Got Many Labels? Deriving Topic Labels from Multiple Sources for Social Media Posts using Crowdsourcing and Ensemble Learning

Shuo Chang[1], Peng Dai[2], Jilin Chen[2], Ed H. Chi[2]

[1]GroupLens Research
University of Minnesota
schang@cs.umn.edu

[2]Google Research
Google Inc.
daipeng, jilinc, edchi@google.com

ABSTRACT

Online search and item recommendation systems are often based on being able to correctly label items with topical keywords. Typically, topical labelers analyze the main text associated with the item, but social media posts are often multimedia in nature and contain contents beyond the main text. Topic labeling for social media posts is therefore an important open problem for supporting effective social media search and recommendation.

In this work, we present a novel solution to this problem for Google+ posts, in which we integrated a number of different entity extractors and annotators, each responsible for a part of the post (e.g. text body, embedded picture, video, or web link). To account for the varying quality of different annotator outputs, we first utilized crowdsourcing to measure the accuracy of individual entity annotators, and then used supervised machine learning to combine different entity annotators based on their relative accuracy. Evaluating using a ground truth data set, we found that our approach substantially outperforms topic labels obtained from the main text, as well as naive combinations of the individual annotators. By accurately applying topic labels according to their relevance to social media posts, the results enables better search and item recommendation.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Classifier design and evaluation

Keywords

Crowdsourcing; Topic annotator

1. INTRODUCTION

On social media, users discover interesting posts via both search and recommendation algorithms. Since both search and recommendation are often based on topical analysis, accurate topic labeling of social media content is of great importance. Prior work on topic labeling of social media posts often heavily relies on Natural Language Processing and Topic Modeling approaches that analyzes only the text of the post [6, 13, 18]. These techniques face great challenges since social media text is often short and noisy, while they are typically designed originally for longer documents [3]. These approaches also leave out other valuable information in social media posts, such as associated pictures, links to web pages, YouTube videos, etc. In this paper, we propose to perform topic labeling on Google+ posts by utilizing all the multimedia content available.

Our topic labeling approach builds on multiple automatic topic annotators that work on different parts of a Google+ (G+ for short) post, such as the text body, embedded picture, video and web link (Figure 1). Manual inspection of the topic labels from different annotators shows varying reliability of the annotators. For example, labels from images tend to be less reliable than labels from texts. In addition, the perception of label relevance is subjective and complex. For instance, for a G+ post from TechCrunch (a tech media company headquartered in Silicon Valley) talking about the history of real estate development in San Francisco, an annotator based on author name may label "tech news", an annotator based on text content may label "real estate", and an annotator based on picture may label "city street". It is difficult for an algorithm to decide which label(s) best describe the post, even though each individual annotator is somewhat accurate.

In this paper, we propose to solve the topic labeling problem for social media posts by combining crowdsourcing with supervised ensemble learners. We first utilize crowdsourcing to collect a ground truth data set on the labels' relevancies for randomly-sampled posts, so as to quantify the reliability of each topic annotator. Using ground truth data from crowdsourced labels, we then utilize supervised ensemble models that combine the outputs of various topic annotators, thus further filter and classify topic labels based on their degree of relevance. In an evaluation, we demonstrate that the ensemble model improves over a baseline that naively aggregates topic labels from all annotators. Moreover, the ensemble model is also capable of accurately classifying topic labels into more fine-grained relevance categories.

In summary, the contributions of our paper are:

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2745401.

Figure 1: An example of a Google+ post with a Youtube video attached. The text of the post does not match with the attached video, which makes topic labeling difficult.

- To the best of our knowledge, we are the first that fully utilizes the various portions of multimedia content for topic labeling of social media posts. Prior work on topic labeling has been limited primarily to text-based approaches.
- To effectively utilize multimedia content, we propose a solution that combines distinct topic annotators through an ensemble model trained from relevance judgments obtained via crowdsourcing.
- In an evaluation, we show that the combination of crowdsourcing and ensemble learning led to a better topic labeling solution for Google+ posts.

This paper is structured as follows. We first survey related work in Section 2. In Section 3, we give an overview of our approach. We first describe the problem of topic labeling for social media posts with multimedia content in Section 3.1. Then we focus on both how we performed crowdsourcing to obtain relevance judgment for the individual annotators in Section 3.2. Finally, we describe how we built the ensemble models in Section 3.3.

In Section 4, we evaluated our method on two tasks: Main or important / not main nor important binary classification of labels, and main or important / relevant / off-topic / Don't Know four-class classification of topic labels. We conclude with a discussion of the result and implication to similar applications in Section 5.

2. RELATED WORK

Our work draws from two broad areas of prior research: one being topic extractors and annotators (for text, image, video), and the other being crowdsourcing. Here we briefly introduce these two areas, and end with a detailed introduction of [18], which described a Twitter-based topic labeling problem that is closely related to our work.

2.1 Topic Extractors and Annotators

The research of topic extractors and annotators aims to assign meaningful topic labels to various types of content, including text, image, and video. Traditionally these research efforts are separated by these content types, each of which requires entirely different underlying algorithms and approaches. We briefly survey research on these annotators here.

For text, topic extraction has a long history, particularly in semantic web research. In social media, prior work has mostly relied on text-based approaches, where the primary challenge is the shortness and noisiness of social media text. Bontcheva and Rout [3] gave a detailed summary of prior work. To name a few, Ramage et al. [13] used semi-supervised labeled Latent Dirchlet Allocation (LDA) model to map Tweets into topic dimensions. Ritter et al. [14] used labeled LDA in natural language processing tasks, i.e., Part of Speech Tagging, Named Entity Recognition (NER) and taxonomy of entities, on Tweets. More recently, Gattani et al. [6] introduced a system that extracts entities from Tweets, links entities to concepts in Wikipedia, and classifies Tweets into 23 predefined categories.

