Simple Dynamic Emission Strategies for Microblog Filtering

Luchen Tan, Adam Roegiest, Charles L. A. Clarke, and Jimmy Lin
David R. Cheriton School of Computer Science
University of Waterloo, Ontario, Canada
{luchen.tan, aroegies, jimmylin}@uwaterloo.ca, claclark@gmail.com

ABSTRACT

Push notifications from social media provide a method to keep up-to-date on topics of personal interest. To be effective, notifications must achieve a balance between pushing too much and pushing too little. Push too little and the user misses important updates; push too much and the user is overwhelmed by unwanted information. Using data from the TREC 2015 Microblog track, we explore simple dynamic emission strategies for microblog push notifications. The key to effective notifications lies in establishing and maintaining appropriate thresholds for pushing updates. We explore and evaluate multiple threshold setting strategies, including purely static thresholds, dynamic thresholds without user feedback, and dynamic thresholds with daily feedback. Our best technique takes advantage of daily feedback in a simple yet effective manner, achieving the best known result reported in the literature to date.

1. INTRODUCTION

Filtering topical events from social media streams, such as Twitter, provides a means for users to keep up-to-date on topics of interest to them. If care is taken, these updates may even be pushed directly to the user through notifications on mobile devices or desktops. However, for push notifications to be successful, the user must be given means to control the frequency and volume of updates, avoiding indiscriminate and unwanted notifications. This frequency and volume depends both on the interests of the user, with topics of greater interest updated in greater volume, and on the topics themselves, with some topics naturally receiving updates more frequently than others.

We might update a user interested in polls for the 2016 U.S. presidential elections many times a day during the election cycle itself, but with updates stopping altogether after November 8. We might update a user interested in California residential water restrictions only when these restrictions change, perhaps a few times a year, but interest in the topic might persist for many years, as long as the user is a resident of the state. For causal sports and entertainment topics (cricket or the Kardashians), a user may not desire more than a few of the most significant updates per day, regardless of events taking place. For topics of great personal importance (a tornado warning or friend's wedding), we might push all updates.

The TREC Microblog tracks provide an experimental forum for research groups working in this area. In 2015, the track [4] required participating groups to monitor the live "spritzer" stream provided by Twitter over a period of ten days in July, selecting tweets relevant to 225 pre-defined interest profiles, each expressed through statements modeled after TREC *ad hoc* topics. Each tweet returned by participating systems was accompanied by the clock time at which the system would have pushed it to the user. This information was then used by the track organizers to compute various official evaluation measures for a subset of 51 interest profiles. The evaluation measures considered both the relevance of selected tweets and the time at which they were putatively pushed. In addition, the measures accounted for retweets, near-duplicate tweets, and other redundancies, reflecting an expectation that a user would not want to receive notifications about the same thing over and over again.

Along with this mobile notification scenario (called "scenario A") the evaluation supported a daily digest scenario (called "scenario B"). At the end of each of the ten days, participating systems returned a ranked list of tweets from that day, just as if a user was sent a summary of the day's events by email. Standard evaluation measures for ranked retrieval may be applied, provided that they also appropriately account for redundant content.

To achieve good results for scenario A, systems must successfully address three requirements implicit in the task:

1. A requirement to score individual tweets with respect to relevance. As a simplification, the evaluation focused on topical relevance. Social signals, such as the prominence of the source or its connection to the user, were not considered, so that the relevance of a tweet is primarily determined by its content.
2. A requirement for novelty, so that the system does not push redundant information. This requirement was operationalized by the assessors considering tweets chronologically: if two textually similar tweets arrive at different times, the later tweet is considered redundant if it does not contain substantive information beyond that found in the earlier tweet [9]. Again, only tweet content is considered, so that a duplicate tweet from a more prominent or authoritative source would still be considered redundant.

SIGIR '16, July 17 - 21, 2016, Pisa, Italy

ISBN 978-1-4503-4069-4/16/07…$15.00

DOI: http://dx.doi.org/10.1145/2911451.2914704

3. A requirement to avoid pushing non-relevant information altogether. The evaluation measures for scenario A explicitly rewarded systems that avoided pushing non-relevant information. Thus, appropriate selection of thresholds was critical to success. In some cases, the ideal response for a given day was to push nothing, and the "empty" strategy of never pushing anything formed a challenging baseline that many systems failed to beat.

