ClusTop: A Clustering-based Topic Modelling Algorithm for Twitter using Word Networks

Conference Paper · December 2017

DOI: 10.1109/BigData.2017.8258147

3 authors, including:

Kwan Hui Lim
University of Melbourne
46 PUBLICATIONS **448** CITATIONS

SEE PROFILE

ClusTop: A Clustering-based Topic Modelling Algorithm for Twitter using Word Networks

Kwan Hui Lim, Shanika Karunasekera, and Aaron Harwood
School of Computing and Information Systems
The University of Melbourne
Parkville, Victoria, Australia
Email: {kwan.lim, karus, aharwood}@unimelb.edu.au

***Abstract*—Twitter is a popular microblogging service, where users frequently engage in discussions about various topics of interest, ranging from popular topics (e.g., music) to niche topics (e.g., politics). With the large amount of tweets, a key challenge is to automatically model and determine the discussion topics without having prior knowledge of the types and number of topics, or requiring the technical expertise to define various algorithmic parameters. For this purpose, we propose the Clustering-based Topic Modelling (ClusTop) algorithm that constructs various types of word network and automatically determines the discussion topics using community detection approaches. Unlike traditional topic models, ClusTop is able to automatically determine the appropriate number of topics and does not require numerous parameters to be set. The ClusTop algorithm is also able to capture the syntactic meaning in tweets via the use of bigrams, trigrams and other word combinations in constructing the word network graph. Using three Twitter datasets with labelled crises and events as topics, ClusTop has been shown to outperform various baselines in terms of topic coherence, pointwise mutual information, precision, recall and F-score.**

***Index Terms*—Topic Modelling; Clustering; Twitter; Microblogs**

I. Introduction

Twitter is a popular microblogging service that has seen wide-spread usage in everyday life, as evidenced by the high daily volume of 500 million tweets posted [1]. On Twitter and similar microblogging services, users frequently perform discussions and debates on topics of interest, ranging from popular and general topics (e.g., music, TV, movies and entertainment) to specialized and niche topics (e.g., politics, special events, crises and incidents). Detecting and knowing the discussion about such topics serve numerous useful purposes, such as understanding general sentiments and trends, and providing accurate and relevant content recommendations. However, the large volume of tweets and high intensity of generated content create a challenge for users to understand the discussion topics in these tweets [2], [3].

One solution is to employ the use of topic modelling algorithms to automatically determine the topics discussed in a set of traditional documents (e.g., news articles, academic papers, etc), where topics are typically represented by a set of keywords. Examples of such algorithms are the original Latent Semantic Analysis [4], Probabilistic Latent Semantic Analysis [5] and Latent Dirichlet Allocation [6]. With the advent of microblogging services and hence short documents (i.e., tweets), variants of these traditional topic models have been proposed, using various aggregation scheme to combine multiple tweets into larger documents [7], [8]. While Latent Dirichlet Allocation and its variant has been shown to model topics well, the number of topics must be defined in advance and they do not account for the syntactic structure of sentences.

In this work, we aim to overcome these limitations by introducing a topic modelling algorithm that is able automatically determine the appropriate number of topics, via the adaptation of community detection algorithms on a network graph where vertices are words and edges are relations between words. Our algorithm is also able to capture the syntactic nature of language via the use of bigrams, trigrams and other word combinations in constructing our word network graph. In addition, we perform an empirical study on the effects of different types of network graphs on the accuracy and quality of topics modelled.

A. Main Contributions

In this paper, our main contributions are as follows:

1) We propose the Clustering-based Topic Modelling (ClusTop) algorithm that utilizes community detection approaches to model topics on Twitter based on a word network graph. Unlike other topic models, ClusTop automatically determines the number of topics by maximizing a modularity score among words in the network.
2) In addition to using a traditional co-word usage network, we experiment with different variants of our ClusTop algorithm based on numerous definitions of words and relations (unigrams, bigrams, trigrams, hashtags, nouns from part-of-speech tagging), and different aggregation schemes (individual tweets, hashtags and mentions).
3) Using three Twitter dataset with labelled topics, we evaluate ClusTop and its variants against various LDA baselines based on measures of topic coherence, pointwise mutual information, precision, recall and F-score. Experimental results show that ClusTop offers superior performance based on these evaluation metrics, compared to the various baselines.

B. Structure and Organization

The rest of this paper is structured as follows. Section II discusses key literature on studying topics on microblogs and topic modelling algorithms. Section III describes our CLUSTOP algorithm. Section IV outlines our experimental methodology in terms of the dataset used, baseline algorithms and evaluation metrics. Section V highlights the results from our evaluation and discusses our main findings. Section VI concludes this paper and highlights possible future directions for this work.

II. RELATED WORK

In this section, we review two lines of research related to our work, namely the study of topics on microblogs and topic modelling algorithms.

A. Studying Topics on Microblogs

In the context of understanding research themes in the Human Computer Interaction domain, [9] used hierarchical clustering on co-keywords usage in papers to identify the main research clusters in two different time period. Researchers have also proposed approaches for identifying communities that frequently talk about common interest topics, using community detection algorithms alongside topological links to celebrities [10] or interaction links via mentions [11]. Researchers like [12] and [13] have also used community detection algorithms on word networks to identify topics with a focus on network analysis and visualization, and detection of spammer topics, respectively. Fried et al. [14] used topic modelling on a series of food-related tweets to understand health information such as overweight rate and diabetes rate. Apart from studying topics on microblogs, topic models have also been used to enhance other task such as distinguishing between personal and corporate accounts [15] and identifying fake follower accounts [16].

