TUCAN: Twitter User Centric ANalyzer

TUCAN: Twitter User Centric ANalyzer

Luigi Grimaudo*, Marco Mellia† and Maurizio Munafò‡
Politecnico di Torino, Italy
{*luigi.grimaudo, †mellia, ‡munafo}@polito.it

Mario Baldi§ and Han Song¶
Narus Inc.
{§mbaldi, ¶hsong}@narus.com

***Abstract*—Twitter has attracted millions of users that generate a humongous flow of information at constant pace. The research community has thus started proposing tools to extract meaningful information from tweets. In this paper, we take a different angle from the mainstream of previous work: we explicitly target the analysis of the timeline of tweets from "single users". We define a framework - named TUCAN - to compare information offered by the target users over time, and to pinpoint recurrent topics or topics of interest. First, tweets belonging to the same time window are aggregated into "bird songs". Several filtering procedures can be selected to remove stop-words and reduce noise. Then, each pair of bird songs is compared using a similarity score to automatically highlight the most common terms, thus highlighting recurrent or persistent topics. TUCAN can be naturally applied to compare bird song pairs generated from timelines of different users.**

By showing actual results for both public profiles and anonymous users, we show how TUCAN is useful to highlight meaningful information from a target user's Twitter timeline.

I. Introduction and motivation

Twitter is nowadays part of everyone's life, with hundreds of millions of people using it on regular basis. Originally born as a microblogging service, Twitter is now being used to chat, to discuss, to run polls, to collect feedback, etc. It is not surprising then that the interest of the research community has been attracted to study the "social aspects" of Twitter. User and usage characterization [1], [2], topic analysis [3]–[5], community-level social interest identification [1] have recently emerged as hot research topics. Most of previous works focus on the analysis of "a community of twitters", whose tweets are analysed using text and data mining techniques to identify the topics, moods, or interests.

In this paper we take a different angle: first, we focus on the analysis of a Twitter *target user*. We consider set of tweets that appear on his Twitter public page, i.e., the target user's timeline, and define a methodology to explore exposed content and extract possible valuable information. Which are the tweets that carry the most valuable information? Which are the topics he/she is interested into? How do this topics change over time? Our second goal is to compare the Twitter activity of two (or more) target users. Do they share some common traits? Is there any shared interest? How important is for one user a topic of interest for the other user? What is the most common interest of these two users, regardless of the time they are interested in it?

We propose a graphical framework which we term as TUCAN- Twitter User Centric ANalyzer. TUCAN highlights correlations among tweets using intuitive visualization, allowing exploration of the information exposed in them, thus enabling the extraction of valuable information from user's timeline. Given a number of limitations on the topic analysis of Twitter messages, such as limited length of messages, prevalent use of non-dictionary words (*i.e.*, abbreviations, mentions, hashtags, re-tweets, slang, and cultural words), and lack of contextual resources (*e.g.*, due to extensive use of Twitter for "private" purposes [6]), lots of ingenuity is required to automatically extract significant information out of tweets. From a methodology stand-point, we build upon text mining techniques, adapting them to cope with the specific Twitter characteristics.

As input, we group a user's tweets based on a window of time (e.g., a day, or a week) so to form *bird songs*, one for each time window. At the next step, filtering is applied to each bird song using either simple stop-word removal, stemming, lemmatization, or more complicated transformations based on lexical databases. Next, terms in bird songs are scored using classic Term Frequency-Inverse Document Frequency (TF-IDF) [7] to pinpoint those terms that are particularly important for the target user. Each pair of birds songs are finally compared by computing a similarity score, so to unveil those bird songs that contain overlapping, and thus persistent, topics. The output is then represented using a coloured matrix, in which cell colour represents the similarity score. As a result, TUCAN offers a simple and natural visual representation of extracted information that easily unveils the most interesting bird songs and the persistent topics the target user is interested into during a given time period. Moreover, comparisons among bird songs gives intuitions on the transition of user interests as well as the significance of topics to the user.

The framework is naturally extended to find and extract similarities among tweets of two or more target users. TUCAN computes and graphically shows the similarity among bird songs generated from the timelines of the pairs of target users, revealing similarities and common interests that are present possibly during different time periods.

TUCAN demonstrates to be useful to highlight correlation among tweets, which in turn proves very valuable in identifying topics of interest in the Twitter timeline of a user. This is very instrumental in generic individual profiling or surveillance applications, where the information hidden inside the target user's flow of tweets has to naturally emerge. TUCAN is also very powerful to compare individuals, to examine their timelines in parallel, hunting for similarities, pinpointing common interests, and observing changes, deviations, etc. For instance, comparing a well-known public profile timeline, e.g., President Barack Obama, against a generic target user would unveil if they share common political interests. Alternatively, two casual targets can be compared to see if some common trait/interest

exist (possibly at different time), e.g., to evaluate the success of an Internet dating or marriage.

To demonstrate the effectiveness of TUCAN on real-world microblogs, we applied it to two month long history of 712 Twitter users. Results show that the correlation among tweets turns out to be a key point in the identification and analysis of twitter users over time; analyzing tweet messages of a politician, we were able to confirm that his topics and topic durations well matched with ongoing political events at the time. Comparing his tweets against tweets from the US government, a subset of topics that are in-line with the government's positions were picked up. Analysis on topic changes revealed transitions in users' social relationships.

II. Related work

The increasing availability of valuable information from microblogging platforms pushed the research community to investigate efforts for mining textual information from them.

Text topic extraction and modeling. A plurality of works ([6], [8]–[12]) is based on a well known topic modeling technique called, the Latent Dirichlet Allocation (LDA) [13]. [10] extends LDA to infer descriptions of entities (*e.g.*, authors) separately from their relationships. [6] incorporates supervision to LDA, leveraging hashtags of Twitter for topic labeling. Generalizing topic extraction to Tweets without hashtags, [11] directly applies LDA to individual sentence within each Tweet message.

