A Novel Regularizer for Temporally Stable Learning with an Application to Twitter Topic Classification

Yakun Wang[†‡]*, Ga Wu[†], Mohamed Reda Bouadjenek[†], Scott Sanner[†], Sen Su[‡], and Zhongbao Zhang[‡]
[†]The University of Toronto, Department of Mechanical and Industrial Engineering
wuga@mie.utoronto.ca, mrb@mie.utoronto.ca, ssanner@mie.utoronto.ca
[‡]Beijing University of Posts and Telecommunications
wangyakun@bupt.edu.cn, susen@bupt.edu.cn, zhongbaozb@bupt.edu.cn

Abstract

Supervised topic classifiers for Twitter and other media sources are important in a variety of long-term topic tracking tasks. Unfortunately, over long periods of time, features that are predictive during the training period may prove ephemeral and fail to generalize to prediction at future times. For example, if we trained a classifier to identify tweets concerning the topic of "Celebrity Death", individual celebrity names and terms associated with these celebrities such as "Nelson Mandela" or "South Africa" would prove to be temporally unstable since they would not generalize over long periods of time; in contrast, terms like "RIP" (rest in peace) would prove to be temporally stable predictors of this topic over long periods of time. In this paper, we aim to design supervised learning methods for Twitter topic classifiers that are capable of automatically downweighting temporally unstable features to improve future generalization. To do this, we first begin with an oracular approach that chooses temporally stable features based on knowledge of both train and test data labels. We then search for feature metrics evaluated on only the training data that are capable of recovering the temporally stable features identified by our oracular definition. We next embed the top-performing metric as a temporal stability regularizer in logistic regression with the important property that the overall training objective retains convexity, hence enabling a globally optimal solution. Finally, we train our topic classifiers on 6 Twitter topics over roughly one year of data and evaluate on the following year of data, showing that logistic regression with our temporal stability regularizer generally outperforms logistic regression without such regularization across the full precision-recall continuum. Overall, these results establish a novel regularizer for training long-term temporally stable topic classifiers for Twitter and beyond.

1 Introduction

Twitter represents a massively distributed information source over topics ranging from social and political events to entertainment and sports news [1, 2]. While recent work has suggested this content can be filtered for the personalized interests of individual users by training standard classifiers as topical filters [3, 4, 5, 2, 6], there remain many open questions about the long-term accuracy of such classification-based filtering approaches. Specifically, over long periods of time, features that are predictive during the training period may prove ephemeral and fail to generalize to prediction at future times.

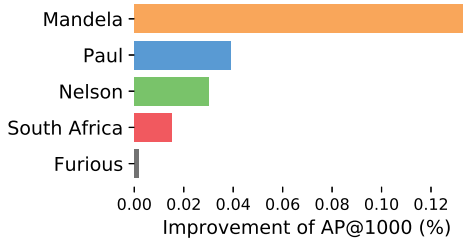


Figure 1: From top to bottom, the effect of successively removing the most temporally unstable features on the average precision of the topic classifier "Celebrity Death". Except "South Africa" which is a location feature, all shown features are term features.

In this work, we argue that *temporally unstable* features may significantly impact the quality of any classifier trained and evaluated over longitudinal data. Moreover, given the trend-driven nature of social networks and in particular Twitter, we argue these data sources are especially susceptible to this problem of temporal feature instability. In this paper, we aim to study the impact of temporally stable and unstable features on topic classification in Twitter and leverage the insights gained to explore novel regularization approaches for learning temporally stable classifiers that are robust to features which do not generalize well over time.

To provide insight into the stability of features, we refer to Figure 1. Here, we assume the use of different feature types such as *terms* (simple word tokens), *hashtags* and *locations* for the topical classification of English Tweets over

*This work has been primarily completed while the author was a visiting student at The University of Toronto.

two years spanning 2013 and 2014. Here we have identified and ranked features based on a measure of temporal instability that we develop later in this paper; we show the effect of the successive removal of these features on the average precision of a topic classifier related to "Celebrity Death". For example, the term "Mandela" has been identified as the most unstable feature, and its removal from the training set boosts the average precision by over 0.13%. Indeed, "Mandela" is clearly a feature that is related to a particular event (i.e., the death of the president of South Africa, which happened on December 5, 2013) that does not generalize well to the prediction of future events under the topic of Celebrity Death. In contrast, a term like "RIP" (for "rest in peace") would generalize well to future events for the Celebrity Death topic. Here we refer to "Mandela" as a temporally unstable feature, while "RIP" would be temporally stable.

As the paper proceeds, we will observe three notable trends: (i) first, the removal of unstable features improves classifier performance, (ii) second, unstable features can be of any type – in this example terms and locations, and (iii) third, unstable features are often contextualized to specific events. Although each removal may only provide a small boost, the cumulative improvement of removing multiple temporally unstable features may be substantial.