B. Topic Modelling Algorithm

Latent Dirichlet Allocation (LDA) [6] is a popular topic model that is used to determine the set of latent topics associated with a set of documents. In LDA, each document is represented as a bag-of-words, each topic as a distribution of words, and each document is assigned a distribution of topics via a generative process. LDA has been applied to Twitter where each tweet is considered a document, and researchers have used aggregation schemes where tweets by the same author or with the same terms, hashtags, posted time are combined as one document [7], [8], while others [17] performed topic modelling based on the posted time and place of tweets. Zhao et al. have also used LDA to study the differences between Twitter and New York Times in terms of the discussed topics and content [18], while Aiello et al. [19] applied LDA for the purpose of trending topics detection in sports and politics, using different textual pre-processing steps. Similarly, researchers have modified LDA to capture the temporal nature of documents, such as the Topic over Time (TOT) algorithm [20] for detecting topical trends over continuous time, and Temporal-LDA [21] for modelling topics and their transitions in streaming documents.

C. Discussion

These earlier works provided interesting insights into the application of topic models on microblogs and they proposed various novel topic modelling algorithms with good performance. However, our research differ from these earlier works in the following ways:

1) While there are researchers that study topics on microblogs, these works focus on the application of topic modelling algorithms on microblogs to understand topical trends in the microblogging community from a social perspective. These works focus less on classifying individual tweets into specific topics and conduct limited or no performance evaluation on these algorithms.
2) In contrast to works that employ community detection algorithms to understand topics, we perform an empirical study based on an extensive range of network types (with multiple definitions of vertices and edges), instead of using only word co-occurence. In addition, we also focus on validating the performance of our proposed algorithm on a set of labelled tweets, instead of only understanding the broad topical trends.
3) Although existing topic models are shown to perform well, they are dependent on setting the appropriate values for various algorithmic parameters, such as the number of topics to model. In contrast, our CLUSTOP algorithm automatically determines the number of topics and does not require any parameter to be set, due to its local maximization of modularity.

III. PROPOSED ALGORITHM

We first introduce some basic notations and preliminaries used in this paper. In traditional network theory, V and E denote the set of vertices and edges, respectively, and an undirected graph $G = (V, E)$ is represented as a collection of vertices V that are connected by a set of edges E. In turn, each edge $e \in E$ is denoted by $e = (\{v_i, v_j\}, w)$, which represents a link between vertices v_i and v_j with a weight w. For our application of community detection algorithms to topic modelling, we denote an undirected graph as $G = (U, R)$, where U is the set of unigrams (vertices) and R is the set of relations (edges) between the unigrams.

In this work, we propose the Clustering-based Topic Modelling (CLUSTOP) algorithm that uses community detection approaches to topic modelling, based on the undirected graph $G = (U, R)$ and different definitions of unigrams and relations. Our basic CLUSTOP algorithm comprises the following steps:

1) **Network Construction**. First, we build a unigram network, i.e., an undirected graph $G = (U, R)$, based on a particular definition of unigrams and relations. This step will be elaborated further in Section III-A, where we will describe the various types of unigrams and relations modelled in this work.

2) **Community Detection**. Using the network graph constructed in Step 1, we then apply community detection approaches to identify the main communities (topics), which we further describe in Section III-B.

3) **Topic Assignment**. Each detected community from Step 2 corresponds to a topic and this step aims to determine which community a tweet belongs to, i.e., assign a topic to a tweet. Refer to Section III-C for more details.

A. Network Construction

In this section, we will describe the step of network graph construction, which in turn depends on: (i) the type of network based on different definitions of unigrams (vertices) and their relations (edges); and (ii) the type of document aggregation, i.e., individual tweets, aggregated by hashtag or mentions, for constructing the network graph.

1) Types of Network: The first stage of our algorithm involves constructing a network graph of word usage, as shown in Algorithm 1. This algorithm involves the following: (i) examining all tweets and tokenizing all words in each tweet based on whitespaces; (ii) for each word-pair in each tweet, build a weighted edge e linking the two words; and (iii) repeating Steps 1 and 2 for all tweets, until we obtain a network graph, where the vertices represent uni-grams and edges represent a relation between two unigrams. We experiment with various forms of relations between different types of unigram, including the following:

- **Co-word Usage (WORD)**. A relationship where two words (uni-grams) is used in the same tweet. That is, co-word usage models all pair-wise word co-occurence in a tweet, regardless of where the word appeared.
- **Co-hashtag Usage (HASH)**. A relationship where two hashtags is used in the same tweet. Twitter users typically use hashtags to categorize their tweets into themes and topics [22], [23], and thus serve as a suitable form of unigram relation.
- **Co-noun Usage (NOUN)**. A relationship where two nouns is used in the same tweet. For determining the noun in a tweet, we utilize the part-of-speech tagging component from Apache OpenNLP library [24], which has been used by many researchers for similar natural language processing [25], [26], [27].
- **Bigram occurrence (BIG)**. A relationship for two words of each bigram in the tweet. Unlike the co-word usage, this bigram occurence only considers a relation/edge between two words if they are used one after another in sequence.
- **Trigram occurrence (TRIG)**. Similar to the earlier bigram occurrence, except that we model a relationship between three words in a trigram, i.e., there is an additional edge between the first and third word.
- **Bigram + Hashtag (BIHA)**. A combination of bigram occurrence and co-hashtag usage, we consider each bigram occurrence and add a relation/edge between each word of a bigram and all hashtags in the tweet.

Algorithm 1: Network Construction

input : T: Collection of tweets in corpus.
output: $G = (U, R)$: Network graph of unigrams (vertices) and relations (edges).

```
begin
    Initialize an empty graph G;
    for each tweet t ∈ T do
        for each word-pair (p1, p2) ∈ t do
            e ← ({p1, p2}, 1);
            if edge e exists in graph G then
                increment edge e in graph G by 1;
            else
                add edge e to graph G;
    Return G;
```

2) Types of Document Aggregation: In the above examples, we are treating each tweet as a single document for topic modelling purposes. For traditional topic modelling purposes, each document usually corresponds to a lengthy piece of text (such as a news article, website or abstract) but on Twitter, each document typically corresponds to a short tweet of 140 characters. Researchers have found that aggregating multiple tweets into a single document improves the performance of LDA on Twitter [7], [8]. As such, we also experiment with different forms of document aggregation scheme for our CLUSTOP algorithm, including:

- **No Aggregation, i.e., individual tweets (NA)**. The basic representation where each tweet is considered a single document, i.e., no aggregation as per traditional topic modelling.
- **Aggregate by Hashtags (AH)**. Each document comprises a set of tweets that are aggregated based on common hashtags used.
- **Aggregate by Mentions (AM)**. Each document comprises a set of tweets that are aggregated based on common mentions of Twitter users.

