Sentiment and Topic Analysis on Social Media: A Multi-Task Multi-Label Classification Approach

Shu Huang[†], Wei Peng[‡], Jingxuan Li[§], Dongwon Lee[†]

[†] College of IST, The Pennsylvania State University, University Park, PA, 16802, U.S.A.
[‡] Xerox Innovation Group, Xerox Corporation, Rochester, NY, 14580, U.S.A.
[§] School of Computer Science, Florida International University, Miami, FL, 33199, U.S.A.

{shuang,dlee}@ist.psu.edu, wei.peng@xerox.com, jli003@cs.fiu.edu

ABSTRACT

Both *sentiment analysis* and *topic classification* are frequently used in customer care and marketing. They can help people understand the brand perception and customer opinions from social media, such as online posts, tweets, forums, and blogs. As such, in recent years, many solutions have been proposed for both tasks. However, we believe that the following two problems have not been addressed adequately: (1) Conventional solutions usually treat the two tasks in isolation. When the two tasks are closely related (e.g., posts about "customer care" often have a "negative" tone), exploring their correlation may yield a better accuracy; (2) Each post is usually assigned with only one sentiment label and one topic label. Since social media is, compared to traditional document corpus, more noisy, ambiguous, and sparser, single label classification may not be able to capture the post classes accurately. To address these two problems, in this paper, we propose a *multi-task multi-label* (MTML) classification model that performs classification of both sentiments and topics concurrently. It incorporates results of each task from prior steps to promote and reinforce the other iteratively. For each task, the model is trained with multiple labels so that they can help address class ambiguity. In the empirical validation, we compare the accuracy of MTML model against four competing methods in two different settings. Results show that MTML produces a much higher accuracy of both sentiment and topic classifications.

Author Keywords

multi-task; multi-label; classification; sentiment analysis; topic analysis.

ACM Classification Keywords

H.2.8 Database Management: Database Applications

[data mining];

H.1.0 Information Systems: Models and Principles

General Terms

Algorithms; Experimentation.

WebSci'13, May 1 – May 5, 2013, Paris, France.

INTRODUCTION

As online Social Network Services (SNS) become more popular in recent years, user-generated contents (UGC) within such SNS have exploded. Since UGC can potentially contain valuable information to many applications, a lot of research has been conducted to investigate how to extract useful knowledge from UGC. Among a variety of SNS, in particular, micro-blogging such as Twitter, has been rapidly growing recently. Users post short texts, called *tweets*, about any topic of interest, reply to others' tweets, and disseminate information to other users by re-tweeting. Although tweets are limited to no more than 140 characters, Twitter has become an extremely popular platform where people freely express and exchange opinions. Businesses in particular has noticed the potential of Twitter and used it in a variety of applications, such as marketing promotion, brand campaign, and customer care [24]. For instance, a lot of companies have started to poll relevant tweets to help understand trending topics among their customers and the sentiments towards their products.

Among all knowledge that can be extracted from tweets, in this paper, we focus on two aspect: (1) *sentiment* of a tweet that captures the subjective mood of a user, such as "positive" and "negative"; and (2) *topic* of a tweet that indicates the scope of subject content from pre-determined aspects, such as "Compliment", "News", and "Promotion". In general, techniques known as *sentiment analysis* and *topic analysis* respectively are used to infer latent sentiments and topics of a given text corpus. Furthermore, in this paper, we employ the following class schemes. The sentiment classes are "positive", "negative", and "neutral". The topic classes include "Care/Support", "Lead/Referral", "Mention", "Promotion", "Review", "Complaint", "Inquiry/ Question", "Compliment", "News", and "Company/Brand". We focus on the problem of *classification*, i.e., given a set of pre-determined classes, how to identify which classes an instance belongs to.

Given a collection of tweets regarding a certain common subject, a topic classification method can reveal the particular aspects that users are talking about and which are dominant, while a sentiment classification method tells the proportion of users who feel positive or negative toward the subject. For instance, Figure 1 shows example tweets related to "virgin mo-

Figure 1. Tweets related to "virgin mobile", with topic and sentiment labels.

bile", with their identified sentiment and topic labels. In this example, some users are talking about promotions, and others are complaining about customer service and payment. Meanwhile, some tweets show positive sentiment about the brand, while others are negative. As one can see, therefore, the analysis of tweet sentiments and topics can help businesses to get a sense of user opinion towards their products and services. Due to the practical implication, in recent years, a lot of studies (e.g., [12, 21, 1, 24, 19]) have been conducted towards sentiment and topic classifications of tweets (see Section 2 for details).

However, by and large, existing solutions have the following issues. First, conventional solutions usually treat sentiment and topic classification tasks separately, though the two tasks are often closely related. For instance, tweets about some topics usually tend to have certain sentiment. In Figure 1, a user who tweets about "promotions" shows positive sentiment, while two other users who complain about the "care/support" appears to be negative. It implies that often tweet topics can help promote the sentiment classification, and vice versa. On the other hand, the same words could present different sentiments in different topics. Therefore, one can exploit such an inter-relationship between two classification tasks to improve the overall classification accuracy. Second, compared to traditional document corpus where sentiment or topic classification occurs, micro-blog data such as tweets are very short, noisy, and ambiguous. For instance, a tweet mentioning a broken mobile device may be assigned to either the topic of "complaint" or "care/support". Therefore, instead of insisting on the assignment of a single class label to a tweet, sometimes, one can flexibly assign multiple class labels to an ambiguous tweet.

Based on the two limitations of existing methods, in this paper, we propose a novel model, termed as the **Multi-Task Multi-Label (MTML)**, which performs the classification of both sentiments and topics of tweets concurrently, and incorporates each other's results from prior steps to promote and reinforce current results iteratively. The learned class labels of one task are incorporated as part of predicting features of the other task. For each task, the model is trained with the maximum entropy by using multiple labels to learn more information and handle class ambiguity. In addition, the MTML model produces probabilistic results, instead of binary results, so that multi-label prediction is allowed and labels can be ranked accordingly.

