clustering algorithms has been downloaded from the testbed [26, 27]. Table 1 shows the properties of this testbed. **Reu_01** dataset contains 1000 documents classified in 5 classes. It is represented over 5 countries (UK, USA, Japan, Canada, and Switzerland). The **Re0** dataset consists of 1500 web documents among 10 classes. It is represented over 5 countries (UK, USA, Japan, Canada, and Switzerland) and among 5 topics (coffee, cocoa, plywood, corn, and Rice). Finally, the **20Newsgroup** dataset is organized into 20 different newsgroups, each is corresponding to a different topic: graphics, atheism, windows. misc, Mac hardware, IBM hardware, windows.x, for sale, motorcycles, autos, rec.sport.hockey, sport.baseball, crypt, electronics, sci.space, sci.med, religion Christian, talk politics Mideast, talk politics guns, politics.misc, talk.religion. Some of the newsgroups are very closely related to each other (e.g. comp.sys.mac.hardware/ comp.sys.ibm.pc.hardware), while others are highly unrelated (e.g soc.religion.christian/ misc.forsale).

TABLE 1 PROPERTIES OF TEXT DOCUMENT DATASETS

Dataset	# Documents	# Classes
Reu_01	1000	5
Re0	1500	10
Mini_Newsgroup	2000	20

B. Software

Preprocessing of the text documents has been implemented by PHP-MySQL Search Engine. Sphider is a lightweight web spider and search engine. It is written in PHP and uses MySQL as its backend database. It includes an automated crawler which can follow the links found on a site, and an indexer that builds an index of all the search terms found in the pages. To build semantic similarity and NLP procedures, Visual Studio and WordNet.Net library have been used. WordNet ® is a large lexical database of English. Nouns, verbs, adjectives, and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. In the suggested model, the fuzzy clustering procedure has been implemented using Matlab Fuzzy Toolbox.

C. Experiment

The presented experiment validates the benefits of implementing the suggested model (IFcm) for trending topics building. This experiment compares this model with other related text clustering models— including incremental K-means (IKMeans) [1] , gravitational search algorithm and k-harmonic means (GSA-KHM) and gravitational search algorithm and k-means (GSA-KM) [3]. The experiment is performed on the three datasets: Reu_01, Re0, and Mini Newsgroup while varying the number of clusters (K).

D. Results

In terms of Entropy and F-measure, the experiment results of the proposed IFcm in comparison with IKMeans, GSA-KM, and GSA-KHM are reported on those datasets mentioned in Table 1. It is clearly observable that the results of the IFcm based trending topics aggregation models are better than those that depend on IKMeans, GSA-KM, and GSA-KHM clustering. Fig. 2 and Fig. 3 for the Reu_01 dataset reveal the superiority of the suggested model for both F-measure and entropy. The recommended trending topics aggregation model achieves an approximately 304 % decreasing in entropy compared to the later ones. While it achieves an increase in F-measure of approximately 53 % compared to them. The results of both Mini_Newsgroup and Re0 datasets are shown in Fig. 4 to Fig. 7. These results confirm the superiority of the suggested model for trending topics aggregation regarding different datasets. In general, for all datasets, the suggested model yields on average 148 % decrease in entropy whereas it achieves a 41 % increase in F-measure. One possible explanation of these results is that the suggested model relies on semantic similarity instead of traditional similarities to build clusters. Furthermore, implementing the incremental clustering increase high cohesion between clusters items in addition to nice-detecting of the semantic characteristics in the incremental round.

Compared to the non-fuzzy IKMeans clustering procedure, for both Reu_01 and Re0 datasets, IFcm produce san approximately 57.5 % decrease in entropy. For all Re_01, Re0 and Mini_Newsgroup datasets, the value of F-Measure of IFcm was better with on average 26 % increase. In some cases IKMeans's entropy is not as good as in Mini_Groups dataset, this is because of great overlap between classes for the same item. In contrast the IFcm handles this overlap when increasing K, allows their belonging to multiple classes, and achieves better entropy. For all values of K expect K =5 and K =6, the entropy has been decreased by approximately 49 % in average, and the F-Measure has been increased by approximately 24 % on average. The superiority of the fuzzy clustering comes from the fact that the fuzzy algorithm minimizes intra-cluster variance, and the results depend on the initial choice of weights only not k. The fuzzyfier m determines the level of cluster fuzziness. A large m result in smaller membership values w, and hence, fuzzier clusters. For K =5 and K=6, one possible reason for higher entropy is that the resulting membership values do not always correspond well to the degree of belonging of the data. As shown by these experiments, illustrated in the mentioned figures, IFcm produces distinctive results that are better than IKMeans, GSA-KM, and GSA-KHM in terms of the entropy and the F-measure for all used datasets.

