Predicting Human Activities from User-Generated Content

Abstract

The activities we do are linked to our interests, personality, political preferences, and decisions we make about the future. In this paper, we explore the task of predicting human activities from user-generated content. We collect a dataset containing instances of social media users writing about a range of everyday activities. We then use a state-of-the-art sentence embedding framework tailored to recognize the semantics of human activities and perform an automatic clustering of these activities. We train a neural network model to make predictions about which clusters contain activities that were performed by a given user based on the text of their previous posts and self-description. Additionally, we explore the degree to which incorporating inferred user traits into our model helps with this prediction task.

Introduction

What a person does says a lot about who they are. Information about the types of activities that a person engages in can provide insights about their interests BIBREF0 , personality BIBREF1 , physical health BIBREF2 , the activities that they are likely to do in the future BIBREF3 , and other psychological phenomena like personal values BIBREF4 . For example, it has been shown that university students who exhibit traits of interpersonal affect and self-esteem are more likely to attend parties BIBREF5 , and those that value stimulation are likely to watch movies that can be categorized as thrillers BIBREF6 .

Several studies have applied computational approaches to the understanding and modeling of human behavior at scale BIBREF7 and in real time BIBREF8 . However, this previous work has mainly relied on specific devices or platforms that require structured definitions of behaviors to be measured. While this leads to an accurate understanding of the types of activities being done by the involved users, these

methods capture a relatively narrow set of behaviors compared to the huge range of things that people do on a day-to-day basis. On the other hand, publicly available social media data provide us with information about an extremely rich and diverse set of human activities, but the data are rarely structured or categorized, and they mostly exist in the form of natural language. Recently, however, natural language processing research has provided several examples of methodologies for extracting and representing human activities from text BIBREF9 , BIBREF10 and even multimodal data BIBREF11 .

In this paper, we explore the task of predicting human activities from user-generated text data, which will allow us to gain a deeper understanding of the kinds of everyday activities that people discuss online with one another. Throughout the paper, we use the word "activity" to refer to what an individual user does or has done in their daily life. Unlike the typical use of this term in the computer vision community BIBREF12 , BIBREF13 , in this paper we use it in a broad sense, to also encompass non-visual activities such as "make vacation plans" or "have a dream" We do not focus on fine-grained sequences actions such as "pick up a camera", "hold a camera to one's face", "press the shutter release button", and others. Rather, we focus on the high-level activity as a person would report to others: "take a picture". Additionally, we specifically focus on everyday human activities done by the users themselves, rather than larger-scale events BIBREF14 , which are typically characterized by the involvement or interest of many users, often at a specific time and location.

Given that the space of possible phrases describing human activities is nearly limitless, we propose a set of human activity clusters that summarize a large set of several hundred-thousand self-reported activities. We then construct predictive models that are able to estimate the likelihood that a user has reported that they have performed an activity from any cluster.

The paper makes the following main contributions. First, starting with a set of nearly 30,000 human activity patterns, we compile a very large dataset of more than 200,000 users undertaking one of the

human activities matching these patterns, along with over 500 million total tweets from these users. Second, we use a state-of-the-art sentence embedding framework tailored to recognize the semantics of human activities and create a set of activity clusters of variable granularity. Third, we explore a neural model that can predict human activities based on natural language data, and in the process also investigate the relationships between everyday human activities and other social variables such as personal values.

Data

While we do not expect to know exactly what a person is doing at any given time, it is fairly common for people to publicly share the types of activities that they are doing by making posts, written in natural language, on social media platforms like Twitter. However, when taking a randomly sampled stream of tweets, we find that only a small fraction of the content was directly related to activities that the users were doing in the real world – instead, most instances are more conversational in nature, or contain the sharing of opinions about the world or links to websites or images. Using such a random sample would require us to filter out a large percentage of the total data collected, making the data collection process inefficient.

Therefore, in order to target only those tweets that are rich in human activity content, we formulate a set of queries that allows us to use the Twitter Search API to find instances of users tweeting about common human activities. Each query contains a first-person, past-tense verb within a phrase that describes a common activity that people do. Using this approach, we are able to retrieve a set of tweets that contains a high concentration of human activity content, and we also find that users who wrote these tweets are much more likely to have written other tweets that describe human activities (Table TABREF1). We build our set of human activity queries from two sources: the Event2Mind dataset BIBREF15 and a set of short activity surveys, which we collect ourselves, to obtain nearly 30K queries (Table TABREF2) .

