# Task-driven Information Valuation in Web Communities

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

# Task-driven Information Valuation in Web Communities

Master Thesis

Maria I. Han Veiga

May 10, 2015

Advisors: Prof. Dr. T. Hofmann, Dr. C. Eickhoff
Department of Mathematics, ETH Zürich

**Abstract**

The increased popularity of online social networks and globalised Internet access have affected the way people share content. The information that users willingly share in these websites can be used for various purposes, from building consumer models for advertising, to inferring personal information that could be invasive.

In this thesis, we use Twitter and Instagram timelines to infer which venue types the user frequents. We show that for some venue types, there is a specific vocabulary associated to these venues.

In order to study the information leak that occurs in these social networks, we present an information score function which estimates the informative value contained in a shared content item, with respect to an inference task. This measure is validated using a framework which actively chooses content with a high information score. We show that by doing this, we can, in some cases, attain a better performance in the inference task than when using the full timeline of the user.

# Contents

Chapter 1

# Introduction

User privacy has been a topic that increasingly gained traction with the rise of online social networks (OSNs). In 2011, the World Economic Forum set out on a multiyear project to study the nature and impact of private personal data.

Recent successful initial public offerings (IPOs) and high market valuations underline the value of OSNs. However, the relation between the number of registered users, their activities online, and these valuations is not entirely clear. Furthermore, although the value of personal data is well accepted, there have not been many studies which concretely assign a tangible commercial value to social profiles.

It has been shown in several studies that user characteristics, such as personal traits [11] or future route intentions [15], can be inferred from their profiles in online social networks.

In this thesis, we use profiles from different OSNs to infer which types of venues a user is likely to visit. With this, we aim to study three fundamental questions:

1. Does adding several profiles from different OSNs give a more accurate representation of the user?

2. Can we quantify the amount of information that a piece of shared content carries, with respect to a concrete inference task?

3. Can we provide an estimate of the value of a user's profile for the social network it belongs to?

## 1.1 Background information

### 1.1.1 Social networks

Online social networks are platforms or websites which allow users to communicate, connect with other users and share content. While originally OSNs focused on the first two aforementioned aspects, nowadays the term OSN also includes platforms which are primarily user-centric, where users can broadcast personal thoughts and content.

With the increased popularity of OSNs, their impact on the user online presence is visible. In 2010, [12] find that online social networks are among the top visited websites for a large population of users.

Thus, due to their prevalence and abundance in personal content, OSNs can be used to study how human society behaves at a large scale [13].

A description of the OSNs used in this thesis follows.

#### Twitter

Twitter is a microblogging platform. It is currently ranked as the $2^{nd}$ largest OSN, with approximately 310 million active users monthly [4].

The main content of Twitter comes in *tweets*, which are posts limited to 140 characters. These can contain text, media (video or images), links to external websites, references to other users and *hashtags* (words starting with # which are used to mark keywords or topics in a tweet). The collection of a user's tweets is called a timeline. In addition to the user's timeline, there is also a (optional) description of the user, a user profile picture, list of followers and a list of users that the user follows.

#### Instagram

Instagram is a photo sharing platform. It is currently ranked as the $7^{th}$ largest OSN, with an estimated 100 million active users monthly [4].

The main content type of Instagram are photos or videos that the user can post. In addition to the photo, the user can add a textual description and share these posts on other OSNs.

#### Foursquare

Foursquare is a location service platform. Its community is comprised of more than 55 million users worldwide [6].

The main content of Foursquare are *check-ins*. Check-ins correspond to venues that the user has visited. In addition to the venue name, there is

more information available, such as location and venue type. With nearly 500 types of venues, these can range from specific restaurant types to outdoor places.

### 1.1.2 eCommerce and advertisement

Given the pervasive nature of the Internet in everyday life, advertising also started shifting from traditional channels - such as newspapers or television, to the online medium.

Websites sell advertisement spots to businesses. *Adwords* by Google, or *Yahoo!Advertising* by Yahoo! are examples of such services. These platforms offer different packages for businesses. For instance, in the context of searching through these web search services, businesses can buy certain keywords to which their websites will be shown first in the results page.

*Twitter Ads* is an advertising platform which allows businesses to advertise within the Twitter platform. A business interested in starting an advertising campaign on Twitter can choose a tailored advertising package based on their campaign objective. This could be gaining followers, website clicks and conversions, tweet engagement, etc... [21].

For the objective of Website clicks and conversions, Twitter Ads offers the the possibility to target Twitter users in several ways: *user interests*, *keywords*, *followers* and *television interests*.

#### Pricing models

The way advertising platforms generate revenue is by selling the advertisement spots. There are three main pricing models for online display advertising, which are introduced below [26]:

- **C**ost-per-click (CPC): the pricing model is based on the number of times the advertisement (ad) is clicked.
- **C**ost-per-action (CPA): the pricing model is based on the number of specific actions performed by the user (e.g. purchases, filling a form, etc.) which are directly linked to the advertisement.
- **C**ost-per-impression (CPI): the pricing model is based on the number of users who view the ad.

## 1.2 Motivation

Motivating the first research question, we want to understand whether users in certain OSNs are more prone to share personal content than in other OSNs.

Secondly, we want to study how much information about the user is unintentionally being exposed through shared content. For example, in the context of venue type visits, if a user shares a venue check-in, it is relatively easy to infer, algorithmically, which venue types the user has visited. However, if the user shares some content without explicitly mentioning a place, as shown in Figure 1.1, it might still contain information about a potential behaviour or visit intention.

In particular, we want to understand whether there is a certain vocabulary on a user's Twitter and Instagram timelines which are related to visits to certain venue types.

Third, we want to study the possibility of, programmatically, finding the content which contains the highest information value for a given task, within the timeline of a user.

As an example, consider two tweets from the same user:

a) Lol should start heading to the gym #fitness

b) Great sunny day!

It it clear that tweet a) provides us more information about the user's intent of visiting a venue than tweet b).