Image annotation is closely related to the larger area of image recognition. There are several pieces of work on annotating images with topic labels [7, 11, 17]. In particular, Weston et al. [17] proposed a scalable and efficient method applied that annotates image by learning joint embedding space for images and topic labels.

For annotating videos, topic labels is extremely challenging [2, 4]. In one example research that relates more directly to our work, Aradhye et al. [2] used both meta data, e.g., video title, description and tags, and audiovisual features to find topic labels for YouTube videos.

Our work does not attempt to improve these specialized topic extractors and annotators; instead, we utilize these annotators and focus on how we can intelligently integrate them for topic labeling of social media posts.

2.2 Crowdsourcing

Crowdsourcing has been a widely-applied approach for human evaluation [9] and for obtaining large-scale training data for machine learning systems. We will present a brief survey of some recent work.

Ensuring quality of crowdsourced work is an important issue, because individual crowd workers are not always reliable. For instance, Kittur and Chi [9] showed that some crowd workers are not reliable because they optimize for maximum profit with minimum work. They improved quality of crowd work by adding verifiable questions to HIT template on Amazon Mechanical Turk (MTurk for short). Kazai et al. [8] studied various factors that affect the quality of relevance judgment task for web search. These factors include conditions of pay, required effort, and selection of workers based on proven reliability. In addition, they found the in-

trinsic factors of workers, e.g., motivation, expertise, etc., also relate to work quality.

A number of prior works have in particular studied using crowdsourcing for annotating social media content, such as text, image and video. Finin et al. [5] studied how to efficiently annotate Named Entities in large volumes of Tweets at low cost using MTurk and CrowdFlower. They found both MTurk and CrowdFlower easy to use, cost effective and capable of producing qualified data. On the other hand, Alonso et al. [1] used crowdsourcing to annotate the level of interestingness of Tweets and found the task very challenging, because agreement among workers is low. Our work build upon these works by following the insights and best practices in utilizing crowdsourcing for training data collection.

2.3 Topic Labeling for Twitter

Perhaps the most relevant work to our work is recent research by Yang et al. [18], which proposed a topic labeling system deployed in Twitter. This system has several components: non-topical tweet detection, automatic labeled data acquisition, evaluation with human computation, diagnostic and corrective learning and topic inference. As part of this system, an integrative model aggregates signals from different parts of the tweets, i.e., text, web page, author, hashtag and user interest. The model assigns weights to different sources, where weights come from human-labeled data.

Similar to their work, we train an ensemble model to aggregate topic labels that are generated by different annotators. Our work differs from theirs in one fundamental aspect: our work deals with versatile multimedia posts in Google+ as opposed to short textual tweets. In recent social media system redesigns, multimedia content is becoming more and more common in posts. Therefore, our work is timely in investigating this challenging problem.

In the next section, we describe the specific challenges of topic labeling for multimedia posts and rationale behind our solution to these challenges. In particular, we explain how the multimedia nature of posts complicates the relevance judgment of topic labels.

3. OUR APPROACH

Here we introduce our approach to label multimedia posts for Google+. The central piece of our system integrates different annotators for various parts of a post, i.e., author name, text boy, attached image, attached video and etc., and merges the varying signals from these different annotators by using a ensemble learner, with ground truth topic labels that are evaluated by crowdsourced workers.

Because the judgment of relevance of topic labels for posts can be highly subjective, we employ many crowd workers to evaluate the relevance of topic labels to aggregate a variety of opinions. This, however, raises the challenge of ensuring the quality of the work on subjective tasks as observed in the early work on crowdsourcing [9]. We will describe our crowdsourcing process that carefully addresses this issue in Section 3.2.

To harness all topic annotators with varying accuracy, we build a supervised ensemble model to filter topic labels from each annotator based on its accuracy and other features from posts. We train the supervised learning model on data from crowdsourcing process. We cover the details of the ensemble model in Section 3.3.

The work flow is summarized in Figure 3: we first leverage the crowd to evaluate relevance of topic labels from different annotators on randomly sampled posts, then train the supervised learning model with data from crowdsourcing, and finally use the ensemble model to classify topic labels from different annotators on unseen Google+ posts.

3.1 Single-Source Annotators

A post on G+ contains author name, body text, comments and optional multimedia attachment of image, video or link to web page. Figure 3.1 shows the interface for creating a new post.

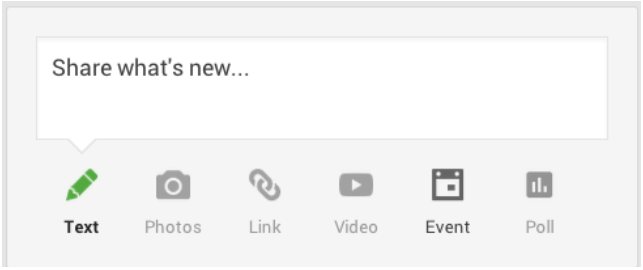

Figure 3: Google+ allows users to attach multiple types of attachment (i.e., image, video and link to web page) to a post.

Intuitively, we can use single-source annotators to annotate each part of a post and combine the labels from these annotators.

To analyze each of the parts of a post, entity/topic annotators map media content onto a particular set of topic keywords. In this work we rely on the Freebase[1] knowledge base system that provides a shared collection of topic keywords that all the individual text annotators, image annotators, and video annotators use [2, 17, 18]. Regardless of the underlying implementation of these single-source annotators, the output of the annotators can be represented as pairs of label and relevance as shown in Equation 1:

$$\{< l_1 : T_{p,l_1} >, ..., < l_i : T_{p,l_i} >, ...\} \quad (1)$$

where l_i is the label, and T_{p,l_i} is the topical relevance of the label l_i to post p. We call T_{p,l_i} topicality score.