In this paper we aim to answer the following two research questions: (RQ1) *how can we identify temporally unstable features*? And (RQ2) *how can we design a learning algorithm that automatically downweights the influence of temporally unstable features*? To address RQ1, we employ 6 different metrics to measure the instability of features and evaluate these features using a purpose-built ground truth dataset. To address RQ2, we propose Temporal Stability Aware Logistic Regression (TSALR), where we introduce a novel reguarizer based on the analysis in RQ1. TSALR is able to automatically reduce the weight of temporally unstable features during classifier training.

In summary, we make the following contributions:

1. We propose to study the temporally stability of features in Twitter. Solving this problem is critical for building a long-term robust classifier for Twitter. To the best of our knowledge, this work is the first study of temporal feature stability in a multi-year dataset.

2. We present an empirical analysis of feature stability using different metrics on 40 TB of uncompressed data from Twitter spanning 2013-2014 with 6 labeled topics. We found that temporally unstable features are usually event related, and we show that removing temporally unstable features improves the performance of the classification task.

3. We introduce a novel Temporal Stability Aware Logistic Regression (TSALR) method using a novel temporal stability regularizer to downweight temporally unstable features during the learning process. We empirically demonstrate the superiority of TSALR with respect to standard Logistic Regression.

Table 1: Feature Statistics of our $829,026,458$ tweet corpus.

#Unique Features

From	Hashtag	Mention	Location	Term
95,547,198	11,183,410	411,341,569	58,601	20,234,728

Feature Usage in #Tweets

Feature	Max	Avg	Median	Most frequent
From	10,196	8.67	2	running_status
Hashtag	1,653,159	13.91	1	#retweet
Mention	6,291	1.26	1	tweet_all_time
Location	10,848,224	9,562.34	130	london
Term	241,896,559	492.37	1	rt

Feature Usage by #Users

Hashtag	592,363	10.08	1	#retweet
Mention	26,293	5.44	1	dimensionist
Location	739,120	641.5	2	london
Term	1,799,385	6,616.65	1	rt

Feature Using #Hashtags

From	18,167	2	0	daily_astrodata
Location	2,440,969	1,837.79	21	uk

2 Dataset Description

We crawled Twitter data using the Twitter Streaming API for two years spanning 2013 and 2014. We collected more than 40 TB of uncompressed data, which contains a total of 829,026,458 English tweets. In the context of Twitter, we consider five feature types for each tweet. Each tweet has a From feature (i.e., the person who tweeted it), a possible Location (i.e., a string provided as meta-data), and a time stamp when it was posted. A tweet can also contain one or more of the following: Hashtag (i.e., a topical keyword specified using the # sign), Mention (i.e., a Twitter username reference using the @ sign), Term (i.e., any non-hashtag and non-mention unigrams). We provide more detailed statistics about each feature in Table 1. For example, there are over 11 million unique hashtags, the most frequent unique hashtag occurred in over 1.6 million tweets, a hashtag has been used on average by 10.08 unique users, and authors (*From* users) have made a median value of 2 tweets.

A critical bottleneck for learning targeted topical social classifiers is to achieve sufficient supervised content labeling. Following the approach of [2, 3], we manually curated a broad thematic range of 6 topics shown in the top row of Table 2 by annotating hashtag sets H^t for each topic $t \in T$. We used 4 independent annotators to query the Twitter search API to identify candidate hashtags for each topic, requir-

Table 2: Test/Train Hashtag samples and statistics.

Topics	Iran	Human Disaster	Celebrity Death	Social Issues	Natural Disaster	Health
#TrainHashtags	12	49	28	31	31	52
#TestHashtags	5	29	16	19	18	33
#TopicalTweets	8,762	408,304	163,890	230,058	42,987	210,217
Sample Hashtags	#irandeal	#gazaunderattack	#robinwilliams	#policebrutality	#earthquake	#ebola
	#iranfreedom	#childrenofsyria	#ripmandela	#michaelbrown	#storm	#virus
	#irantalk	#iraqwar	#ripjoanrivers	#justice4all	#tsunami	#vaccine
	#rouhani	#bombthreat	#mandela	#freetheweed	#aboods	#chickenpox
	#nuclearpower	#isis	#paulwalker	#newnjgunlaw	#hurricanekatrina	#theplague

ing an inner-annotator agreement of 3 annotators to permit a hashtag to be assigned to a topic set.

To split our dataset into train, validation and test sets, we split H^t into three disjoint sets H^t_{train}, H^t_{val} and H^t_{test} according to two time stamps t^{val}_{split} and t^{test}_{split} for each topic and the first usage timestamp h_{time*} of each hashtag $h \in H^t$. In short, all hashtags $h \in H^t$ with $h_{time*} < t^{val}_{split}$ are used to generate positive labels in the training data, those with $h_{time*} \geq t^{test}_{split}$ are used for positive labels in the test data and the remainder are used for validation data. The purpose of this design is to ensure hyperparameters are tuned to encourage generalization to unseen topical hashtags that did not occur during training. We remark that a classifier that simply memorizes training hashtags will fail to correctly classify the validation data except in cases where a tweet contains both a training and validation hashtag. We provide detailed statistics of Hashtags for each topic in Table 2.