Thus, it is our objective to build a score function which will give an information score to a shared content item. From the example before, we expect this measure to give a higher score to example a) than example b) in the task of inferring *"Does the user visit the gym?"*.

Finally, having established such a score function, the next objective is to provide an estimate of how much information a user is giving away by sharing content which is seemingly void of commercial value. In order to do this, we attribute an estimated monetary value to a user's timeline on a hosting online social network, which is given by whether content can be effectively advertised to the user, given their online foot-print.

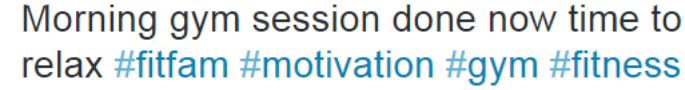



Figure 1.1: Tweet expressing user's habits or lifestyle.

## 1.3 Problem statement

### 1.3.1 Cross OSN inference

To gauge how much information can be obtained on habits or lifestyle of users through their online footprint, we look at whether it is possible to predict venue type visits of a user, which are logged on Foursquare, given their Twitter or Instagram feed. Let $\mathcal{V}$ be the set of venue types we want to predict and $\mathcal{U}$ a set of users. Assume we have a set of functions $\{f_A\}_{A\in\mathcal{V}}$, $f_A : \mathcal{U} \to \{0,1\}$ which are defined in the following way:

$$f_A(u \in \mathcal{U}) = \begin{cases} 1 & \text{if user has visited A} \\ 0 & \text{if user has not visited A.} \end{cases}$$

We then use the user's OSN timelines to infer these functions.

### 1.3.2 Information content

The second problem we are concerned with is understanding the information content of a single post from the timeline of a user. Intuitively, we expect some posts to give more information about the user, with respect to a certain inference task. First, we test this hypothesis by making venue type predictions on the user's incomplete timeline. As we add information about the user, we expect the prediction's accuracy to increase.

We then find a function which, given a post and the inference task at hand, can return an information score of the post. We validate the usefulness of this score by measuring the classifier's performance increase in the inference task over random post sampling.

## 1.4 Thesis goals and contributions

A summary of the key contributions of this thesis follows:

- **Novel data set:** Due to the nature of the problem, there was no data set publicly available that fitted our needs. We assemble the data set from scratch which contains profiles of the same user across different OSNs. To our knowledge, a data set of this type has not been publicly released in the past.
- **Cross OSN inference:** We study the effects of using user information from more than one social network to better model the user.
- **Content information score:** We formulate a score function to quantify the value of shared content, from the perspective of improving the per-

formance in the inference task and providing new information about the user and present an evaluation method.

## 1.5 Outline

The remainder of this thesis is structured as follows:

In Chapter 2, we present related work to this thesis. In particular, we describe various contributions in social network inference and cross social network modelling, as well as some basic theoretical background.

In Chapter 3, we present the methods used in this thesis. Firstly, the cross network inference task is formalized. Secondly, we formulate the information score function and present the evaluation framework to validate this measure. Thirdly, we propose a simple pricing model for the user profile, derived from pricing models in advertising.

In Chapter 4, the data set collection procedure is described and a description of the data set used throughout the experiments is provided.

In Chapter 5, the results of the experiments are presented.

Finally, in Chapter 6, the main conclusions of this thesis and future work is discussed.

Chapter 2

# Literature review

In this chapter we present an overview of different research efforts that fall under the area of this thesis and a provide a short introduction to mathematical methods which will be used throughout the thesis.

## 2.1 Inferring OSN user behaviour

With the emergence of the field of Computational Social Science, many scientists have looked at OSNs as a primary source for human data.

We observe that commonly, authors have limited their study to a single social network. There is a vast amount of papers which study the information content contained in user profiles in different OSNs. For example, in [8] and [18], the authors use Twitter to predict the personality of users in the light of the *Big Five* personality model. The features used in both papers were not derived from the content of the tweets but from metrics such as number of followers and following, number of mentions, number of hashtags, etc...

In [11], the authors correlate Facebook likes to personality traits, again using the *Big Five* personality model, as well as other sensitive personal attributes such as sexual orientation, ethnicity, religious and political views, intelligence, and so on. Instagram has not been featured much in academic research yet, but some attempts to understand its contents and user types are presented in [9].

From the perspective of studying cross-OSN user behaviour, the authors in [5] study the macro-scale patterns of activity across Twitter and Pinterest[1], in particular, how users distribute their activity across the sites and the dissemination of content from one site to another.

[1]Pinterest is another OSN which primarily allows users to "pin" media content, such as videos, images and so on, to their pin board and serves as a content aggregator.

## 2.2 Background in methods

In this section, a short introduction of concepts is provided for the sake of completeness.

### 2.2.1 Classification

Given an input set $X$ and a set $K = \{k_j\}_{j=1}^{w}$ of classes, the goal of classification is to assign an input element $x \in X$ to a class $k_j$, based on a training set of data containing elements whose class membership is known.

A concrete implementation of an algorithm which performs classification is known as a classifier. Formally, a trained classifier can be defined as $\mathcal{C} : X \to \{k_1, ..., k_w\}$, a function that maps input data $X$ to a class in $K$.

The particular choice of a concrete classifier is often dependent on the type of input data and problem to be solved. A good source for more information is [2].

### 2.2.2 Vector space models

A vector space model is an algebraic model commonly used in Information Retrieval (IR) tasks, where text documents and queries are represented as vectors in the term space. The basis of this vector space is given by a set of terms.

Formally, let $V$ be a vector space in $\mathbb{R}^n$ and $T = \{t_1, ..., t_n\}$ be a set of linearly independent vectors in $V$. A document $D$ can be represented by the vector $(w_1, w_2, ..., w_n)$, where a weight $w_i$ is related to whether the $i^{th}$ term appears or not in document $D$.

There are several ways to compute the weights $w_i$. One of the most used ways is the term frequency inverse document frequency (TF-IDF) weighting. More information about this can be found in [16].