B. Community Detection

After constructing the network graph in the previous section, we now describe our approach to modelling the topics in this graph using community detection approaches. We mainly focus on adapting the Louvain algorithm [28] for this purpose, as the Louvain algorithm was shown to be one of the best performing algorithm in a comprehensive survey of community

detection algorithms [29].[1]

For this step, we apply the Louvain algorithm [28] for our purpose of topic modelling, which is described by the pseudo-code in Algorithm 2. In this algorithm, there are the following steps:

1) Initially, each unigram is placed in their own community/cluster (Line 2).
2) Following which, for each unigram, we examine each neighbour of this unigram and combine two unigrams into the same community/cluster if their modularity gain is the greatest among all of the neighbours (Lines 4 to 16).
3) Next, we build a new network graph where unigrams in the same community/cluster are combined as a single vertex (unigram), and Step 2 is repeated until the modularity score is maximized (Lines 17 to 20).

One of the reason for the Louvain algorithm's good performance is due to its local adjustment of unigrams (vertices) into communities/clusters, by maximizing the gain in the following modularity function [28]:

$$Q = \left[\frac{\sum_{in}+k_{i,in}}{2m} - \left(\frac{\sum_{tot}+k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right] \quad (1)$$

where $\sum_{in}$ and $\sum_{tot}$ represents the total weight of all links inside a community/cluster and total weight of all links to a community/cluster, respectively. Similarly, the terms k_i and $k_{i,in}$ denote the total weight of all links to i and total weight of links to i within the community/cluster. Lastly, m denotes the total weight of all links in the network graph.

At the end of this step, we will obtain a set of communities/clusters based on the provided network graph. Each community/cluster will represent a particular topic, where the members of each community/cluster serve as the representative words of each topic. For each topic, we also rank the keywords (i.e., members of each community) based on the total weight of all links to a unigram/vertex.

Algorithm 2: Topic Modelling using Louvain

input : $G = (U, R)$: Network graph of unigrams (vertices) and relations (edges).
output: $A = (U, C)$: Assignment of unigrams (vertices) U into communities C.

1 **begin**
2 Assign all unigrams u into their own community;
3 **repeat**
4 **for** each unigram $u \in U$ **do**
5 $MaxModularity \leftarrow -1$;
6 $MaxModNeighbour \leftarrow NULL$;
7 **for** each neighbour u_n of unigram u **do**
8 $ShiftMod \leftarrow$ Modularity score of shifting unigram u to neighbour u_n's community;
9 **if** $ShiftMod > MaxModularity$ **then**
10 $MaxModularity \leftarrow ShiftMod$;
11 $MaxModNeighbour \leftarrow u_n$;
12 $OriginalMod \leftarrow$ Modularity score of unigram u in its original community;
13 **if** $OriginalMod > MaxModularity$ **then**
14 Shift unigram u to the community of MaxModNeighbour;
15 **else**
16 Keep unigram u in its original community;
17 $U_n \leftarrow$ New unigrams (vertices) U_n based on the newly-formed communities;
18 $R_n \leftarrow$ New relations (edges) R_n based on edge weights between nodes in two communities;
19 $G_n \leftarrow$ New network graph $G_n = (U_n, R_n)$;
20 The algorithm iterates again (Lines 4 to 19) with network graph G_n as input;
21 **until** *Community structure stabilizes and modularity score is maximized*;
22 Return a_n;

C. Topic Assignment

Given the detected communities/topics C from Section III-B and a tweet $t = \{u_1, ...u_n\}$, we define the most likely topic for this tweet as:

$$\arg\max_{c \in C} \sum_{u \in c} k_u \delta(u = u_t), \qquad \forall\ u_t \in t \quad (2)$$

where $\delta(u = u_t) = 1$ if an unigram u of a community/topic $c \in C$ is the same as an unigram u_t of a tweet t and $\delta(p = c) = 0$ otherwise, and k_u denotes the total weight of links to unigram u (as previously described in Section III-B).

In short, we assign a tweet t to a community/topic c that has the highest co-occurrence of unigrams in both the tweet and community/topic, where the unigram in the community/topic is weighted based on its co-occurrence to other unigrams.

IV. Dataset and Evaluation Methodology

In this section, we give an overview of our experimental dataset and describe our evaluation methodology in terms of the CLUSTOP algorithm variants, baseline algorithms and evaluation metrics.

[1]Based on the generated network graph, our approach can also be easily generalized to use other community detection algorithms but we utilize the Louvain algorithm due to its good performance. We have also experimented with other algorithms such as the Infomap [30] and Label Propagation [31] algorithms but these algorithms have a tendency to generate a large number (hundreds to thousands) of small communities, thus making it unfeasible for our topic modelling purpose.

A. Dataset

For our experimental evaluation, we utilize three Twitter datasets with labelled topics [32], [33], [34], which enables us to better evaluate our algorithm and baselines against the ground truth topics compared to an unlabelled dataset. In total, these datasets comprise close to 8 million tweets, from which we focus on the subset of tweets with annotated and verified topics. These topics are in the form of 60k labelled tweets about 6 crisis [32], 27.9k labelled tweets about 26 crisis [33], and 3.6k labelled tweets about 8 events [34]. Refer to Table I for more details. The annotation of these tweets into the respective topics (crises and events) were performed via the CrowdFlower crowdsourcing platform, and more details can be found in the respective papers.