Given that we have a total of $538,365,507$ unique features in our Twitter corpus, it is critical to pare this down to a size amenable for efficient learning and robust to overfitting. To this end, We empirically select the top 1000 features using Pearson's chi-squared test χ^2 for each topic.

3 Temporally Stable/Unstable Feature Analysis

Events and topics that are commonly discussed on Twitter and other media sources tend to be of short-term interest. Therefore, they are often intensively discussed during a short period of time with discussion frequency decaying thereafter. Consequently, some extracted high chi-squared features may be irrelevant if they were only associated with a short-lived event. These temporally unstable features generally do not contribute to the long-term accuracy of the classifier.

While there is no single agreed-upon definition of a *temporally stable feature*, assuming we have oracular knowledge of the train and test data (i.e., we can see into the future to observe the test labels) we propose the following high-level working definitions for the purpose of this paper. We will later clarify our use of "associated" in the context of logistic regression.

DEFINITION 1. (TEMPORALLY STABLE FEATURE (TSF)) *A temporally stable feature is a measurable characteristic of the data being observed, which is associated with the topic label at all times. More precisely, if a feature x is consistently associated with the corresponding topic label in both the training data and the test data, then x is a temporally stable feature.*

By analogy, we propose the following definition for *temporal unstable features*:

DEFINITION 2. (TEMPORALLY UNSTABLE FEATURE (TUF)) *A temporally unstable feature is a measurable characteristic of the data being observed, which is only associated with the topic label at a specific time. More precisely, if a feature x is associated with the topic label in only the training data or only the test data (i.e., not both), then x is a temporally unstable feature.*

In practice, we will not have oracular knowledge of the future and hence we must attempt to to distinguish TUF from TSF solely on the basis of the training data. Hence, in the following, we attempt to identify the metric that only uses the training data and best recovers TUF and TSF features identified through our previously defined oracular definition. Following this, we then introduce a learning method with a novel temporal regularizer that embeds this best metric as a method to downweight TUFs and improve temporal stability in learning.

3.1 Temporally (Un)Stable Feature Analysis. As our goal is to build a classifier that is robust to Temporally Unstable Features, it is critical to automatically identify them from the training data only. A straightforward approach is to (i) divide the training dataset into several time windows, (ii) measure the importance of each feature in each time slice, and finally (iii) measure the stability of each feature's importance over time.

For the purpose of measuring feature importance in classification, we propose to evaluate feature weights in Logistic Regression (LR), as done commonly in the literature [7]. Briefly, the loss function to optimize for an LR-based binary

classifier can be defined as

$$L(\theta) = \frac{1}{|D|}\left[\sum_{i=1}^{|D|}\log\left(1+e^{-y^{(i)}\theta^T\mathbf{x}^{(i)}}\right)+\frac{\lambda}{2}||\theta||_2^2\right], \tag{3.1}$$

where $\mathbf{x}$ is a vector of inputs with K features $(x_1, \cdots, x_K)$, $y \in \{-1(\text{false}), 1(\text{true})\}$ is a binary label, θ is the learned vector of K weights $(\theta_1, \cdots, \theta_K)$ associated to the K features, and $D = \{(\mathbf{x}, y)\}$ is the training set of tweets (features $\mathbf{x}$ are tweet token frequencies for the top 1,000 feature tokens previously selected). The final term is a standard L2 norm regularizer included to avoid overfitting. A large positive weight value θ_k of a feature k indicates strong positive association with the corresponding topic label, whereas a large negative θ_k value indicates a strong negative association with the topic label. However, a value of θ_k that is close to 0 typically indicates a very low association of the corresponding feature with the topic label.

We extend this analysis to the temporal stability context, where we identify the strongly associated features for *different* disjoint time windows of data. The temporal LR loss function for T separate time windows is then defined as

$$L(W)=\sum_{t=1}^{T}\frac{1}{|D|_t}\left[\sum_{i=1}^{|D|_t}\log\left(1+e^{-y^{(i)}\mathbf{w}_t^T\mathbf{x}^{(i)}}\right)+\frac{\lambda}{2}||\mathbf{w}_t||_2^2\right], \tag{3.2}$$

where W is a $T \times K$ *temporal* weight matrix, with entry $w_{t,k}$ being the weight of the feature k *during* the time window t. Later, we will use $\mathbf{w}_{:,k}$ to indicate the T-dimensional vector of weights of feature k for all time windows.

Given the ability to learn time-dependent feature weights $w_{t,k}$, we propose to use different metrics to measure the temporal instability of these feature weights. The hypothesis behind each of the following metrics is that they capture some measure of a feature's weight variation over time, where higher variation is a stronger indicator of a temporally unstable feature (TUF) defined previously.

Below, we describe each six different metrics to measure the temporal instability of features:

- **Deviation Divide Mean (DDM)** is defined as the average absolute deviation of the weights of a feature k from it's mean, normalized by its mean:
$$DDM(\mathbf{w}_{:,k}) = \frac{\sum_{t=1}^{T}|w_{t,k}-\mu_k|}{T\times\mu_k}, \quad \forall k \in \{1\cdots K\}, \tag{3.3}$$
where μ_k denotes the mean value of weight vector $\mathbf{w}_{:,k}$ during T time windows.