To further enhance the performance of topic extraction from short and sparse messages, author-topic (AT) model was proposed [14], [15]. By creating topic mixture at the level of authors rather than individual documents, AT is claimed to obtain more stable set of topics than LDA. [5] conducts empirical comparisons of LDA, AT, and simple TF-IDF on aggregates of Tweet messages. The work discovers that the accuracy of the topic models are highly influenced by the length of the documents. It also finds that with long enough documents, the model based approaches become less effective compared to the baseline TF-IDF. Based on the observations, we design TUCAN to flexibly aggregate messages into bird songs. With effectively formed bird songs, TUCAN can provide powerful topic analysis even with generic TF-IDF.

Time-series analysis in microblogs. Many literatures on topic analysis ([3], [6], [11]) focus on detecting emergence of anomalous topics or prominent shifts on topic trends. Leveraging groups of semantically associated document tags, [3] discovers temporally emergent topics from Twitter data stream. [6] defines four types of Tweet categories and classifies streamed messages into them. Because these time-series analysis work on the entire group of users as a whole and do not distinguish single users, they cannot express topical relationships across individuals. We, on the other hand, focus on building dynamic relationships among the users. Aimed at similar goal, [16] proposes to detect topical relationships across entities over time. However, they only focus on time correlated co-occurring events. Instead, TUCAN aims to detect topic correlations even if they occur at different time frames.

III. Framework

The TUCAN architecture includes three modules: (i) bird song generator, (ii) cross-correlation computation engine, and (iii) dashboard visualizer. A set of target Twitter users, *e.g.*, their screen names or user-ids, is provided to the system as an input. The system collects tweets related to such users on which various analytics are executed. Their outcome is visualized to enable the operator to gain knowledge about the users and the topics they are twitting about.

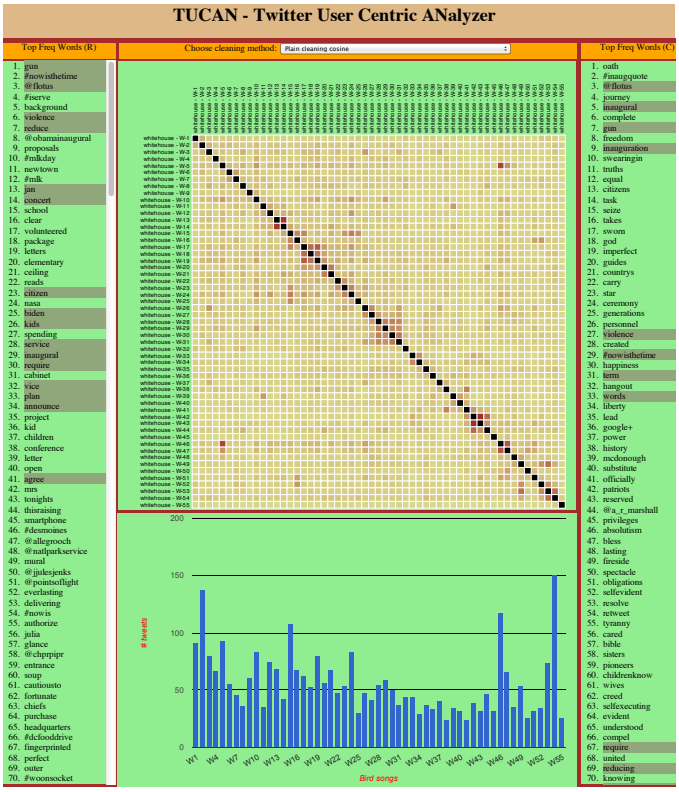
A. Bird song generation and cleaning process

Let $TW(u)$ be the set of tweets of a single user u that are retrieved from Twitter, time stamped with their generation time, stored and organized in a repository in binary format, to be easily accessed and further analyzed when necessary. Bird songs are created by aggregating tweets from $TW(u)$ generated within a time period T, to then be analyzed. We define the i-th *bird song* for the user u, $BS(u,i)$, as the subset of tweets in $TW(u)$ that appear in the i-th time period of duration T, i.e., the set of tweets that are generated in the $[(i-1)T,(i)T),\ i>0$ window of time. For each user u, $S(u)=\{BS(u,i)\ \forall i \| BS(u,i)\neq\emptyset\}$ is the set of all non-null bird songs. Let $N(u)=|S(u)|$ be the number of bird songs for user u.

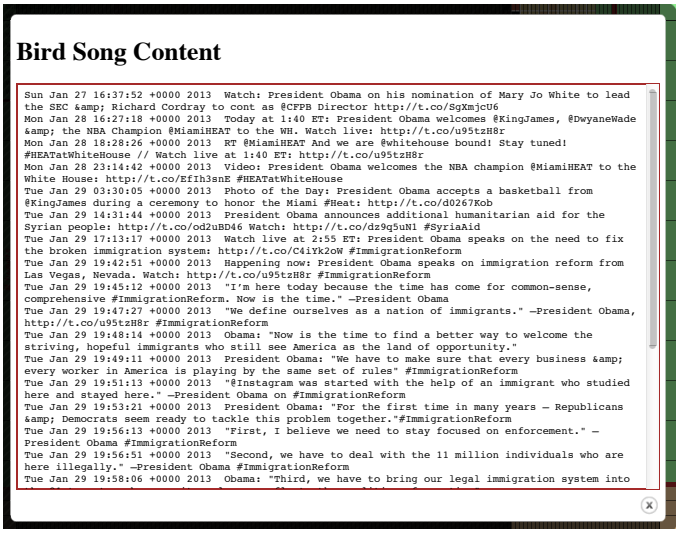
A "plain cleaning" pre-processing is applied to bird songs to discard stopwords, HTML tag entities, and links. Plain cleaning can be possibly substituted by more advanced text cleaning mechanisms; the following are also considered in this work: (i) removal of Twitter 'mentions', (ii) stemming, (iii) lemmatization, and (iv) ontology-based lexicon generalization. TUCAN allows the analyst to select the most appropriate cleaning method to take advantage of different effects of them in different contexts. Twitter mentions are words that begins with @ signs representing the mentioning of some named entities. The intuition behind removing the mentions comes from the fact that they do not provide insight in the topics being addressed, being just Twitter-ID of other users. Stemming and lemmatization are common text processing techniques aiming at reducing a word to its root form to lower sparseness present in a text document. The main difference between stemming and lemmatization is that the former is based on the heuristic of removing the trailing part of a word, while the latter brings a word to a canonical form based on a vocabulary and a morphological analysis the word. Here the Porter stemming algorithm [17] was deployed, while lemmatization is derived from the well-established Wordnet lexical database [18]. At last, our ontology-based lexicon generalization method leverages the Wordnet database to derive the most general concept for each word in the bird song. For instance, "gun" and "rifle" are replaced by the more generic term "weapon". The impact of the different cleaning methods will be exemplified by the experimental results presented in Section IV.

