Frequent Itemsets as a summary of social media streams

Younos Aboulnaga[1]

David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada
yaboulna@uwaterloo.ca

1 Problem Statement

Making sense of Social Media has been an area of active research, because as noisy as the medium is many people turn to it as their source of information specially on recent news or events. Recurring stories of success include the Tweet that saved the life of a UC Berkley journalism student who got detained while covering the unrest in Egypt in 2008, the photographs of the emergency landing of the Delta Airliner on Hudson river in 2009 that wouldn't have made it to the public if not for Twitter, Tweets about earthquakes arriving to people before they feel the earthquakes, celebrity deaths and so on and so forth. However, these are all success stories of people rather than the platform. Actually the Tweet that saved the life of the UC Berkley student could have gone unnoticed in the noisy medium that he happened to have access to briefly write "detained". That a friend of his saw the Tweet and took action based on it was mainly because the friend was following him closely, given that he is an adventurous person doing something interesting. This is not to say that the platform doesn't deserve any merit; in fact, it is unprecedented that individuals are given broadcasting abilities that exceed those of news agencies, which are in turn now using the same medium to compete with individuals for being the first source of news stories. However, this broadcasting abilities are acquired through building a large network of followers, whose accounts are in turn being followed.

Identification of influential accounts, classifying them according to topic and recommending them is a large body of work which we consider only tangentially related. We take a different view point at social media, which actually takes the *social* part out of it by looking at content and disregarding the source. Actually the social aspect is really absent in many cases, such as Facebook group posts with hundreds of comments from tens of users having long intermingling discussions while they are all strangers to each other. If anyone wants to join the discussion or if the owner of the group wants to understand what the post has stirred, they cannot filter posts by people's rapport not only because this is not readily available, but also because this would be inappropriate as it would lead to neglecting posts subjectively. A similar situation arises when following the Twitter public stream, currently available only to applications through the streaming API but not made available for viewing neither by Twitter nor by any application. Rather than making the public stream available directly, many applications provide digests focusing on certain aspects, such as the most Retweeted content, or the presence of links to certain types of content. Detecting real world events and creating a calendar has attracted a lot of attention in academia, and is topped by Ritter's Status Calendar [1]. Another areas of interest in academia is topic modelling, but it is usually done in batch mode using techniques such as LDA. Petrovic proposes one of the few online mining techniques by using locality sensitive hashing for first story detection; Tweets are clustered and a new cluster is started if the Tweet's distance from all centroids is larger than a certain threshold.

We are also focus on techniques suitable for online mining, and we also intend to finally present the user by highlights from the social media stream. However, we address the problem in two steps, an online step and an on demand step. The online step uses Frequent Itemsets Mining (FIM) to keep an up to date synopsis of the stream, stripped down of the noise and keeping only *interesting* itemsets. The on demand step can be searching, clustering or any other mining operation; it operates on the synopsis data and is thus more efficient. However, the definition of *interesting* itemsets has to be precisely defined first of all. We propose a few characteristics for judging interestingness of itemsets in the list below.

1. **Multi word expressions and named entities are not interesting in themselves:** FIM algorithms are meant to operate on retail purchase data, where every item is a product and there is no undesired items. However, the nature of language is that many terms or even sets of terms are not informative and thus mining them is undesired; namely, multi-word expressions like phrasal verbs, and any occurrence of stop words. These are analogous to packaging of items in a retail context, which is luckily not charged separately on receipts, except for plastic bags in some countries like Canada and Brazil. Named entities made up of more than one proper noun are only slightly interesting in themselves (indicating that the entity is under discussion), however it is highly interesting as part of a bigger itemset. We propose that this type of itemsets should not be reported unless they are part of a bigger itemset, however this would break the monotone property. Therefore, an alternative is to collapse such itemsets into single items; e.g., “Justin Bieber and Katy Perry” would become Justin-Bieber, Katy-Perry where the “and” would be pruned out.
2. **Time domain characteristics:** Itemsets that (re)occurred recently are more interesting than those whose most recent occurrence is further in the past. Itemset that don't recur for a certain period has dropped off the radar of interest. This doesn't happen gradually, but rather abruptly and regardless of the support of the itemset when it was interesting.
3. **Frequency domain characteristics:** Frequency of an itemset is not defined as its support divided by the interval between the first and last occurrence, but rather it should be defined as frequency components reflecting how frequency changed over time. Wavelets, Discrete Time Fourier Transform, and other curve fitting techniques can extract such components. Another alternative is dynamic histograms that create bins of certain mean values, minimizing variance. Itemsets of upward tendency in the frequency domain are interesting, and those of downward tendency are not.

[1] http://statuscalendar.cs.washington.edu/

4. **Bursty itemsets are interesting:** Bursts are not necessarily accompanied by upward tendency in the frequency domain. Kleiberg (2003) showed that bursts can happen as dispersed periods of slightly increased interest, and proposed a method that uses an infinite state machine to indicate the state of a term as levels of burstiness. Moving up the burstiness levels has a cost, while moving down happens with time. The cost function depends on the characteristics of the timeseries, and minimizing the overall cost reveals truly bursty features. For itemsets with flat frequency domain functions, we will use this model to detect bursts, devising the appropriate cost function.