Summarizing World Speak: A Preliminary Graph Based Approach

Nikhil Londhe
University at Buffalo
nikhillo@buffalo.edu

Rohini K. Srihari
University at Buffalo
rohini@buffalo.edu

Abstract

Social media platforms play a crucial role in piecing together global news stories via their corresponding online discussions. Thus, in this work, we introduce the problem of automatically summarizing massively multilingual microblog text streams. We discuss the challenges involved in both generating summaries as well as evaluating them. We introduce a simple word graph based approach that utilizes node neighborhoods to identify keyphrases and thus in turn, pick summary candidates. We also demonstrate the effectiveness of our method in generating precise summaries as compared to other popular techniques.

1 Introduction & Background

The popularization of social media has fundamentally transformed how news stories are reported and shared (Kwak et al., 2010; Hermida, 2010). As the news stories like the disappearance of flight MH370 (#MH370), or the British referendum to exit EU (#Brexit), or the release of a mobile game based on the popular Pokemon cartoon series (#PokemonGO) – capture global attention, Twitter conversations about them swell in volumes yet vary widely in opinion as well as language, as illustrated in Table 1.

Thus, this begs the question as to how does one begin to *understand* such hashtags in their entirety? Can *true* summaries be generated for such multilingual datasets? Further, how would such summaries circumvent the inherent challenges of language bias, subjectivity of posts and potential Spam (Stafford and Yu, 2013)? Thus, in this work, we explore two primary questions with regards to such multilingual text streams : (a) what constitutes an *ideal* summary? and (b) how can such summaries be evaluated?

1.1 Microblog Summarization

Typically multi-document summarization takes one of two approaches : *abstraction* or *extraction*. Abstractive methods generate a summary by incorporating key information (Kim et al., 2011), whereas extractive methods on the other hand simply aim to choose the most representative sentences (Radev et al., 2002). Typically, longer documents like news stories and blogs have lend themselves better to abstractive summaries whilst shorter documents like social media posts tend to do better with extractive summaries.

As such microblog summarization has received sustained interest in the past few years (Inouye and Kalita, 2011). Overall, two simple techniques based on term frequencies, namely SumBasic (Nenkova and Vanderwende, 2005) and HybridTfIdf (Sharifi et al., 2013) seem to be unanimous choices in most circumstances (Mackie et al., 2014). Thus, for the example hashtags introduced above, we examine the generated summaries for these methods in Table 2. Overall, the problems with the extracted summaries can be enumerated as:

- **Language bias:** Most summaries contained disproportionate number of English posts despite the dataset being more language balanced.[1]
- **Objective vs Subjective posts:** Although in a sense, a representative summary should stay close to the underlying split between subjective and objective posts – for our use case, we would prefer more objective posts, that present an information of some sort beyond just a simple opinion

[1] See section 1.2

Proceedings of Recent Advances in Natural Language Processing, pages 452–458,
Varna, Bulgaria, Sep 4–6 2017.
https://doi.org/10.26615/978-954-452-049-6_060

Hashtag	Tweets	Notes
MH370	RT @HuffPostQuebec: #MH370: l'Australie n'a rien détecté près de ses côtes	- Multiple languages : redundancy across languages - Spam and irrelevant data - Mixture of subjective and objective posts
	The Pilots Flight Simulator files were deleted in February. Fishy? #MH370	
	RT @BlackIrishI: St. Jude Pray For Us #MH370 #Flight370	
PokemonGO	RT @BaronVonGamez: Caught my first pokemon...in the gulag #PokemonGO	
	rt gyms pokemongo	
	#PokemonGo - yet another virtual drug for the phone-face generation.	
Brexit	El meme más viral ahora mismo #Brexit	
	British submarine docks in Gibraltar as Spain try to claim sovereignty after Brexit vote	
	#Followback Brexit + uncertainty = market chaos: #TeamFollowBack	

Table 1: Sample tweets for three global hashtags

Hashtag	Algo	Summary
MH370	S	• mh370 • rt mh370 • flight mh370
	H	• rt prayers for the families of missing flight mh370 • rt on the pilots crew and passengers of mh370 • rt what was the cargo of flight mh370
Brexit	S	• brexit • rt brexit • fascinated to see if facebook
	H	• rt on the lessons of brexit for academics • rt in the midst of all brexit fiasco • rt brexit why the british said no to
PokemonGO	S	• pokemongo • rt pokemongo • rt this is the best pokemongo gif so far
	H	• rt pokemongo is out now in the uk • rt caught my first pokemon in the gulag pokemongo • rt this is the magic of pokemongo

Table 2: Initial summarization experiments using SumBasic (S) and HybridTfIdf (H)

Statistic	**Min**	**Max**	**Avg**
Number of tweets	2,153	18,488	7,353
Average Length	6.85	12.46	10.00
Number of tokens	2,445	23,734	10,476
Stopword % (en)	17.91	44.45	30.21
Non english %	24.39	58.10	38.64
Number of languages	38	60	50

Table 3: A summary of the collected tweets

Thus, in essence we seek a summarization algorithm that can be language agnostic whilst preferring objective posts over subjective ones whilst minimizing redundancy within and across languages in the selected posts.

