Text Summarization: The Benefits of Argumentation Mining

Marcus Hughes, jmh3

Abstract

Summarization of Twitter presents a unique challenge because tweets are unstructured and short. However, Twitter summarization can be a useful way to probe public opinion. Here I convey that the graph based abstractive summarization algorithm Opinosis shows much promise for Twitter summarization. Furthermore, I show that preprocessing by only considering tweets that have statements supporting a claim proposed by a hashtag creates more meaningful and rich summaries.

1. Introduction

There are roughly 500 million tweets sent each day [2]. Many of these tweets pertain to social movements and become grouped under a topical heading, a hashtag. For example, #LoveWins trended after the United States Supreme Court's legalization of same-sex marriage in 2015 and #Ferguson trended after the social unrest following a police shooting in Ferguson, MO in 2014. Hashtags can come in pairs for and against a movement, e.g. #DefundPP and #StandWithPP in opposition and support of Planned Parenthood's funding respectively. Tweets with movement hashtags are often, but not always, accompanied with some statement explaining and/or supporting the poster's view of the movement. Previous work [10] in argumentation mining proposed that these hashtags can be seen as a premise for an argument and that a classifier can determine which tweets support the premise with additional text and which are merely tweets with no suppor. Thus, Twitter can be used to probe public opinion and reasoning over popular issues. Due to the large volume of tweets, it is impossible to summarize the arguments without using an automated summarization technique.

Many such techniques have been proposed in literature. I test several here and discuss others that I did not have time to implement. This is the first step toward answering whether preprocessing tweets to select only meaningful ones benefits summarization algorithms or not.

2. Previous Work

Summarization of multiple documents, in this case tweets, can be categorized into two main groups: extractive and abstractive. The difficult first step of summarization is identifying which parts of a document are important. Thus, extractive summarization only identifies important portions of the text and groups them together without any further processing. This clearly does not lend itself for an easily readable summary. An extractive summary will likely have redundant words. For example consider three imaginary tweets as a corpus: 1. "Bill 2017 is too expensive but it'll get the job done." 2. "Bill 2017 is too expensive but it's inherently racist." 3. "Bill 2017 gets the job done, no more problems ever!". An extractive summary selecting two out of three of these tweets will repeat words, e.g. Bill 2017, too expensive, get the job done. Further as [4] illustrates, there are intrinsic biases in this kind of summary. Any extractive summary forced to pick between these tweets will miss important caveats presented by the others.

Abstractive summarization on the other hand can meld the three tweets into one summary, e.g. "Bill 2017 will get the job done but it's expensive and racist", to provide a more complete summary without redundant words. Abstractive summarization can range from anything as simple as exanding abbreviations to generating novel natural English summaries. More advanced abstractive summarization is much more difficult than extractive summarization, and thus the majority of work in summarization has been in extractive approaches up to this point.

Previous work in twitter summarization concluded that simpler algorithms, i.e. SumBasic and Hybrid TF-IDF, that only utilize the probability of words instead of more complicated approaches tend to perform better [7]. This was acredited to the "unstructured, unconnected and short characteristics of Twitter posts that are not like traditional documents." I found no other work on twitter summarization that conducted such a comparison of techniques. There have been more recent approaches that claim to outperform the simpler approaches by exploiting socio-temporal context [6], mutual reinforcement graphs [1], stream summarization [15], and information extraction [14]. Most twitter summarization algorithms focus on real-time characterization of events or identifying trends in the twitter community at large. Constructing logical arguments from tweets over a refined topic is not as common.

3. Algorithms

I utlized the following algorithms in my experiment. This is by no means exhaustive or necessarily even the most modern. I would like to have also included LexRank[3], a neural network approach [13], and phrase reinforcement [12], but I did not have the time.

3.1. Random

Without using any additional information from the tweets, the most basic approach to extractive summarization would be to pick random tweets from the corpus and call that a summary. That does not guarantee a representative sample. All the tweets in the summary may repeat each other. I have provided an implementation of it here.

3.2. MostRecent

This utilizes only one portion of the tweet data: the date. Arguably, recent tweets are more relevant for understanding the current opinion on a subject. However, they can become highly biased toward only tiny details of the most recent events especially if the corpus has a narrow scope. I provided this implementation.