#### Curating the term vocabulary

It is often the case that some sort of curation is performed in the raw term vocabulary to determine the appropriate term vocabulary. The most common methods are the following:

- **Stop Word removal:** removes words which are too common in the language and thus, have little discriminating value (e.g. a, an, and, the, to, with...);
- **Normalization:** maps words which point to the same concept but might have different spelling (e.g. anti-discriminatory, antidiscriminatory $\to$ antidiscriminatory);

- **Case folding:** reduces all letters to lower case (e.g. Ferrari → ferrari);
- **Stemming:** reduces inflected (or sometimes derived) words to their word stem, base or root form using a heuristic. This does not guarantee that the word stem is a word (e.g. cats → cat, ponies → poni);
- **Lemmatization:** groups together different inflected forms of a word using a vocabulary and morphological analysis of words (e.g. cats → cat, ponies → pony). A popular corpus for lemmatization is WordNet [23].

### 2.2.3 Balancing unbalanced data sets

When applying machine learning algorithms to real world problems, it is often to encounter problems where the classes of labels are not balanced, i.e. some classes occur much less frequently than others. This becomes a problem because traditional performance measures, such as accuracy, do not represent the real performance of the classifier. Consider a unbalanced problem where 98% of the examples belong to class A and only 2% to class B. The classifier's accuracy would be 98% by just predicting class A in all examples [1].

When dealing with imbalanced data sets, sampling techniques have been used to counter the effect of these data sets. There are two ways to sample:

- Under-sampling the majority class
- Over-sampling the minority class

While random oversampling can sometimes cause over-fitting, under-sampling can remove important examples from the data set. In survey [10], the author notes that there is no major improvement when using more sophisticated methods to over-sample the majority class (or under-sample the minority class) over randomly selecting elements to duplicate (or remove). However, there were no methods which introduced artificial data (such as SMOTE) in the evaluation.

Chapter 3

# Methods

The methods used in the thesis are presented in this chapter. In Section 3.1, the method to perform cross OSN inference is described. In Section 3.2 we describe the methodology used to quantify information in shared content items, construct the information score function and propose the evaluation method. In Section 3.3 we describe a simple pricing model for a social profile, based on advertising pricing models. The methods for data set creation are outlined in Chapter 4.

## 3.1 Cross OSN inference

In this section, we describe the method to perform cross OSN inference, where we use different OSNs (separately or in groups) to quantify the differences between the types of content shared, in the task of predicting user visits to different types of venues.

Let $\mathcal{U}$ be a set of users. A user is determined by the set of his profiles on $q$ considered OSNs. However, because in practice we only work with the timelines of these OSN profiles, we can define a user $u$ by the set of his $q$ associated timelines (on the considered OSNs), thus: $u = \{S_k^u\}_{k=1}^q$. When the user we are referring to is obvious from the context, we drop the over-script notation.

In this thesis, we use the set of posts from these profiles to estimate the probability of user $u$ visiting venue $A$ given a set of posts $M$. This probability is denoted by $\mathrm{p}_u(A|M)$, where A is "*user u visits venue type A*" and M is a set of posts from user $u$.

We use a binary classifier for this classification task of predicting if user goes to venue type A or not. The features are *extracted* from the timelines. Timelines can be seen as a text containing posts as sentences. We map these *texts* to a vector space model and use a TD-IDF representation. The TD-IDF term

vocabulary is, in principle, generated from terms of all the timelines. We use the AdaBoost algorithm with decision trees as weak learners as our classifier, because it generally works well without much parameter adjustment [3]. The classifier is trained on the timelines of a subset of users and tested on the remaining users' timelines. The classifier's performance is evaluated under 10 fold cross-validation. Let $\mathcal{C}$ denote the trained classifier as this will be used further on.

We compare a classifier's performance, when trained on one single OSN at a time and when trained using all timelines across all considered OSNs, i.e. $\bigcup_{k=1}^{q} S_k$.

To evaluate the classifier's performance when using data from different OSNs, we use the Wilcoxon Signed-Rank test, a non-parametric test for paired samples, to test whether there are significant differences in the classifier's performance. We can do this because we fix the cross-validation folds before we choose which OSN (or set of OSNs) to be used in the classifier's training and testing phases.

## 3.2 Measuring information content

The second objective of this thesis is to find a function which quantifies the information content in a post. The information contained in a post can be modelled in two ways:

- Relevance of the post with respect to the inference task;
- Novelty of the post with respect to the user's previously seen content.

We want to model these two distinct quantities: *relevance* of a post as being dependent on the current post and on a trained classifier, and *novelty* of the post, which depends only on the current post and on the previously seen posts.

We model the information content as a convex combination between these two quantities, thus introducing a modelling parameter $\lambda \in [0, 1]$. Let us define the information score function $\mathbb{I} : \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{C} \rightarrow \mathbb{R}^+$, where $n$ denotes the dimension of the feature space and $\mathcal{C}$ a trained classifier.

$$\begin{aligned} \mathbb{I}(\text{post}, \text{user timeline}, \text{classifier}) = \lambda\ &\text{novelty}(\text{post}, \text{user timeline}) \\ &+ (1 - \lambda)\ \text{relevance}(\text{post}, \text{classifier}) \end{aligned} \tag{3.1}$$

As expected, $\mathbb{I}$ depends on a current post, previously seen posts and a classifier $\mathcal{C}$.

### 3.2.1 Novelty

For a fixed user $u \in \mathcal{U}$, let $\vec{s}_1$, $\vec{s}_2 \in \mathbb{R}^n$ be the vector of features of shared content $s_1$, $s_2 \in S_k^u$, user $u$'s timeline in one the considered OSNs. $\vec{s}_1, \vec{s}_2 \in \mathbb{R}^n$ can be, for instance, elements of a vector space model.

Informally, the function $novelty : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$ for shared content items $\vec{s}$ should have the following properties:

1. Two elements should have novelty 0 if one is contained in the other.
2. Two elements should have small novelty if they are similar.
3. Two elements should have high novelty if they are very distinct.