- **Max Absolute Deviation (MAD)** is defined as the maximum absolute deviation of the weights of a feature k from its mean among T time windows:
$$MAD(\mathbf{w}_{:,k}) = max(|\mathbf{w}_{:,k}-\mu_k|), \quad \forall k \in \{1\cdots K\}. \tag{3.4}$$

- **Average Absolute Deviation (AAD)** is defined as the average absolute deviation of weights of a feature k from its mean over T time windows:
$$AAD(\mathbf{w}_{:,k}) = \frac{\sum_{t=1}^{T}|w_{t,k}-\mu_k|}{T}, \quad \forall k \in \{1\cdots K\}. \tag{3.5}$$

- **Max Divide Min (MDM)** is defined as the ratio of the maximal weight of a feature k to its minimal weight:
$$MDM(\mathbf{w}_{:,k}) = \frac{\max\{\mathbf{w}_{:,k}\}}{\min\{\mathbf{w}_{:,k}\}}, \quad \forall k \in \{1\cdots K\}. \tag{3.6}$$

- **Variance** is defined as the average squared deviations of the weights of a feature k from its mean μ_k over T time windows:
$$Variance(\mathbf{w}_{:,k}) = \frac{\sum_{t=1}^{T}(w_{t,k}-\mu_k)^2}{T}, \quad \forall k \in \{1\cdots K\}. \tag{3.7}$$

- **Z-score** is defined as the average of the ratio between the absolute deviation and standard deviation of the weights of a feature k over T time windows:
$$\text{Z-score}(\mathbf{w}_{:,k}) = \frac{1}{T}\sum_{t=1}^{T}\frac{|w_{t,k}-\mu_k|}{\sqrt{Variance(\mathbf{w}_{:,k})}}, \quad \forall k \in \{1\cdots K\}. \tag{3.8}$$

3.2 Ground Truth TSFs and TUFs. To evaluate the ability of the above described metrics to identify TUFs, we construct a ground truth set of TSFs and TUFs (recalling our previous respective oracular Definitions 1 and 2) in the following procedure: (1) We used LR to learn the importance of each feature on the training data and the test data. (2) We rank features using the weights learned by the LR in both training data and test data. (3) If a feature x_k appears in the Top-n features in both training data and test data, x_k is considered as a TSF. However, if x_k appears only in the Top-n of the training data but not test data (or vice versa), then we consider x_k as a TUF. Note that in our experiment we used $n = 100$ features, a value that we determined empirically by running LR on a validation data set with TUF features removed (for different thresholds n) and choosing the n that provided the best validation accuracy.

An overview of the top five features generated for both TSF and TUF using the above described process for our 6 topics are shown in Table 3. For example, terms like

Table 3: Top 5 Ground truth temporally stable features (TSFs) and temporally unstable features (TUFs) for each topic. Features starting with 'H_', 'M_', 'L_', and 'U_' respectively denote hashtag, mention, location and user name, and features without these marks are general terms.

Iran Nuclear Deal

TSF	TUF
H_iran	H_freethe7
Iran	H_news
Iranian	H_iranelection
H_syria	H_obama
H_iraq	president

Human Disaster

TSF	TUF
H_terrorism	H_raqqa
U_rk70534	H_iraq
M_statedept	H_speakup4syrianchildren
M_ifalasteen	ghouta
displace	H_saa

Celebrity Death

TSF	TUF
rip	Mandela
sad	Nelson
tribute	M_nelsonmandela
inspire	Avery
angel	uncle

Social Issue

TSF	TUF
police	abuse
abort	cost
black	H_nj2as
christy	veto
kill	M_hindarifka

Natural Disaster

TSF	TUF
M_weatherchannel	M_jonleebrody
H_smem	H_floodph
H_flooding	H_lightning
quake	M_twc_hurricane
M_usgs	H_phillipines

Health

TSF	TUF
M_who	H_h7n9
H_nurses	H_globalhealth
outbreak	H_worldaidsday
alert	H_pregnancy
virus	allergic

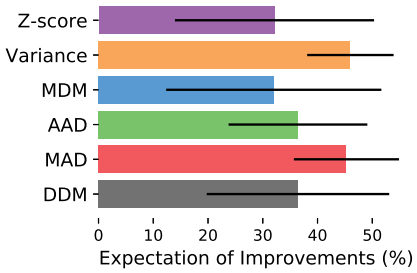


Figure 2: Average improvement percentage when removing the top 100 features identified by each metric over all topics. 95% confidence intervals are shown.

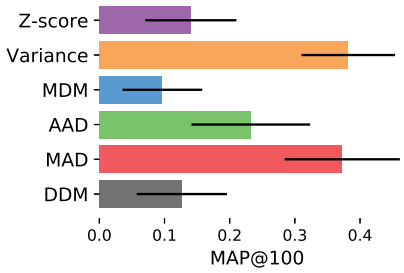


Figure 3: MAP for different metrics over all topics. 95% confidence intervals are shown.