There are two major challenges that complicate the aggregation of these topic labels:

First, the single-source annotators have varying levels of reliability, because they are optimized for single source of input (text, image or video). For instance, recognizing a cat from image is technically more difficult than extracting the word "cat" from text. Moreover, annotators provide inconsistent topicality scores due to contextual differences in how they are applied. For instance, textual entity extractors generally work better with longer pieces of text than social media posts, which tend to be quite short. For another example, an annotator applied to the author-name field value of "Big Cat Rescue" is likely to return the 'cat' label, when the account really promotes protection on 'tigers'.

Second, humans may perceive topic of a post differently based on their background knowledge and understanding of the context. When judging whether a topic label is relevant,

[1]https://www.freebase.com/

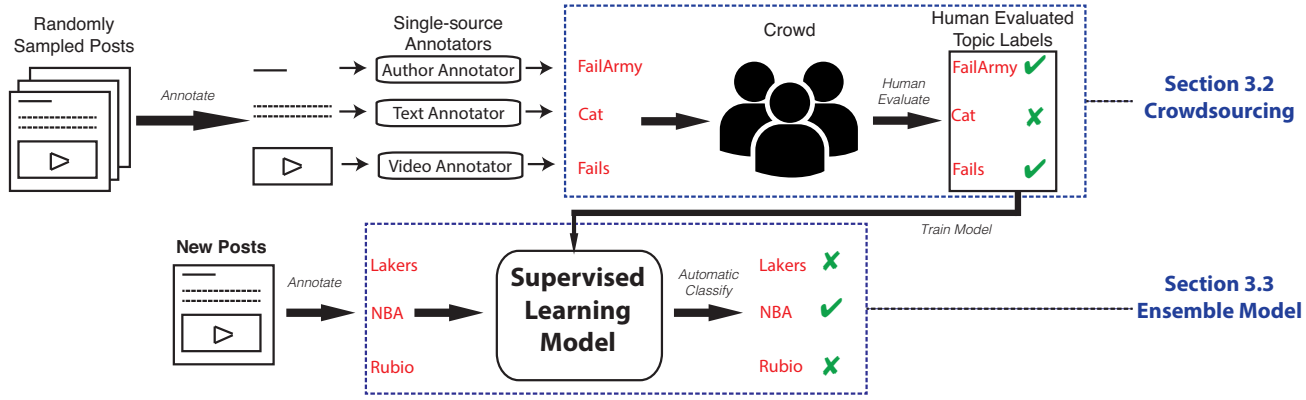

Figure 2: Approach overview. We crowdsource relevance judgment of topic labels (described in Section 3.2). Then we train a supervised ensemble model with human evaluated topic labels (described in Section 3.3)

.

humans consider a post as an integral object and weigh text, image, and video in the post differently. For example, Figure 1 shows a Google+ post with a YouTube video: "Cat" is a label for the body text while both the author name and video show the topic is about "Fails". When presented to people, "Cat" is more likely to be perceived as irrelevant, since the video attracts more attention than the text.

An ensemble model combines the topic labels from different annotators, by applying an informed decision-making process based on reliability of annotators and various features of the post, such as the attachment type. For this reason, we choose to train and apply an ensemble model to aggregate topic labels from multiple annotators. To do this, we must first obtain ground truth labels for the supervised machine learning algorithms.

3.2 Crowdsourcing Training Labels

We use crowdsourcing to obtain relevance judgment on entities as topic labels, generated from multiple single-source annotators, on randomly sampled posts. These human evaluated labels serve as the training data for the ensemble models.

3.2.1 Task UI

We use Amazon's Mechanical Turk (MTurk), a popular crowdsourcing platform, to collect topic label evaluation. In one task, we ask workers to evaluate all entities extracted by all single-source annotators for a single Google+ post. Figure 4 shows the crowdsourcing template UI we used. We pay workers 15 cents for the 1 to 2 minutes tasks.

We ask the worker which category best describes how each label is related to the post, choosing from four categories, "Main or Important", "Related", "Off-topic" and "Don't know", each with detailed definitions.

We request fine-grained relevance category judgments to provide some flexibility later in how the labels are used. For example, a search engine may index posts using only "Main or Important" labels, due to its stringent requirements on accuracy.

We also provide toggleable instructions (partial text below) and examples to minimize ambiguity. To gain context, a worker can optionally hover her mouse over a label to read its English description (shown in black popover).

> *A Google+ post may contain text, images, videos or links to external web pages. Please consider the entire content of the post when answering the questions in this task. If the post contains a link or a video, please click through to get a better understanding of the post.*

3.2.2 Quality Control for Crowdsourcing

Crowdsourcing pipelines are subject to spamming and other low quality work, and the task of relevance judgments for topic labels is no exception. Irresponsible workers, aiming to maximize their profit, game the system by filling the forms with random answers. This is especially difficult, since we do not have any ground truth data set to measure worker performance. To control the quality of crowd work, therefore, we need to control worker quality by avoiding the spammers. We accomplish this chiefly in two ways:

First, a single MTurk worker can perform work on at most 5% of the labeling tasks. A spammer typically performs many tasks as quickly as possible to maximize their profit without actually paying any attention to the task. Though being the minority [16], spammers usually complete more tasks than honest workers due to low effort in spamming behavior. Note that MTurk does not provide this limiting feature. As a result, we set up a gateway server to keep track of the work histories of workers and disable their access to new tasks once they reach our preset limit.