1.2 Evaluating Summaries

Unfortunately, no standard datasets exist for evaluating microblog summarization, largely due to Twitter's data redistribution policies. Although some interest in real-time microblog summarization has been seen in the recent past (Lin et al., 2016), such tasks largely involve stream filtering than summarization. Most researchers in the past have relied on using a variety of trending hashtags to crawl a large number of tweets over an extended period of time. We thus, follow a similar approach and present a summary of the collected datasets between June and July 2016 in Table 3. Note that the crawling process involved using the streaming API for a fixed interval of time. Thus, the number of posts varied between datasets based on the popularity of the hashtag at the time of collection and an additional drop due to deduplication.

Further, there is no consensus on the number of tweets to be considered for a summary. Typically this number has varied from as little as one (Sharifi et al., 2010) to as many as 70 (Chakrabarti and Punera, 2011) and is usually dependent on the method of evaluation. For example, when human summaries are available and recall oriented metrics like ROUGE (Lin, 2004) are used, the target summary size tends to be conservative. However, when precision based metrics are utilized that merely judge the generated outputs, a larger summary size is evaluated.

Given that it is intractable to produce human generated summaries for our use case, we propose an alternate two-pronged approach that seeks to evaluate both precision and recall. First, to evaluate recall, we employ automatic evaluation techniques as proposed by Louis and Nenkova (2009). The basic principle of such evaluation is treating both the input dataset and the generated summary as word distributions and quantifying the quality of the summary as a function of the divergence between these distributions. Although as argued by Saggion et al. (2010), the method through not always reliable works well for multilingual documents as in our case.

We also manually evaluate the precision of the generated summaries for different target summary sizes. The relevance of a post in an ordered set is measured purely in terms of whether it adds any information to the summary. That is, subjective posts or posts that are redundantly similar to previously presented information are deemed as irrel-

Algorithm 1 NEIGHBORHOOD SUMMARIZATION

Input: Dataset D, target summary length n, length parameter λ and phrases P
Output: Representative n tweets
Let $C \leftarrow \emptyset$ be a cache of tweets, $numToks = 0$, $allToks \leftarrow \emptyset$ be a set of unique tokens
for Tweet T in D **do**
    toks = tokenize(T)
    for Token t in $toks$ **do**
        incrementTf(t)
        addEdges(t, $toks \setminus t$)
        $numToks++$, $allToks.add(t)$
        if $t.tf > \eta \times avgTokenProb$ **then**
            $C.add(T)$
        end if
    end for
end for
Let $S \leftarrow \emptyset$ be a set of seen tweets, $H \leftarrow \emptyset$ be a heap of size n
for Tweet T in C **do**
    if $getMaxOverlap(S, T) < 0.8$ **then**
        Compute $\tau(T)$
        Let $\omega(T) = \tau(T)$
        $p = getPhrase(T, P)$
        if p is not null **then**
            $\omega(T) += w_2(p)$
        end if
        $\omega(T) /= max(\|T\|, \lambda)$
        Add $< T, \omega(T) >$ to H
        Add T to S
    end if
end for
Return H

evant. We contend that this two fold evaluation allows us to measure two equally important properties for the generated summaries – how true they are to the underlying term and topic distribution as well as their utility.

Thus, having presented the preliminaries of both the problem and evaluation techniques, we present our graph based summarization technique in the following section.

2 Word Graphs & Keyphrase Extraction

Before we present our summarization technique, it is pertinent that we present how we arrived at a word graph based solution to begin with. Essentially, we were seeking to solve two problems:

- Reduce the language bias by somehow assigning some notion of language independent *importance* to a token
- Incorporate keyphrase extraction (that has shown to produce better summaries (D'Avanzo and Magnini, 2005; Boudin and Morin, 2013)) in a language agnostic setting

We claim that vertex neighborhoods for such graphs are immensely useful in not only determining *important* words (or keyphrase constituents) but also in supplementing context for rarer languages. Algorithm 1 presents a programmatic summary and as can be seen, it proceeds in two stages – graph construction and summary extraction. We present each in the following subsections.