B. Cross-correlation computation

Each pre-processed bird song is tailored in a Bag-Of-Words (BoW) model, a common representation used in information retrieval and natural language processing. The bird song is tokenized in an unordered set of words, disregarding their sequence and position. Each word is then scored according



(a) TUCAN Main Interface



(b) Bird song detail

Fig. 1. TUCAN Web Interface showing the analysis of the WhiteHouse official account. $T = 7$ days, plaing cleaning and Cosine similarity are considered.

to a weighting scheme. In this work, the Term Frequency-Inverse Document Frequency (TF-IDF) score is adopted as past literature has shown it to produce good results [5]. TF-IDF is computed as the product of the frequency of a term in its bird song and the inverse of the frequency of the term in the set of documents (i.e., all bird songs) being analyzed. TF-IDF provides a measure of the importance of a term in a specific bird song (first factor) put in perspective with how common the term is in the whole collection of bird songs. The intuition behind this weighting scheme is that, if a word appears in a huge number of bird songs in a given collection, its discriminative power is very low and is probably not useful to represent the content of the bird song, even if it often appears in it. Hence, words that are frequent in a bird song but rare in the collection are assigned with higher weights.

Bird songs are then transformed into a vector space model $VS(u, i)$, in which each word is given a fixed position. In this space, each word in the bird song $BS(u, i)$ is characterized by its TF-IDF score. Words that do not appear in $BS(u, i)$ are characterized by a null score.

Two indexes are deployed to evaluate the similarity $VS(u, i) \otimes VS(v, j)$ among a pair of bird song vectors: Cosine similarity and Mean Reciprocal Rank (MRR).

Given any two term document vectors, the Cosine similarity is the cosine of the angle between them. The closer two vectors are to one other, the smaller the angle between them will be, i.e., the higher their similarity. Intuitively, the Cosine similarity of two very similar bird songs will be close to one. Instead, if no common words appear in two bird songs, their Cosine similarity will be 0.

MRR is commonly proposed in the literature to score a list of potential responses to a query, ordered by probability. MRR is defined as:

$$MRR_{R_1,R_2} = \frac{1}{n(R_1)} \sum_{w \in R_1} \frac{1}{rank(w, R_2)}$$

in which R_1 and R_2 are ranked term vectors, $n(R_1)$ is the number of words in R_1 whose TF-IDF score is not null, and the $rank(w, R_2)$ returns the rank of the word w in R_2. In case w is not present in R_2, $\frac{1}{rank(w,R_2)} = 0$.

In the context of this work, terms in $VS(u, i)$ are sorted by decreasing TF-IDF values to form $RS(u, i)$, the ranked term vector, and

$$VS(u, i) \otimes VS(v, j) = MRR_{RS(u,i),RS(v,j)}$$

Notice also that $n(RS(u, i))$ is the number of words in $BS(u, i)$.

Note that the commutative property holds true for the Cosine similarity, while in general $MRR_{R_1,R_2} \neq MRR_{R_2,R_1}$. Indeed, the MRR of R_1 with respect to R_2 shows how terms that are present in R_1 are important (in rank sense) in R_2. For instance, consider two bird songs of 10 words each, $BS1$ and $BS2$. Assume only the word "violence" appears in both; it is the top ranked word in $RS1$, but is ranked tenth in $RS2$; then $MRR_{RS1,RS2} = 1/100$ while $MRR_{RS2,RS1} = 1/10$, reflecting that in $BS1$ "violence" is much more important than in $BS2$. Section IV demonstrates how non-commutative property of MRR is used to discover particular relationship between a pair of bird songs.

C. Dashboard visualizer

In order to pinpoint similarities among bird songs, independently of the time the user posted them, TUCAN computes the similarity score for all possible pairs of bird songs. In total, N^2 similarity scores are computed and stored in a matrix form, where each cell represents $VS(u,i) \otimes VS(u,j), i,j \in [1,N]$. To help identifying correlation, the matrix is presented to the analyst in a graphical format using a web interface. Each cell is represented by a square whose color reflects the similarity score between the i-th and j-th bird songs. In particular, let

$$m = \max_{i,j,i \neq j} VS(u,i) \otimes VS(u,j),$$

cells are colored with different intensity, using a linear scale, so that the cell with similarity equal to m has the darkest color (see Figure 1(a) for an example). Bird songs are organized in increasing time window from left to right (and top to bottom).

As shown in Figure 1(a), when a cell is clicked, the web interface displays the top-ranked words appearing in $BS(u,i)$ and $BS(u,j), i \neq j$ on the left and right panes next to the matrix. Words that appear in both bird songs are highlighted. When clicking on the cells in the main diagonal (presented always in black[1]), the analyst is offered a popup showing the content of the original tweets of the i-th bird song. The GUI also shows a histogram below the matrix reporting $n(u,i)\ \forall i$ to allow the analyst to easily gauging variations in the bird song size, e.g., due to the user changing his twitting habits during a holiday period. At the top of the matrix, the analyst is offered a drop-down menu to select the cleaning pre-processing to be applied.