Second, we follow the approach introduced by Kittur et al. [9] and add in verifiable questions for each task. As shown in Figure 4, we ask workers each topic label is relevant to which part of the post. We inform workers that we will check the correctness of their answers with this instruction:

> *You are expected to spend on average 1 to 2 minutes on each HIT. If you don't spend enough time, your HIT will be rejected. Some of the questions have correct answers. If your answers don't match with correct answers, your HIT will be rejected.*

To verify the effectiveness of verifiable questions, we conducted an quick A/B experiment. We randomly sampled 300 Google+ posts, created identical tasks (same payment

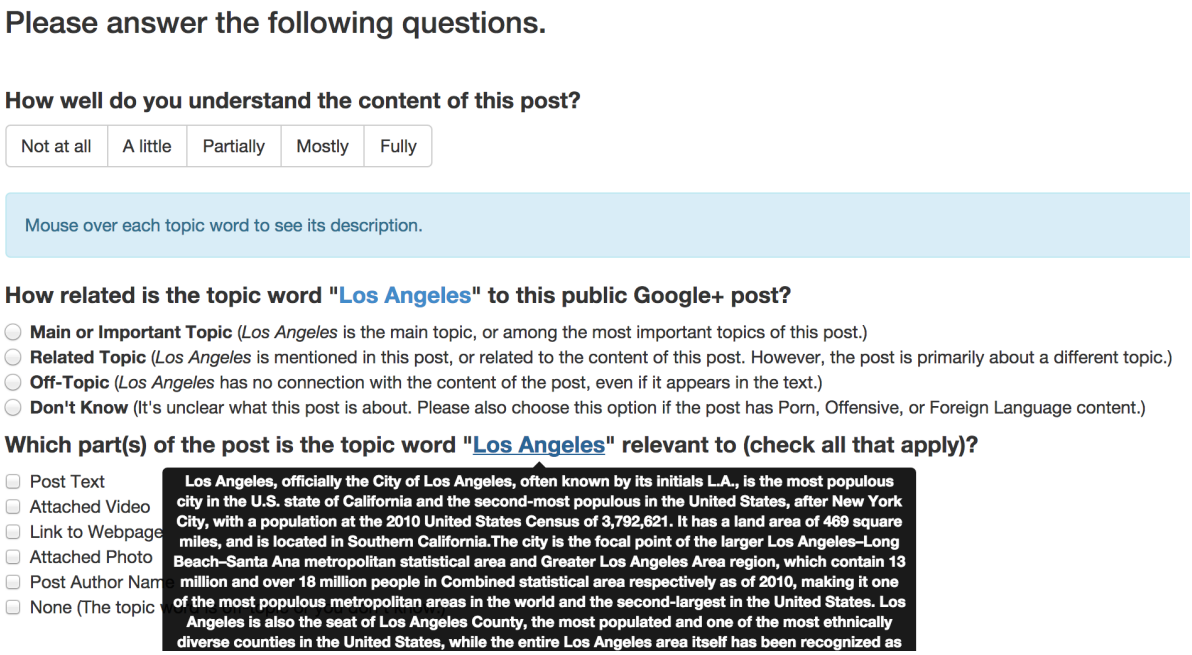

Figure 4: Task template used on MTurk to evaluate relevance of topic labels to a Google+ post. The post, left out from the screen shot, is embedded in the task in similar way as Figure 1. Workers can play videos and click onto web page in the post.

and description) for these posts with two UIs, but one with verifiable questions and the other without. We had three independent workers judge each post.

	No VQ	VQ
Median time per post	1.3 min	**1.6 min**
Chances of unanimous agreement	34.9%	**35.5%**
Chances of majority agreement	88.0%	**90.1%**

Table 1: Comparison of two answer statistics between with and without verifiable questions("VQ" and "No VQ"). Workers spend longer time on tasks and have higher chance to reach agreement when there are verifiable questions.

The result, as summarized in Table 1, indeed shows the benefit of adding verifiable questions. The median time workers spend on tasks is longer and workers have better agreement when verifiable questions exist. Without ground truth answers, better agreement makes us more confident of answer quality. The verifiable questions helped encourage workers to carefully evaluate each label, as part of their decision process of relevance. To correctly answer verifiable questions, spammers need to spend same amount of effort to finish tasks. Together with clear warning about rejection, verifiable questions discourage spamming behavior.

The output of this crowdsourcing process helps us prepare a human-labeled ground truth data set for the ensemble model in next section.

3.3 Supervised Ensemble Model

With human-evaluated topic labels from the crowdsourcing process, we train an ensemble model to combine topic labels from different topic annotators. In this section, we describe the details of the ensemble model.

For each G+ post, the ensemble model takes in the topic labels generated from various single-source annotators and generates a filtered set of topic labels for the post. We have single-source annotators for author name, body text, comment, image, video and link to web page respectively.

The entire process can be modeled as a classification task: predicting the relevance class of the candidate topic labels for a post. Depending on the applications of the ensemble model, the classification task can be configured differently to either generate two-class or multi-class classification of the topic labels.

Binary Classification for Only "Important" Labels: We want to be able to select a topic label only when it is central and important to the post. From all topic labels generated from various annotators, select only "Main or Important" topic labels. In other words, predicting a topic label as positive when it is "Main or Important" and as negative when it is any one of "Relevant", "Off-topic", or "Don't Know" categories. These topic labels most accurately describe the content of posts, helpful in a number of applications. For example, search engine may index posts with these topic labels for highly relevant search result.

Multiclass Classification into All Categories: In this case, we want to actually categorize topic labels into all four categories: "Main or Important", "Relevant", "Off-topic" and "Don't Know", corresponding to the choices in the crowdsourcing template. This can be naturally modeled as multiclass classification problem. Such categorization provides useful information about the quality of topic labels, allowing applications selectively use topic labels based on their need. For example, for accuracy-critical tasks like search we may

only use "Main or Important" topic labels for search indexing, while for recommendation tasks we may also include "Relevant" topic labels so as to include a broader range of related topics. For posts with "Don't Know" labels, they should be excluded from being shown to end-users.