2.1 Graph Construction

Unlike the other graph based summarization algorithms (Olariu, 2014) that index bigrams and trigrams, we instead index an entire post as a clique. Our graph $G = (V, E)$ indexes tokens as vertices (V) and token co-occurrence establishes an edge (E). That is, for a post P with k distinct words, we add it to the graph as k distinct vertices with an edge connecting every possible pair of vertices (i.e. N_2^k edges total). The primary reason for this choice is to circumvent both the relatively free word order associated with hashtag usage but also present a larger context to every token. Further, while adding to the graph, we perform some rudimentary tokenization like lowercasing all text, dropping usernames, URLs, emojis, symbols as well as removing the # prefix. The primary reason for this is to prune the overall graph size. We found that most of such discourse tokens had a low degree and did not affect the system output.

For every token added to the graph, we maintain its term frequency tf as a vertex property that is updated on each subsequent occurrence of the token. Additionally, we maintain a cache of tweets C as we build the graph. A given tweet is added to the cache if it contains any token that occurs more frequently[2] than average. The average token frequency can simply be computed by keeping track of number of tokens ($numToks$) and the number of unique tokens (size of set $allToks$). The cache allows us to reduce the computation time on the next phase.

[2]The parameter η allows to control and reduce the cache size

2.2 Summarization

Having described the first stage, i.e., graph construction, we now turn our attention to summarization. In this stage, we iterate over the cached tweets and re-rank them based on two parameters (a) the average importance of all tokens within the tweet and (b) token overlap with previously seen tweets. For a given vertex v, with the set X_v representing all neighbors of v, we assign the following weights to each vertex:

$$w_1(v) = tf(v)/numToks \tag{1}$$

$$w_2(v) = \frac{1}{n}\sum_{i=1}^{n} w_1(x_i) \tag{2}$$

where $numToks$ represents the total number of tokens as before , and $x_i \in X_v$. In essence, this is equivalent to weighing each vertex based upon the average probability of occurrence of its neighbors. Finally, given a tweet T, it can be represented numerically as:

$$\tau(T) = \sum_{t}^{T} w_2(t) \tag{3}$$

We claim that similar to other tf based formulations (Mackie et al., 2014), we are in turn quantifying each post by the *importance* of its constituent tokens. Finally, we additionally boost a tweet if it contains a keyphrase. In defining a keyphrase, we utilize the MWE extraction methodology as outlined by Londhe et al. (2016). They essentially construct a similar graph and use pairwise evaluation of vertices to determine MWEs that include weak MWEs like Named Entities. Although our current algorithm expects the phrases as an external input, a combined algorithm could be developed that simultaneously keeps track of MWE candidates during the graph construction phase.

Note that although we do not explicitly filter out stopwords, they are implicitly controlled by two things. Firstly, by weighing each tweet and not individual tokens, it prevents dominance by specific tokens. Secondly, by restricting permissible tweet overlap prevents the same set of words from reappearing multiple times. We also argue that the implicit inclusion of importance effectively only boosts keyphrases. That is, the boost on MWEs only impacts the final summary if the MWE is important. We now present our experimental results

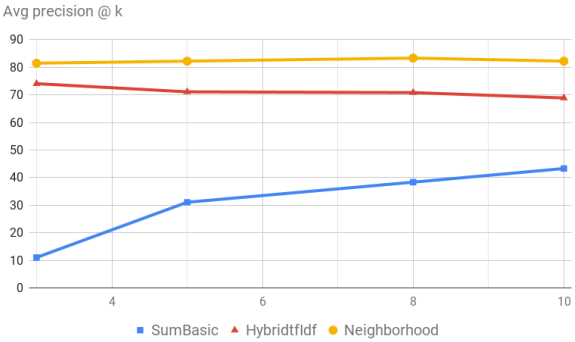

Figure 1: Average Precison @ k

to prove that the said method generates better summaries for our use case as compared to traditional methods.

3 Experiments & Results

3.1 Experiments

As discussed in Section 1.2, we use two fold evaluation for our techniques. Firstly, we compare the divergence statistics and present them in Table 4. The actual divergence values are unbounded, and typically a lower divergence score implies a better summary. However, as described before, a higher divergence score may not imply a bad summary in our case.

Thus, we additionally compare precision at k for a range of values of k and present the results in Figure 1. As described in Section 1.2, precision is manually computed by evaluating information added by a given post. We also list some generated summaries and their assigned relevance labels in Table 5. Note that a grayed out cell indicates a non relevant post.