IV. Experiments

Applying TUCAN to real world data from Twitter, we conducted an extensive study examining its capability on analyzing user centric topics. We begin by presenting a description on our dataset and how we collected it. Then we provide a series of sensitivity evaluation on various parameters of TUCAN, followed by a number of use cases with emphasis on different aspects of user centric topic analysis.

A. Dataset description

To perform user centric analysis through TUCAN, we monitor 712 randomly selected Twitter users for two or more months starting from the Summer 2012. The actual Tweet period covered for each user depends on the combination of the user's activity and crawling limitations imposed by Twitter API. Additionally, we monitor 28 well-known public figures, selected among politicians, news media, tech blogs, etc. In total, we collect 740 twitter timelines leveraging Twitter REST APIs[2]. Specifically, we access each user's public timeline and retrieve tweet STATUS objects which contains monograms (messages he puts on his page with no destined user), mentions of other users, conversations with follower/followees, and status updates.

From a total of 810,655 tweets, it emerges that 15% of them contain hashtags, 25% contain replies and 12% hyperlinks to other web pages. Similar proportions of message types are reported in the literature, suggesting our dataset presents no bias towards any particular types of tweets. About 300 users (40%) twitted more than twice in each week. Out of them, 20 users posted more than 400 tweets per week (i.e., more than 57 tweets/day). This already suggests that the window size parameter T has to be tailored to each user twitting habit when forming bird songs. Section IV-C presents sensitivity tests on T.

[1]Note that by definition, $VS(u,i) \otimes VS(u,i) = 1$.

[2]https://dev.twitter.com/docs/api

Rank	single Tweet	$T = 1$ day	$T = 7$ days	$T = 14$ days
1	photo	lead	#immigrationreform	#immigrationreform
2	day	international	**immigration**	**gun**
3	bo	@cfpb	**gun**	**immigration**
4	snow	cordray	**violence**	**violence**
5		mary	comprehensive	comprehensive
6		snow	@whlive	@whlive
7		nominates	broken	broken
8		sec	@vp	reform
9		richard	representative	representative
10		white	reform	@vp

TABLE I. Top-words ranked by TF-IDF, Barack Obama.

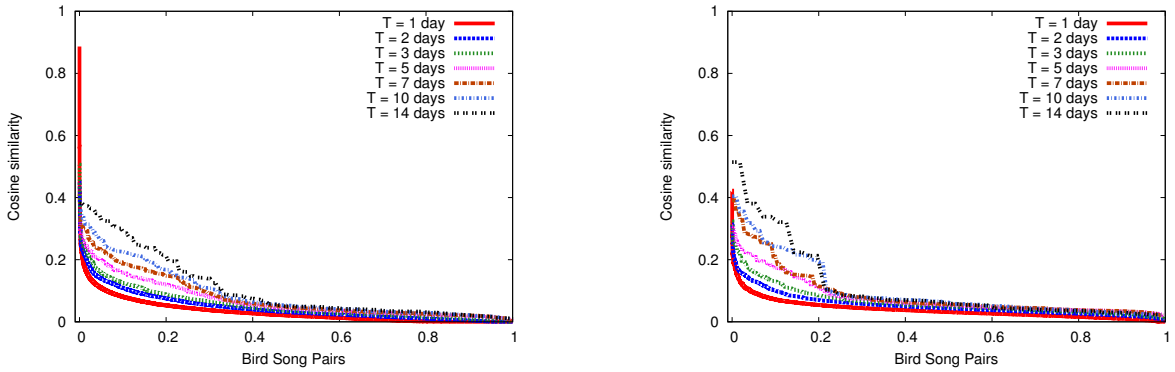
B. The TUCAN GUI

Revisiting Figure 1, we present how TUCAN GUI is used for our analysis with an example of 56 week long history of official White House tweets. The reader can appreciate the correlation that TUCAN highlights among bird songs along the main diagonal. The darker areas indeed show that the correlation among top-words in bird songs is high, unveiling persistent topics. For instance, the top-words presented in the left and right lists easily allow to see the topics the White House was twitting about, i.e., violence and inauguration (Week-41). Those tweets refer to the second half of January 2013 during (i) the Inaugural Address by President Barack Obama, and (ii) the debate on violence and weapon possession started after the Newtown school tragedy. For reference, consider (part of) the tweets that form the bird song referring the 21st of January 2013 on Figure 1(b). Intuitively, extracting and summarizing information from the original tweets is much more complicated than by observing TUCAN output. Other areas of high correlation are clearly visible. Those refer to the Sandy hurricane, London Olympics games, etc. TUCAN allows to easily spot these major events that last for several weeks. Notice the Week-6/Week-46 dot with high similarity. Topics in those weeks refer to bills, insurance, gas price, and cost of education.