3.3.1 Training Features in the Model

Our goal is to learn a general model that is based on the features of a post and the topic labels, essentially the same information used by humans to judge relevance. We extracted features that are readily available and contain helpful information for judging relevance of topic labels. We do not utilize low-level features such as word tokens and image pixels, because these features are already utilized by the single-source annotators. These features, though not comprehensive, are common to most topic label system for social media posts, serving the purpose of validating our proposed system.

- **Topicality scores from single-source annotators (denoted as *topic*).** Single-source annotators generate topic labels with topicality score (between 0 and 1) of the topics as shown in Equation 1. For instance, for a short post like "Go Wolves!", text annotator is not confident that "Wolves" refer to the NBA basketball team "Timberwolves" due to lack of context. Therefore, the topic label "Timberwolves" may have a low topicality score. Furthermore, different annotators assign topicality scores according to different standards: some annotators may be conservative while others may be more optimistic. With human evaluation, the supervised ensemble model can learn how much to place confidence in topicality scores from various annotators. In our system, we use six single-source annotators, resulting in the topicality scores feature vector in Equation 2. A topicality score of 0 means the topic label is not generated from a particular annotator.

$$< T_{author}, T_{comment}, T_{photo}, T_{text}, T_{video}, T_{webpage} > \quad (2)$$

- **Conditional probability with other topic labels on a post (denoted as *prob*).** Because one post mostly talks about one topic, it is unlikely to have very different topic labels to co-appear on one post. For example, "NBA" is very unlikely to appear with "Knitting" on same post. More formally, assuming we already have a topic label L_1 for a post, then the conditional probability of L_2 co-appear with L_1 is computed as Equation 3. With more topic labels we approximate the combined probability using geometric mean.

$$P(L_2|L_1) = \frac{Freq(L_1, L_2)}{Freq(L_1)} \quad (3)$$

- **Length of text in post (denoted as *length*).** We also extract length of text in post, with the intuition that single-source annotators tend to be more accurate on longer posts. We apply log transformation and normalize this feature to be in range of 0 to 1 using the max post length.

- **Type of attachment (denoted as *type*).** The type of attachment in posts also affects how people perceive the topic of the post. For instance, users tend to pay more attention to images and videos when they are present in a post. This is a categorical feature.

- **Whether topic label is a user-provided hashtag (denoted as *hashtag*).** There is good chance that humans would perceive a topic label as relevant if it is one of author provided hashtags. This feature is a binary variable.

For the rest of the paper, we will refer to features by their shorthand notations. For example, $topic, prob$ represents a set of features consists of topicality score vector and conditional probability with other labels.

3.3.2 Classification Algorithm

We wanted to understand the performance of the ensemble model under different classification algorithms. We picked several popular classification algorithms, implemented in a popular machine learning library called 'scikit-learn' [12], for both the binary and multiclass classification problem. The set of algorithms we experimented with are:

- Random Forest (RF). RF is an ensemble learning model that computes the average decision of many decision trees (200 in our case) trained on random samples of data set. RF, same with basic decision trees, can handle multiclass classification.

- Gradient Boosting Classifier (GBC). GBC is an additive model that iteratively adds decision trees (200 total trees in our case) using boosting. Similar with RF, GBC can classify multiple classes.

- Logistic Regression. We use logistic regression with L2 regularization. For multiclass classification, we use the one-vs-rest strategy. This strategy fits a binary classifier for each class, with that class as positive and other classes as negative, and decides the class for data as majority decision of all binary classifiers.

- Support Vector Machine (SVM). We use linear SVM with L2 regularization. Same with Logistic Regression, we apply SVM to multiclass classification using the one-vs-rest strategy.

In the evaluation section below, we picked the best performing algorithm from the above list to compare with baseline method.

4. EVALUATION

In this section, we evaluate the ensemble model on the ground truth constructed from a gold standard data set. In our evaluation, we try to answer following questions:

- How does the ensemble model compare with a naive baseline methods? For binary classification, the baseline is an union predictor, i.e., predicting topic labels by aggregating all annotator outputs. For multiclass classification, the baseline is predicting the most common category for a label.

- What are the different performance of classification algorithms, and what is the best performing ensembling technique?

- How do different features contribute to the ensemble model?

4.1 Evaluation Setup

Data. Using the crowdsourcing process described in Section 2.2, we created a training data set for the ensemble model, as well as a gold standard data set to test the ensemble model.

We first group recent G+ posts (in August 2014) by types of annotators used to generate topic labels. Then we sample uniformly randomly from each group and form a stratified G+ post sample, representative of all annotators. Next, we create crowdsourcing tasks from each sampled post and have N distinct workers answer each task. Finally, we aggregate answers from N workers by taking the majority vote for each task.

For the training data set, we have 5104 topic labels on 2550 posts, with $N = 10$, for a total of 51040 judgments. For the gold standard data set, we have 592 topic labels on 300 posts, with $N = 20$ for more reliability, for a total of 11840 judgments. By increasing number of independent workers on each task, we get more reliable judgements. In a pilot study, we find that quality of work done by 7 Mturk workers is comparable to quality of work done by 3 trained corporate experts. Therefore, we are confident that judgements from 20 workers on Mturk can serve as the gold standard data set.

Experiment. Our evaluation is carried out in several steps:

1. We first train the four classification algorithms (described in Section 3.3.2) on all 31 (2^5-1) combinations of the 5 features (described in Section 3.3.1).
2. Then, we find the best model – the classification algorithm trained on one feature combination that yields the best performance, and compare it to baseline method.
3. Next, we compare the best performing model of each classification algorithm to baseline method.
4. Finally, for the best classification algorithm, we compare the varying performance of different combinations of features.