3.3. SumBasic

SumBasic was intended by its initial authors as a baseline for summarization but turned out to perform quite well, in fact out performing other systems [9]. SumBasic tabulates the frequency of terms across the documents assinging each term a probability. Each tweet or sentence is assigned a probability proportional to the sum of the probabilities of its terms. Then, the highest weighted tweet containing the most probable term is added to the summary. The probability of the highest word is decreased and this iterates until the summary is of the appropriate length. This means it is very sensitive to stopwords potentially. One must exclude the hashtag as a term or else it automatically becomes the first entry. I implemented this.

3.4. Hybrid TF-IDF

TF-IDF or term frequency inverse document frequency has been used in many portions of natural language processing [**?**]. It considers the product of the probability of the terms and the logarithm of the inverse document frequency (tweets in which a certain term appears). These two portions play a balancing act: words that occur more frequently are likely important but common, non-meaning words (e.g. to, the, it) are likely not important and are thus decreased for by the logarithm of the inverse. A document is primarily a single tweet. However, when computing the term frequencies, the document is the entire collection of tweets because tweets are too short for it to matter otherwise. Thus, the term frequency and document frequency portions can both still contribute. I provided this implementation.

3.5. TF-IDF

TF-IDF performs the same as Hybrid TF-IDF except the document is always only a tweet. Therefore, it really should not provide much information. I provided this implementation only since it was a very simple modification of Hybrid TF-IDF.

3.6. Opinosis

Opinosis is a graph based approach to abstractive summarization developed in 2010 by [4]. Each word is treated as a node in this approach. The strength of each edge grows as more documents include a similar phrases. The approach essentially tries to find an optimal set of partial paths through the graph. A partial path is defined as valid if it matches some predefined set of valid part of speech patterns, more on this later. The benefit of this approach is that different views from tweets that complement each other can be combined into one summary by traveling through the graph, e.g. “the pills are large and too expensive” and “the pills are expensive but still worth while” can be combined into “the pills are large and expensive but worth while.” This simplification is achieved by recognizing that some nodes, words, can be collapsed and removed while others help coordinate phrases together into one. I utilized an existing implementation of the algorithm provided by the author that executed with no problem because it was distributed as a .jar file. Opinosis is optimized for short, highly redundant documents but not necessarily tweets.

3.7. Mead

I was interested in MEAD because it is open source, has many adjustable parameters, and is mentioned in many papers [11]. Mead incorporates a multitude of features that can be turned on and off: length consideration, KeyWordMatch, cosine overlap with the centroid vector of the cluster, LexPageRank, among other details. However, Mead was a bit finicky to get up and running. It is no longer maintained and relied on perl packages that had numerous levels of dependencies. When it’s finally up and running, it works marvelously. However, I must admit that I do not understand all the options it provides yet. As such, I utilized only the default options. This means it functions as a centroid approach and tries to cluster tweets by meaning and find minimal overlap.

4. Experiment

I worked with three corpora of tweets based on three hashtags: #DefundPP, #StandWithPP, and #FightFor15. Togehter, the corpora comprise 7977 tweets: 1301 for #DefundPP, 1031 for #FightFor15, 5645 for #StandWithPP. These were provided to me by Jon Park as he had previously mined them [10]. They came with labels for which tweets had support, which tweets were statements without support, and which tweets were discarded (they were spam or had no coherent theme or argument). For each of the previously described algorithms, I generated two 15 tweet summaries: one based solely on tweets identified as having supporting reasoning another from the entire corpus of tweets with no selection based on label. 15 tweets were selected as a reasonable percentage length of each of the corpora: 1% for #DefundPP, 1.5% for #FightFor15, and 0.3% for #StandWithPP. This nmber could be adjusted as desired depending on the application of the summary.

My initial plan for this experiment was to look at the summaries pairwise. For each corpus in a randomized and blind fashion, each summary would compared to the other summaries pairwise. So for #DefundPP I would present myself with two random summaries. Both summaries may be based on filtered, with-reason tweets; both summaries may be based on the entire corpus with no filtering; or one tweet may be from an filtered summarizer and one from an unfiltered summarizer. I would note which summary I thought was better overall considering ease of reading, lack of repetition, number of meaningful words, and breadth of the summary. While recording the “best” tweet of the two I wrote a one sentence comment that could be used for later analysis. This technique allows me to determine simultaneously which summairzer performs best on tweets and how filtering impacts summarization. A drawback to this technique is the evaluator will have to repeatedly reread summaries. However, this technique could theoretically far outperform other approaches (e.g. reading all the summaries and ranking them) for large scale, distributed human evaluation (e.g. Mechanical Turk evaluation) where evaluators may randomly assign values if presented with a large amount of data at one time.