C. Parameter sensitivity analysis

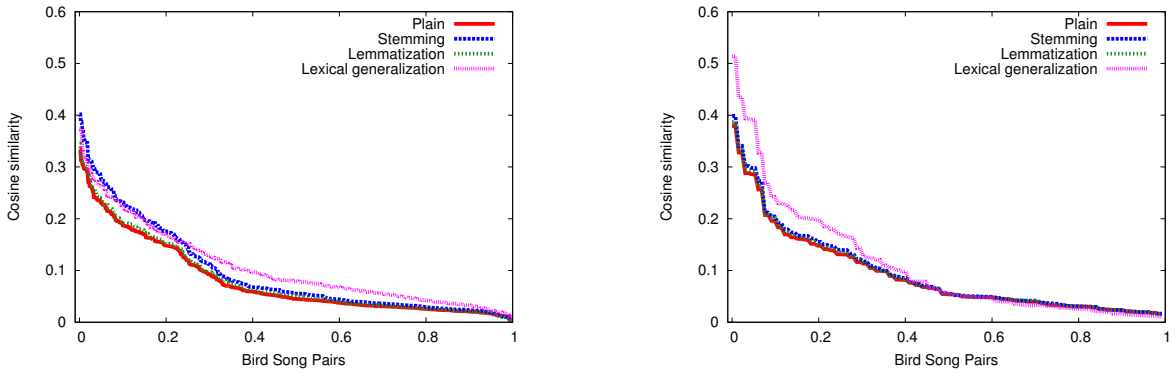
We begin our analysis on TUCAN by showing effects of tuning different parameters: time window sizes, preprocessing methods, and inclusion of Twitter mentions. Results are presented showing, for all bird songs pairs of user u, the similarity score sorted in decreasing order. The X-axis displays the bird song rank normalized to the number of bird songs $N(u)$. The Y-axis shows absolute values of similarity score.

Effect of different time window sizes. The time window size T determines the size of bird song – a highly important parameter for topic models to perform optimally [5]. Figure 2 shows comparisons of time windows sizes for a public figure (Barack Obama, on the left) and a randomly chosen normal user (User A, on the right). As we vary window size from



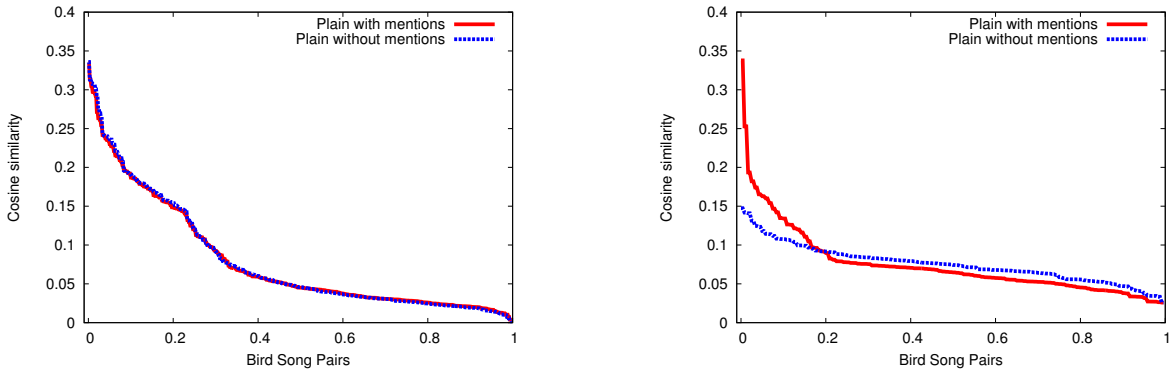
(a) Barack Obama (b) User A

Fig. 2. Effect of different time window sizes T. Plain cleaning and Cosine similarity.



(a) Barack Obama (b) User B

Fig. 3. Effect of different cleaning methods. Cosine similarity and $T = 7$ days.



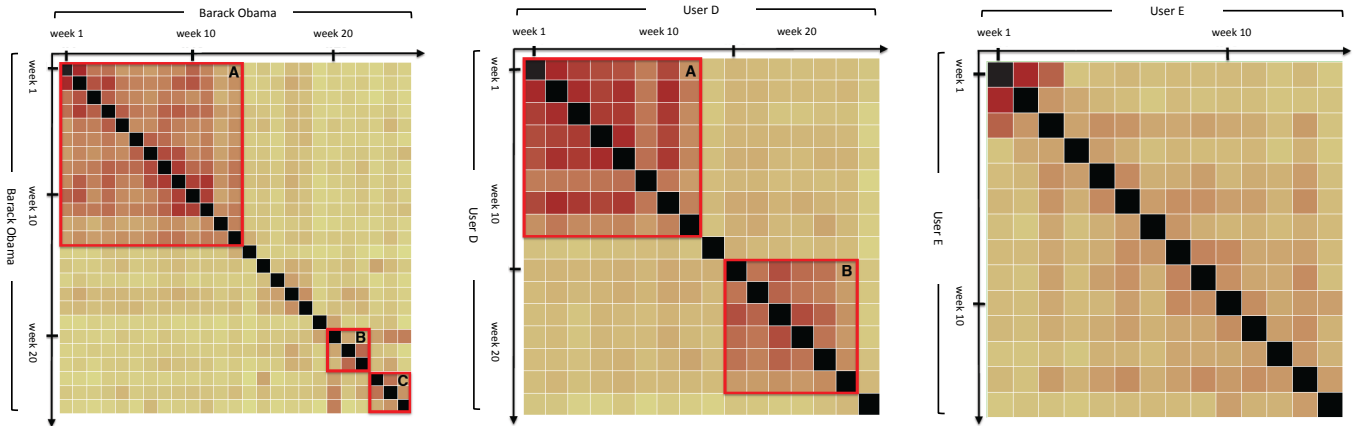
(a) Barack Obama (b) User C

Fig. 4. Effect of mention removal. $T = 7$ days, plain cleaning, and Cosine similarity.

$T = 1$ day to $T = 14$ days, we expect that the overall similarity scores become strictly higher. Indeed, Figure 2 clearly shows this; for instance, for Barack Obama, the average (max) score of $T = 1$ day is 0.03 (0.87), average score of $T = 14$ days is 0.11 (0.38). Same observation holds for normal users as shown in Figure 2(b). Notice that higher similarity score is not always welcome; a too large aggregation time window tends to create very large birds songs, in which similarity is artificially inflated, and the analysis blurred. As previously stated, T should be matched to twitting habits of the target.

On the other end, too short aggregation time window makes similarity interesting only on a small subset of bird song pairs, focusing the analysis on a too small groups of bird songs. Artifacts are also possibly created. For instance, notice the high similarity score at $x = 0$ in Figure 2(a) when $T = 1$ day. The reason for this outlier is that bird songs are formed by only a handful of terms; if three or four of those happened to co-occur in two bird songs, their similarity score turns out to be extremely high.