Our contributions in this paper are as follows:

- By combining both sentiment and topic classifications of tweets, we develop a novel probabilistic MTML model.
- Using real tweets and crowdsourcing based ground truth data, we validate the MTML model. Compared to four state-of-the-art classification methods, overall, the MTML improves the classification accuracy by 5% for sentiment and 12% for topic, respectively.
- We also compare the MTML against the four classification methods after the problem is converted to the single-task single-label setting by two class re-organization methods (LP and DMI) and show the superiority of the MTML remained.

The rest of this paper is organized as follows: we discuss related work in Section 2. The problem statement and definition is introduced in Section 3. In Section 4, we present details of the MTML classification model. The experimental results and analysis are shown in Section 5, followed by the conclusion in Section 6.

RELATED WORK

Multi-Label Classification

Multi-label classification is concerned with categorizing instances into multiple classes, while the associated classes are not exclusive. Each associated class of an instance is called a "label". Existing multi-label classification methods can be generally grouped into two categories: class reorganization and algorithm innovation.

Class reorganization methods reorganize classes to transform the multi-label classification into single-label classification. Three approaches are proposed for this purpose in [4]. They include: randomly selecting one from the multiple labels, ignoring all multi-label instances, and constructing multiple single-label classifiers. Another approach extends classes by constructing a label power set (LP) and considering each different label combination as a new class [20]. The disadvantages of this approach are that it may lead to a large number of reorganized classes and each class has too few instances. Another widely used reorganizing method is to construct a binary classifier for each class, and then the classification results on all classes are combined into a multi-label result [14]. In a methodology overview [23], an undocumented method is

Table 1. Example Tweets of "Virgin Mobile" with Sentiments and Topics

ID	Content	Sentiments	Topics
1	Virgin Mobile's #Sparah campaign is genius! Love the episodes!	Positive	Compliment
2	I love the new phone u came out with for virgin mobile. i love the samsung restore.	Positive	Compliment
3	@virginmobileus Care to answer???	Negative	Complaint, Care/Support
4	is seriously annoyed with Virgin Mobile. Get your crap together and fix my account!!!!!	Negative	Complaint, Care/Support
5	@anonymizedName get the hell out of here with virgin mobile crap!	Negative	Complaint

introduced, which decomposes instances by using only single labels and then merges the single-label classification results.

Algorithm innovation methods focus on modifying single-label classification models to adapt to multi-label classification. In [16], a mixture model is used to represent the multiple classes with training documents labeled by EM. An algorithm innovation with decision tree algorithm C4.5 adopts a new entropy measure that allows multiple labels in leaves [8]. After that, an algorithm MMAC is proposed, which learns a set of association rules first and then combine these rules into a multi-label classification model [22]. Jin et al. study a special kind of classification in which each instance is given a set of candidate labels and only one of them is correct [13]. In this work, a log-likelihood based approach is used together with EM to handle the multiple-label. Most existing multi-label classification methods cannot be directly applied to address multi-task classification. At the same time, the association between different tasks are not explored either.

Multi-Task Classification

Multi-task classification utilizes the correlation between related tasks to improve classification by learning tasks in parallel. Existing work mostly falls into two groups. The first group uses kernels and regularizers, while the second group investigates common features and task similarity measures.

Many algorithms are proposed to solve multi-task learning with various kernels and regularizer. In [5], k-nearest neighbor and kernel regression are introduced to learn tasks in parallel. Evgeniou et al. present a multi-task learning approach based on the minimization of a regularization function similar to the one of SVM [10]. Later, a multi-task kernel function is derived to help estimate multiple task functions at one time [9]. In [6], a multi-task learning algorithm based on gradient boosted decision tree is proposed for web-search ranking over multiple datasets.

Exploring common features and task similarity also helps with multi-task learning. Ben-David et al. define and exploit task relatedness by the similarity between distributions generated by examples of tasks [3]. Later, a common feature selection method is derived for SVM when multiple tasks exist over a common input space [11]. To learn some common features across multiple related tasks, a 1-norm regularization method with a new regularizer is introduced in [2]. In [25], a dirichlet process based model is proposed to identify similar tasks and solve both symmetric and asymmetric multi-task learning. Another study of features uses hashing to reduce feature dimension and apply it on very large scale multi-task learning.

These methods focus on multi-task classification but do not consider multiple labels in each task. The study of multi-label multi-task learning still remains open.

Tweet Sentiment and Topic Analysis

Tweet sentiment and topic analysis becomes very popular recently. However most state-of-the-art studies address only sentiment classification or topic classification. To determines tweet sentiment, query-based dependent features and related tweets are explored and incorporated in [12]. In [1], POS-specific prior polarity features are introduced and applied with a tree kernel for sentiment analysis. Tan et al. find that including the influence of social connections can improve accuracy of sentiment classification [21]. In addition, a graph model is introduced to classify sentiment of hashtags in a time period [24].

To classify topics of noun phrases in tweets, a community-based method is presented to identify their boundaries within the context and classify them to a specific category [7]. After that, a model that switches between two probability estimates of words is proposed, which can learn from stationary words and also respond to bursty words [19]. In [18], another method is introduced to determine whether a tweet is related to a topic or not by using data compression. Furthermore, a Bag-of-Words approach and a network-based approach are evaluated in classifying twitter trending topics into 18 general categories [15].

These approaches focus on single-label classification on either sentiment or topic classes. Among the state-of-the-art work, none of them studies multi-label classification that analyzes both sentiments and topics at the same time. To address the problem of multi-label multi-task classification, we propose an algorithm based on multi-label learning and utilize association between tasks to promote classification accuracy.