3.2 Observations

1. The divergence statistics for our method do not vary much from the other techniques, except the KL divergence between the generated summary and the input.
2. However, our method consistently outperforms both on the metrics of precision and compactness.
3. While the precision on HybridTfIdf is comparable to our method, the method is more likely to present posts in the dominant language, i.e English.

Algorithm	JS (Unsmoothened)	JS (Smoothened)	KL (Input to Summary)	KL (Summary to Input)
SumBasic	0.42	0.21	1.60	0.89
HybridTfIdf	0.37	0.18	1.43	0.59
Neighborhood	0.41	0.23	1.69	**1.14**

Table 4: Average Divergence Statistics

Dataset	SumBasic	HybridtfIdf	Neighborhood
PokemonGO	pokemongo	rt pokemongo is out now in the uk	rt icymi pokemongo fiasco ruins the circle of life
	rt pokemongo	rt caught my first pokemon in the gulag pokemongo	pokemons rollen uit de 3dprinter in technobiel pokemongo
	rt this is the best pokemongo gif so far	rt this is the magic of pokemongo	jenkeiss nuoriso saatu liikkeelle paremmin kuin yhdelläkään terveysohjelmalla vau pokemongo lisääliikett
	pokemon pokemongo	rt pokemon go all the time pokemongo	pokemongo oyununda pokemon falan avlamam bur özelliklerime aykır benim ayağıma gelsin
	rt any caamember pokémon trainers find our tiny trucks and get free ride around town to help with your quest pokemongo goo	rt what is the world coming to	all my pokemon stops are churches what you trying to say pokemongo
MH370	mh370	rt prayers for the families of missing flight mh370	rt results of the crowdsourced search for malaysia flight mh370 malaysiaairlines via
	rt mh370	rt on the pilots crew and passengers of mh370	rt the baseless rush to blame pilots of flight 370 via mh370
	flight mh370	rt what was the cargo of flight mh370	rt and this is what the hunt for mh370 has come down to
	malaysia mh370	rt this is who hijacked the malaysian airlines flight mh370	rt prayers for the families of missing flight mh370
	rt breaking news on mh370 plus were ranking your questions right now at 370qs cnn	mh370 where in the world is this plane	rt australia to take charge of southern search for missing flight mh370
Brexit	brexit	rt on the lessons of brexit for academics	rt brexit britain now to suffer the uncertainty that curses switzerland
	rt brexit	rt in the midst of all brexit fiasco	rt how the hokey cokey made me ashamed to be londoner brexit
	fascinated to see if facebook	rt brexit why the british said no to	rt the latest mylbrook vehicle finance news thanks to brexit porsche
	rt watching brexit politics this morning makes me wonder if the snp could field candidates in england and wales sturgeon re	rt brexit poll this is the beginning of end for	rt brexit big fuck to the merchants selling us nwo allwhitesunite
	but were litist for saying brexit is about racism	rt in light of brexit and the ensuing chaos	rt the startling human toll of brexit

Table 5: Summary samples & relevance

4. Although SumBasic seems to perform poorly on precision, its overall performance is comparable for larger values of k. Rather, it seems to produce better summaries for larger values of k.
5. However, we do need a better and ideally a single automatic method of evaluation that could potentially combine these three metrics

4 Related Work

Overall, the work related to hashtag summarization can be divided into two broad categories: (a) Stream clustering (b) Microblog summarization. Stream clustering involves summarizing a live stream of data like the classic CluStream (Aggarwal et al., 2003) method or the more recent Sumblr (Shou et al., 2013) pipeline. However, these methods are suited for large steam processing and require continuous data streaming to create and compare historical summaries. Also, the summaries are query driven implemented as adhoc queries or drill-downs. Thus, we mention them here only for the sake of completeness as they are not directly comparable to our use case.

Microblog summarization, as partially discussed in Section 1.1, has largely evolved from traditional summarization techniques that follow one of the *extraction* or *abstraction* routes. However, although several methods like TextRank (Mihalcea and Tarau, 2004), LexRank (Erkan and Radev, 2004) and MEAD (Radev et al., 2004) have been proposed over the past few years, the

two methods compared, viz SumBasic (Nenkova and Vanderwende, 2005) and HybridTfIDF (Sharifi et al., 2013) have been shown to outperfom them (Mackie et al., 2014).

Finally, improvements in parsers and taggers specific to Twitter have also given rise to a new family of Information Extraction based methods for summarization (Xu et al., 2013). However, most such tools are language specific and may not be applied to multilingual datasets without loss of generality and accuracy.