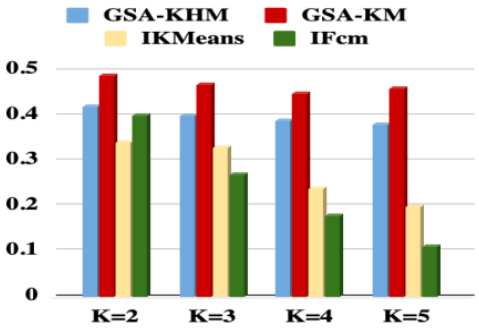

Fig 2: The entropy comparison between GSA-KM, and GSA-KHM, and IFcm clustering methods on Reu 01 dataset

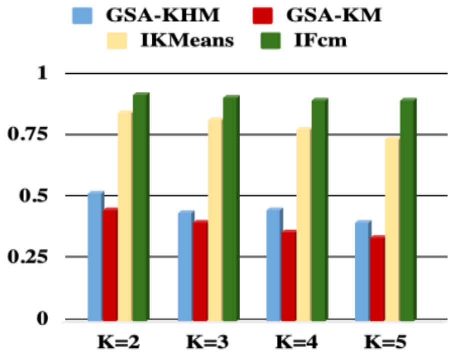

Fig 3: The F-measure comparison between GSA-KM, and GSA-KHM, and IFcm clustering methods on Reu 01 dataset

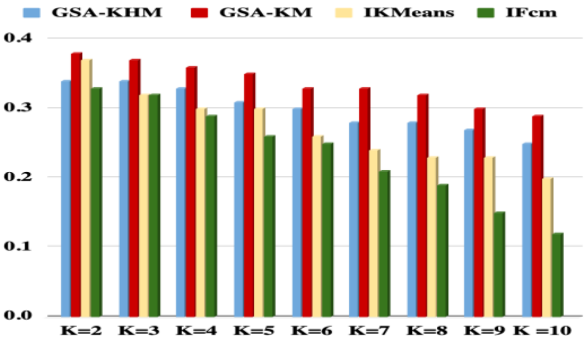

Fig 4: The entropy comparison between GSA-KM, and GSA-KHM, and IFcm clustering methods on Re0 dataset

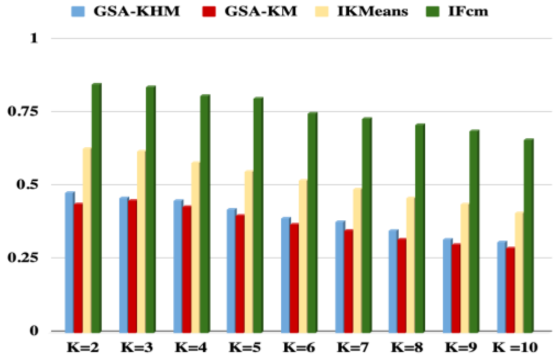

Fig 5: The F-measure comparison between GSA-KM, and GSA-KHM, and IFcm clustering methods on Re0 dataset

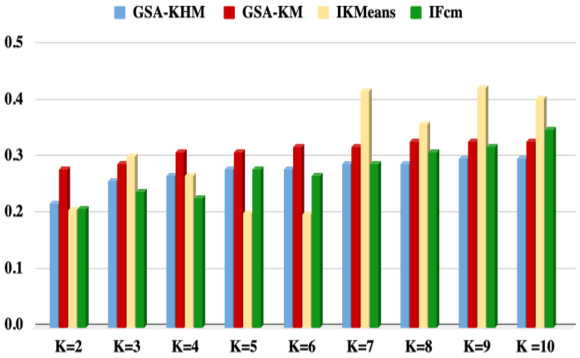

Fig 6: The entropy comparison between clustering methods on Mini_Newsgroup dataset