PROBLEM STATEMENT

Sentiment and topic analysis of social media have a wide application in business marketing and customer care. For instance, when promoting a new policy or a product, the company wants to know how customers comment about it so that they can respond properly and timely to address criticisms and issues. For this purpose, monitoring the current sentiment trend and topics towards a certain product or brand name is both necessary and important. However, as a lot of posts may be generated in a short time, hiring human experts to work on them is too expensive. To address this problem, it requires some techniques that can classify tweet topics and sentiments automatically and quickly.

However, sentiment and topic analysis of social media involves a lot of challenges. As tweets are very short and may contain incomplete sentences, their meaning could be ambiguous and interpretations highly rely on the context. At the same time, people tend to use informal language or even

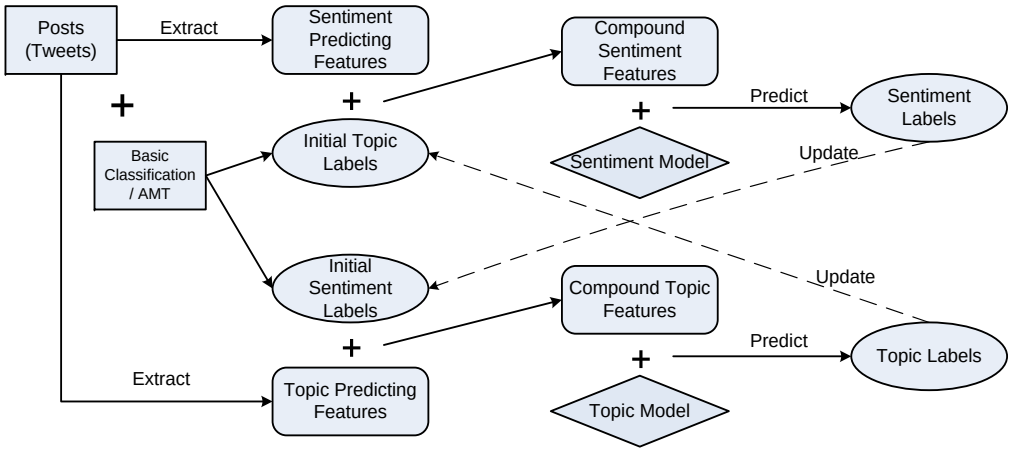

Figure 2. Multi-task multi-label classification model for both sentiment and topic classifications

bad syntax in tweets. This makes classic methods of natural language processing not well applicable in many situations. What is more, topic classification is hard even if done by human experts. On one hand, topics of tweets may not be perfectly exclusive. On the other hand, the content of a tweet may cover multiple topics. Therefore, binary classification may not produce satisfactory results. To solve this problem, multi-label classification is required.

As we have introduced, tweet topics and sentiments are not completely independent. By observing a collection of tweets, we find that certain association exists between tweet topics and sentiments. In addition, the appearance of some terms may also serve as strong indicators of certain classes. As an example, Table 1 shows some real tweets regarding "Virgin Mobile", with user names anonymized. Tweet sentiments are positive, negative, and neutral. Tweet topics are 10 predefined classes. As shown in the table, tweets 1 and 2 indicate an association between Positive sentiment and topic Compliment. These tweets both contain the term "love", which gives a strong indication for both Positive sentiment and topic Compliment. Tweets 3-5 are negative, while their topics include Complaint and Care/Support. They imply that these two topics are likely to appear together with Negative sentiment. Meanwhile, the term "crap" appears in both tweets 4 and 5, implying an association with Negative sentiment and those two topics.

As observed above, sentiment classification and topic classification are associated. What is more, these two tasks are also connected with certain indicating terms. Considering the association between tasks, co-classification of multiple tasks can help reinforce each other and produce better results than doing them independently. Meanwhile, each task may involve multiple labels, i.e. a tweet refers to more than one topic. Classifying with multi-label can help handle the class ambiguity and improve classification accuracy. Therefore, we propose to incorporate multiple labels into multi-task classification. In this way, we can make good use of the latent information in predicting features, and at the same time, employ the results of multiple tasks to promote each other.

To incorporate both multi-task and multi-label classification, we investigate the following questions: *how to make use of multi-task classification to promote each task? How to incorporate and process multiple labels in multi-task classification? In particular, how to apply the method on sentiment and topic classifications?*

Formally, the multi-task multi-label (MTML) classification is defined as follows:

Problem 1 (MTML Classification) *Given an instance x and classification tasks $T = \{T_j : j = 1, ...M\}$, where the j-th classification task T_j has a finite set of classes $L_j = \{l_{jk} : k = 1, ...K_j\}$, the goal of MTML classification is to find a collection of class label sets $Y = \{Y_1, ..Y_j..\}$ that x belongs to, $Y_j = \{l_{j1}, ..l_{jq}\} \subseteq L_j$ is the set of class labels of x for the j-th classification task.* ■

MULTI-TASK MULTI-LABEL CLASSIFICATION

Overview

By classifying both sentiments and topics at the same time, in the MTML model, we incorporate the results into predicting features, so that labels of the two tasks can promote and reinforce each other. For each task, the model is trained with maximum entropy on different predicting feature spaces. To learn with multiple labels, model coefficients are estimated with an optimization of multi-task likelihood and the prior label distributions.

Figure 2 illustrates an overview of the classification using the MTML model for classifying sentiment and topic of tweets. With a tweet collection, first, we extract sentiment and topic predicting features. Meanwhile, by using an existing classification method or Amazon Mechanical Turk based crowdsourcing, initial class labels can be obtained. Then, incorporating initial labels with predicting features, we get compound sentiment and topic features. The model can be trained by estimating coefficients with the training dataset. Once the model is trained, given compound features, it can generate new sentiment and topic labels. Repeating the two classifications iteratively can keep the class labels updated until it converges.