"nelson" and "mandela" have been identified as being TUF for the Celebrity Death topic; this is quite reasonable as most tweets mentioning the death of the former South African president Nelson Mandela were generated during a short period of time (mainly during December 2013). In general, we observed that for the topic of Celebrity Death topic, proper nouns (e.g., names) tend to be TUF as they fail to generalize to new celebrity death events. In contrast, TSF are usually general, not proper nouns such as "sad" and "inspire", which are more appropriate features for building a temporally stable classifier for this topic.

3.3 Metrics Evaluation. Now, we proceed to evaluate the ability of the different training data metrics we defined in Section 3.1 to identify our oracular ground truth TUFs defined in Section 3.2. Here we measure Mean Average Precision for the top 100 ranks (MAP@100) with the mean taken over the 6 topics – metrics which rank more of the ground TUFs in the top-ranks will achieve a higher MAP@100. The results obtained for this analysis are reported in Figures 3. In short, these results suggest that Variance and MAD perform best, followed by AAD, with their performances statistically indistinguishable.

Next, we evaluate whether or not the removal of the top TUFs identified by the 6 metrics can improve the performance of the learning algorithm. In order to answer this question, for each metric, we measure the average accuracy improvement of the classifier if the top-100 unstable features identified by that metric are removed. The results are shown in Figure 2, where we note that the best metric is again Variance, though this is not statistically significant compared to other methods given the large confidence intervals. The obtained results here are consistent with the results of MAP in Figure 3, where the Variance metric also had the highest

mean performance. Since Variance also has convenient convexity properties that allow global optimization, we select it as a temporal stability regularizer for logistic regression that we detail next.

4 TSALR: Temporal Stability Aware Logistic Regression

Motivated by the ability of Variance to identify TUFs, we now move beyond simple feature selection to describe a new method that automatically embeds the Variance as a temporal stability regularizer in logistic regression. Specifically, this approach uses feature Variance to encourage lower weights for unstable features without entirely removing them from the classifier as a hard feature selection approach would do.

4.1 Model Description. As described in Section 3.1, Logistic Regression gives a high weight to important features during the learning process of a classification. Thus, we leverage this property and extend it to develop a Temporal Stability Aware Logistic Regression (TSALR). TSALR *combines* the global LR model in Equation 3.1 with the temporal LR models in Equation 3.2 and introduces a temporal regularization term that causes a *global* feature weight θ_k to be downweighted when it deviates significantly from the time-dependent weights $\mathbf{w}_{:,k}$.

To understand the rationale for this behavior, recall that according to the definition of TSF, the temporal weight $w_{t,k}$ of a temporally stable feature k in a time window t should be consistent with its global weight θ. On the other hand, based on the definition of TUF, the temporal weight $w_{t,k}$ of a temporally unstable feature k in a time window t should significantly deviate from θ_k. In addition, we previously observed in Section 3.3 that the "Variance" metric performs among the best methods for TUF identification. Hence we add a regularization term that penalizes Variance of $\mathbf{w}_{:,k}$ from the effective mean value θ_k, the logistic regression learner can only reduce this penalty by downweighting all $\mathbf{w}_{:,k}$ and θ_k for feature k as desired for TUF features. Formally, the objective to optimize in TSALR is defined as:

(4.9)
$$\begin{aligned} L(\boldsymbol{\theta}, W) =& \frac{1}{|D|}\sum_{i=1}^{|D|}\log\left(1+e^{-y^{(i)}\boldsymbol{\theta}^T\mathbf{x}^{(i)}}\right) \\ &+\sum_{t=1}^{T}\left[\frac{1}{|D|_t}\sum_{i=1}^{|D|_t}\log\left(1+e^{-y^{(i)}\mathbf{w}_t^T\mathbf{x}^{(i)}}\right)\right] \\ &+\lambda_1\sum_{t=1}^{T}||\boldsymbol{\theta}-\mathbf{w}_t||_2^2+\lambda_2(||\boldsymbol{\theta}||_2^2+\sum_{t=1}^{T}||\mathbf{w}_t||_2^2) \end{aligned}$$

There are four components in the proposed model. The first part is the global Logistic Regression model that is parameterized by $\boldsymbol{\theta}$. The second part is a set of temporal LR models that parameterized by $\mathbf{w}_t$ associated to each month t (i.e., each temporal LR model is fitted to data for disjoint subsets of time in the training data). The third part is a mutual constraint that regularizes the global weights $\boldsymbol{\theta}$ and temporal $\mathbf{w}_t$ to be similar. This part models our *novel temporal regularizer* and we critically note that when weights for a feature tend to be unstable over different periods of time, only a small (or near-zero) weight for the feature in each $\mathbf{w}_t$ and $\boldsymbol{\theta}$ will prevent a large penalty from this regularizer. Hence, temporally unstable feature weights are inherently downweighted. In contrast, when weights are stable across all time subsets, the feature weight is not penalized and may be large. The last part is an L_2 regularizer for all parameters in this model to prevent overfitting. λ_1 and λ_2 are hyperparameters that are tuned using the validation dataset. Note that we only use the global model to do the prediction after training. In other words, the time-dependent models are only used to temporarily regularize the global model used for final prediction.