Event2Mind Activities

The Event2Mind dataset contains a large number of event phrases which are annotated for intent and reaction. The events themselves come from four sources of phrasal events (stories, common n-grams found in web data, blogs, and English idioms), and many of them fall under our classification of human activities, making Event2Mind a great resource in our search for concrete examples of human activities. We consider events for which a person is the subject (e.g, "PersonX listens to PersonX's music") to be human activities, and remove the rest (e.g., "It is Christmas morning"). We then use several simple rules to convert the Event2Mind instances into first-person past-tense activities. Since all events were already filtered so that they begin with "PersonX", we replace the first occurrence of "PersonX" in each event with "I" and all subsequent occurrences with "me". All occurrences of "PersonX's" become "my", and the main verb in each phrase is conjugated to its past-tense form using the Pattern python module. For example, the event "PersonX teaches PersonX's son" becomes the query "I taught my son". Since Event2Mind also contains wildcard placeholders that can match any span of text within the same phrase (e.g., "PersonX buys INLINEFORM0 at the store") but the Twitter API doesn't provide a mechanism for wildcard search, we split the event on the string INLINEFORM1 and generate a query that requires all substrings to appear in the tweet. We then check all candidate tweets after retrieval and remove any for which the substrings do not appear in the same order as the original pattern.

Short Survey Activities

In order to get an even richer set of human activities, we also ask a set of 1,000 people across the United States to list any five activities that they had done in the past week. We collect our responses using Amazon Mechanical Turk, and manually verify that all responses are reasonable. We remove any duplicate strings and automatically convert them into first-person and past-tense (if they were not in that form already). For this set of queries, there are no wildcards and we only search for exact matches.

Example queries obtained using this approach include "I went to the gym" and "I watched a documentary".

Query Results

Using our combined set of unique human activity queries, we use the Twitter Search API to collect the most recent 100 matches per query (the maximum allowed by the API per request), as available, and we refer to these tweets as our set of queried tweets. We then filter the queried tweets as follows: first, we verify that for any tweets requiring the match of multiple substrings (due to wildcards in the original activity phrase), the substrings appear in the correct order and do not span multiple sentences. Next, we remove activity phrases that are preceded with indications that the author of the tweet did not actually perform the activity, such as "I wish" or "should I ...?". We refer to the set of tweets left after this filtering as valid queried tweets (see Table TABREF8 for more details).

In order to gather other potentially useful information about the users who wrote at least one valid queried tweet, we collect both their self-written profile and their previously written tweets (up to 3,200 past tweets per user, as allowed by the Twitter API), and we refer to these as our set of additional tweets. We ensure that there is no overlap between the sets of queried tweets and additional tweets, so in the unlikely case that a user has posted the same tweet multiple times, it cannot be included in both sets.

Further, we use a simple pattern-matching approach to extract additional activities from these additional tweets. We search for strings that match I <VBD> .* <EOS> where <VBD> is any past-tense verb, .* matches any string (non-greedy), and <EOS> matches the end of a sentence. We then perform the same filtering as before for indications that the person did not actually do the activity, and we refer to these filtered matches as our set of additional activities (see Table TABREF11 for more information). Note that since these additional activities can contain any range of verbs, they are naturally noisier than our set of

valid query tweets, and we therefore do not treat them as a reliable "ground truth" source of self-reported human activities, but as a potentially useful signal of activity-related information that can be associated with users in our dataset.

For our final dataset, we also filter our set of users. From the set of users who posted at least one valid queried tweet, we remove those who had empty user profiles, those with less than 25 additional tweets, and those with less than 5 additional activities (Table TABREF12).

Creating Human Activity Clusters

Given that the set of possible human activity phrases is extremely large and it is unlikely that the same phrase will appear multiple times, we make this space more manageable by first performing a clustering over the set of activity phrase instances that we extract from all valid queried tweets. We define an activity phrase instance as the set of words matching an activity query, plus all following words through the end of the sentence in which the match appears. By doing this clustering, our models will be able to make a prediction about the likelihood that a user has mentioned activities from each cluster, rather than only making predictions about a single point in the semantic space of human activities.

In order to cluster our activity phrase instances, we need to define a notion of distance between any pair of instances. For this, we turn to prior work on models to determine semantic similarity between human activity phrases BIBREF16 in which the authors utilized transfer learning in order to fine-tune the Infersent BIBREF17 sentence similarity model to specifically capture relationships between human activity phrases. We use the authors' BiLSTM-max sentence encoder trained to capture the relatedness dimension of human activity phrases to obtain vector representations of each of our activity phrases. The measure of distance between vectors produced by this model was shown to be strongly correlated with human judgments of general activity relatedness (Spearman's INLINEFORM0 between the model and

human ratings, while inter-annotator agreement is INLINEFORM1).