Feature Extraction and Selection

To train the MTML classification model, we first extract predicting features from tweets. Given a collection of tweets, we remove stopping words and select all keywords and bi-grams. For each tweet, its predicting feature vector $X_i = [a_1, a_2, ...a_m]$ consists of keywords and bi-grams in it. Since there are a tremendous amount of predicting features, feature selection is necessary to obtain the optimal predicting accuracy. For this purpose, using the Mallet [17], we measure the predicting accuracy of Support Vector Machine, Naive Bayes, and Maximum Entropy with different numbers of predicting features. Then we compare the results and determine the optimal number of predicting features accordingly. Feature extraction and selection are conducted on both sentiment and topic classifications. The optimal predicting feature sets are selected separately on the two tasks. On different tasks, the number of optimal predicting features may vary.

The MTML Model

Within the predicting feature space, each tweet can be mapped to a feature vector. As we have introduced, each tweet instance is associated with a set of class labels. Assume that there are a total of K classes and N training instances. Let X_i denote the feature vector of the i-th instance x_i, where $i = 1, 2, ..., N$, and L_i denotes its label set. We apply Maximum Entropy (ME) to estimate the class distribution, which allows flexibility in model construction and also produces probabilistic classification result.

Let θ_k represent the coefficient vector of the k-th class, $k = 1, 2, ..., K$ and Y_i represent the class that instance x_i is assigned. Then, the probability of x_i to be classified into the k-th class becomes:

$$P(Y_i = k|X_i, \theta) = \frac{e^{\theta_k \cdot X_i}}{1 + \sum_{j=1}^{K} e^{\theta_j \cdot X_i}} \quad (1)$$

When solving the multi-task classification, we do not assume the independence of each task any more. By extending equation (1), we propose to incorporate classification labels of another task to make use of the latent task associations. Given an instance x_i, assume LS_i is its sentiment labels, and LT_i is its topic labels. Then, the feature vectors can be extended by including labels of another task. For the multi-task classification, let xs_i represent the sentiment feature vector and XS_i be the extended sentiment feature vector. Then, $XS_i = [xs_i, LT_i]$. Similarly, use xt_i and XT_i to denote the initial and extended topic feature vector, $XT_i = [xt_i, LS_i]$. Based on them, let P_s and P_t denote the sentiment and topic distribution of an instance. Then, for the sentiment classification, we get:

$$P_s(Y_i = k|xs_i, LT_i, \theta s) = \frac{e^{\theta s_k \cdot XS_i}}{1 + \sum_{j=1}^{K} e^{\theta s_j \cdot XS_i}} \quad (2)$$

For the topic classification, next, we get:

$$P_t(Y_i = k|xt_i, LS_i, \theta t) = \frac{e^{\theta t_k \cdot XT_i}}{1 + \sum_{j=1}^{K} e^{\theta t_j \cdot XT_i}} \quad (3)$$

Now, we incorporate multi-labels into the classification. While learning with multi-label, our goal is to find the parameters θs and θt that maximize the probability of instance x_i to be labeled with LS_i and LT_i. Formally, let Θ denote the optimal values of $(\theta s, \theta t)$. Then, the objective function to estimate parameters can be written as:

$$\Theta = \arg\max_{\theta s, \theta t} \Pi_i P_s(Y_i \in LS_i|xs_i, LT_i, \theta s) \cdot P_t(Y_i \in LT_i|xt_i, LS_i, \theta t) \quad (4)$$

Let $\hat{P}_s$ and $\hat{P}_t$ be the prior probability generated from the labels. Then, P_s and P_t are the posterior probability produced by the classification model. To estimate parameters, one approach is to make the model based classification match the distribution from prior labels, i.e., minimize the difference between them. For each instance x_i, $\hat{P}_{s_i}$ can be calculated by the proportion of each label in LS_i out of all labels in LS_i; and similarly for $\hat{P}_{t_i}$. Both $\hat{P}_{s_i}$ and $\hat{P}_{t_i}$ are calculated with constraints of probabilities, $\sum_{k \in LS_i} \hat{P}_{s_i}(Y = k|x_i) = 1$, and $\sum_{k \in LT_i} \hat{P}_{t_i}(Y = k|x_i) = 1$.

Based on equation (4), a widely accepted method of parameter estimation is to minimize the KL-divergence between the prior and posterior probabilities of each instance. Denote S as all of the sentiment classes and T as all of the topic classes. Then, following the KL-divergence, the objective function can be furthermore written as:

$$\Theta = \arg\min_{\theta s, \theta t} \begin{cases} \sum_i \sum_{k \in S} \hat{P}_{s_i}(Y = k|x_i) log \frac{\hat{P}_{s_i}(Y=k|x_i)}{P_{s_i}(Y=k|xs_i, LT_i, \theta s)} \\ \sum_i \sum_{k \in T} \hat{P}_{t_i}(Y = k|x_i) log \frac{\hat{P}_{t_i}(Y=k|x_i)}{P_{t_i}(Y=k|xt_i, LS_i, \theta t)} \end{cases} \quad (5)$$

Since for any class k that is not in LS or LT, the prior probability $\hat{P}_{s_i}(Y = k|x_i) = \hat{P}_{t_i}(Y = k|x_i) = 0$, having no influence on the parameter estimation. Therefore, equation (5) can be simplified to the following:

$$\Theta = \arg\max_{\theta s, \theta t} \begin{cases} \sum_i \sum_{k \in LS_i} & \hat{P}_{s_i}(Y = k|x_i) \\ & \cdot log P_{s_i}(Y = k|xs_i, LT_i, \theta s) \\ \sum_i \sum_{k \in LT_i} & \hat{P}_{t_i}(Y = k|x_i) \\ & \cdot log P_{t_i}(Y = k|xt_i, LS_i, \theta t) \end{cases} \quad (6)$$

with constraints $\sum_{k \in LS_i} \hat{P}_{s_i}(Y = k|x_i) = 1$, and $\sum_{k \in LT_i} \hat{P}_{t_i}(Y = k|x_i) = 1$.