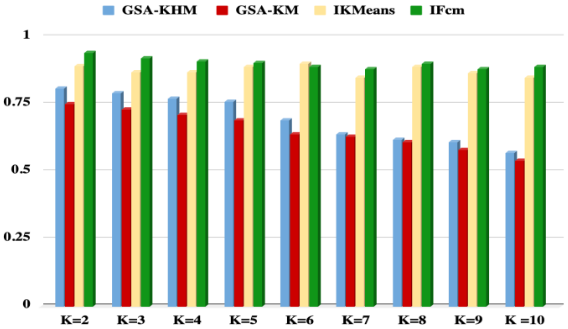

Fig 7: The F-measure comparison between clustering methods on Mini_Newsgroup dataset

VI. CONCLUSIONS AND FUTURE WORK

The proposed trending topic detection model resembles a magazine with hot-topics which merges data from several data resources, such as user-defined resources or prepared categories. Commonly, identifying all the possible trends based on user's preferences might search for information on a topic that can be very difficult. The suggested model can help to identify search terms that are a solid representation of the possible ways users may search around a topic. This model targets the identification of semantically related topics from user preferences. Their textual contexts are collected from the news search. Then, clustering and periodical tracking of trending topics are applied.

At first, the model builds the main document repository by the use of different data resources. Then the summarized documents are obtained using the crawler and the summarizer. Next, the similarity between documents is measured in order to classify the web trending topics with similar patterns using the soft fuzzy C-Means. This model tracks all daily news using an incremental building user profiles module that updates user profiles. The main advantages of the suggested model are that the running time has been reduced twice without any data loss or dependence by providing summarized documents for clustering and by using a searching scheme in an incremental manner. Also, all language issues have been resolved by using a set of well-known semantic techniques.

By quantitative experiments on manually annotated trends, the practice of the suggested model was validated. The comparative study confirms the superiority of the suggested model compared with two other well-known algorithms, using three different online datasets. It yields at least 40% improvement in terms of entropy and 20% improvement in F-measure.

The remaining key challenges – and basis of future work– include the integration of the other appropriate context sources for different languages with the right-to-left like Arabic style. They also include the usage of big data which boost the ability of the model implementation in the real world. Distributed computing technology that can be applied to the dynamically updated user selection algorithm to speed up the training is another interesting area.

REFERENCES

[1] M. Abou-Of, H. Saad, and S.M. Darwish, "Smart and Incremental Model to Build Clustered Trending Topics of Web Documents", International Conference on Advanced Machine Learning Technologies and Applications, Springer, pp. 888-897, Cham, 2019

[2] S. Fuchs, D. Borth, and A. Ulges, “Trending Topic Aggregation by News-Based Context Modeling”, Proceedings of the 39th Annual German Conference, Advances in Artificial Intelligence, pp. 162–168, Germany, Springer, 2016.

[3] M. Mirhosseini, “A clustering approach using a combination of the gravitational search algorithm and k-harmonic means and its application in text document clustering”, inter Turkish Journal of Electrical Engineering & Computer Sciences, Iran, 2016.

[4] M. S. C. Sapul, T. H. Aung and R. Jiamthapthaksin, “Trending topic discovery of Twitter Tweets using clustering and topic modeling algorithms”, Proceedings of 2017 14th International Joint Conference on Computer Science and Software Engineering, Thailand, IEEE, 2017.

[5] T. Georgiou, A. El Abbadi, and X. Yan, “Privacy-Preserving Community-Aware Trending Topic Detection in Online Social Media”, Ch.11 of DBSec 2017: Data and Applications Security and Privacy XXXI, pp 205-224, USA, Springer, 2017.

[6] L. Recalde, D. F. Nettleton, R. Baeza-Yates, “Detection of Trending Topic Communities: Bridging Content Creators and Distributors”, Proceedings of the 28th ACM Conference on Hypertext and Social Media, pp 205-213, Prague, Czech Republic, ACM, 2017.