We remark that we jointly train $\boldsymbol{\theta}$ and $\mathbf{w}_t$ rather than pretraining and freezing $\mathbf{w}_t$. We do this since our aim is *not* to penalize $\boldsymbol{\theta}$ for deviating from $\mathbf{w}_t$, which would just encourage $\boldsymbol{\theta}$ to settle at the average of the $\mathbf{w}_t$'s. Rather, the intent is to allow the joint model over all time windows to find a compromise $\boldsymbol{\theta}$ and $\mathbf{w}_t$ that balance improvements in likelihood with penalties from variance.

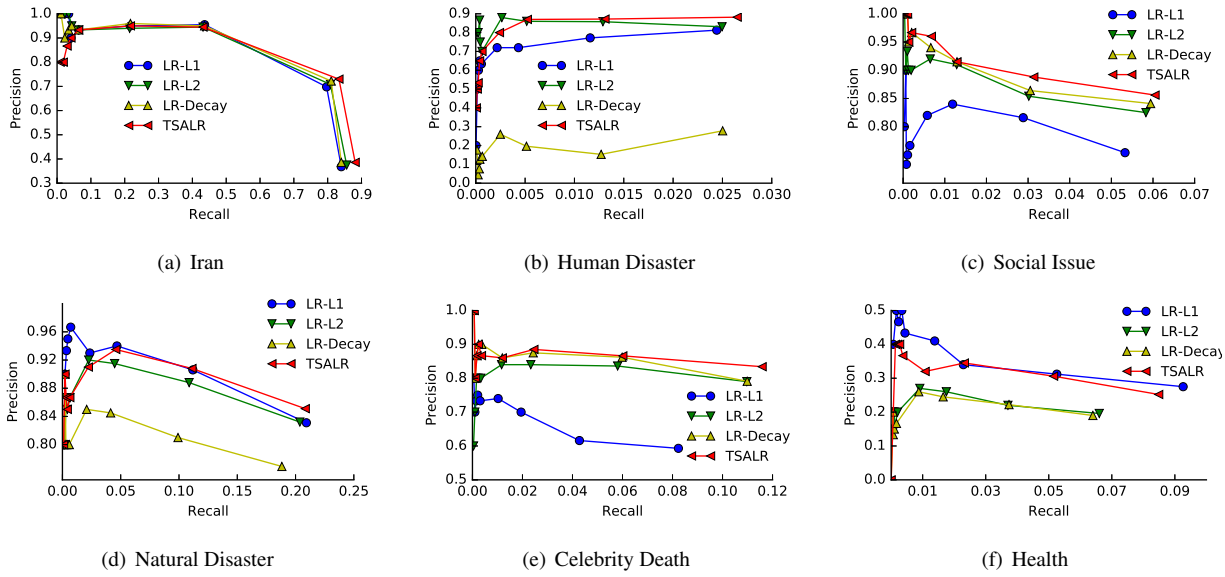
A final critical observation for effective learning is that *TSALR is convex* since each additive component in TSALR is convex (notably, including the Variance-based temporal regularizer), which guarantees a *global minimum* can be found efficiently.

4.2 Evaluating TSALR. In this section, we evaluate TSALR by comparing it to LR with regularization via the L2 norm and the L1 norm (i.e., absolute values of weights). In addition, we also compare TSALR to LR-Decay, a baseline method based on the intuition that the newer (more recent) a feature is, the more temporally stable it is for future prediction. LR-Decay is formulated as follows:

(4.10)
$$L(\theta) = \frac{1}{|D|}\left[\sum_{i=1}^{|D|}\log\left(1+e^{-y^{(i)}\sum_k[\theta_k x_k^{(i)} z_k^{(i)}]}\right)+\frac{\lambda}{2}||\theta||_2^2\right]$$

where $\mathbf{z}$ is a vector of decay factors $(z_1, \cdots, z_k)$ with each element $z_k = e^{-\gamma(t_0-t_k)}$, t_0 denotes the current time slice, t_k denotes the first time the feature x_k appeared in the dataset, and γ is a parameter that controls the decay rate of the contribution of feature x_k. In the experiments, t ranges over the $T = 12$ months. All hyperparameters are tuned on held-out validation data.

First, we show in Table 4 a comparison of TSALR against the baselines in terms of average precision for the top 1000 results ($AP@1000$). Briefly, we observe that TSALR



(a) Iran (b) Human Disaster (c) Social Issue

(d) Natural Disaster (e) Celebrity Death (f) Health

Figure 4: Precision-Recall at k curve of LR and TSALR. TSALR generally improves future predictive performance across the Precision-Recall continuum for all topics.

Table 4: Performance comparison using AP@1000.