The cosine distance is a standard measure for document dissimilarity in Information Retrieval. It is defined as follows:

$$\text{cosine distance} = 1 - \frac{\vec{s_1} \cdot \vec{s_2}}{\|\vec{s_1}\| \|\vec{s_2}\|} \tag{3.2}$$

For two vectors $\vec{s_1}, \vec{s_2}$ belonging to a normed vector space, it returns $1 - cos(\omega)$ where $\omega$ is the angle between the two vectors. Although it fulfils (2) and (3), (1) is left unfulfilled, which is important for our *novelty* score.

The proposed function to measure novelty is the following: Let $\nu$: $\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$, be a non-symmetric function defined by:

$$\nu(\vec{s_1}, \vec{s_2}) = \frac{\sum_{i=1}^{n} \mathbb{1}[s_{1,i} \neq s_{2,i} \wedge s_{2,i} = 1]}{\sum_{i=1}^{n} \mathbb{1}[s_{2,i} \neq 0]} \tag{3.3}$$

This function compares vector $\vec{s_2}$ to $\vec{s_1}$ and provides the percentage of dimensions which are non-zero in $\vec{s_2}$ but zero in $\vec{s_1}$. This models the new terms appearing from $\vec{s_2}$ over $\vec{s_1}$ if these belong to a vector space model.

### 3.2.2 Relevance

Measuring the relevance of shared content can be thought of as finding out which shared content contains features (in our case, words) that are important for the classifier.

In ensemble methods, where decision trees are used as the base classifiers, a function which is often used to describe feature importance is the *Gini Importance* ($I_g$).

In the next paragraphs, we will provide an intuition of how the Gini Importance is calculated. For a more formal treatment of this topic, we refer to [20] and [17].

Let us consider an ensemble method with trees as base classifiers. For a node $\tau$, the impurity of $\tau$ measures how well classes are separated. Ideally, all examples in a node should belong to the same class, i.e impurity would be zero for this node.

A measure of impurity is the *Gini Impurity* $i(\tau)$. For a feature $\theta$ and the node split threshold $t_\theta$ on the variable $\theta$, we can calculate the decrease in Gini Impurity $\Delta_i(\tau)$. The classifier aims to find the pairs $(\theta, t_\theta)$ which minimize $\Delta_i(\tau)$. The decrease in Gini Impurity given $\theta$, a node $\tau$ and a tree T is $\Delta_{i_\theta}(\tau, T)$. The Gini Importance for a feature $\theta$ is thus given by:

$$I_g(\theta) = \sum_T \sum_\tau \Delta_{i_\theta}(\tau, T). \tag{3.4}$$

The Gini Importance indicates how often a particular feature $\theta$ was selected for a split, and how large its overall discriminative value was for a particular classification problem.

In this work, we estimate the relevance of a post by summing up the Gini Importances of the words contained in a post:

$$\text{relevance(post)} = \sum_{\theta \in \text{post}} I_g(\theta) \tag{3.5}$$

### 3.2.3 Evaluation method

In this section we set up the framework to evaluate the information score function introduced above.

Given a user $u \in \mathcal{U}$, we produce an estimate of the probability of $u$ visiting venue type A, as in Section 3.1, given a subset of posts $M \subseteq \bigcup_{k=1}^{q} S_k$ of his timelines, denoted as $p_u(A|M)$. We emphasize that $M$ does not necessarily contain the entire timeline $S_k$.

We train a classifier as described in Section 3.1. For every test user $u$, we initiate the procedure by randomly sampling $n_i$ posts from his timeline $S_k$ of length $n_l^k$ and create the truncated timeline. Then, at each iteration, we sample a constant number $d$ of posts from his timeline, add them to the truncated timeline and make a prediction using truncated timeline[1]. We iterate this process until user $u$ has no more posts left (or until the truncated

[1]Rigorously speaking, we convert the truncated timeline to a TF-IDF representation first before we use it to make a prediction.

timeline reaches a fixed amount of posts). Thus, we obtain an ordered sequence of predictions: $(y_0, y_1, ..., y_{n_{end}})$, where $n_{end}$ represents the number of iterations. If $d = 1$, i.e. we add one post at a time, then $n_{end} = n_l^k - n_i$ (if we add all posts).

An example code is shown in Algorithm 1.

**Algorithm 1** Evaluation method

```
1:  function EVALUATION METHOD(classifier, user, n_i)
2:      timeline ← user's timeline
3:      truncated timeline ← RandomlySample(timeline, n_i)
4:      y_0 ← classifier.predict(truncated timeline)
5:      remaining posts ← timeline \ truncated timeline
6:      while remaining posts > 0 do
7:          post ← Sample(remaining posts, d)
8:          remaining posts ← remaining posts \ post
9:          truncated timeline ← truncated timeline ∪ post
10:         y_i ← classifier.predict(truncated timeline)
11:         i ← i + 1
12:     end while
13:     return (y_1, y_2, ..., y_{n_end})
14: end function
```

We evaluate the performance of the information score by defining a sampling function which samples posts with higher information scores and compare the classifier's performance to when we randomly sample posts.

For simplicity, assume we fix k i.e. we fix the OSN we are considering, we add 1 post per iteration and we start with 0 initial posts. Because users might have different timeline lengths, we average the results in the following way: the maximum timeline length $n_{l_{max}}$ is calculated. Then, for each user whose timeline is shorter than $n_{l_{max}}$, we duplicate the last prediction $y_{n_{end}}$ and generate a sequence of predictions of length $n_{l_{max}}$ for each user.

## 3.3 Pricing model

In this section we propose a pricing model to estimate a commercial value of a profile to its hosting OSN.