Further inspection on topic words also supports the importance of aggregation of tweets into bird songs. Table I shows up to ten top-words extracted from Barack Obama's bird songs (as previously mentioned). When tweets are used as they are (without aggregation), we not only observe that the number of common words are small, *i.e.*, the tweet has too few words to allow successful analysis; but we also observe



(a) Barack Obama - Max similarity = 33% (b) User D - Max similarity = 31% (c) User E - Max similarity = 26%

Fig. 5. Similarity among bird songs for different type of users. $T = 7$ days, plain cleaning, Cosine similarity.

that the relationship among the words are loose. Similarly, for $T = 1$ day, no clear topic emerge. In contrary, when $T = 7$ or $T = 14$ days, the top-words are much more coherent (especially between 'gun', 'violence', and 'broken') pinpointing to a clear topic.

In summary, both the general trend of small similarity, and possible existence of outliers suggest to use quite large time window for analysis. As observed in Table I, and from other tests run on a large number of users, $T = 7$ days usually gives the same amount of meaningful keywords as larger window sizes (*e.g.*, $T = 14$ days). For that reason, from here on, we use $T = 7$ days unless otherwise noted. Once similarity has been pinpointed, the analyst can drill down by lowering T.

Effect of different pre-processing methods. Many researchers on information extraction have proposed different pre-processing methods to sanitize original documents. On the particular application to Twitter document analysis, however, no work identified the optimal method. Therefore, we evaluate the performance of three well-known sanitization methods – stemming, lemmatization, lexical generalization – applied on the top of plain cleaning. Figure 3 compares the cleaning methods considering one public and one normal Twitter users as before. For the profile of Barack Obama, Figure 3(a) suggests that stemming and lexical generalization work better than lemmatization and plain cleaning. However, the overall gap between the two groups of curves is less than 0.05 in similarity score. In the case of a normal user, Figure 3(b) shows that lexical generalization tend to perform better than other pre-processing methods (by about 0.05). Notice that the increase in similarity is predominant for those pairs whose similarity is already quite large, and thus possibly less useful. By investigating further, we notice that lexical generalization tends (by definition) to return more general topics.

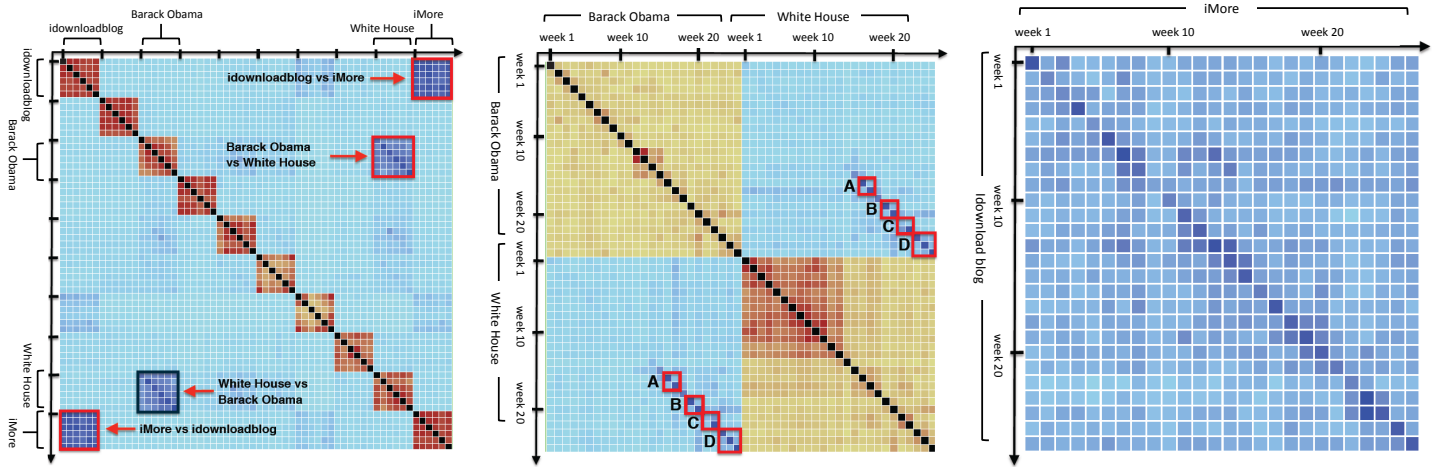
In summary, we observe small impact of the filtering process, and results are marginally affected by this choice. As such, TUCAN has been designed to offer the analyst the choice of the cleaning method that he consider the best for the case under analysis. Plain cleaning is the default choice.

Effect of including Twitter mentions. Among many specific mechanisms Twitter offers, "mentions" play an important role in the analysis of user conversations [19]. From our analysis, we noticed an interesting contrasts when mentions are included or excluded. As Figure 4(a) shows, for public figure's tweets (Barack Obama), results of including and not including mentions do not make much difference in similarity distributions. This is because of the usage of mentions by public profiles: either those are rarely used (e.g., in news media), or they are used to mention i) to lots of different users, or ii) to always the same group of users (this is the case for Barack Obama). However, for a normal user, as seen in Figure 4(b), proportion of mentions can get up to 70% and clearly makes distinction on the similarity distribution. The reason for similarity being higher when mentions are included is because the mentions themselves works as keywords (as in the case of '@whlive' or '@vp' from Table I that are however the Twitter profiles of White House Live and of the Vice President), resulting in (unnaturally) increased similarity scores. In Section IV-D, we will demonstrate cases where inclusion of mentions can indicate a particular pattern of a normal user's social relationship. Unless explicitly denoted, however, we include mentions in our analysis.