	LR-L1	LR-L2	LR-Decay	TSALR
Iran Nuclear Deal	0.757	0.777	0.778	**0.801**
	5.81%	3.08%	2.95%	Improvement
Human Disaster	0.019	0.021	0.021	**0.023**
	21.05%	9.52%	9.52%	Improvement
Celebrity Death	0.054	0.091	0.094	**0.100**
	85.18%	9.89%	6.38%	Improvement
Social Issue	0.043	0.050	0.052	**0.054**
	25.58%	8.00%	3.84%	Improvement
Natural Disaster	0.175	0.180	0.153	**0.188**
	7.42%	4.44%	22.87%	Improvement
Health	**0.030**	0.015	0.015	0.026
	-13.33%	73.33%	73.33%	Improvement

outperforms the baselines on 5 topics. In particular, TSALR improves $AP@1000$ by roughly 73% over LR-L2 and LR-Decay on the Health topic, and 10% over LR-L2 and LR-Decay on the Celebrity Death topic. We also observe that LR-L1 performed the best on the Health topic, but performed the worst on the other 5 topics, which means its performance is not consistent. LR-Decay is better than LR-L2 on the topics Iran Nuclear Deal, Celebrity Death and Social Issue, but is not better than LR-L2 on the other 3 topics. While LR-Decay is sensitive to time, it effectively only uses the subset of most recent features when doing prediction; however, new features can be temporally unstable features, which may mislead LR-Decay's predictions. These results confirm our general hypothesis that reducing the contribution of TUF through Variance-based temporal regularization can improve generalization over future data.

Finally, we show in Figure 4 the comparison using precision-recall curves for the 6 topics. In summary, we clearly observe that TSALR outperforms the competitors in most cases. In general, we observe that TSALR outperforms LR-L2 and LR-Decay at almost all points on the precision-recall curve. While LR-L1 sometimes provides better precision at low recall, it performs very poorly on Social Issue and Celebrity Death and poorly on Human Disaster. Thus LR-L1 proves to be much less consistent than TSALR, which is always near the top performance of any method.

5 Related Work

Below, we review the major work related to feature selection and topic classification, specifically for social network data that was the target application of this paper.

5.1 Feature Selection. Feature selection algorithms rank features according to metrics such as Pearson correlation coefficient, Pearsons chi-squared test χ^2 and Mutual Information as the principle criteria for selecting the top-ranked features [8]. Those algorithms generally belong to one of the three categories: filter approaches [9], wrapper approaches [10] and embedding approaches [11].

Filter methods are usually applied as a preprocessing method to filter out the less informative features independently of any classification algorithm [9]. They are particularly effective in terms of computational complexity by ignoring the impact of the selected features on the actual clas-

sification task objective.

Wrapper methods treat the classification algorithm as a black box, and wrap the algorithm in various search methods that look for a feature subset to maximize the classification performance [10, 12, 13]. There are two types of wrapper methods: Sequential selection [14, 15] and heuristic search algorithm [12]. Sequential selection algorithms are iterative methods in which we start with no feature (or all the features) and add features (or remove features) until the best performance of the learning algorithm is obtained. On the other hand, the heuristic search algorithms adopt greedy heuristic-guided strategies to evaluate different subsets to optimize the overall objective function. Despite being effective, the computational complexity of wrapper methods is their key drawback and would prevent them from being applied to our 40 TB of Twitter data.

Embedded feature selection methods [16, 11] combine the advantages of filter and wrapper methods, which incorporate feature selection as part of the classifier training process. A typical example of this method is LASSO [17] regression whose objective inherently encourages weight sparsity that has the ultimate effect of performing feature selection as a byproduct of training.

The above methods, as generic feature selection techniques, are not specialized to any notion of temporal stability in feature selection. While there is a lot of active research aiming to select features that are insensitive to varying conditions such as data perturbations [18, 19, 20], none of these works directly address the long-term temporal stability of features that we address in this paper. The consequence is that features selected by those methods do not explicitly control for temporal stability over long-time horizons that is crucial to topical classification on persistent social media platforms such as Twitter.

5.2 Twitter Topic Classification. The first challenge of classification on Twitter is labeling a sufficient quantity of data to enable reliable and generalizable training. Previous related work has assigned labels to the tweets either with a single hashtag [3, 2], a user-defined query for each topic [5], or co-training based on the URLs and text of the tweet [4]. We slightly expand on [3] by labeling with a set of hashtags instead of a single hashtag.

The next challenge of classification on Twitter is defining appropriate features. Sriram et al. [21] leverages the user profile as features in addition to the sparse linguistic features of tweets (i.e., bag of words) for classification. Kurka et al. [22] explored the use of retweet information. Li et al. [23] use an Entity knowledge base (Entity KB) to enrich the Twitter features. Mehrotra et al. [24] build augmented document representations from tweets to improve density of the training data. More recently, supervised Latent Dirichlet Allocation (LDA) is proposed by augmenting tweets with content from embedded URLS [25]. While all of these works address different ways to construct and select rich features for social media topic classifiers, they all focus on how to build a strong classifier for a given training dataset and neglect the temporal stability of features explored in this paper. However, the temporal stability regularization proposed in this paper can be applied to any set of (rich) features including the feature sets defined in these works.