In equation (6), $\hat{P}_{s_i}$ and $\hat{P}_{t_i}$ are calculated from the labels. P_{s_i} and P_{t_i} are model-based probabilities, which vary with θs and θt. By solving equation (6), θs and θt can be determined. When the data is sparse, ME may have the problem of "overfitting." To reduce such an overfitting, we integrate the Gaussian prior into ME for parameter estimation, with mean at 0 and variance of 1.

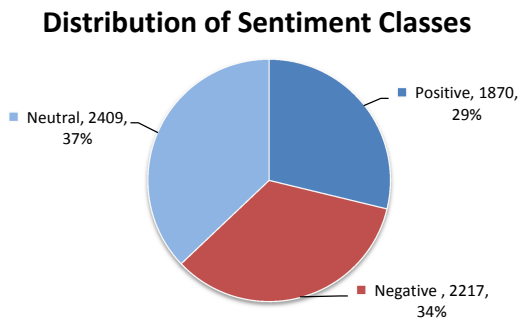

Figure 3. Percentages and numbers of tweets on sentiment classes

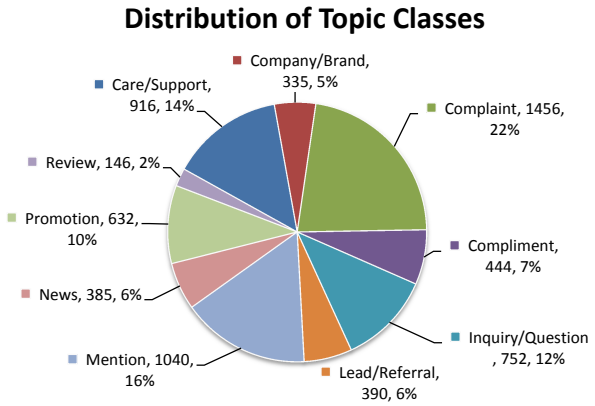

Figure 4. Percentages and numbers of tweets on topic classes

After the model is trained, given a tweet and the feature vector, its sentiment and topic classes can be determined by equation (2) and (3). Since extended feature vectors of the two tasks make use of labels from each other, it is necessary to obtain the initial labels. They can be generated from the classic ME model or any other classification approach. After that, during the process of multi-task classification, the sentiment labels obtained from equation (2) can be applied in equation (3) for topic classification, and vice versa. Repeating the two tasks iteratively will keep updating the classification results until it converges.

As a summary, the MTML classification proceeds as follows:

1. Given an instance x_i, extract its topic feature vector xt and sentiment feature vector xs.
2. Generate initial topic labels LT and sentiment labels LS of x_i by using a simple classification method or crowdsourcing.
3. Integrate LT with xs to obtain the compound sentiment feature vector XS, and obtain the compound topic feature vector XT similarly out of LS and xt.
4. Apply XS to the MTML *sentiment* classification model and generate sentiment labels LS' of x_i.
5. Apply XT to the MTML *topic* classification model to generate topic labels LT'.
6. Plug in LT' to update XS, and also use LS' to update XT.
7. Repeat steps 4-6 until the classification result converges.

EXPERIMENTAL VALIDATION

Set-Up

Dataset. The proposed MTML model is evaluated using real tweets crawled from 8/31/2010 to 4/26/2011. They contain at least one of the keywords "virginmobile", "VMUcare", "boostmobile", and "boostcare." Our target is to classify sentiments and topics of these tweets towards "boost mobile" and "virgin mobile". After removing tweets that are posted by company customer services, we get a total of 6,496 user-generated tweets for the experiment. For classification, we take 3 sentiment classes and 10 topic classes, which are preset by professionals from the agent of the company "Virgin Mobile." The sentiment classes are "Positive", "Negative", and "Neutral". Figure 3 shows the number of tweets in each sentiment class and their percentage in the distribution. Topic classes include "Care/Support", "Lead/Referral", "Mention", "Promotion", "Review", "Complaint", "Inquiry/ Question", "Compliment", "News", and "Company/Brand". The number of tweets in each class and their percentages in the distribution are shown in Figure 4.

Ground-Truth. Initial sentiment labels and topic labels of tweets are assigned by crowdsourcing via Amazon Mechanical Turk (AMT). AMT is a crowdsourcing marketplace which allows collaboration of people to complete tasks that are hard for computers to do but easy for human workers to do. AMT has two types of users: requesters and workers. Requesters post Human Intelligence Tasks (HITs) with monetary incentives, while workers can browse HITs and complete them for monetary incentives. Requesters may accept or reject the result submitted by workers. With certain quality control mechanisms (e.g., majority voting or controlled HIT) requesters can obtain high-quality results for the submitted HITs through AMT.

Using the AMT, we collect 3 sentiment labels and 3 topic labels for each tweet. Labels may be identical or different. For each tweet, if at least two labels agree with each other, then this label is selected as the majority-voted label. Out of all 6,496 tweets, 6,143 of them have majority-voted sentiment labels, and 4,466 of them have majority-voted topic labels. Among 4,257 tweets that have both sentiment and topic majority-voted labels, we randomly select 500 for testing. The remaining ones and all other tweets that have 3 different labels are used as training instances, which contain 5,996 tweets. Since our MTML model can train with multiple labels, we make use of all labels in training. For testing, the majority-voted label is employed as the ground truth.