Since the first two requirements are inherent in any ranking task, we focus on the third requirement. After demonstrating the impact of ignoring the third requirement, we consider various strategies for establishing and maintaining thresholds for pushing tweets. We examine strategies under two assumptions: with and without user feedback. When feedback is not available, thresholds are established from historical information. When feedback is provided, it is limited to once-per-day judgments based on the scenario B output. Of particular importance is the establishment of a global score threshold, applied across all topics in the absence of feedback. Our best technique takes advantage of daily feedback in a simple yet effective manner, achieving the best known result reported in the literature to date.

2. RELATED WORK

Filtering has been a longstanding subject for information retrieval research. Earlier work on Topic Detection and Tracking (TDT) investigated algorithms for the discovery of new topics and maintaining awareness when these topics reappear in newswire streams or broadcast news [2]. Experimental results from TREC from the mid-1990s to early-2000s indicated that simple IR techniques can achieve high quality results in TDT domains [1, 3, 10, 11]. For example, Allan et al. [1] used the most common words from 1 to 16 training stories to generate short queries which were compared to documents using TF-IDF based similarity.

More recently, researchers have examined filtering and tracking problems in the context of social media. However, Twitter introduces several issues not present in previous topic tracking tasks, especially in relation to tweets. In particular, tweets have a maximum length of 140 characters and this length limitation implies that meaningful words rarely occur more than once, suggesting that TF-IDF weighting schemes may be less useful. Lin et al. [5] investigated the use of four language modeling smoothing techniques to filter tweets, mitigating issues with sparse terms, i.e., the zero-probability problem. Zhao and Tajima [12] framed a retweet recommendation problem as a multiple-choice secretary problem. They examined Twitter "portal accounts", which retweet selected tweets for their followers, and considered a number of strategies for tweet selection. They proposed and compared a number of online and near-online decision methods, including a history-based threshold algorithm, a stochastic threshold algorithm, a time-interval algorithm, and an "every k-tweets" algorithm.

3. TREC 2015 MICROBLOG TRACK

Topics (called "interest profiles") provided to participants in the TREC 2015 Microblog track included a short query-like description of the information need, called the "title" in TREC parlance. For example, topic MB235 has the title "California residential water restrictions". Other fields in a topic elaborate on the information need, providing a more complete indication of what is and is not relevant. While track participants were permitted to use these other fields for filtering purposes, we focused on the more realistic task in which the user provides only the short query.

Track participants filtered the so-called Twitter "spritzer" stream over ten days in July 2015, selecting those relevant to 225 pre-defined interest profiles and recording their tweet ids for submission. In addition to the tweet ids, the push time was recorded for each tweet, indicating the time the system decided to push the notification. After all experimental runs were submitted to the track organizers, 51 of the interest profiles were selected for judging. Tweets were pooled and judged on a three-point scale. In evaluating a run, a tweet was considered redundant, providing no gain, if it did not contain substantial new information not found in previously pushed tweets [9].

The primary evaluation measure for scenario A is expected latency-discounted gain:

$$\text{ELG} = \frac{1}{N}\sum_{i=1}^{N} G_i D_i \quad (1)$$

For N pushed tweets, G_i is the gain associated with tweet i based on the assigned relevance grade, after adjusting for redundancy. The temporal discount applied to tweet i is $D_i = \max(0, (100 - L_i)/100)$, where L_i is the latency in minutes between the time the tweet appears in the stream and the time the system decides to push it. ELG is computed on a daily basis, over the tweets pushed by a system that day, with the system's overall score averaged across all topics and all days.

ELG has an interesting discontinuity when a system decides not to push anything. For some topics on some days, when no relevant tweets appear in the stream, this is the correct action. On such days, a system pushing *any* tweet receives a score of zero (since none can possibly be relevant). To reward systems that push nothing on such days, the value of ELG is defined to be one. On the other hand, for systems that push nothing on days when relevant tweets appear in the stream, the value of ELG is defined to be zero. Since no relevant tweets appeared for many topics on many days, the "empty" strategy of never pushing anything receives a non-zero ELG score. Indeed, this strategy forms a challenging baseline which many participating systems failed to beat.