TABLE I
DESCRIPTION OF DATASET

Dataset	Paper Reference	Number of Topics	Total Tweets	Labelled Tweets
A	[32]	6	7.67mil	60k
B	[33]	26	0.28mil	27.9k
C	[34]	8	4.8k	3.6k

For each of the dataset, we split them into four partitions and perform a 4-fold cross validation [35]. At each evaluation iteration, we use three partitions as our training set and the last partition as our testing set. After completing all evaluations, we compute and report the mean results for each algorithm based on the metrics of topic coherence, pointwise mutual information, precision, recall and f-score, which we elaborate further in the rest of the paper.

1) Topic Quality Metrics: For determining the quality of the detected topics, we measure the topic quality based on the topic coherence and pointwise mutual information metrics. These two metrics have also been widely used in many topic modelling research [36], [37], [8]. For both evaluation metrics, we denote a detected topic t that comprises a set of n representative unigrams/keywords $U^{(t)} = (u_1^{(t)}, ..., u_n^{(t)})$ for each topic.

1) **Topic Coherence (TC)**. Given that $D(u_i, u_j)$ denotes the number of times both unigrams u_i and u_j appeared in the same document/tweet, and similarly, $D(u_i)$ for a single unigram u_i, topic coherence is defined as:

$$TC(t, U^{(t)}) = \sum_{u_i \in U^{(t)}} \sum_{u_j \in U^{(t)}, u_i \neq u_j} log \ \frac{D(u_i, u_j)}{D(u_j)} \quad (3)$$

2) **Pointwise Mutual Information (PMI)**. Given that $P(u_i, u_j)$ denotes the probability of a unigram pair u_i and u_j appearing in the same document/tweet, and $P(u_i)$ for the probability of a single unigram u_i, pointwise mutual information is defined as:

$$PMI(t, U^{(t)}) = \sum_{u_i \in U^{(t)}} \sum_{u_j \in U^{(t)}, u_i \neq u_j} log \ \frac{P(u_i, u_j)}{P(u_i)P(u_j)} \quad (4)$$

In both the TC and PMI metrics, it is possible for a division by 0 or taking the log of 0 when the appropriate numerator or denominator are 0, i.e., when a particular word or word pair has not been previously observed. As such, we adopt a similar strategy as [36], [8] by adding a small value $\epsilon = 1$ to both components to avoid the situation of a division by 0 or log of 0.

2) Topic Relevance Metrics: Precision, recall and f-score are popular metrics used in Information Retrieval and other related fields, such as in topic modelling [19], [38], tour recommendation [39], [40], location prediction and tagging [41], [42], among others. In contrast to the previous topic quality metrics (TC and PMI), these metrics allow us to evaluate how relevant and accurate the detected topics are, compared to the ground truth topics. In topic modelling, researchers typically manually curate a set of ground truth keywords to describe a specific topic, then evaluate how well the detected keywords from their topic models match these ground truth keywords [19]. For our evaluation, we adopt a similar methodology except that we automatically determine the ground truth keywords from the respective Wikipedia article for each topic.

Given that $U^D = (u_1^D, ..., u_n^D)$ and $U^G = (u_1^G, ..., u_n^G)$ denotes the set of detected unigrams and ground truth unigrams for a specific topic, the metrics we use are as follows:

- **Precision**. The proportion of unigrams for the detected topic U^D that also appears in the ground truth unigrams U^G. For a topic t, precision is defined as:

$$P(t) = \frac{|U^D \cap U^G|}{|U^D|} \quad (5)$$

- **Recall**. The proportion of ground truth unigrams U^G that also appears in the unigrams for the detected topic U^D. For a topic t, recall is defined as:

$$R(t) = \frac{|U^D \cap U^G|}{|U^G|} \quad (6)$$

- **F-score**. The harmonic mean of precision $P(t)$ and recall $R(t)$, which was introduced in Equations 5 and 6, respectively. For a topic t, F-score is defined as:

$$F(t) = \frac{2 \times P(t) \times R(t)}{P(t) + R(t)} \quad (7)$$

In our experiments, we compute the precision, recall and F-score derived from the testing set, in terms of the top 5 and 10 keywords of each topic modelled.

TABLE II
COMPARISON OF CLUSTOP ALGORITHM AGAINST VARIOUS BASELINES, IN TERMS OF TOPIC COHERENCE (TC) AND POINTWISE MUTUAL INFORMATION (PMI). THE TOP THREE SCORES FOR EACH METRIC ARE BOLDED AND HIGHLIGHTED IN BLUE.