Baseline. To validate our model, we use 2 class reorganization methods: Label Power set (LP) [20] and Decompose-Merge Instance (DMI) [23], as well as 4 existing classification models as baselines. They include Naive Bayes (NB), Maximum Entropy (ME), Support Vector Machine (SVM), EM with Prior on Maximum Entropy (EPME) [13]. First,

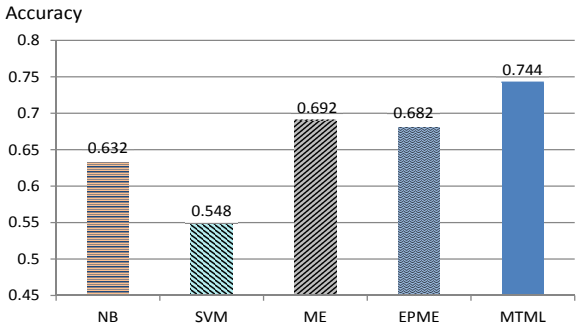

Figure 5. The accuracy of sentiment classification of five methods

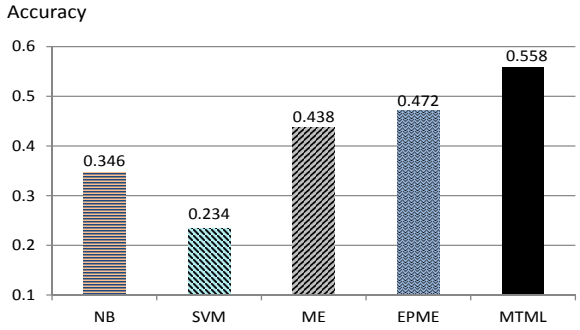

Figure 6. The accuracy of topic classification of five methods

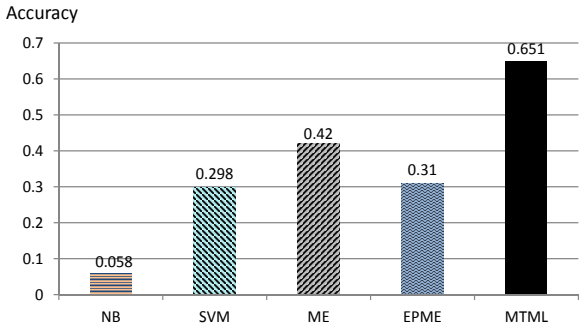

Figure 7. The accuracy of multi-task classification of five methods *after* class reorganization is applied

the MTML model is compared against the baseline models on both tasks. After that, we apply LP with DMI to convert the multi-task multi-label classification into single-task single-label classification first, and then measure the performance of baselines accordingly.

Feature Selection. Predicting features are first generated by extracting keywords from tweet contents. Hashtags are treated the same as other keywords, without any special weighting or discrimination. Initially, 50,553 keywords (thus feature dimensions) are extracted. Instead of doing dynamic feature reduction using conventional methods such as PCA, we used a simple empirical approach. We first measured the accuracy while varying the number of features from 400 to 5,000. For the sentiment classification task, the highest accuracy was obtained with 3,400 features, while for the topic classification task, 2800 features produce the best result. As a result, in the experiment, we simply adopted the 3,400 and 2,800 features for both sentiment and topic classification tasks, respectively. Note that these two sets of features are independent. They are not combined together in the evaluations of our model and baselines.

Evaluation Metrics. We use classification accuracy to measure the performance of model. It is defined as follows:

$$Accuracy_{classification} = \frac{1}{N}\sum_{i=1}^{N} I(Z_i = Y_i)$$

where $I(true) = 1$ and $I(false) = 0$.

Evaluation

In the experiment, we evaluate MTML on both sentiment and topic classification tasks. The results of MTML are compared against baselines respectively. After that, we measure the average accuracy of MTML on multi-task and compare it against baseline results on the LP-converted dataset. In particular, we look into the classification accuracy on each class. By associating the class distribution with the accuracy improvement, we analyze their correlation and how the class properties affect accuracy.

First, we measure the MTML model on sentiment classification. The training dataset contains 5,996 tweets and the testing data contains 500 tweets. Each training tweet is associated with 3 training labels. Meanwhile, MTML is evaluated against NB, ME, SVM, and EPME. Figure 5 shows the accuracy of MTML and baselines on sentiment classification. As shown in the figure, MTML outperforms all baselines, achieving the accuracy of 0.744. Compared to ME and EPME, MTML makes an improvement of 5%. Although sentiment classification achieves a fairly good accuracy with baselines already, therefore, using multi-task and multi-label enables a reasonable improvement.

Second, our MTML model is validated with topic classification on the same dataset. Classification accuracies of our model and baselines are shown in Figure 6. Since there are a total of 10 topic classes and their distribution is not even, the accuracies of both MTML and baselines are not very high. However, MTML still outperforms the baselines and achieves an accuracy of 0.558.

Next, we use LP to transform the dataset into single-task classification with 30 classes (i.e., 3 sentiments $\times$ 10 topics). Furthermore, for each instance with multi-label, we apply DMI to convert it into multiple instances with single labels. Then, the accuracies of NB, ME, SVM, and EPME are measured on this converted dataset. Figure 7 shows the performance of MTML on multi-task classification against baselines after this class reorganization. Among all the methods performed, NB has the lowest accuracy while our proposed MTML still outperforms all baselines.