The drawback of rereading became too large. While I was trying it, I began to realize which summary was which and my answers started to get a bit dry. I found that for the amount of time I was spending reading, I wasn't getting much benefit. While I was attempting an unbiased view, I clearly was not achieving that. Therefore, I instead compared only each unfiltered summary with its filtered counterpart on a step by step basis. I then looked at all the filtered approaches against each other and similarly all the unfiltered approaches against each other. I kept it blind during this process.

5. Results

Rank	#StandWithPP	#FightFor15	#DefundPP
1	Mead	Opinosis	Opinosis
2	Opinosis	TF-IDF	Random
3	TF-IDF	SumBasic	Mead
4	SumBasic	Random	TF-IDF
5	Random	MostRecent	SumBasic
6	Most Recent	Mead	MostRecent
7	Hybrid TF-IDF	Hybrid TF-IDF	Hybrid TF-IDF

Figure 1: Ranking without filtering

Rank	#StandWithPP	#FightFor15	#DefundPP
1	Opinosis	Hybrid TF-IDF	Opinosis
2	Mead	Random	SumBasic
3	TF-IDF	Opinosis	MostRecent
4	Hybrid TF-IDF	TF-IDF	Random
5	SumBasic	SumBasic	Mead
6	Random	MostRecent	Hybrid-TF-IDF
7	MostRecent	Mead	TF-IDF

Figure 2: Ranking with pre-filtering

Here is an example of my thought process during the ranking process. For #StandWithPP with no filter, I ranked Mead first because its most important tweet was "I support planned parenthood . Its about personal freedom to choose , responsibility , humanrights , law & social justice . I #StandWithPP". Opinosis was a close second as its first entry was "i #standwithpp / making a political statement , because health is a human , /' shouldn't be a statement and especially today ." This is close to having clear meaning, but something got lost in translation; I'm not sure they there are " / "in the summary. Hybrid TF-IDF was least ranked because its summary included the following tweet: "#StandWithPP #StandWithPP #StandWithPP #StandWithPP #StandWithPP #StandWithPP #StandWithPP #StandWithPP #StandWithPP #StandWithPP !!!!!!!!!" That has no added meaning for the summary. If I were more careful, this tweet should have a weight of 0. With the filter, Opinosis was able to summarize meanings in short concise ways: "pregnant , her body , her choice . their own choices . care in safe supportive environment ." While it may not be the most gramatically correct summary, you quickly capture what I understand to be the theme of the tweets.

For #StandWithPP I preferred the filtered summary for these summarizers: TF-IDF, Hybrid TF-IDF, Random, MostRecent. I preferred the unfiltered summary for these summarizers: Mead, Opinosis. For

#FightFor15 I preferred the filtered summary for all except Opinosis. For #DefundPP I preferred the filtered summary for these summarizers: SumBasic, Random, Opinosis, TF-IDF. I preferred the unfiltered summary for these summarizers: Mead, MostRecent, Hybrid TF-IDF. Therefore, I preferred the filtered summary 13 times and the unfiltered summary 6 times. That seems to indicate to me that filtering helps.

```
i #standwithpp '/'' making a political statement , because health is a human , '/'' shouldn't be a state
this is #antichoice #terrorism , american terrorism and domestic terrorism .
terrorist attack .
t_user is american terrorism and my new shero .
we proudly #standwithpp of colorado and colorado springs .
for affordable health care .
my heart is heavy .
access to safe , legal .
access to safe , affordable .
their own bodies .
access to safe , legal abortions .
#scotus to decide biggest #abortion case in decades -/: sign letter now :/: t_url .
t_user you for real ? who else shoots up pp clinics #standwithpp against the willfully daft .
our thoughts with everyone involved in this terrible and in this terrible situation .
i am proud to #standwithpp .
```

Figure 3: Opinosis with no filtering on #StandWithPP

```
for affordable health care .
access to safe , affordable .
their own bodies .
pregnant , her body , her choice .
their own choices .
care in safe supportive environment .
terrorist attack in and colorado .
reproductive health & rights .
reproductive health is important .
parenthood is good .
t_user so you're outlining progress ? it is abortion is legal b/c it is a right .
i know that pp patients could be any of us #standwithpp because health is a human and because a human .
i'm so glad t_user will be defending women's rights on the planned parenthood and parenthood panel .
domestic terrorists .
a provider of transition care to many @ppact @ppact is essential to the trans community .
```