	Top 5 Keywords / Unigrams						**Top 10 Keywords / Unigrams**					
	Dataset A		**Dataset B**		**Dataset C**		**Dataset A**		**Dataset B**		**Dataset C**	
Algorithm	TC	PMI	TC	PMI	TC	PMI	TC	PMI	TC	PMI	TC	PMI
ClusTop-Word-NA	-37.61	-5.55	-34.05	-7.70	-37.93	-14.39	-171.00	-49.17	-160.82	-39.50	-173.41	-67.55
ClusTop-BiG-NA	-36.56	7.27	-35.91	1.25	-42.50	-16.39	-153.40	-29.60	-158.16	-25.83	-194.84	-63.40
ClusTop-TriG-NA	-30.85	**10.75**	-35.75	-2.56	-41.98	-18.16	-122.63	-16.09	-166.46	-25.10	-194.20	-73.49
ClusTop-BiHa-NA	-23.27	**19.60**	-32.29	4.70	-37.87	-11.19	-81.38	**7.14**	-140.81	-14.93	-169.93	-50.68
ClusTop-Hash-NA	**-7.14**	5.77	**-14.80**	0.35	**-14.11**	2.63	**-19.40**	2.27	**-54.93**	-6.91	**-47.82**	**4.44**
ClusTop-Noun-NA	-17.06	10.60	-21.42	**6.94**	-22.79	-0.32	-64.68	**2.86**	-90.50	**-3.55**	-97.77	-14.06
ClusTop-Word-AH	-30.93	-1.62	-40.23	-27.56	-24.21	**10.26**	-137.65	-57.69	-198.33	-131.14	-88.55	**9.07**
ClusTop-Noun-AH	-28.68	-11.38	-41.81	-19.85	**-17.64**	**4.58**	-132.37	-72.25	-185.55	-102.49	**-63.76**	-2.51
ClusTop-Hash-AH	**-6.27**	5.36	**-12.84**	0.94	**-13.11**	2.64	**-16.22**	1.54	**-47.35**	-7.31	**-43.68**	2.23
ClusTop-Word-AM	-34.36	8.31	-30.91	**11.02**	-37.69	-14.34	-146.09	-5.42	-126.84	**8.06**	-179.93	-69.00
ClusTop-Hash-AM	-19.71	**11.36**	**-18.53**	**16.25**	-33.71	-7.25	-73.94	**8.47**	**-52.93**	**14.30**	-153.65	-30.35
ClusTop-Noun-AM	**-11.15**	4.76	-19.21	0.30	-22.55	**4.02**	**-29.92**	-0.30	-70.13	-9.04	-70.27	**11.87**
LDA-Orig	-74.68	-74.42	-66.87	-62.16	-54.17	-43.07	-323.47	-307.51	-297.11	-269.17	-251.33	-191.86
LDA-Hash	-51.20	-43.84	-55.14	-42.94	-41.46	-23.44	-247.09	-185.40	-256.85	-199.15	-206.60	-112.48
LDA-Ment	-52.84	-45.85	-54.09	-45.31	-47.35	-27.74	-250.24	-198.83	-258.59	-206.48	-225.52	-136.37

B. Variants of ClusTop Algorithm

Based on the six types of unigram network and three types of document aggregation (introduced in Section III-A), there can be mutliple variants of our CLUSTOP algorithm. For our evaluation, we experiment with the following 12 variants of our CLUSTOP algorithm, namely:

- **ClusTop-Word-NA**. CLUSTOP based on a co-word usage network, with no tweet aggregation.
- **ClusTop-BiG-NA**. CLUSTOP based on a bigram occurence network, with no tweet aggregation.
- **ClusTop-TriG-NA**. CLUSTOP based on a trigram occurence network, with no tweet aggregation.
- **ClusTop-BiHa-NA**. CLUSTOP based on a bigram occurence + co-hashtag usage network, with no tweet aggregation.
- **ClusTop-Hash-NA**. CLUSTOP based on a co-hashtag usage network, with no tweet aggregation.
- **ClusTop-Noun-NA**. CLUSTOP based on a co-noun usage network, with no tweet aggregation.
- **ClusTop-Word-AH**. CLUSTOP based on a co-word usage network, with tweets aggregated based on common hashtags.
- **ClusTop-Hash-AH**. CLUSTOP based on a co-hashtag usage network, with tweets aggregated based on common hashtags.
- **ClusTop-Noun-AH**. CLUSTOP based on a co-noun usage network, with tweets aggregated based on common hashtags.
- **ClusTop-Word-AM**. CLUSTOP based on a co-word usage network, with tweets aggregated based on common mentions.
- **ClusTop-Hash-AM**. CLUSTOP based on a co-hashtag usage network, with tweets aggregated based on common mentions.
- **ClusTop-Noun-AM**. CLUSTOP based on a co-noun usage network, with tweets aggregated based on common mentions.

Note that we did not use the CLUSTOP variants based on bigrams and trigrams combined with the hashtag and mention aggregation schemes, as these variants provide minimal improvements compared to their original non-aggregated variants. Consider a simple example of three tweets with a common hashtag, the hashtag aggregation scheme with bigrams will only produce an additional two bigrams resulting from the first and second tweet as well as the second and

TABLE III

COMPARISON OF CLUSTOP ALGORITHM AGAINST VARIOUS BASELINES, IN TERMS OF PRECISION (PRE), RECALL (REC) AND F-SCORE (FS) FOR THE TOP 5 KEYWORDS/UNIGRAMS OF EACH TOPIC. THE TOP THREE SCORES FOR EACH METRIC ARE BOLDED AND HIGHLIGHTED IN BLUE (IN THE EVENT OF A TIED RESULT, ALL TIED SCORES ARE HIGHLIGHTED).