One way online social networks generate revenue is by selling advertisement spots to businesses. The typical models to price advertisement campaigns are the following: Cost-Per-Impression (CPI), Cost-Per-Click (CPC) and Cost-Per-Action (CPA). In the CPI billing system, businesses are charged every time the ad is viewed by a user, whereas in the CPC/CPA billing sys-

tem, the business is only charged when the user interacts with an ad (e.g. clicks or performs an action).

Traditionally, research has focused on estimating the revenue that these advertising platforms can generate. In this thesis we are interested in estimating the value of a particular profile with respect to its hosting OSN, based on the revenue the OSN makes on advertising materials to their users.

For the CPI billing system, estimating the value of a profile with respect to its hosting OSN is relatively simple:

$$\mathbb{E}[\text{profile value}] = p(\text{user sees ad}) \cdot CPI_{ad}$$

Where $CPI_{ad}$ is the agreed price between service and business and $p(\text{user sees ad})$ is given by the probability of the user seeing the ad: $\frac{1}{\text{\# of users}}$.

For the CPC/CPA billing system, the expected revenue for the OSN for an ad is given by $\mathbb{E}[\text{revenue}] = p_{ad}(\text{click}) \cdot CPC_{ad}$. We can use this to define the quantity which we are interested in: $\mathbb{E}[\text{profile value}] = p_{ad}(click) \cdot CPC_{ad}$ (or as a proportion of $\mathbb{E}[\text{revenue}]$). Instead of estimating $p_{ad}(\text{click})$ using a Click-Through-Rate, which is usually dependent on ad properties [19], we can estimate $p_{ad}(\text{click})$ per user, based on their online foot-print.

For venue type related advertisements, for example, for a particular type of restaurant or sports facility, we can obtain an estimate of the user's interests based on the probability of the user visiting certain venue types, using the classifiers built previously. We use the estimated probability defined earlier, $p_u$ as a proxy for probability of interest and thus, of clicking on a related ad. We propose that $p_{ad}(\text{click}) = \alpha p_u(Y|B)$, where Y denotes '*user visits related venue type*' and B a set of posts by the user. Thus, yielding:

$$\mathbb{E}[\text{profile value}] = \alpha_1 p_u(Y|B) \cdot CPC_{ad}$$

$$\mathbb{E}[\text{profile value}] = \alpha_2 p_u(Y|B) \cdot CPA_{ad}.$$

$\alpha_1$, $\alpha_2 \in [0, 1]$ can be thought of how much we trust on this proxy probability $p_u(Y|B)$.

Some efforts have been made towards compensating users due to their privacy loss [14]. While our proposed model does not take privacy loss into consideration, this idea is discussed in Chapter 6, Section 6.1

Chapter 4

# Data set

In this chapter, the method to create the data set is described, insights from the collected data set are given and how we use the data collected is explained.

## 4.1 Data set creation

Due to the nature of our task, it is necessary to use a fairly unique data set. For this reason, we create a data set from scratch. The requirements for this data set creation are the following:

1. Finding accounts in different social networks for the same person;
2. Possibility of gathering a considerable amount of data.

We use the Twitter Search API to search for users who cross-post content to find their corresponding profiles in other social networks. The Search API allows queries containing regular expressions, enabling us to look for URLs which correspond to the user's posts in other OSNs. Then, using the API from the corresponding OSN, we can find the user's screen name and crawl their profile. A visual depiction of this process can be found in Figure 4.1.

The cross posting volume in December 2014 for different social networks can be found in Table 4.1. Some OSNs incentivise users to cross-post more than others.

The online social networks featured in our data set are Twitter, Foursquare and Instagram. They were chosen due to the volume available through the Twitter API search and how open their corresponding APIs were.

Data was collected in January and February 2015. We scanned around 2000 profiles but only 618 users fulfilled the criteria of actively using the 3 selected OSNs and posting predominantly in English. From these 618 profiles, we


Figure 4.1: Using Twitter search to find user cross-posting activity

| OSN | Volume per minute |
|---|---|
| Instagram | ∼550 |
| Facebook | ∼170 |
| Foursquare | ∼70 |
| Pinterest | ∼30 |

Table 4.1: New tweets per minute that link to posts in different OSNs

have approximately 1.1 million tweets, 18000 Instagram pictures and 99000 Foursquare check-ins. In the next section, we give more insights into the data set.

## 4.2 Data set insights

Using the notation introduced in Chapter 3, user $u \in \mathcal{U}$ is defined by $\{S_1, S_2, S_3\}$, $(q = 3)$. For easier readability, we rename these variables so that a user $u$ is defined by: $\{T, I, F\}$, where $T$ corresponds to user's Twitter timeline, $I$ to the user's Instagram timeline and $F$ to the user's Foursquare timeline[1].

Given user $u \in \mathcal{U}$, their twitter timeline $T$ contains the set of tweets $\{t_1, t_2, ..., t_{n_T}\}$, where $n_T$ denotes the number of tweets in their timeline. Similarly, $I$ contains the set of Instagram pictures $\{i_1, i_2, ..., i_{n_I}\}$ and $F$ the corresponding set of venue check-ins $\{f_1, f_2, ..., f_{n_F}\}$. Figure 4.2 shows the distribution of $n_T$, i.e. the number of tweets per user. Figure 4.3 shows the distribution of $n_I$, the number of Instagram posts per user and Figure 4.4 shows the distribution of $n_F$, the number of check-ins per user.