4. BASELINE SYSTEM

The system used for the experiments reported in this paper was deployed for the TREC 2015 Microblog track [7], with the design of the system based on substantial pilot experiments conducted prior to the actual evaluation. The system achieved the best ELG score across the 32 automatic runs submitted by 14 participating groups.

All experiments described in this paper are from *post hoc* runs using a replay mechanism over tweet data captured during the evaluation period. Following the TREC evaluation, we performed error analysis to understand the contributions of various system components, and have distilled our algorithm into simple strategies to address the three requirements listed in the introduction, detailed below.

Relevance. Although we recognize that social signals and other non-content features are important for relevance, as a first step we only consider tweet content. For pre-processing, we apply language detection tools to eliminate non-English

tweets, tokenize the tweets using Twokenize,[1] and then apply some simple tweet quality heuristics, e.g., eliminating tweets containing less than five tokens.

For matching against the tweet stream, titles were tokenized by splitting on space and punctuation, and stopwords were removed. Since relevant tweets may not contain all, or even any, of the title terms, we employed a pseudo-relevance feedback step to expand the title terms with 5 hashtags and 10 other terms. This was accomplished by querying Twitter's own search engine with title terms at the beginning of each day and extracting the top terms using pointwise KL-divergence [6, 8].

For relevance scoring, we applied a simple matching formula developed through pilot testing. Given the short length of tweets, many "standard" features such as query term frequency appear to have limited value. We found that (binary) query term occurrence appears to be the key feature, with the occurrence of title terms having greater importance than the occurrence of expansion terms. To achieve a balance, we score tweet relevance as follows:

$$(w_t \cdot N_t + w_e \cdot N_e) \cdot \frac{N_t}{|T|} \quad (2)$$

where N_t is the number of title terms that appear in the tweet, N_e is the number of expansion terms that appear in the tweet, and $|T|$ is the number of title terms. Based on pilot experiments, weights were set to $w_t = 3$ and $w_e = 1$.

Novelty. We de-duplicated tweets by computing unigram overlap between each new tweet and the tweets previously pushed for a given topic across all days. Tweets with 60% or more overlap were discarded and not further considered. As with the other parameters, we based this overlap setting on pilot experiments conducted prior to the actual TREC 2015 evaluation.

Thresholding. Thresholds for pushing tweets were based on the relevance score in Equation (2). Each day, a threshold is selected, and only tweets with scores greater than or equal to the threshold are considered for delivery. In addition, the evaluation placed a limit of $k = 10$ on the number of tweets that could be pushed each day. Once k tweets are pushed, all further system output is ignored.

Because of its simplicity, Equation (2) has the interesting property that reasonable thresholds are the same across all queries. Our simplest thresholding strategy is thus to select a single static global threshold (GT) across all queries and days. A simple dynamic strategy (without feedback) is to consider the top k from the previous day, selecting as a threshold the score of the kth tweet. However, under this strategy, we do not lower the threshold below a global minimum, selecting as a threshold $\max(GT, \text{top-}k \text{ yesterday})$.

For our TREC 2015 experiments, we used a global threshold of $GT = 5$ (determined based on previous pilot experiments). To better understand the impact of this threshold, Figure 1 shows the effectiveness of our system for these two strategies across a range of global threshold values. The baseline for this plot is the effectiveness of the empty strategy, i.e., never pushing anything, with low threshold values underperforming it. A global threshold of $GT = 6$ slightly outperforms our default threshold of $GT = 5$, which we retain for the remainder of this paper. We find that our simple strategy of dynamically adjusting the threshold without feedback is not effective—the difference compared to using a simple global threshold is negligible for most settings.