	Top 5 Keywords / Unigrams								
	Dataset A			**Dataset B**			**Dataset C**		
Algorithm	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-score
ClusTop-Word-NA	.754±.0018	.031±.0001	.059±.0002	.866±.0031	.043±.0003	.082±.0005	.840±.0093	.027±.0005	.052±.0009
ClusTop-BiG-NA	.786±.0019	**.034±.0001**	**.064±.0002**	.857±.0033	**.046±.0004**	**.086±.0006**	.833±.0099	**.029±.0006**	**.056±.0011**
ClusTop-TriG-NA	.791±.0018	**.034±.0001**	**.064±.0002**	.871±.0031	**.046±.0003**	**.087±.0006**	.822±.0093	**.031±.0006**	**.058±.0011**
ClusTop-BiHa-NA	.784±.0018	.032±.0001	.060±.0002	.886±.0030	.045±.0003	.084±.0006	.820±.0094	**.029±.0005**	.055±.0009
ClusTop-Hash-NA	**.898±.0039**	.023±.0001	.044±.0003	.916±.0075	.032±.0005	.062±.0010	**.936±.0106**	.022±.0004	.042±.0008
ClusTop-Noun-NA	.761±.0019	.028±.0001	.054±.0002	.836±.0032	.043±.0003	.081±.0006	.888±.0081	**.032±.0007**	**.062±.0013**
ClusTop-Word-AH	.741±.0025	.025±.0001	.049±.0001	.845±.0048	.040±.0004	.075±.0007	.844±.0102	.027±.0005	.052±.0009
ClusTop-Noun-AH	.802±.0024	.024±.0001	.046±.0001	.873±.0035	.037±.0003	.071±.0006	.872±.0084	**.029±.0005**	**.056±.0010**
ClusTop-Hash-AH	**.847±.0019**	.026±.0001	.051±.0001	.912±.0038	**.046±.0005**	**.086±.0008**	.896±.0078	.024±.0004	.046±.0007
ClusTop-Word-AM	.748±.0014	**.034±.0001**	**.065±.0002**	**.929±.0030**	.036±.0003	.069±.0006	.758±.0159	.022±.0003	.043±.0005
ClusTop-Hash-AM	.763±.0015	.027±.0001	.052±.0002	.917±.0033	.037±.0003	.072±.0006	.869±.0111	.024±.0004	.047±.0008
ClusTop-Noun-AM	.842±.0018	.025±.0000	.048±.0001	**.950±.0033**	.039±.0004	.074±.0006	**.923±.0085**	.022±.0004	.043±.0008
LDA-Orig	**.925±.0014**	.027±.0001	.052±.0002	**.956±.0022**	.037±.0003	.070±.0006	**.898±.0097**	.025±.0004	.049±.0008
LDA-Hash	.821±.0016	.031±.0001	.059±.0002	.916±.0031	.036±.0003	.069±.0005	.837±.0098	.028±.0005	.054±.0010
LDA-Ment	.830±.0016	.031±.0001	.060±.0002	.900±.0031	.039±.0003	.074±.0005	.814±.0179	.023±.0004	.044±.0007

third tweet. Moreover, these two additional bigrams will be generated from the last word of the first tweet and first word of the second tweet, which will not be syntactically meaningful in most cases.

C. Baseline Algorithms

LDA is a popular topic modelling algorithm that was used for traditional documents (such as news articles), and more recently for social media (such as tweets on Twitter). Given the popularity of LDA for topic modelling, we compare our CLUSTOP algorithm and its variants against the following LDA-based algorithms, namely:

1) **LDA-Orig**. The original version of LDA introduced by [6], where each document corresponds to a single tweet.
2) **LDA-Hash**. A variant of LDA applied on Twitter, where each document is aggregated from multiple tweets with the same hashtag [8].
3) **LDA-Ment**. An adaptation of the Twitter-based LDA variant proposed by [43], where we aggregate tweets with the same mention into a single document.

V. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we report on the results of our experiments and discuss some implications of these findings.

A. Topic Coherence and Pointwise Mutual Information

Table II shows the performance of our CLUSTOP algorithm and its variants against the various LDA baselines, in terms of Topic Coherence and Pointwise Mutual Information, based on the top 5 and 10 keywords in the detected topics.

The results generally show that all variants of our CLUSTOP algorithm outperform the various LDA baselines, in terms of both evaluation metrics of Topic Coherence and Pointwise Mutual Information. In particular, we note the following:

- The performance of CLUSTOP could be largely attributed to its usage of the various types of word network graphs, which retain the syntactic meaning and association between words in a tweet via the use of bigrams, trigrams, word co-usage and its variants.
- All CLUSTOP variants that utilizes hashtags (CLUSTOP-HASH-NA, CLUSTOP-HASH-AH and CLUSTOP-HASH-AM) offer better overall performance compared to its counterparts that utilizes other forms of unigram and relation, i.e., words, bigrams, trigrams, nouns.

TABLE IV

COMPARISON OF CLUSTOP ALGORITHM AGAINST VARIOUS BASELINES, IN TERMS OF PRECISION (PRE), RECALL (REC) AND F-SCORE (FS) FOR THE TOP 10 KEYWORDS/UNIGRAMS OF EACH TOPIC. THE TOP THREE SCORES FOR EACH METRIC ARE BOLDED AND HIGHLIGHTED IN BLUE (IN THE EVENT OF A TIED RESULT, ALL TIED SCORES ARE HIGHLIGHTED).

	Top 10 Keywords / Unigrams								
	Dataset A			**Dataset B**			**Dataset C**		
Algorithm	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-score
ClusTop-Word-NA	.690±.0019	.033±.0001	.062±.0002	.804±.0033	**.051±.0004**	**.095±.0007**	.764±.0101	.033±.0007	.062±.0013
ClusTop-BiG-NA	.707±.0020	**.035±.0001**	**.065±.0002**	.789±.0036	**.052±.0004**	**.096±.0007**	.765±.0103	**.035±.0008**	**.067±.0015**
ClusTop-TriG-NA	.717±.0020	**.034±.0001**	**.064±.0002**	.811±.0034	**.053±.0004**	**.098±.0007**	.746±.0096	**.036±.0008**	**.068±.0014**
ClusTop-BiHa-NA	.719±.0019	**.034±.0001**	**.064±.0002**	.782±.0035	**.051±.0004**	**.095±.0007**	.765±.0096	**.035±.0008**	**.065±.0014**
ClusTop-Hash-NA	**.860±.0043**	.023±.0001	.045±.0003	**.896±.0076**	.032±.0005	.061±.0010	**.925±.0116**	.022±.0004	.043±.0008
ClusTop-Noun-NA	.736±.0019	.033±.0001	.062±.0002	.803±.0033	.049±.0004	.091±.0006	.802±.0093	**.035±.0008**	**.065±.0014**
ClusTop-Word-AH	.692±.0025	.025±.0001	.049±.0001	.784±.0045	.043±.0004	.080±.0008	.778±.0101	.030±.0005	.057±.0010
ClusTop-Noun-AH	.735±.0022	.024±.0001	.045±.0001	.827±.0036	.041±.0004	.076±.0007	.828±.0088	.031±.0005	.059±.0010
ClusTop-Hash-AH	.833±.0018	.027±.0001	.051±.0001	.842±.0039	.041±.0004	.078±.0007	**.857±.0088**	.025±.0004	.048±.0008
ClusTop-Word-AM	.734±.0014	**.036±.0001**	**.068±.0002**	.828±.0037	.039±.0003	.073±.0005	.709±.0130	.023±.0003	.044±.0006
ClusTop-Hash-AM	.731±.0015	.030±.0001	.058±.0002	.845±.0036	.043±.0004	.081±.0007	.823±.0107	.028±.0006	.053±.0010
ClusTop-Noun-AM	**.860±.0013**	.024±.0000	.047±.0001	**.932±.0031**	.040±.0004	.076±.0006	**.908±.0080**	.024±.0004	.047±.0008
LDA-Orig	**.848±.0016**	.029±.0001	.056±.0002	**.885±.0030**	.040±.0003	.076±.0006	.808±.0105	.026±.0005	.051±.0009
LDA-Hash	.759±.0018	**.034±.0001**	**.064±.0002**	.847±.0034	.041±.0003	.078±.0006	.778±.0098	.034±.0007	.064±.0012
LDA-Ment	.752±.0018	.033±.0001	.063±.0002	.820±.0035	.044±.0003	.082±.0005	.787±.0126	.024±.0004	.047±.0007