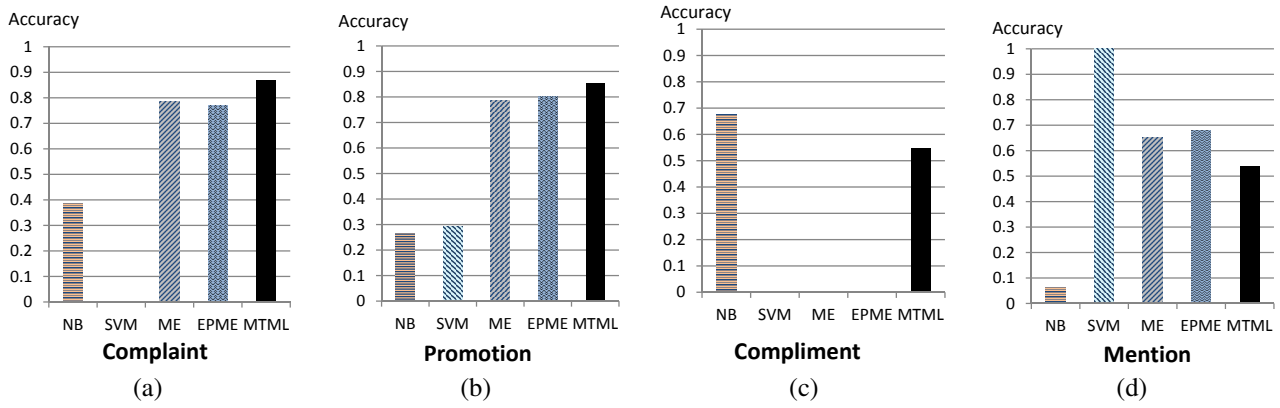

Figure 8. The accuracy of topic classification of the MTML model per four topic classes

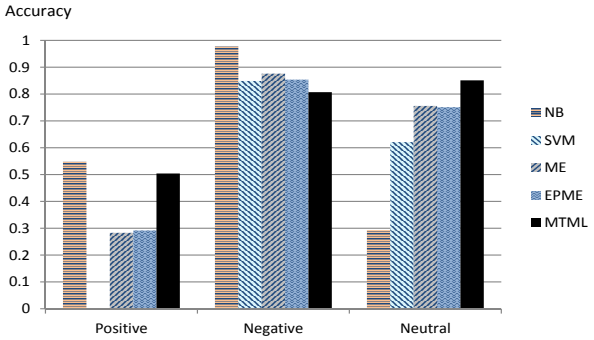

Figure 9. The accuracy of sentiment classification of the MTML model per three sentiment classes

Since different classes take different proportions out of the whole dataset, next, we look into sentiment and topic classes and measure accuracy per each class. Figure 9 shows accuracies of all methods on each sentiment class. Overall, MTML performs well on all sentiment classes, especially the class of "Neutral". For topic classes, we show the comparison of all methods on 4 most interesting topics in Figure 8. Among all 10 topic classes, MTML tends to classify better on those relatively large-sized and explicit ones, such as "Complaint" and "Promotion." Figure 8(a) and (b) shows that MTML has an accuracy of 0.869 and 0.853 on these two classes, respectively. Another interesting observation is that NB outperforms the MTML model on small-sized classes, such as "Compliment" and "Review." As shown in Figure 8(c), NB is 13% better than MTML on "Compliment" class. Finally, Figure 8(d) illustrates that SVM performs the best on the class "Mention." However, it performs poorly on all other classes, because it classifies a majority of instances into "Mention" class.

Case Study

To investigate the advantages of multi-task classification in details, we look at a few sample tweets and their classification results with different methods. Tables 2 and 3 show 4 sample tweets with their sentiment and topic classification labels. Besides the ground truth label in the Truth column, we list classification results by NB, SVM, ME and our MTML model.

Case 1 *Tweet #1 has topic Complaint and sentiment Negative. NB, SVM and ME all classify it to the wrong topic and wrong sentiment classes. However, by using the multi-task approach and incorporating the association between Complaint and Negative, our MTML model successfully classifies it to the right topic and sentiment.*

Case 2 *Tweet #2 has topic Compliment and sentiment Positive. Keyword "love" is a strong indicating feature, but neither SVM nor ME classifies it right. MTML introduces multi-task and multi-label based on ME, therefore, MTML generates the correct classification results.*

Case 3 *Tweet #3 has topic Promotion and sentiment Positive. Both ME and SVM fail to classify on topic or sentiment. NB classifies with only right sentiment. As a comparison, MTML benefits from multi-task and makes right classifications on both tasks.*

Case 4 *Tweet #4 has topic Mention and sentiment Neutral. Among baselines, only SVM classifies its topic correctly. NB classifies with an incorrect association between topic and sentiment. ME does not classify correctly on either task. Only MTML utilizes multi-task labels to promote each other, and successfully classifies both topic and sentiment accurately.*

Summary of Experiments

The experiment shows that the MTML model performs better than baseline methods on both sentiment and topic classification. It produces classification accuracies of 0.744 on sentiment and 0.558 on topic. Compared with ME, MTML improves the accuracy by 5% on sentiment and 12% on topic classification, which indicates that using multi-label and multi-task is effective to improve both classifications. In particular, topic classification obtains a higher accuracy increase than sentiment classifications. It appears that incorporating sentiment labels seems to be of more help to distinguish topics. Looking into accuracies per each class also reveals some insights. Among all classes, for instance, MTML has a higher accuracy on large-sized ones, such as "Complaint", "Mention", and "Promotion." Since topic classes have unbalanced distributions and some of them have very few instances, increasing the dataset size may help increase the classification accuracy.