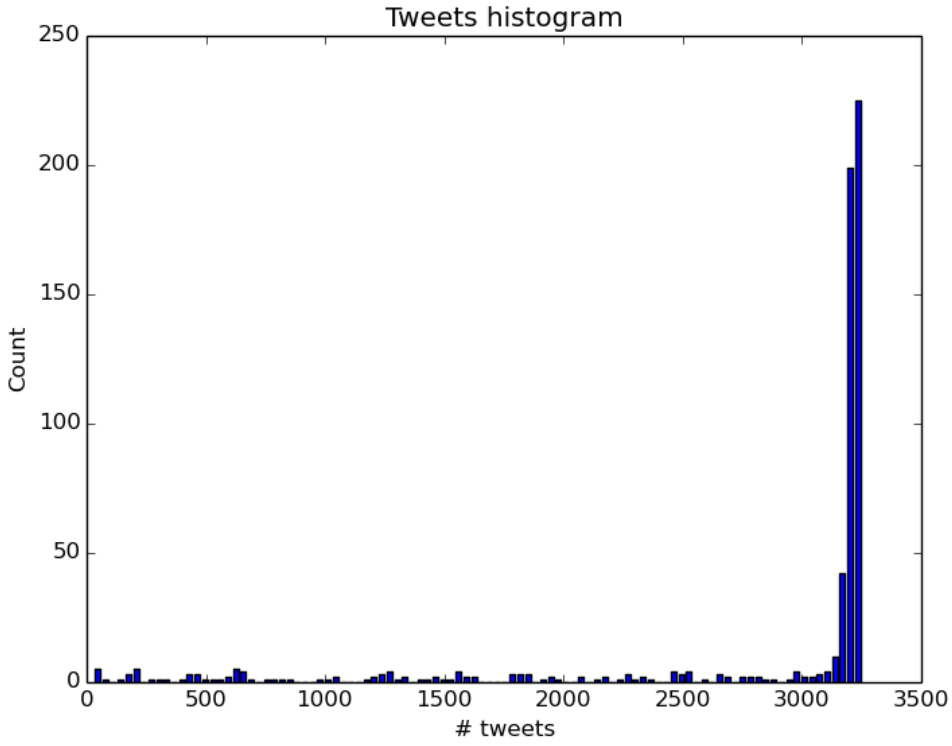



Figure 4.2: Histogram of number of tweets per user

The Twitter API restricts the access to 3200 tweets per profile (including

[1]In reality, the Foursquare timeline is the set of venues visited by the user which have been also posted on Twitter.

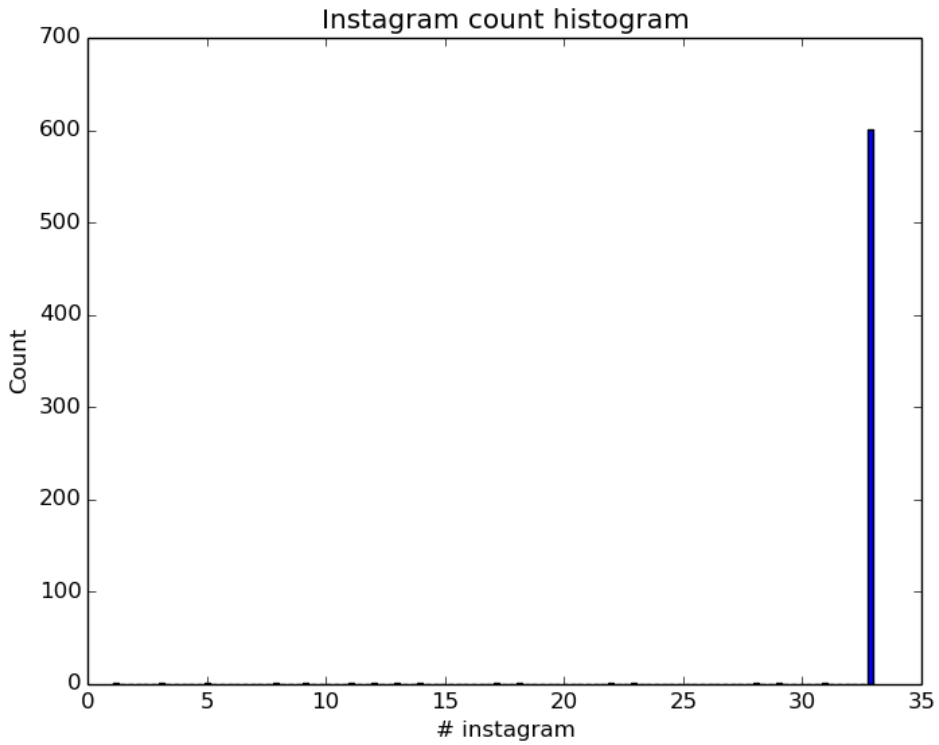



Figure 4.3: Histogram of number of Instagram posts per user

retweets) [22], because we exclude direct retweets from our data set, the majority of the Twitter profiles we collect contain between 3000 to 3200 tweets. For each Instagram profile, we recover the 33 most recent Instagram posts due to API restrictions. For each Foursquare profile, we recover the check-ins which were cross-posted on the retrieved Twitter timeline.

Each check-in has plenty of information associated to the venue, as it can be found in [7]. In this thesis, we work with venue types. Venue types are hierarchically organised in three layers. For example, one of the main categories of venue types is *Arts & Entertainment*, which is parent to *Aquarium*, *Arcade*, *Museum*, and so on. Under *Museum*, there are *Art Museums*, *History Museums*, and so on. The distribution of main venue types visited by users can be seen in Figure 4.5. The distribution of the subcategories of the main category *Arts & Entertainment* can be seen in Figure 4.6.