- The aggregation schemes employed by LDA (LDA-HASH and LDA-MENT) generally outperform its original counterpart (LDA-ORIG), thus showing that LDA works better on larger documents.
- In addition to all CLUSTOP variants outperforming the LDA baselines, the aggregation schemes employed by CLUSTOP showed better performance compared to its non-aggregated counterparts.

B. Precision, Recall and F-score

Table III shows the Precision, Recall and F-score of our CLUSTOP algorithm and variants, and the various LDA baselines based on the top 5 keywords of detected topics, while Table IV shows the same results based on top 10 keywords of detected topics.

In terms of Precision, Recall and F-score, there are specific variants of CLUSTOP that outperform the LDA baselines for each of the evaluation metric. Our main observations are as follows:

- The top three algorithms with the highest Recall and F-score are all variants of our CLUSTOP algorithms[2], in particular the CLUSTOP-BIG-NA and CLUSTOP-TRIG-NA variants.
- Based on the Precision metric, the overall best performers are CLUSTOP-NOUN-AM, CLUSTOP-HASH-NA and LDA-ORIG based on their frequency of appearance as the top three algorithms.

[2]Except for one instance where LDA-HASH is tied with CLUSTOP-TRIG-NA and CLUSTOP-BIHA-NA at third place for Dataset A.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed the CLUSTOP algorithm for topic modelling on Twitter, using community detection approaches on a network graph with multiple definitions of vertices and edges. Unlike traditional topic modelling algorithms, CLUSTOP does not require the setting of numerous parameters and is able to automatically determine the appropriate number of topics based on a local maximization of modularity among the word network graph. We also performed an empirical study on the effects of using different types of vertices and edges (unigrams, bigrams, trigrams, hashtags, nouns from part-of-speech tagging, and different aggregation schemes (individual tweets, hashtags and mentions). This empirical study resulted in different variants of our CLUSTOP algorithm, which we use to compare against various LDA baselines based on the evaluation metrics of topic coherence, pointwise mutual information, precision, recall and F-score. Based on three Twitter datasets

with labeled topics (crises and events), the experimental results show that our CLUSTOP algorithm out-performs the various LDA baselines in terms of these evaluation metrics.

This work explored the use of community detection approaches for automated topic modelling on Twitter and discussed the implications of different types of network graphs via an empirical study. There still remain various directions for future research, which include:

- One key challenge in evaluating topic models is to obtain a dataset with annotated labels of the ground truth topics. Future work can automate the labelling of this ground truth topic by using the semantic similarity between tweets and Wikipedia or news articles to determine appropriate topic labels.
- This work focused on using community detection approaches for topic modelling purposes. Future work can utilize a joint modelling of social relations between users and the various types of word network graph to detect topic-coherence communities, i.e., communities of users based on topical interests.
- Another future direction is to extend our CLUSTOP algorithm to incorporate temporal and spatial attributes associated with geo-tagged tweets. This extension will allow us to model topics that are associated with specific time periods or physical locations.

VII. ACKNOWLEDGMENTS

This research is supported by the Defence Science and Technology Group. The authors thank Lucia Falzon for the preliminary discussions, and the anonymous reviewers for their useful comments.

REFERENCES

[1] Internet Live Statistics, "Twitter usage statistics," Internet, 2016, http://www.internetlivestats.com/twitter-statistics/.

[2] S. Kumar, F. Morstatter, and H. Liu, *Twitter Data Analytics*. New York, NY, USA: Springer, 2013.

[3] Y. Liao, M. Moshtaghi, B. Han, S. Karunasekera, R. Kotagiri, T. Baldwin, A. Harwood, and P. Pattison, *Social Networks: Computational Aspects and Mining*. Springer-Verlag, ch. Mining Micro-Blogs: Opportunities and Challenges.

[4] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *Journal of the American Society for Information Science*, vol. 41, no. 6, p. 391, 1990.

[5] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*, 1999, pp. 289–296.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[7] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the First Workshop on Social Media Analytics (SMA'10)*, 2010, pp. 80–88.

[8] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie, "Improving LDA topic models for microblogs via tweet pooling and automatic labeling," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*, 2013, pp. 889–892.

[9] Y. Liu, J. Goncalves, D. Ferreira, B. Xiao, S. Hosio, and V. Kostakos, "CHI 1994-2013: mapping two decades of intellectual progress through co-word analysis," in *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI'14)*, 2014, pp. 3553–3562.

[10] K. H. Lim and A. Datta, "A topological approach for detecting twitter communities with common interests," in *Ubiquitous Social Media Analysis*. Springer Berlin Heidelberg, 2013, pp. 23–43.