Table 2. Sample tweets and topic classification results of NB, SVM, ME and MTML

ID	Tweet Content	Truth	NB	SVM	ME	MTML
1	Brought to you by boost mobile unlimited plan.....now with shrinkage????	Complaint	Compliment	Mention	Mention	Complaint
2	I am loving #Sparah and my @virginmobileus LG Optimus!!! @anonymized is so beautiful and ready for the spotlight.	Compliment	Compliment	Mention	Inquiry/ Questions	Compliment
3	New Boost Mobile Android phone for sale! The New Galaxy Prevail Touch Screen! If u want it get @me!	Promotion	Lead/ Referral	Mention	Mention	Promotion
4	That's top-up card It's a phrase which I believe was coined by virgin mobile for its prepaid phone service.	Mention	Compliment	Mention	Complaint	Mention

Table 3. Sample tweets and sentiment classification results of NB, SVM, ME and MTML

ID	Tweet Content	Truth	NB	SVM	ME	MTML
1	Brought to you by boost mobile unlimited plan.....now with shrinkage????	Negative	Positive	Neutral	Neutral	Negative
2	I am loving #Sparah and my @virginmobileus LG Optimus!!! @anonymized is so beautiful and ready for the spotlight.	Positive	Positive	Negative	Neutral	Positive
3	New Boost Mobile Android phone for sale! The New Galaxy Prevail Touch Screen! If u want it get @me!	Positive	Positive	Neutral	Neutral	Positive
4	That's top-up card It's a phrase which I believe was coined by virgin mobile for its prepaid phone service.	Neutral	Negative	Negative	Negative	Neutral

CONCLUSIONS AND FUTURE WORK

In this paper, we study the sentiment and topic classification problem on online posts. By exploring the latent association between tweet sentiments and topics, we propose a *multi-task multi-label* (MTML) classification model. The model utilizes the correlation between related classes across two tasks, and incorporates the result of each classification task to promote the other. In addition, the MTML model integrates multi-label in training to learn from ambiguous expressions and to classify such accordingly. Experiments on a collection of real tweets using crowdsourced ground truth reveal that our proposed model can classify both sentiments and topics of tweets accurately and outperforms other four competing methods.

Many research questions remain open for future work. First, we plan to explore other classification tasks and extend our current model to incorporate more related tasks. Second, while the performance of our model is promising and superior to competing methods, we will investigate other ideas to further improve the accuracy. For instance, a hybrid approach is promising where crowdsourcing is used to solve a small set of challenging classification tasks while algorithm-based classification is used for the remaining tasks.

REFERENCES

1. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., and Passonneau, R. Sentiment analysis twitter data. In *Workshop on Languages in Social Media* (2011), 30–38.
2. Argyriou, A., odoros Evgeniou, and Pontil, M. Multi-task feature learning. In *Conf. on Neural Information Processing Systems (NIPS)* (2006), 41–48.
3. Ben-David, S., and Schuller, R. Exploiting task relatedness for mulitple task learning. In *Annual Conf. on Computational Learning Theory* (2003), 567–580.
4. Boutell, M. R., Luo, J., Shen, X., and Brown, C. M. Learning multi-label scene classification. *Pattern Recognition* (2004), 1757–1771.
5. Caruana, R. Multitask learning. *Mach. Learn. 28* (1997), 41–75.
6. Chapelle, O., Shivaswamy, P., Vadrevu, S., Weinberger, K., Zhang, Y., and Tseng, B. Multi-task learning for boosting with application to web search ranking. In *ACM KDD* (2010), 1189–1198.
7. Chua, F. C. T., Cohen, W. W., Betteridge, J., and Lim, E.-P. Community-based classification noun phrases in twitter. In *ACM CIKM* (2012).
8. Clare, A., and King, R. D. Knowledge discovery in multi-label phenotype data. In *5th European Conf. on Principles Data Mining and Knowledge Discovery* (2001), 42–53.
9. Evgeniou, O., Micchelli, C. A., and Pontil, M. Learning multiple tasks with kernel methods. *J. Machine Learning Research 6* (2005), 615–637.
10. Evgeniou, O., and Pontil, M. Regularized multi–task learning. In *ACM KDD* (2004), 109–117.
11. Jebara, T. Multi-task feature and kernel selection for svms. In *Int'l Conf. on Machine learning (ICML)* (2004).
12. Jiang, L., Yu, M., Zhou, M., Liu, X., and Zhao, T. Target-dependent twitter sentiment classification. In *ACL* (2011), 151–160.
13. Jin, R., and Ghahramani, Z. Learning with multiple labels. In *Conf. on Neural Information Processing Systems (NIPS)* (2003).
14. Lauser, B., and Hotho, A. Automatic multi-label subject indexing in a multilingual environment. In *7th European Conf. in Research and Advanced Technology for Digital Libraries* (2003), 140–151.
15. Lee, K., Palsetia, D., Narayanan, R., Patwary, M. M. A., Agrawal, A., and Choudhary, A. Twitter trending topic classification. In *IEEE IDCM Workshops* (2011), 251–258.
16. McCallum, A. K. Multi-label text classification with a mixture model trained by em. In *AAAI Workshop on Text Learning* (1999).
17. McCallum, A. K. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

18. Nishida, K., Banno, R., Fujimura, K., and Hoshide, T. Tweet classification by data compression. In *Int'l Workshop on DETecting and Exploiting Cultural diversiTy on social web* (2011), 29–34.

19. Nishida, K., Hoshide, T., and Fujimura, K. Improving tweet stream classification by detecting changes in word probability. In *ACM SIGIR* (2012).

20. Tai, F., and Lin, H.-T. Multilabel classification with principal label space transformation. *Neural Comput. 24* (2012), 2508–2542.

21. Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., and Li, P. User-level sentiment analysis incorporating social networks. In *ACM KDD* (2011), 1397–1405.

22. Thabtah, F. A., Cowling, P., and Peng, Y. Mmac: A new multi-class, multi-label associative classification approach. In *IEEE ICDM* (2004), 217–224.

23. Tsoumakas, G., and Katakis, I. Multi-label classification: An overview. *Int J. Data Warehousing and Mining 3* (2007), 1–13.

24. Wang, X., Wei, F., Liu, X., Zhou, M., and Zhang, M. Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In *ACM CIKM* (2011), 1031–1040.

25. Xue, Y., Liao, X., Carin, L., and Krishnapuram, B. Multi-task learning for classification with dirichlet process priors. *J. Machine Learning Research 8* (2007), 35–63.