While the relationship between two activity phrases can be defined in a number of ways BIBREF10 , we we chose a model that was optimized to capture relatedness so that our clusters would contain groups of related activities without enforcing that they are strictly the same activity. Since the model that we employed was trained on activity phrases in the infinitive form, we again use the Pattern python library, this time to convert all of our past-tense activities to this form. We also omit the leading first person pronoun from each phrase, and remove user mentions (@<user>), hashtags, and URLs. We then define the distance between any two vectors using cosine distance, i.e., INLINEFORM0 , for vectors INLINEFORM1 and INLINEFORM2 .

We use K-means clustering in order to find a set of INLINEFORM0 clusters that can be used to represent the semantic space in which the activity vectors lie. We experiment with INLINEFORM1 with INLINEFORM2 and evaluate the clustering results using several metrics that do not require supervision: within-cluster variance, silhouette coefficient BIBREF18 , Calinski-Harabaz criterion BIBREF19 , and Davies-Bouldin criterion BIBREF20 . In practice, however, we find that these metrics are strongly correlated (either positively or negatively) with the INLINEFORM3 , making it difficult to quantitatively compare the results of using a different number of clusters, and we therefore make a decision based on a qualitative analysis of the clusters. For the purpose of making these kinds of predictions about clusters, it is beneficial to have a smaller number of larger clusters, but clusters that are too large are no longer meaningful since they contain sets of activities that are less strongly related to one another. In the end, we find that using INLINEFORM4 clusters leads to a good balance between cluster size and specificity, and we use this configuration for our prediction experiments moving forward. Examples of activities that were assigned the same cluster label are shown in Table TABREF15 , and Table TABREF16 illustrates the notion of distance within our newly defined semantic space of human activities. For example, two cooking-related clusters are near to one another, while a photography-related cluster is very distant from

both.

Methodology

Given a set of activity clusters and knowledge about the users who have reported to have participated in these activities, we explore the ability of machine learning models to make inferences about which activities are likely to be next performed by a user. Here we describe the supervised learning setup, evaluation, and neural architecture used for the prediction task.

Problem Statement

We formulate our prediction problem as follows: for a given user, we would like to produce a probability distribution over all activity clusters such that: INLINEFORM0

where INLINEFORM0 is a set of activity clusters, INLINEFORM1 , INLINEFORM2 , and INLINEFORM3 are vectors that represent the user's history, profile, and attributes, respectively, and INLINEFORM4 is the target cluster. The target cluster is the cluster label of an activity cluster that contains an activity that is known to have been performed by the user.

If a model is able to accurately predict the target cluster, then it is able to estimate the general type of activity that the user is likely to write about doing in the future given some set of information about the user and what they have written in the past. By also generating a probability distribution over the clusters, we can assign a likelihood that each user will write about performing each group of activities in the future. For example, such a model could predict the likelihood that a person will claim to engage in a "Cooking" activity or a "Pet/Animal related" activity.

The ability to predict the exact activity cluster correctly is an extremely difficult task, and in fact, achieving that alone would be a less informative result than producing predictions about the likelihood of all clusters. Further, in our setup, we only have knowledge about a sample of activities that people actually have done. In reality, it is very likely that users have participated in activities that belong to a huge variety of clusters, regardless of which activities were actually reported on social media. Therefore, it should be sufficient for a model to give a relatively high probability to any activity that has been reported by a user, even if there is no report of the user having performed an activity from the cluster with the highest probability for that user.

Model Architecture

As input to our activity prediction model, we use three major components: a user's history, profile, and attributes. We represent a history as a sequence of documents, INLINEFORM0 , written by the user, that contain information about the kinds of activities that they have done. Let INLINEFORM1 , and each document in INLINEFORM2 is represented as a sequence of tokens. We experiment with two sources for INLINEFORM3 : all additional tweets written by a user, or only the additional activities contained in tweets written by a user, which is a direct subset of the text contained in the full set of tweets.

A user's profile is a single document, also represented as a sequence of tokens. For each user, we populate the profile input using the plain text user description associated with their account, which often contains terms which express self-identity such as "republican" or "athiest."

We represent the tokens in both the user's history and profile with the pretrained 100-dimensional GloVe-Twitter word embeddings BIBREF21 , and preprocess all text with the script included with these embeddings.

Finally, our model allows the inclusion of any additional attributes that might be known or inferred in order to aid the prediction task, which can be passed to the model as a INLINEFORM0 dimensional real-valued vector. For instance, we can use personal values as a set of attributes, as described in Section SECREF26 .