Figure 4: Opinosis with filtering on #StandWithPP

6. Discussion and Analysis

6.1. Filtering by arguments helps disproportionately

For Opinosis, I find that filtering and only summarizing those tweets labeled "with-reason" often led to a moderately worse summary. You can see this some in Figures 3 and 4. The filtered summary feels more choppy with shorter entries. Some of this shortness is addressed in Section 6.2 since it stems not from general algorithm but lower level implementation details. On the other hand methods like Hybrid TF-IDF

```
Christian Extremism. #StandWithPP
I stand for baby #StandWithPP
I #StandWithPP  support women’s right to health.
Basic healthcare, people. #PlannedParenthood #StandWithPP
Stop the gynoticians!! #StandWithPP https://t.co/bVjs3OqEC2
#StopDomesticTerrorism #StandWithPP #EnoughIsEnough https://t.co/AwjpCwd1lu
#standwithpp and denounce terrorism.
@CeceCastilloo you’re why i’m pro-choice #StandWithPP
I stand for baby #StandWithPP
Healthcare is a human right! #standwithpp https://t.co/0LeKC6D5Rg
Christian terrorism is why I #StandWithPP
Thinking of my colleagues in Colorado. #StandWithPP
I stand for baby #StandWithPP
Healthcare is dangerous in America. #StandWithPP
#WhyINeedYouIn5Words "Planned Parenthood condoms didn’t work" #StandWithPP
```

Figure 5: SumBasic with Filter

```
#StandWithPP
I #StandWithPP .
#StandwithPP  https://t.co/SEoWD5B0xN
I continue to #StandWithPP
@topWave_ @JesseLaGreca #StandWithPP
Today, everyday, I #StandWithPP
#StandWithPP against the terrorists
I #StandWithPP
#StandWithPP and #StopAmericanISIS
#StandwithPP Cecile is a badass  https://t.co/7rSMhr0Q3s
#StandWithPP regardless of truth
This is terrorism #StandWithPP  https://t.co/urr9d0uEcy
"pro-life" #StandWithPP  https://t.co/gydWAMudpM
I #StandWithPP in #ColoradoSprings
#StandWithPP for @LadyPJustice
```

Figure 6: SumBasic without Filter

and SumBasic appear to benefit greatly from filtering. This is most clearly demonstrated in Figures 5 and 6. Without filtering, the tweets mainly repeat that they shey stand with Planned Parenthood without clearly converying why. On the other hand, the filtering only allows meaningful tweets through.

6.2. Problems with Opinosis

As mentioned earlier, Opinosis is an abstractive summarization tool giving it many advantages of its own. However, Opinosis was developed assuming regular English written language, i.e. no convention of hashtags, @ symbols, or abbreviations. It is designed to perform on Penn Treebank part of speech tags. However, tweets are better tagged with an alternative part of speech system [5]. This system can account for the nuances of twitter: urls, emoticons, hashtags, interjection abbreviations (WTF), shorthand (substituting u for you). Opinosis utilizes part of speech tags when determining what a valid path through its constructed graph is. It would be best to develop a new version of Opinosis more specifically adapted to tweets. Unfortunately the Opinosis source code is not available, and I did not have time to develop it from scratch. I did find another implementation of Opinosis available on GitHub by another author https://github.com/fannix/opinosis. When I went to test it, I discovered that the results it produced did not match those produced by the original version. They did not properly combine sentences but instead produced something more akin to an extractive summary, just repeating document entries (tweets) in their entirety. I did not feel comfortable going forward with it as such.

7. Novel Summarization

While I did not have time to implement a novel summarization method, I did have some thoughts I wished to convey.

7.1. N-gram mining

The Pyramid automated summarization evaluation method [8] works by checking if some set of important atomic facts (the most basic facts) that a human evaluator deemed important enough to place in their summary also appear in the automated summary. In a sense, these facts are expressions about a theme, e.g. consider the atomic fact "my favorite color is red" is an expression on the theme "my favorite color." A good summary would convey the strongest (most common) expressions on the most critical themes (themes that occur most often). Therefore, I could imagine an approach similar to SumBasic but instead of searching for the highest probability term it looks for the highest probability theme.