4.2 Binary Classification for Main Topic Labels

We use standard precision/recall metrics to evaluate performance of binary classification. We train supervised learning models and make predictions on the gold standard data — 592 pairs of topic label and post. The category distribution of the gold standard data is summarized in Table 2. For positive class, i.e., topic label being "Main or Important", we compute precision, recall and $F1$-score, which is

$$\frac{2 \times precision \times recall}{precision + recall}$$

.

Main or Important	Relevant	Off-topic	Don't Know
47.7%	29.5%	18.8%	3.9%

Table 2: Unbalanced class distribution in gold standard data set.

Best Ensemble Model. For each one of the four classification algorithms, we trained 31 models on various combinations of features. The best performing model, as measured by $F1$-score, is Gradient Boosting Classifier (GBC) trained on all five features, i.e., topic, prob, length, hashtag, and type.

To understand the relative performance of the ensemble models in aggregating various single-source annotators, we compare the performance of the ensemble models to single-source annotators and a baseline method that classifies topic labels from all annotators as "Main or Important", which we refer as the All Annotator Baseline.

Annotator	F1	Precision	Recall
Author Name	0.007	0.111	0.003
Comment	0.007	0.250	0.003
Photo	0.049	0.242	0.027
Post Text	0.402	0.497	0.337
Video	0.630	0.708	0.568
Web Link	0.293	0.390	0.235
All Annotator Baseline	0.664	0.497	1.000
Ensemble Model	**0.717**	**0.691**	**0.745**

Table 3: Comparison of the best performing ensemble model and single-source annotators, listed above. "All annotator" predicts the union of topic labels from single-source annotators as positive. The best performing algorithm is GBC trained on all five features, having the best overall $F1$-score here.

Table 3 summarizes the results of this comparison, which shows that the best ensemble model has the highest $F1$-score. There are three findings from this table:

- Our best ensemble model has the best overall performance (0.717 $F1$-score) in comparison with baseline method and all single-source annotators. The ensemble model is significantly more precise than the baseline method (0.691 compared to 0.497), close to the most precise video annotator (0.708). Though having worse recall than baseline, the ensemble model has the best recall compared to any single-source annotator.
- The first six rows of the table shows varying reliability of different annotators. The annotator that extracts topic labels from author name has precision as low as 0.111, while the most accurate video annotator has precision of 0.708. A big portion of Google+ posts, however, does not contain videos, resulting in the poor recall of 0.568.
- The baseline method that blindly takes union of topic labels from all annotators, though having perfect recall, has unsatisfying precision (0.497). This can be attributed to the fact that baseline method also includes inaccurate topics from unreliable annotators.

In summary, the ensemble model can aggregate topic labels from unreliable annotators and identify relevant labels based on features from post and topic labels.

Classification Algorithm. After exhaustively training on all combinations of features, we obtain the best models for four algorithms when trained on the set of all 5 features. We find that the best models of the four classification algorithms all outperform the baseline method, as is shown in Figure 5. The four models consistently have higher $F1$-score than baseline, with GBC having the highest $F1$-score. In other words, regardless of implementations of ensemble model, we can improve over the naive baseline.

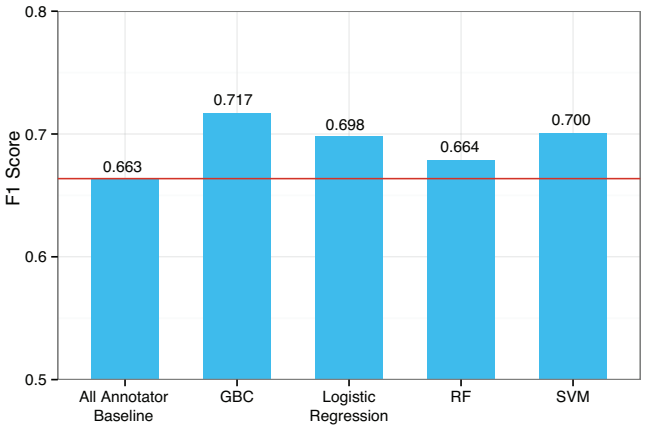

Figure 5: Comparison of best models of each classification algorithm on binary classification task. The four best models are trained on all five features after exhaustively searching feature combinations. Y axis shows the $F1$-score. GBC has the highest $F1$-score.

Feature Analysis. Both features about topic labels and features about post provide useful information. Due to space constraint, we only show the comparison results of models trained on 31 feature combinations for the best performing GBC algorithm. Figure 6 depicts the GBC models trained on combinations of one to five features. We find that topicality scores to be the most powerful single feature, resulting in 0.689 $F1$-score. Adding hashtag post features (*hashtag*) and type of attachment (*type*) further improved performance. Conditional probability consistency with other labels (*prob*) and length of text (*length*) did not have significant effects.

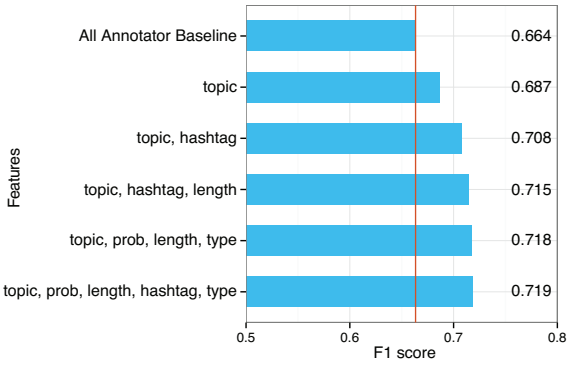

Figure 6: Feature analysis for GBC model on binary classification task. We show best performing models trained on 1 to 5 features. The x axis shows the $F1$-scores of models.

Combining these results, we find that our ensemble model, which aggregates single-source annotators using supervised machine learning, is better at classifying topic labels that are central to posts than any single-source annotator and the naive baseline method.