D. User centric analysis

To demonstrate the effectiveness of TUCAN on user analysis, we present results of case studies. Unless mentioned otherwise, we use the following settings by default: (i) windows size of 7 days, (ii) pre-processing with plain cleaning, and (iii) similarity scoring using Cosine similarity measure.

Analysis on timeline of a single user. Figure 5 shows correlation matrices representing similarities between pairs of bird songs of a single user. Figure 5(a) shows a matrix on the bird songs of *Barack Obama*. It highlights three blocks of highly correlated period of Tweets. The larger block [A] at the upper left corner represents Obama tweets during US presidential election in 2012. With a maximum Cosine similarity score of 0.33, it is clear that he has been tweeting a lot on a few correlated topics (voting, Romney, convention, health, etc. being among the most recurrent top terms). Block [B] refers to periods when Obama was interested in fiscal cliff. Finally, block [C] relates to the shooting in the Newtown elementary school, during which Obama's major topic terms were gun, violence, and weapon.

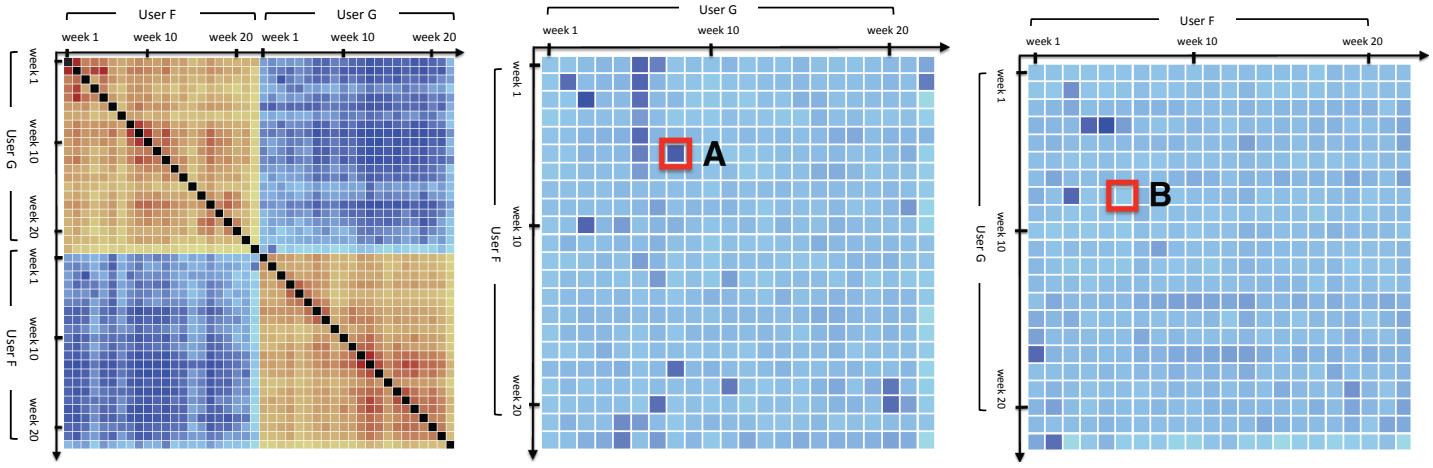


(a) Famous users vs famous users. $T = 14$ days.

(b) Barack Obama vs White House. $T = 7$ days.

(c) idownloadblog vs iMore. $T = 7$ days.

Fig. 6. Similarity among users over different bird songs. Plain cleaning and Cosine similarity.



(a) Cosine similarity

(b) MRR - User F vs user G

(c) MRR - User G vs user F

Fig. 7. Comparison between user F vs user G. $T = 7$ days, plain cleaning.

The correlation matrix in Figure 5(b) shows an interesting behavior of a normal "user D" (as opposed to a public figure or news media). As discussed in Section IV-C, mentions are very frequent among common users. Analyzing user D's bird songs without filtering out mentions, the plot highlights two blocks, [A] and [B]. The similarity of bird songs are dominated by the use of mentions to particular follower/followee of his. Investigating key terms in the time period of block [A], user D was exchanging messages with one of his follower. After one week of pause, in block [B], user D then mentions about another follower of his (and never refers to the follower in [A]). We suppose that user D's sudden change in his mentions indicates a change in his social relationship, e.g., change of his dating partner.

Lastly, Figure 5(c) shows a typical correlation matrix of generic "normal users". Compared to a public figure's correlation matrix (Figure 5(a)), the size of correlated blocks is small and more uniform. Likewise, the maximum similarity score is also lower at 0.26. This can be explained by different use of Twitter between public figures and normal users; public figures use Twitter to deliver messages with substantial topics ([2], [6]), whereas normal users use Twitter to socialize (with messages on status updates, social signals, messages indicating mood, etc.) as noted in [6].

Finally, TUCAN can also be instrumented to highlight artificial similarity among a user's tweet that were generated by automatic tools like Foursquare check-in, auto-tweet tools, etc. We do not report their examples for sake of brevity.

Analysis across different users. Besides the per-user analysis, TUCAN can infer semantic relationships across a multiple of users when applied to a group of target users. We select ten public figures and media blogs and report the cross-similarity matrix in Figure 6. The latest six bird songs with $T = 14$ days are considered, referring to a common period of time. Each bird song is checked against each other. Results are represented as a colored matrix, using different color scales (and normalization) for blocks outside the main diagonal and in the main diagonal (where same-user's bird songs are compared). Focusing on the former, two pairs of users emerge as mostly correlated: $\{$*Barack Obama*, *White House*$\}$ and $\{$*idownloadblog*, *iMore*$\}$.

Zooming in and increasing the resolution by selecting $T = 7$ days, Figure 6(b) compares {*Barack Obama*, *White House*} in detail over 25 weeks of tweeting. First, notice that during Barack Obama's campaign (ref. Figure 5(a)) the correlation with White House is marginal. After elections, four periods of high correlations are pinpointed, highlighting the periods Barack Obama and White House publicize similar topics. The block [A] indicates the period of educational cost cut. [B] indicates the massacre at Newtown. [C] refers to fiscal cliff, and [D] on reformation of US immigration laws. The discovery of both well-correlated and non-correlated periods allows us to quantify periods of time the President spoke for himself (and his political party) and the government of the US.