We train a deep neural model, summarized in Figure FIGREF21 , to take a user's history, profile, and attributes, and output a probability distribution over the set of INLINEFORM0 clusters of human activities, indicating the likelihood that the user has reported to have performed an activity in each cluster. There are four major components of our network:

This is applied to each of the INLINEFORM0 documents in the history– either an activity phrase or a full tweet. For document INLINEFORM1 in INLINEFORM2 , it takes a sequence of token embeddings as input and produces a INLINEFORM3 dimensional vector, INLINEFORM4 as output.

This layer takes the sequence INLINEFORM0 as input and produces a single INLINEFORM1 dimensional vector, INLINEFORM2 , as output, intended to represent high-level features extracted from the entire history of the user.

Takes each token in the user's profile as input and produces a single INLINEFORM0 dimensional vector, INLINEFORM1 as output.

As input, this module takes the concatenation INLINEFORM0 , where INLINEFORM1 is the predefined attribute vector associated with the user. Then, a prediction is made for each of the INLINEFORM2 clusters, first applying softmax in order to obtain a probability distribution. We refer to the dimension of the output as INLINEFORM3 .

For any of the three encoder layers, several layer types can be used, including recurrent, convolutional, or self-attention based BIBREF22 layers. The classifier layer is the only layer that does not take a sequence as input and we implement it using a simple feed-forward multi-layer network containing INLINEFORM0 layers with INLINEFORM1 hidden units each. The network is trained with cross-entropy loss, which has been shown to perform competitively when optimizing for top-k classification tasks BIBREF23 .

Incorporating Personal Values

While the attributes vector INLINEFORM0 can be used to encode any information of interest about a user, we choose to experiment with the use of personal values because of their theoretical connection to human activities BIBREF6 . In order to get a representation of a user's values, we turn to the hierarchical personal values lexicon from BIBREF24 . In this lexicon, there are 50 value dimensions, represented as sets of words and phrases that characterize that value. Since users' profiles often contain value-related content, we use the Distributed Dictionary Representations (DDR) method BIBREF25 to compute a score, INLINEFORM1 for each value dimension, INLINEFORM2 , using cosine similarity as follows: INLINEFORM3

where INLINEFORM0 is a representation of a set of vectors, which, for the DDR method, is defined as the mean vector of the set; INLINEFORM1 is a set of word embeddings, one for each token in the user's profile; and INLINEFORM2 is another set of word embeddings, one for each token in the lexicon for value dimension INLINEFORM3 . Finally, we set INLINEFORM4 where INLINEFORM5 , the number of value dimensions in the lexicon. Examples of profiles with high scores for sample value dimensions are shown in Table TABREF27 .

Further, we explore the types of activity clusters that contain activities reported by users with high scores

for various value dimensions. For a given value, we compute a score for each cluster INLINEFORM0 by taking the average INLINEFORM1 of all users who tweeted about doing activities in the cluster. For each value INLINEFORM2 , we can then rank all clusters by their INLINEFORM3 score. Examples of those with the highest scores are presented in Table TABREF28 . We observe that users whose profiles had high scores for Family were likely to report doing activities including family members, those with high scores for Nature tweeted about travel, and those with high Work-Ethic scores reported performing writing related tasks.

Evaluation

We evaluate our activity prediction models using a number of metrics that consider not only the most likely cluster, but also the set of INLINEFORM0 most likely clusters. First, we evaluate the average per-class accuracy of the model's ability to rank INLINEFORM1 , the target cluster, within the top INLINEFORM2 clusters. These scores tell us how well the model is able to make predictions about the kinds of activities that each user is likely to do.

Second, we test how well the model is able to sort users by their likelihood of having reported to do an activity from a cluster. This average comparison rank (ACR) score is computed as follows: for each user in the test set, we sample INLINEFORM0 other users who do not have the same activity label. Then, we use the probabilities assigned by the model to rank all INLINEFORM1 users by their likelihood of being assigned INLINEFORM3 , and the comparison rank score is the percentage of users who were ranked ahead of the target user (lower is better). We then average this comparison rank across all users in the test set to get the ACR. The ACR score tells us how well the model is able to find a rank users based on their likelihood of writing about doing a given activity, which could be useful for finding, e.g., the users who are most likely to claim that they "purchased some pants" or least likely to mention that they "went to the gym" in the future.

Experiments and Results

We split our data at the user-level, and from our set of valid users we use 200,000 instances for training data, 10,000 as test data, and the rest as our validation set.