Algorithm	Features	F1
Baseline	N/A	0.308
Logistic	topic,hashtag,length	0.498
SVM	topic,hashtag	0.475
RF	topic,hashtag,prob,length	0.526
GBC	**topic,hashtag,prob**	**0.547**

Table 4: The $F1$-scores of the best-performing model of every classification algorithm for multiclass classification along with their feature combinations. GBC performs the best. All ensemble learning algorithms outperform the baseline algorithm, which consistently predicts the most popular category, "Main or Important".

4.3 Multiclass Classification of Topic Labels

For this part of the evaluation, we extend our evaluation metrics so as to handle multiclass classification of all classes (i.e., "Main or Important", "Relevant", "Off-topic" and "Don't Know"). As the distribution of topic labels is unbalanced across classes (Table 2), we compute precision, recall and $F1$-score for each one of the four classes and take a weighted average to get averaged precision, recall and $F1$-score, where the weights are frequencies of the four classes (as shown in Equation 4).

$$F1 = \frac{n_1}{n}F1_1 + \frac{n_2}{n}F1_2 + \frac{n_3}{n}F1_3 + \frac{n_4}{n}F1_4 \quad (4)$$

In the equation, $F1_i$ denotes $F1$-score of classifying class i, n_i denotes the number of test data in class i and n denotes total the number of test data.

Under this setup, we introduce a baseline method that always predict the most common label "Main or Important" for all input data, which we refer as the Common Label Baseline.

Best Ensemble Model. Similar with the evaluation for binary classification, we train the four classification algorithms on 31 combinations of five features. Comparing to the binary classification case, the best ensemble model has more significant improvement over the baseline method. The best model, GBC trained on topic, hashtag, prob shown in Table 4, has highest $F1$-score of 0.547 in comparison to 0.308 $F1$-score of baseline.

Classification Algorithm. Best models of four classification algorithms all have significant improvement over the baseline method, more details shown in Figure 7. We achieve the best performance of each classification algorithm on different combinations of features as shown in Table 2. This is different from binary classification, where all classification algorithms perform best on full set of features. Overall, ensemble models outperform the baseline for all classification algorithms.

Feature Analysis. Consistent with the result in binary classification, we observe the strong predictive power of topic label feature, topicality scores (0.529 $F1$-score with single feature), and post feature, length of post (0.542 $F1$-score with *(*topic, length), as shown in Figure 8.

However, we notice that *hashtag, prob, type* do not contribute to the classification. One possible explanation is that topicality score is most informative about degree of relevance, which is the basis for multiclass classification task.

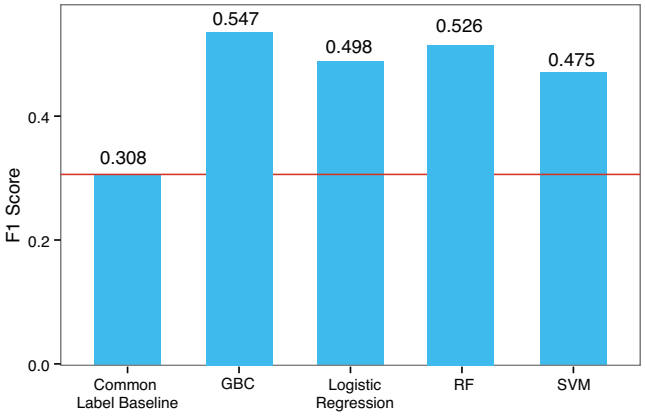

Figure 7: Comparison of best models of each prediction algorithm for multi-class classification of topic label relevance. The features used by best models are summarized in Table 4. Y axis shows the $F1$-score. GBC has the highest $F1$-score.

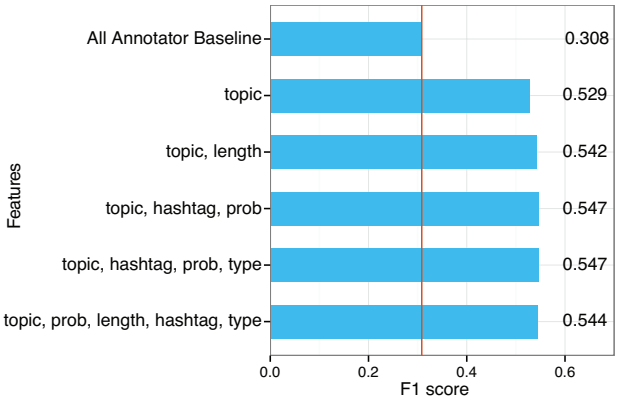

Figure 8: Feature analysis for multi-class classification of topic label relevance. The x axis shows $F1$-score of GBC algorithm trained on different sets of features. Feature of topicality scores *topic* has the most predictive power. Somewhat surprisingly, *prob, hashtag, type* bring little improvement.

Other features do not contain much information about degree of relevance.

In summary, our ensemble model is significantly better than naive baseline method in classification of topic labels. Trained with labeled data from crowdsourcing process, the ensemble model is capable of classify the relevance of unseen topic labels on new posts with decent performance.

5. DISCUSSION

The results show that our crowdsourced ensemble model significantly outperforms baseline methods for both binary and multiclass classification of topic labels. In other words, by integrating topic annotators on different parts of the posts, we have substantially improved topic label quality for Google+ posts.

Our approach also allows downstream applications like search and recommendation to selectively use the topic labels based on their specific needs. Users care very much about accuracy when they search for a topic word, therefore, search engine can index Google+ posts with only "Main or Important" topic labels. On the other hand, recommendation engines often aim to encourage users to explore more broadly, so may prefer to use topic labels more loosely, including topic labels that are either "Main or Important" or "Relevant". In this way, users will able to see more novel and serendipitous recommendations [15].