Similar consideration holds when zooming in {*idownloadblog*, *iMore*} comparison in Figure 6(c). Both users are blogs reporting news on Apple products. Also in this case $T = 7$ days, for 25 bird songs. Only the cross-similarity macro block is shown for the sake of brevity. Notice the large similarity in the main diagonal; it indicates that the two profiles report the same news, whose duration last for short period of time. The behavior is justified by the fact that both accounts work as sources of technology news.

Analysis on different similarity metrics. So far, we focused on the correlation among bird songs using Cosine similarity measure. Using MRR metric, on the other hand, TUCAN is able to highlight relative degree of interests (or focus on the correlated topic) between users being compared. In Figure 7 we consider a randomly picked pair of normal users {*User F*, *User G*}. Applying Cosine similarity (Figure 7(a)), no pair of bird songs appear to be significantly correlated, indicating that no particular topic is shared between the two users. Maximum cosine similarity is indeed 0.06. Applying MRR (Figure 7(b)), however, the matrix highlights a small number of bird songs with clearly higher similarity score than the rest. In macro blocks [A] and [B], bird songs reveal the two users' common interests on "final exams", indicating that the two users are possibly students. Moreover, we can see that the topic is more important for user F than user G because $MRR_{F,G} > MRR_{G,F}$. By inspecting tweet messages around [A], we inferred that user F is a teenager repeatedly tweeting her round of finals preceding summer vacation. By inspecting tweet messages around [B], user G does not mention as frequently on his exam as he is a graduating student. As seen in its non-commutative property, with MRR, the different significance of the same topic (exams) is expressed as skewed to one side of the matrix.

V. Conclusion

In this paper we presented TUCAN, a framework to graphically represent semantic correlations of individual Twitter users' timelines. Building on text mining techniques, TUCAN analyses "bird songs", *i.e.*, group of tweets belonging to the same time period, and compares their similarity. The analyst is offered a GUI to investigate the impact of different preprocessing and similarity definitions. Experiments conducted on actual Twitter users show the ability to pinpoint recurrent topics, or correlations among users.

There are several avenues for future work. First, we would like to expand our framework to be able to model patterns of topic durations and transitions. Leveraging the measurements revealing the correlation durations of topics, accumulating the statics for long-term can reflect changes in the user's interests. Second, we are interested in inferring users' social relationships based on their topical relations. Our evaluation on MRR shows the possibility of quantifying inequivalence between pairs of topics. Extending the metric, we expect to obtain finer-grained relational information than just the degree of similarities.

References

[1] A. Java, X. Song, T. Finin, and B. Tseng, "Why We Twitter: Understanding Microblogging Usage and Communities," *Workshop on Web Mining and Social Network Analysis*, pp. 56–65, 2007.

[2] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a Social Network or a News Media?" *WWW*, pp. 591–600, 2010.

[3] F. Alvanaki, S. Michel, K. Ramamritham, and G. Weikum, "See What's enBlogue - Real-time Emergent Topic Identification in Social Media," in *EDBT*. Berlin, Germany: ACM, 2012.

[4] M. Mathioudakis and N. Koudas, "TwitterMonitor: Trend detection over the twitter stream," in *SIGMOD '10*. New York, NY, USA: ACM, 2010, pp. 1155–1158.

[5] L. Hong and B. D. Davison, "Empirical Study of Topic Modeling in Twitter," in *Workshop on Social Media Analytics*, New York, NY, USA: ACM, 2010, pp. 80–88.

[6] D. Ramage, S. T. Dumais, and D. J. Liebling, "Characterizing Microblogs with Topic Models," in *ICWSM*, W. W. Cohen and S. Gosling, Eds. The AAAI Press, 2010.

[7] G. Salton and M. J. Mcgill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.

[8] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled LDA: a Supervised Topic Model for Credit Attribution in Multi-labeled Corpora," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*Stroudsburg, PA, 2009, pp. 248–256.

[9] Y. Liu, A. Niculescu-Mizil, and W. Gryc, "Topic-link LDA: Joint Models of Topic and Author Community," in *Annual International Conference on Machine Learning*, New York, NY, USA: ACM, 2009, pp. 665–672.

[10] J. Chang, J. Boyd-Graber, and D. M. Blei, "Connections Between the Lines: Augmenting Social Networks with Text," in *ACM SIGKDD*, New York, NY, USA: ACM, 2009, pp. 169–178.

[11] 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 *ECIR'11* Berlin, 2011, pp. 338–349.

[12] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi, "Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-Scale Data Collections," in *WWW*, New York, NY, 2008, pp. 91–100.

[13] D. M. Blei, A. Ng, and M. Jordan, "Latent Dirichlet Allocation," *JMLR*, vol. 3, pp. 993–1022, 2003.

[14] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The Author-Topic Model for Authors and Documents," in *UAI*, Arlington, VA, 2004, pp. 487–494.

[15] M. Rosen-Zvi, C. Chemudugunta, T. Griffiths, P. Smyth, and M. Steyvers, "Learning Author-Topic Models from Text Corpora," vol. 28, no. 1. New York, NY, USA: ACM, Jan. 2010, pp. 4:1–4:38.

[16] A. Das Sarma, A. Jain, and C. Yu, "Dynamic Relationship and Event Discovery," in *WSDM*, New York, NY, 2011, pp. 207–216.

[17] M. Porter, "An Algorithm for Suffix Stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[18] C. Fellbaum, "WordNet: An Electronic Lexical Database," 1998.

[19] C. Honeycutt and S. C. Herring, "Beyond Microblogging: Conversation and Collaboration via Twitter," in *HICSS'09*, 2009, pp. 1–10.