[11] K. H. Lim and A. Datta, "An interaction-based approach to detecting highly interactive twitter communities using tweeting links," *Web Intelligence*, vol. 14, no. 1, pp. 1–15, 2016.

[12] D. Paranyushkin, "Identifying the pathways for meaning circulation using text network analysis," in *Nodus Labs*, 2011.

[13] S. B. Jr, G. S. Kido, and G. M. Tavares, "Artificial and natural topic detection in online social networks," *iSys - Revista Brasileira de Sistemas de Informacao*, vol. 10, no. 1, pp. 80–98, 2017.

[14] D. Fried, M. Surdeanu, S. Kobourov, M. Hingle, and D. Bell, "Analyzing the language of food on social media," in *Proceedings of the 2014 IEEE International Conference on Big Data (BigData'14)*, 2014, pp. 778–783.

[15] P. Yin, N. Ram, W.-C. Lee, C. Tucker, S. Khandelwal, and M. Salathe, "Two sides of a coin: Separating personal communication and public dissemination accounts in twitter," in *Proceedings of the 18th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'14)*, 2014, pp. 163–175.

[16] Y. Shen, J. Yu, K. Dong, and K. Nan, "Automatic fake followers detection in chinese micro-blogging system," in *Proceedings of the 18th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'14)*, 2014, pp. 596–607.

[17] K. H. Lim, S. Karunasekera, A. Harwood, and L. Falzon, "Spatial-based topic modelling using wikidata knowledge base," in *Proceedings of the 2017 IEEE International Conference on Big Data (BigData'17)*, 2017.

[18] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li, "Comparing twitter and traditional media using topic models," in *Proceedings of the 33rd European Conference on Information Retrieval (ECIR'11)*, 2011, pp. 338–349.

[19] L. M. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, A. Göker, I. Kompatsiaris, and A. Jaimes, "Sensing trending topics in twitter," *IEEE Transactions on Multimedia*, vol. 15, no. 6, pp. 1268–1282, 2013.

[20] X. Wang and A. McCallum, "Topics over time: A non-markov continuous-time model of topical trends," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'06)*, 2006, pp. 424–433.

[21] Y. Wang, E. Agichtein, and M. Benzi, "Tm-lda: Efficient online modeling of latent topic transitions in social media," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12)*, 2012, pp. 123–131.

[22] Z. Ma, A. Sun, and G. Cong, "Will this #hashtag be popular tomorrow?" in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*, 2012, pp. 1173–1174.

[23] J. Lehmann, B. Goncalves, J. J. Ramasco, and C. Cattuto, "Dynamical classes of collective attention in twitter," in *Proceedings of the 21st International Conference on World Wide Web (WWW'12)*, 2012, pp. 251–260.

[24] The Apache Software Foundation, "The Apache OpenNLP library," Internet, 2017, http://opennlp.apache.org.

[25] C. A. Mattmann and M. Sharan, "An automatic approach for discovering and geocoding locations in domain-specific web data," in *Proceedings of the 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI'16)*, 2016, pp. 87–93.

[26] I. S. Vicente, X. Saralegi, and R. Agerri, "Elixa: A modular and flexible absa platform," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval'15)*, 2015, pp. 748–752.

[27] R. Agerri and G. Rigau, "Robust multilingual named entity recognition with shallow semi-supervised features," *Artificial Intelligence*, vol. 238, pp. 63–82, 2016.

[28] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics*, vol. 2008, no. 10, p. P10008, 2008.

[29] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.

[30] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Science*, vol. 105, no. 4, pp. 1118–1123, 2008.

[31] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, no. 3, p. 036106, 2007.

[32] A. Olteanu, C. Castillo, F. Diaz, and S. Vieweg, "Crisislex: A lexicon for collecting and filtering microblogged communications in crises," in *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media (ICWSM'14)*, 2014, pp. 376–385.

[33] A. Olteanu, S. Vieweg, and C. Castillo, "What to expect when the unexpected happens: Social media communications across crises," in *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW'15)*, 2015, pp. 994–1009.

[34] A. Zubiaga, M. Liakata, R. Procter, G. W. S. Hoi, and P. Tolmie, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *PloS one*, vol. 11, no. 3, p. e0150989, 2016.

[35] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95)*, 1995, pp. 1137–1145.

[36] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'11)*, 2011, pp. 262–272.

[37] L. Yao, Y. Zhang, B. Wei, H. Qian, and Y. Wang, "Incorporating probabilistic knowledge into topic models," in *Proceedings of the 19th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'15)*, 2015, pp. 586–597.

[38] A. Ritter, O. Etzioni, and S. Clark, "Open domain event extraction from twitter," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12)*, 2012, pp. 1104–1112.

[39] X. Wang, C. Leckie, J. Chan, K. H. Lim, and T. Vaithianathan, "Improving personalized trip recommendation to avoid crowds using pedestrian sensor data," in *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM'16)*, 2016, pp. 25–34.

[40] K. H. Lim, J. Chan, S. Karunasekera, and C. Leckie, "Personalized itinerary recommendation with queuing time awareness," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'17)*, 2017, pp. 325–334.

[41] D. Zheng, T. Hu, Q. You, H. A. Kautz, and J. Luo, "Towards lifestyle understanding: Predicting home and vacation locations from user's online photo collections," in *Proceedings of the Ninth International AAAI Conference on Web and Social Media (KDD'15)*, 2015, pp. 553–561.

[42] B. Cao, F. Chen, D. Joshi, and S. Y. Philip, "Inferring crowd-sourced venues for tweets," in *Proceedings of the 2015 IEEE International Conference on Big Data (BigData'15)*, 2015, pp. 639–648.

[43] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: Finding topic-sensitive influential twitterers," in *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM'10)*, 2010, pp. 261–270.