6 Conclusion

In this paper, we proposed a study of the temporal stability of features for topical classification in Twitter. For this purpose, we proposed 6 metrics to identify temporally unstable features. We compared the proposed metrics with an oracular method (i.e., using the test data) for generating a ground truth set of temporally unstable features (TUF). Our results showed that the metrics proposed are, to some extent, able to identify TUF features. In particular, the Variance metric has demonstrated strong performance to identify TUFs and also permits incorporation into a convex temporally stable logistic regression framework.

We empirically showed the performance improvement for standard classification by eliminating TUFs over 6 classification topics. Finally, we leveraged the Variance metric to design a novel temporally regularized variant of logistic regression, i.e., Temporal Stability Aware Logistic Regression (TSALR). TSALR avoids some difficulties of feature selection by directly downweighting the influence of temporally unstable features in an embedded approach. We empirically demonstrated that TSALR is effective for learning TSFs for topical classification in Twitter.

Overall, this paper is an initial step towards a new research direction on temporally stable feature selection and learning for tweet classifiers. Our proposed temporal stability regularizer is not restricted to Logistic regression, but can be employed by other machine learning methodologies as well. To this end, we hope this paper paves the way for future studies on long-term temporally stable learning and extends it to other machine learning problems where temporal stability issues also arise.

References

[1] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 591–600, New York, NY, USA, 2010. ACM.

[2] Zahra Iman, Scott Sanner, Mohamed Reda Bouadjenek, and Lexing Xie. A longitudinal study of topic classification on twitter. In *Proceedings of the 11th International AAAI Conference on Web and Social Media (ICWSM-17)*, Montreal, Canada, 2017.

[3] Jimmy Lin, Rion Snow, and William Morgan. Smoothing techniques for adaptive online language models: topic tracking in tweet streams. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011.

[4] Shuang-Hong Yang, Alek Kolcz, Andy Schlaikjer, and Pankaj Gupta. Large-scale high-precision topic modeling on twitter. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014.

[5] Walid Magdy and Tamer Elsayed. Adaptive method for following dynamic topics on twitter. In *ICWSM*, 2014.

[6] Mohamed Reda Bouadjenek, Hakim Hacid, and Mokrane Bouzeghoub. Social networks and information retrieval, how are they converging? a survey, a taxonomy and an analysis of social information retrieval approaches and platforms. *Information Systems*, 2016.

[7] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.

[8] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[9] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proc. of the 20th international conference on machine learning (ICML-03)*, pages 856–863, 2003.

[10] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2):273–324, 1997.

[11] Enrique Romero and Josep María Sopena. Performing feature selection with multilayer perceptrons. *IEEE Trans. on Neural Networks*, 19(3):431–441, 2008.

[12] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[13] Patrenahalli M. Narendra and Keinosuke Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Trans. on Comp.*, 9(C-26):917–922, 1977.

[14] P. Pudil, J. Novoviov, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119 – 1125, 1994.

[15] J. Reunanen. Overfitting in making comparisons between variable selection methods. *Journal of Machine Learning Research*, 3:1371–1382, 2003. cited By 246.

[16] Chenping Hou, Feiping Nie, Dongyun Yi, and Yi Wu. Feature selection via joint embedding learning and sparse regression. In *Proc. of the 22nd International Joint Conf. on Artificial Intelligence*, IJCAI'11, pages 1324–1329, 2011.

[17] Tong Zhang. On the consistency of feature selection using greedy least squares regression. *Journal of Machine Learning Research*, 10(Mar):555–568, 2009.

[18] Petr Somol and Jana Novovicova. Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(11):1921–1939, 2010.

[19] Alexandros Kalousis, Julien Prados, and Melanie Hilario. Stability of feature selection algorithms: a study on high-dimensional spaces. *Knowledge and information systems*, 12(1):95–116, 2007.

[20] Sarah Nogueira and Gavin Brown. Measuring the stability of feature selection. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 442–457. Springer, 2016.

[21] Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short text classification in twitter to improve information filtering. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 841–842. ACM, 2010.

[22] David Burth Kurka, Alan Godoy, and Fernando J Von Zuben. Using retweet information as a feature to classify messages contents. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1485–1491. International World Wide Web Conferences Steering Committee, 2017.

[23] Quanzhi Li, Sameena Shah, Xiaomo Liu, Armineh Nourbakhsh, and Rui Fang. Tweetsift: Tweet topic classification based on entity knowledge base and topic enhanced word embedding. In *Proc. of the 25th ACM Conference on Information and Knowledge Management*, pages 2429–2432, 2016.

[24] Rishabh Mehrotra, Scott Sanner, Wray Buntine, and Lexing Xie. Improving lda topic models for microblogs via tweet pooling and automatic labeling. In *Proc. of the 36th international ACM SIGIR conference*, pages 889–892. ACM, 2013.

[25] Saurabh Kataria and Arvind Agarwal. Supervised topic models for microblog classification. In *Data Mining (ICDM), 2015 IEEE International Conference on*, pages 793–798. IEEE, 2015.