[1] http://www.ark.cs.cmu.edu/TweetNLP/

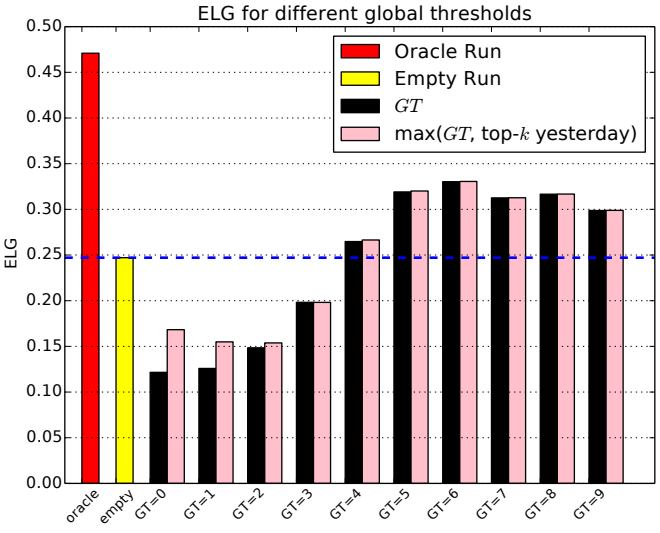

Figure 1: ELG for different global thresholds, the oracle run, and the empty run.

The oracle run in Figure 1 represents the ELG achievable if we made an ideal selection of the global threshold for each topic at the beginning of each day. Clearly, substantial improvements can be achieved through better threshold selection. In the next section, we explore a dynamic emission strategy that uses feedback from each day's digest (scenario B) to select the threshold for the following day.

5. FEEDBACK STRATEGIES

Under scenario B, participants submitted a ranked list of tweets for each topic for each day, providing a daily digest of events for the hypothetical user. Building on the baseline system described above, we use each day's digest to provide relevance feedback for determining thresholds for the following day. While in reality each system saved a ranked list during each day of the evaluation period for later submission, and thus system results were not judged until the evaluation concluded, here we assume that relevance information is provided at the end of each day and available for immediate use. We imagine a user interacting with the results once per day, providing feedback as a way of adjusting the filter for the next day. While in practice this daily interaction might be too onerous, the results provide a sense of what gains could be achieved with ongoing feedback.

For the feedback strategies described below, we used our official scenario B run submitted to TREC 2015 as the daily digest—this ensured that all tweets have relevance judgments. Our method of relevance scoring, Equation (2), often assigns the same score to several tweets. We use the term "score block" to denote a set of tweets with the same score. Accordingly, we say that for a score s_i, it has a corresponding score block SB_i. To determine a dynamic threshold using relevance feedback, we begin by combining all feedback received into a single list of score blocks. At the end of day one for a particular topic, we have 10 tweets worth of feedback, on day two, 20 tweets worth, and so on.

There are two edge cases to consider. In the first case, if there are no relevant tweets in any score block, we take

Baseline Strategies	ELG
$GT = 5$	0.3191
$GT = 6$	0.3303 ($p = 0.3819$)
Feedback Strategies	ELG
avg_gain	0.3257 ($p = 0.5664$)
weighted_avg_gain	0.3510 ($p = 0.0004$)
weighted_avg_gain+r_score	0.3678 ($p \sim 0.0000$)

Table 1: Effectiveness of various emission strategies; p-values are generated from a paired sign test with $GT = 5$.

the maximum of the global threshold (GT) and the highest score seen so far plus w_t, the weight assigned to a title term match. In the second case, if all tweets are relevant, we take the minimum score seen so far.

If we have a mix of relevant and non-relevant tweets, we first compute the average gain for each score as follows:

$$\text{avg\_gain}(s_i) = \frac{\sum_{s_j \geq s_i} \sum_{t \in SB_j} \text{gain}(t)}{\sum_{s_j \geq s_i} |SB_j|}$$

where gain(t) comes from the relevance judgments. We then weight each score's average gain by the proportion of relevant content provided by that score (i.e., the precision of that score):

$$\text{weight}(s_i) = \frac{|\{t | t \in SB_i \wedge \text{gain}(t) > 0\}|}{\sum_{s_j} |SB_j|}$$

$$\text{weighted\_avg\_gain}(s_i) = \text{avg\_gain}(s_i) \cdot \text{weight}(s_i)$$