5 Future Work and Conclusions

We believe that the problem warrants much further work. The largest problem would obviously involve finding better ways to evaluate such multilingual summaries without manual intervention. Although considered while designing the algorithm, we did not find a satisfactory way to quantify language divergence as part of the generated summaries. It would also be worthwhile to consider redundancy across languages, especially as applicable for ranking the summary candidates. That is, should a post in the dominant language be preferred against a post in another language that is similar but contains marginally more information?

It would also be worthwhile to incorporate some prior probabilities (like stopword lists) and other semantic equivalences, probably crosslingual dictionaries to improve the generated summaries. It would also be interesting to measure how the summaries evolve for a fixed target summary size as the algorithm sees more posts and use it to measure topic drift and topic divergence. Finally, we would like to explore temporal relationships in a trending topic and possibly generate sub-topical summaries by automatically partitioning the given text stream.

References

Charu C Aggarwal, Jiawei Han, Jianyong Wang, and Philip S Yu. 2003. A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*. VLDB Endowment, pages 81–92.

Florian Boudin and Emmanuel Morin. 2013. Keyphrase extraction for n-best reranking in multi-sentence compression. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Deepayan Chakrabarti and Kunal Punera. 2011. Event summarization using tweets. *ICWSM* 11:66–73.

Ernesto D'Avanzo and Bernado Magnini. 2005. A keyphrase-based approach to summarization: the lake system at duc-2005. In *Proceedings of DUC*.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22:457–479.

Alfred Hermida. 2010. Twittering the news: The emergence of ambient journalism. *Journalism practice* 4(3):297–308.

David Inouye and Jugal K Kalita. 2011. Comparing twitter summarization algorithms for multiple post summaries. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*. IEEE, pages 298–306.

Hyun Duk Kim, Kavita Ganesan, Parikshit Sondhi, and ChengXiang Zhai. 2011. Comprehensive review of opinion summarization. Technical report.

Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*. ACM, pages 591–600.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*. Barcelona, Spain, volume 8.

Jimmy Lin, Adam Roegiest, Luchen Tan, Richard McCreadie, Ellen Voorhees, and Fernando Diaz. 2016. Overview of the trec 2016 real-time summarization track. In *Proceedings of the 25th Text REtrieval Conference, TREC*. volume 16.

Nikhil Londhe, Rohini Srihari, and Vishrawas Gopalakrishnan. 2016. Time-independent and language-independent extraction of multiword expressions from twitter. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. pages 2269–2278.

Annie Louis and Ani Nenkova. 2009. Automatically evaluating content selection in summarization without human models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*. Association for Computational Linguistics, pages 306–314.

Stuart Mackie, Richard McCreadie, Craig Macdonald, and Iadh Ounis. 2014. Comparing algorithms for microblog summarisation. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, pages 153–159.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. Association for Computational Linguistics.

Ani Nenkova and Lucy Vanderwende. 2005. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005* 101.

Andrei Olariu. 2014. Efficient online summarization of microblogging streams. In *EACL*. pages 236–240.

Dragomir R Radev, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Celebi, Stanko Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam, Danyu Liu, et al. 2004. Mead-a platform for multidocument multilingual text summarization. In *LREC*.

Dragomir R Radev, Eduard Hovy, and Kathleen McKeown. 2002. Introduction to the special issue on summarization. *Computational linguistics* 28(4):399–408.

Horacio Saggion, Juan-Manuel Torres-Moreno, Iria da Cunha, and Eric SanJuan. 2010. Multilingual summarization evaluation without human models. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, pages 1059–1067.

Beaux Sharifi, Mark-Anthony Hutton, and Jugal K Kalita. 2010. Experiments in microblog summarization. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*. IEEE, pages 49–56.

Beaux P Sharifi, David I Inouye, and Jugal K Kalita. 2013. Summarization of twitter microblogs. *The Computer Journal* page bxt109.

Lidan Shou, Zhenhua Wang, Ke Chen, and Gang Chen. 2013. Sumblr: continuous summarization of evolving tweet streams. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. ACM, pages 533–542.

Grant Stafford and Louis Lei Yu. 2013. An evaluation of the effect of spam on twitter trending topics. In *Social Computing (SocialCom), 2013 International Conference on*. IEEE, pages 373–378.

Wei Xu, Ralph Grishman, Adam Meyers, and Alan Ritter. 2013. A preliminary study of tweet summarization using information extraction. *NAACL 2013* page 20.