The main technical challenges in the work is effectively aggregating different annotators, where the topicality score from one annotator might be inconsistent and incomparable to the score from another annotator. To address this problem, we use crowdsourcing to help evaluate topic labels from different annotators, so that the ensemble model trained with these judgments is able to capture the variability of accuracy of annotators. Topicality scores from annotators themselves are often not sufficient. As the example shown in Figure 1, even if we had perfect annotators for all parts of the post, human judgment on topics of the post requires thinking about what the components of the post are and how topics from different parts relate to each other. In the example, the agreement between author name and video content confirms that the topic is about "fail video". Accordingly, we have found that additional features, including length of text, type of attachment and whether is hashtag, often further improve the ensemble model.

There are several limitations of the methods employed in this research. First, most importantly, the performance of our supervised learning model is bounded by the quality of evaluations from crowd workers. Due to the subjective nature of relevance judgment, the quality control of workers is challenging. Though we have employed carefully designed task template and applied several control techniques, it is by no means the best solution. Using more intelligent crowdsourcing systems that adaptively employ workers based on task difficulty and worker expertise can potentially improve the quality of crowd work. Second, in addition, we use out-of-box implementation of several supervised learning algorithm. There is potential of getting better performance by carefully tuning the algorithm.

In the future, we will study how to optimize the performance of the whole system under a fixed budget. We used 10 independent workers to evaluate each post in the crowdsourcing process. Under a fixed budget, we can reduce the number of workers, possibly decrease the overall reliability of human evaluations, instead get more posts evaluated. We should study the tradeoff between quality of human judgments and the number of posts evaluated and how it affects the accuracy of ensemble model. In fact, recent work [10] found that different supervised learning algorithm benefit differently from more labels.

Another direction we will work on is to introduce active learning into the system, making the system intelligently choose posts and topic labels to evaluate in crowdsourcing and then incrementally update the supervised learning model on-the-fly.

6. CONCLUSION

In this paper, we proposed a novel system to provide topic labels for multimedia posts on Google+ by utilizing crowd-

sourcing and supervised ensemble learning. The system aggregates different single-source annotators, each extracting topic labels from one part of the post (e.g., text, picture or video). We use crowdsourcing to evaluate how relevant topic labels are on a sample of Google+ posts. The crowdsourced judgments enable us to understand the varying reliability of the single-source annotators. We train an ensemble model on the data obtained from crowdsourcing process.

Evaluating on a gold standard data set, we find the ensemble model outperforms baseline method that naively combines topic labels from all annotators in classifying topic labels that are "Main or Important" topics. The ensemble model also significantly outperforms a baseline method in multiclass classification of topic labels into relevance categories.

Important user functions such as search and recommendation will benefit from better topic labels. By greatly improving the performance of how we apply topic labels to social media posts, it is our hope that users will enjoy more relevant and interesting posts.

7. ACKNOWLEDGEMENTS

We would like to thank Amazon Mechanical Turk workers for their participation in this study. We also thank Lichan Hong and Zhiyuan Cheng for thoughtful discussions and valuable feedback.

References

[1] O. Alonso, C. C. Marshall, and M. Najork. Are Some Tweets More Interesting Than Others? #HardQuestion. In *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval - HCIR '13*, pages 1–10, New York, New York, USA, Oct. 2013. ACM Press.

[2] H. Aradhye, G. Toderici, and J. Yagnik. Video2text: Learning to annotate video content. In *Proceedings of the 2009 IEEE International Conference on Data Mining Workshops*, ICDMW '09, pages 144–151, Washington, DC, USA, 2009. IEEE Computer Society.

[3] K. Bontcheva and D. Rout. Making sense of social media streams through semantics: a survey. *Semantic Web*.

[4] S. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–1002. IEEE, 2004.

[5] T. Finin, W. Murnane, A. Karandikar, N. Keller, J. Martineau, and M. Dredze. Annotating named entities in Twitter data with crowdsourcing. pages 80–88, 2010.

[6] A. Gattani, D. S. Lamba, N. Garera, M. Tiwari, X. Chai, S. Das, S. Subramaniam, A. Rajaraman, V. Harinarayan, and A. Doan. Entity extraction, linking, classification, and tagging for social media: A wikipedia-based approach. *Proc. VLDB Endow.*, 6(11):1126–1137, Aug. 2013.

[7] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 119–126. ACM, 2003.

[8] G. Kazai, J. Kamps, and N. Milic-Frayling. An analysis of human factors and label accuracy in crowdsourcing relevance judgments. *Information Retrieval*, 16(2):138–178, 2013.

[9] A. Kittur, E. H. Chi, and B. Suh. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 453–456, New York, NY, USA, 2008. ACM.

[10] C. Lin and D. Weld. To Re (label), or Not To Re (label). 2014.

[11] J. Liu, R. Hu, M. Wang, Y. Wang, and E. Y. Chang. Web-scale image annotation. In *Proceedings of the 9th Pacific Rim Conference on Multimedia: Advances in Multimedia Information Processing*, PCM '08, pages 663–674, Berlin, Heidelberg, 2008. Springer-Verlag.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[13] D. Ramage, S. Dumais, and D. Liebling. Characterizing microblogs with topic models. 2010.

[14] A. Ritter, S. Clark, Mausam, and O. Etzioni. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[15] G. Shani and A. Gunawardana. Evaluating recommendation systems. In F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors, *Recommender Systems Handbook*, pages 257–297. Springer US, 2011.

[16] J. Vuurens, A. P. de Vries, and C. Eickhoff. How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *Proc. ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIRâĂŹ11)*, pages 21–26, 2011.

[17] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, pages 2764–2770. AAAI Press, 2011.

[18] S.-H. Yang, A. Kolcz, A. Schlaikjer, and P. Gupta. Large-scale high-precision topic modeling on twitter. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14*, pages 1907–1916, New York, New York, USA, Aug. 2014. ACM Press.