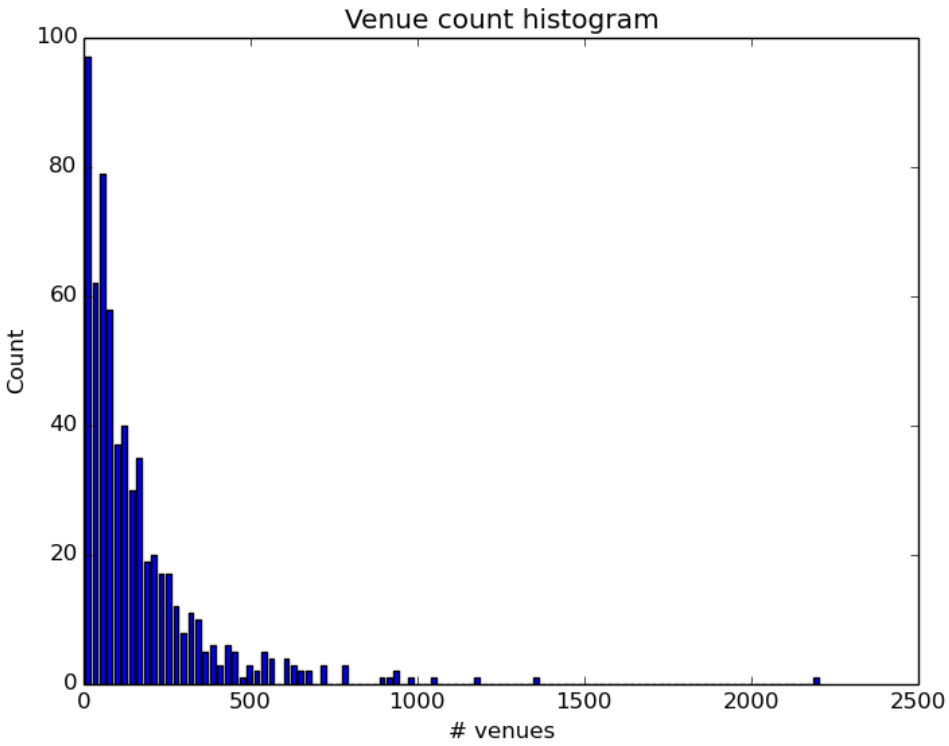



Figure 4.4: Histogram of number of check-ins per user

## 4.3 Classification specification

In this section we describe how we use the described data set to generate the data set to be used by the classifier described in Section 3.1.

### 4.3.1 Feature space generation

User features are generated from their corresponding OSN timelines. The timelines are represented by TF-IDF vectors. The posts in Twitter or Instagram often contain slang, made-up or misspelled words, memes, links or replies to other users. Because of these, we curate the term vocabulary:

1. All links and mentions (other user's nicknames) are removed.
2. English stop words are removed.
3. Words which occur less than 5 times across all timelines and users are removed (to account for misspellings, links which were not removed or noisy tokens, and reduce computational burden).

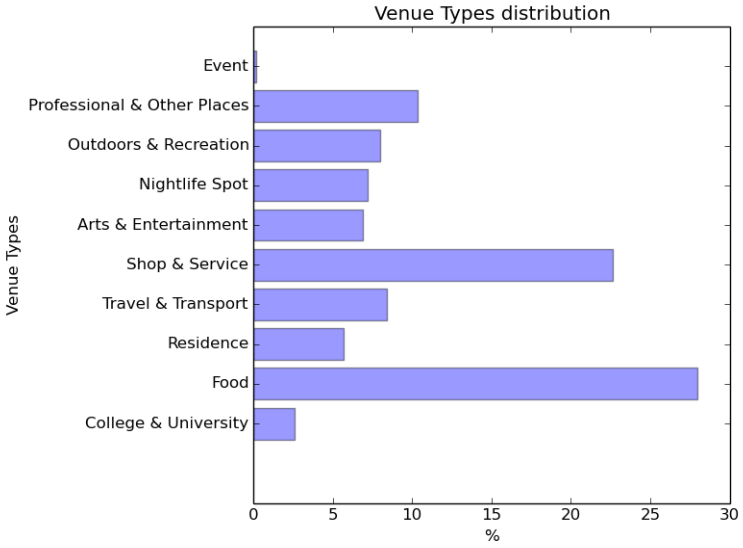



Figure 4.5: User distribution across main categories

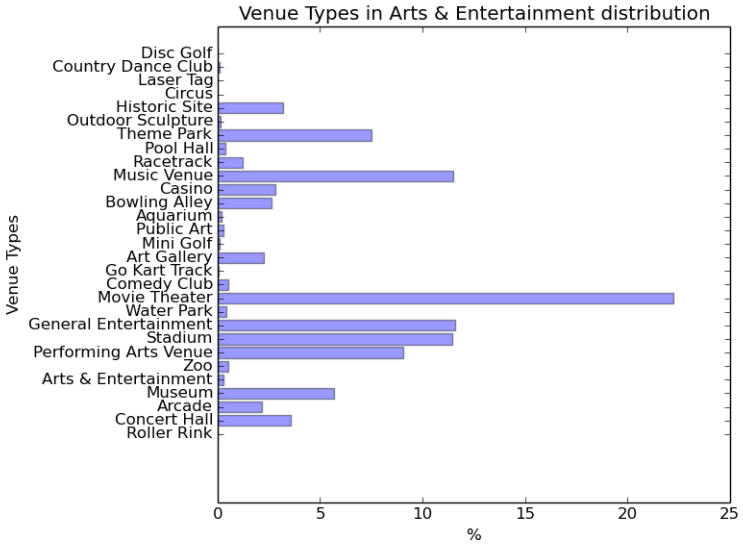



Figure 4.6: User distribution across subcategories of *Arts & Entertainment*

4. Words which are recognized are reduced to their base (lemmatized using WordNet corpus).

### 4.3.2 Label assignment

Let $\mathcal{V}$ be the set of venues we are interested in working with, $A \in \mathcal{V}$ be a particular venue type and $u$ be a user. The label $l_A(u)$ is 1 if $u$ has visited venue type A and 0 otherwise.

Chapter 5

# Experimental results

In this chapter we describe the experimental results using the data set presented in Section 4.2.

To remind the reader, we use a user's Twitter and Instagram timelines to infer user venue type visits.

## 5.1 Measures of classifier performance

To measure classifier performance we use Recall, Precision, Accuracy, Specificity and $F_1$ score, defined as:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{5.1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5.2}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \tag{5.3}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{5.4}$$

$$F_1 \text{ score} = 2\,\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.5}$$

Where $TP$ denotes true positives, $FP$ denotes false positives, $TN$ denotes true negatives and $FN$ denotes false negatives.

Because our data set has imbalanced classes, we optimize our results for the $F_1$ score instead of accuracy.