7.2. The Value of Graphs and Troubles of Twitter

It is my opinion that twitter summarization benefits strongly from graph based summarization. Tweets are very unstructured. Structure does arise through the repetitive and redundant nature of the multitude of voices expressed through twitter. Graphs allow for meaningful connections to be amplified as shown in Opinosis's stellar performance. I chose not to fine tune any of the algorithms strictly to twitter because in their original formulations in literature they are presented in other contexts. I think all the algorithms could have stronger performance if "@" terms were weighted much weaker. From what I can tell, "@" very rarely really contributes to the argumentation. People use many different modes of communication on twitter. One individual may write "govt" while another will write "government." If these are not mapped to the same node in a graph, valuable information can be lost. Therefore, I think tweets should undergo more extensive cleaning to make them of similar style: a common acronym set, removal of urls, etc.

Sometimes additional hashtags are utilized in a tweet in addition to the premise hashtag. These sometimes are synonyms for the premise hashtag. In other cases, they link and compress a more detailed argument further. Hashtags that serve as addional support and not merely synonyms appear, it seems, in the middle of the tweet as part of the main sentence. On the otherhand, synonym hashtags get prepended or appended to the tweet outside of the main sentence.

Finally, I think the summary should be in plain English instead of twitter format. This requires some more processing, however it would be lessened if tweets were already in a similar style with preprocessing. A nested list structure of the summary or some indication of how concepts relate would be even more beneficial instead of just a collection of tweets or ideas. In the summaries, the percentage of people holding a view is somewhat lost (it only lingers through the ordering of the summaries sometimes.)

8. Conclusions

Ultimately, I don't think that any of the summarizers did especially well. They each suffer from unnecessary repetition of words. I think Opinosis shows the most promise. All of the summarizers would perform better if refined for oddities of twitter instead of just their off the self version.

References

[1] Duan, Y., Chen, Z., Wei, F., Zhou, M., and Shum, H.-Y. Twitter topic summarization by ranking tweets using social influence and content quality. *Proceedings of the 24th . . . 4*, 96 (2012), 763–780.

[2] Edwards, J. Leaked twitter api data shows the number of tweets is in serious decline. *Business Insider*, Feb 2 (2016).

[3] Erkan, G., and Radev, D. R. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research 22* (2004), 457–479.

[4] Ganesan, K., Zhai, C., and Han, J. Opinosis : A Graph-Based Approach to Abstractive Summarization of Highly Redundant Opinions. *Proceedings of the 23rd International Conference on Computational Linguistics*, August (2010), 340–348.

[5] Gimpel, K., Schneider, N., O'Connor, B., and Das, D. Part-of-speech tagging for twitter: Annotation, features, and experiments. *Proceedings of the 49th* (2011).

[6] He, R., Liu, Y., Yu, G., Tang, J., Hu, Q., and Dang, J. Twitter summarization with social-temporal context, 2016.

[7] Inouye, D., and Kalita+, J. K. Comparing Twitter Summarization Algorithms for Multiple Post Summaries. *2011 IEEE Third Int'l Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third Int'l Conference on Social Computing* (2011), 298–306.

[8] Nenkova, A., and Passonneau, R. Evaluating content selection in summarization: The pyramid method. *Proceedings of HLT-NAACL 2004* (2004), 145–152.

[9] Nenkova, A., and Vanderwende, L. The impact of frequency on summarization. *Microsoft Research Redmond Washington Tech Rep MSRTR2005101* (2005).

[10] Park, J. Argument Mining in Twitter : Recognizing Premise Tweets for Claim Hashtags. *Unpublished, private correspondence* (2016), 1–9.

[11] Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., and Zhang, Z. {MEAD} — {A} platform for multidocument multilingual text summarization. *Conference on Language Resources and Evaluation (LREC)* (2004), 699–702.

[12] Sharifi, B., Hutton, M.-A., and Kalita, J. K. Experiments in Microblog Summarization. *Social Computing (SocialCom), 2010 IEEE Second International Conference on* (2010).

[13] Wang, L., and Ling, W. Neural Network-Based Abstract Generation for Opinions and Arguments. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (2016), 47–57.

[14] Xu, W., Grishman, R., Meyers, A., and Ritter, A. A Preliminary Study of Tweet Summarization using Information Extraction. *Naacl 2013*, Lasm (2013), 20–29.

[15] Yang, X., and Ruan, Y. A Framework for Summarizing and Analyzing Twitter Feeds. *Kdd12*, Figure 1 (2012), 370–378.