For the document encoder and profile encoder we use Bi-LSTMs with max pooling BIBREF17 , with INLINEFORM0 and INLINEFORM1 . For the history encoder, we empirically found that single mean pooling layer over the set of all document embeddings outperformed other more complicated architectures, and so that is what we use in our experiments. Finally, the classifier is a 3-layer feed-forward network with and INLINEFORM2 for the hidden layers, followed by a softmax over the INLINEFORM3 -dimensional output. We use Adam BIBREF26 as our optimizer, set the maximum number of epochs to 100, and shuffle the order of the training data at each epoch. During each training step, we represent each user's history as a new random sample of INLINEFORM4 documents if there are more than INLINEFORM5 documents available for the user, and we use a batch size of 32 users. Since there is a class imbalance in our data, we use sample weighting in order to prevent the model from converging to a solution that simply predicts the most common classes present in the training data. Each sample is weighted according to its class, INLINEFORM6 , using the following formula: INLINEFORM7

where INLINEFORM0 is the number of training instances belonging to class INLINEFORM1 . We evaluate our model on the development data after each epoch and save the model with the highest per-class accuracy. Finally, we compute the results on the test data using this model, and report these results.

We test several configurations of our model. We use the complete model described in section SECREF19 using either the set of additional tweets written by a user as their history (INLINEFORM0), or only the set of additional activities contained in those tweets (INLINEFORM1). Then, to test the effect of the various

model components, we systematically ablate the attributes vector input INLINEFORM2 , the profile text (and subsequently, the Profile Encoder layer) INLINEFORM3 , and the set of documents, D, comprising the history along with the Document and History Encoders, thereby removing the INLINEFORM4 vector as input to the classifier. We also explore removing pairs of these inputs at the same time. To contextualize the results, we also include the theoretical scores achieved by random guessing, labeled as rand.

We consider two variations on our dataset: the first is a simplified, 50-class classification problem. We choose the 50 most common clusters out of our full set of INLINEFORM0 and only make predictions about users who have reportedly performed an activity in one of these clusters. The second variation uses the entire dataset, but rather than making predictions about all INLINEFORM1 classes, we only make fine-grained predictions about those classes for which INLINEFORM2 . We do this under the assumption that training an adequate classifier for a given class requires at least INLINEFORM3 examples. All classes for which INLINEFORM4 are assigned an "other" label. In this way, we still make a prediction for every instance in the dataset, but we avoid allowing the model to try to fit to a huge landscape of outputs when the training data for some of these outputs is insufficient. By setting INLINEFORM5 to 100, we are left with 805 out of 1024 classes, and an 806th "other" class for our 806-class setup. Note that this version includes all activities from all 1024 clusters, it is just that the smallest clusters are grouped together with the "other" label.

While our models are able to make predictions indicating that learning has taken place, it is clear that this prediction task is difficult. In the 50-class setup, the INLINEFORM0 model consistently had the strongest average per-class accuracy for all values of INLINEFORM1 and the lowest (best) ACR score (Table TABREF31). The INLINEFORM2 model performed nearly as well, showing that using only the human-activity relevant content from a user's history gives similar results to using the full set of content available. When including the attributes and profile for a user, the model typically overfits quickly and

generalization deteriorates.

In the 806-class version of the task, we observe the effects of including a larger range of activities, including many that do not appear as often as others in the training data (Table TABREF34). This version of the task also simulates a more realistic scenario, since predictions can be made for the "other" class when the model does to expect the user to claim to do an activity from any of the known clusters. In this setting, we see that the INLINEFORM0 model works well for INLINEFORM1 , suggesting that the use of the INLINEFORM2 vectors helps, especially when predicting the correct cluster within the top 25 is important. For INLINEFORM3 , the same INLINEFORM4 model that worked best in the 50-class setup again outperforms the others. Here, in contrast to the 50-class setting, using the full set of tweets usually performs better than focusing only on the human activity content. Interestingly, the best ACR scores are even lower in the 806-class setup, showing that it is just as easy to rank users by their likelihood of writing about an activity, even when considering many more activity clusters.

Conclusions

In this paper, we addressed the task of predicting human activities from user-generated content. We collected a large Twitter dataset consisting of posts from more than 200,000 users mentioning at least one of the nearly 30,000 everyday activities that we explored. Using sentence embedding models, we projected activity instances into a vector space and perform clustering in order to learn about the high-level groups of behaviors that are commonly mentioned online. We trained predictive models to make inferences about the likelihood that a user had reported to have done activities across the range of clusters that we discovered, and found that these models were able to achieve results significantly higher than random guessing baselines for the metrics that we consider. While the overall prediction scores are not very high, the models that we trained do show that they are able to generalize findings from one set of users to another. This is evidence that the task is feasible, but very difficult, and it could benefit from

further investigation.

We make the activity clusters, models, and code for the prediction task available at

http://lit.eecs.umich.edu/downloads.html

Acknowledgments