While selecting a threshold that maximizes average gain doesn't perform particularly well, selecting a threshold that maximizes the weighted average gain significantly improves ELG (see Table 1). However, the weighted average gain method still returns too much non-relevant content. To further improve effectiveness, we use the relevance information to adjust the threshold based on the ratio between relevant and non-relevant content:

$$\text{r\_score}(s_i) = \frac{\sum_{s_j \geq s_i} |\{t | t \in SB_j \wedge \text{gain}(t) = 0\}|}{\sum_{s_j \geq s_i} |\{t | t \in SB_j \wedge \text{gain}(t) > 0\}|}$$

For use as a threshold, a score's r_score must be less than some cutoff, σ. We can vary σ to be more or less permissive of non-relevant tweets: via a coarse-grained parameter sweep, we find that $\sigma = 1.75$ achieves substantially improved results on ELG (see Table 1). Note that if no score achieves an r_score less than or equal to σ, we set the threshold for the next day to be the maximum of the global threshold (GT) and the highest score seen so far, across all days.

6. CONCLUSION

Simple techniques for content matching and novelty can achieve good effectiveness for microblog filtering, provided that care is taken to set appropriate thresholds to avoid pushing non-relevant information. Referring back to Figure 1, our most effective technique achieves an ELG of 0.3678, which is still substantially below what might be achieved if the optimal threshold could be determined for each topic at the beginning of each day, i.e., the oracle with an ELG of 0.4709. However, we significantly improve upon our already highly-effective baseline (already the best automatic run at TREC 2015). In fact, our technique achieves the best known result reported in the literature to date, including manual runs. Our experiments highlight the importance of proper threshold setting, and demonstrate that systems can automatically set appropriate thresholds using simple yet effective feedback techniques. While dynamic thresholds can be set from the tweets of previous days without feedback, such a strategy provides little value, at least with the simple technique we tried.

Our experiments show that dynamic thresholding using feedback has the potential to produce substantial and significant gains. While we have explored only one approach to this idea, through end-of-day relevance judgments, we can imagine more realistic interfaces, which might for example allow incremental judgments as tweets are pushed. In addition, we hope to incorporate social signals and other non-content features into the relevance and novelty components of our system, with the goal of retaining our simple approach to thresholding, while improving overall effectiveness.

Acknowledgments. This work was supported in part by the U.S. National Science Foundation under awards IIS-1218043 and CNS-1405688 and the Natural Sciences and Engineering Research Council of Canada (NSERC). Any opinions, findings, conclusions, or recommendations expressed are those of the authors and do not necessarily reflect the views of the sponsors.

7. REFERENCES

[1] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. *SIGIR*, 1998.

[2] J. G. Fiscus and G. R. Doddington. Topic detection and tracking evaluation overview. *Topic Detection and Tracking*. Kluwer, Norwell, MA, 2002.

[3] L. S. Larkey, F. Feng, M. Connell, and V. Lavrenko. Language-specific models in multilingual topic tracking. *SIGIR*, 2004.

[4] J. Lin, M. Efron, G. Sherman, Y. Wang, and E. M. Voorhees. Overview of the TREC-2015 Microblog track. *TREC*, 2015.

[5] J. Lin, R. Snow, and W. Morgan. Smoothing techniques for adaptive online language models: Topic tracking in tweet streams. *SIGKDD*, 2011.

[6] L. Tan and C. L. A. Clarke. Succinct queries for linking and tracking news in social media. *CIKM*, 2014.

[7] L. Tan, A. Roegiest, and C. L. A. Clarke. University of Waterloo at TREC 2015 Microblog Track. *TREC*, 2015.

[8] T. Tomokiyo and M. Hurst. A language model approach to keyphrase extraction. *ACL Workshop on Multiword Expressions*, 2003.

[9] Y. Wang, G. Sherman, J. Lin, and M. Efron. Assessor differences and user preferences in tweet timeline generation. *SIGIR*, 2015.

[10] J. Yamron, I. Carp, L. Gillick, S. Lowe, and P. Van Mulbregt. Topic tracking in a news stream. *DARPA Broadcast News Workshop*, 1999.

[11] Y. Yang, T. Ault, T. Pierce, and C. W. Lattimer. Improving text categorization methods for event tracking. *SIGIR*, 2000.

[12] X. Zhao and K. Tajima. Online retweet recommendation with item count limits. *Web Intelligence and Intelligent Agent Technologies*, 2014.