[7] T. Georgiou, A. El Abbadi, and X. Yan, “Extracting Topics with Focused Communities for Social Content Recommendation”, Proceedings of The 20th ACM Conference on Computer-Supported Cooperative Work and Social Computing, USA, ACM, 2017.

[8] M. Morchid, D. Josselin, Y. Portilla, R. Dufour, G. Linarès, “A Topic Modeling Based Representation To Detect Tweet Locations”, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, France, 2015.

[9] H. Fritz, L.A. GarcíA-Escudero, and A. Mayo-Iscar. “Robust constrained fuzzy clustering”, Information Sciences, 245, pp.38-52, 2013.

[10] J. Yu, and M.S.Yang, “Deterministic annealing Gustafson-Kessel fuzzy clustering algorithm”, Information Sciences, 417, pp.435-453, 2017.

[11] M.S.Yang, and Y.C. Tian, “Bias-correction fuzzy clustering algorithms”, Information Sciences, 309, pp.138-162, 2015.

[12] P. D’Urso, and R. Massari, “Fuzzy clustering of mixed data”, Information Sciences, 505, pp.513-534, 2019.

[13] A. Zhou, Y. Wang, and J. Zhang, “Objective extraction via fuzzy clustering in evolutionary many-objective optimization”, Information Sciences, 2018

[14] A.K. Paul, and P.C. Shill, “New automatic fuzzy relational clustering algorithms using multi-objective NSGA-II”, Information Sciences, 448, pp.112-133, 2018.

[15] A. Saha,andS. Das, “Axiomatic generalization of the membership degree weighting function for fuzzy c means clustering: Theoretical development and convergence analysis”, Information Sciences, 408, pp.129-145, 2017.

[16] J. Wang, A. Zelenyuk, D. Imre, and K. Mueller, “Big Data Management with Incremental K-Means Trees–GPU-Accelerated Construction and Visualization”, Informatics Open Access Journal, USA, 2017.

[17] M. Nazrul, M. Seera, C.K. Loo, “A robust incremental clustering-based facial feature tracking”, Applied Soft Computing, Vol. 53, pp 34–44, Elsevier, 2017.

[18] G. Miller, Princeton Univand NJ. Princeton, “WordNet: a lexical database for English”, Published in Magazine Communications of the ACM CACM, Pages 39-41, ACM, USA, 1995.

[19] M. D. Manning, M. Surdeanum, J. Bauer, J. Finkel, S. J. Bethardm and D. McClosky, “The Stanford CoreNLP Natural language Processing Toolkit”, Proceedings of the 52 nd Annual Meeting of the Association for Computational Linguistics: SystemDemonstrations, pp. 55-60, Maryland, 2014.

[20] K. Merchant and Y. Pande, “NLP Based Latent Semantic Analysis for Legal Text Summarization”, Proceedings of the International Conference on Advances in Computing, Communications and Informatics, pp.1803-1807, India, 2018.

[21] T. Weia, Y. Lu, H. Chang, Q. Zhoua, and X. Bao, “A Semantic Approach for Text Clustering using WordNet and Lexical Chains”, Expert Systems with Applications, Vol. 42, pp. 2264-2275, 2015.

[22] E. Corra, A. Lopes, D. Amancio, “Word sense disambiguation”, Intelligent Systems, Applications: An International Journal archive Volume 442 Issue C, Pages 103-113, Elsevier. New York, NY, USA, 2018

[23] E. Lee, “Partisan Intuition Belies Strong, Institutional Consensus and Wide Zipf’s Law for Voting Blocs in US Supreme Court”, Proceedings of Journal of Statistical Physics, Springer US, 2018.

[24] D. Vu, N. Dao, and S. Cho, “Downlink sum-rate optimization leveraging Hungarian method in fog radio access networks”, Proceedings of International Conference on Information Networking (ICOIN), IEEE, Thailand, 2018.

[25] R. Shamir, Y. Duchin, J. Kim, G. Sapiro, and N. Harel, “Continuous Dice Coefficient: a Method for Evaluating Probabilistic Segmentations”, In Proceedings of Radiotherapy and OncologyVolume 127, Barcelona, Spain, Elsevier, 2018.

[26] https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection, Accessed: 2-nov-2019.

[27] http://qwone.com/~jason/20Newsgroups, Accessed: 2-nov-2019.