## 5.2 Inference

We select 40 venue types to test our methods on. These venue types are selected based on the percentage of the population that has visited them. We restrict this percentage to lie between (approximately) 25% to 35% such that there are enough positive examples to train the classifier on, but that they are still specific enough to have a distinct vocabulary associated to them. For example, venue types such as *Restaurant*, *Coffee Shop* and *Bar* are not included in these 40 venues because a larger percentage of users have visited them, but venue types such as *Japanese Restaurant*, *Gastropub* and *Wine Bar* are included.

For the first task, we do not do any further discrimination, in terms of which venue types are more interesting to predict from a commercial point of view. The table with the 40 venues and their corresponding positive class percentages can be found in the Table A.1.

For the set of the selected 40 venue types $\mathcal{V}$, we build 40 classifier's (one per venue type) as described in Section 3.1. We build the feature spaces from Twitter and Instagram timelines using TF-IDF representation and measure the classifiers' performances at predicting '*user goes to venue type A*' for $A \in \mathcal{V}$.

The classifiers' performance, over 10-fold cross-validation and averaged across the 40 venues, can be found in Table 5.1.

Table 5.1: Average classifier performance using different data sets. Results are averaged over 10 fold cross-validation.

| | Accuracy | Precision | Recall | Specificity | $F_1$ Score |
|---|---|---|---|---|---|
| Twitter | 69.6 | 49.8 | 40.7 | 81.9 | 44.7 |
| Twitter+Instagram | 69.9 | 50.4 | 40.3 | 82.5 | 44.7 |

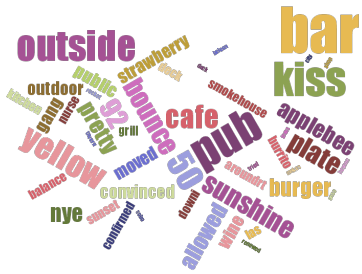
Adding the Instagram data did not change the classifier's performance significantly[1]. However, there are a few venues in which the $F_1$ score improvement was statistically significant under the Wilcoxon signed-rank test ($\alpha = 0.05$). This can be seen in Table 5.2. There was one venue in which the performance degradation was statistically significant, as shown in Table 5.3.

We can also find the vocabulary which is highly correlated with the positive label. In Figure 5.1 we can see examples of the top performing classifiers for venue types *Gym*, *Gastropub* and *Spiritual Center* and one of the worst performing ones (*Concert Hall*), using both data sets.

[1]Due to the reduced size of the Instagram data set, the result using classifiers that were trained and tested only on Instagram data are not included in this thesis.

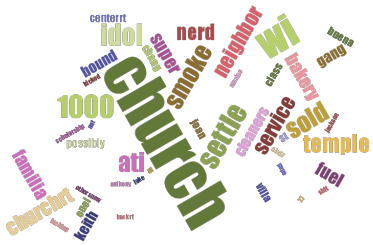

(a) *Gym* classifier using Twitter data


(b) *Gym* classifier using Twitter and Instagram data




(c) *Gastropub* classifier using Twitter data

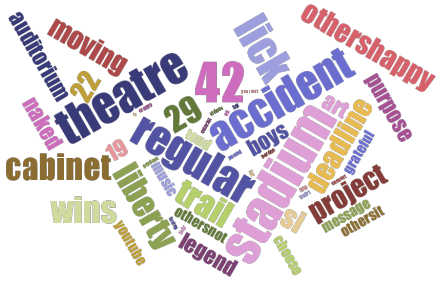

(d) *Gastropub* classifier using Twitter and Instagram data




(e) *Spiritual Center* classifier using Twitter data


(f) *Spiritual Center* classifier using Twitter and Instagram


(g) *Concert Hall* classifier using Twitter data




(h) *Concert Hall* classifier using Twitter data

Figure 5.1: Word clouds showing words which are most correlated with the positive label of each classifier.

Table 5.2: Venues in which using Instagram data improved the performance most over using just Twitter data. The value to the left corresponds to Twitter only data, whereas the value to the right corresponds to Twitter and Instagram data.

| Venues | Accuracy | Precision | Recall | Specificity | $F_1$ Score |
|---|---|---|---|---|---|
| Gym | 65.7 \| 70.7 | 53.0 \| 61.6 | 46.4 \| 51.6 | 76.6 \| 81.8 | 49.5 \| 56.2 |
| Cocktail Bar | 68.0 \| 71.0 | 51.5 \| 58.3 | 41.9 \| 46.8 | 80.4 \| 83.2 | 46.2 \| 51.9 |

Table 5.3: Venues in which using Instagram data degraded the performance most over using just Twitter data. The value to the left corresponds to Twitter only data, whereas the value to the right corresponds to Twitter and Instagram data.

| Venues | Accuracy | Precision | Recall | Specificity | $F_1$ Score |
|---|---|---|---|---|---|
| Performing Arts Venue | 63.7 \| 61.3 | 48.2 \| 44.8 | 43.2 \| 37.2 | 75.3 \| 74.9 | 45.6 \| 40.6 |

The full results can be found in the the Appendix, in Table A.2. We performed experiments with oversampling the minority class, but as the results were not useful we omit these experiments from the thesis.

## 5.3 Progressive Inference

In the following section we present the results from the method described in Algorithm 1, when sampling posts randomly. For each test user, we add 25 posts randomly sampled from their timeline at every iteration. Furthermore, for each user, we run the described method 10 times to account for randomness.

Figure 5.2 shows the classifiers' performance change over all 40 venues. After approximately 60 iterations (1600 posts), the $F_1$ score across the 40 venues.

There are venue types which attain a stable $F_1$ score quicker than others. We make an informal split of the evaluated venue types in terms of being *quickly learned*, *slowly learned* and *hard to learn*, based on how many iterations it takes for the $F_1$ score to stabilize and how it changes when more posts are added. This split is solely made for readability purposes of the following graphs.

In Figure 5.3, we show the performance of classifiers for some venue types which are *quickly learned*.

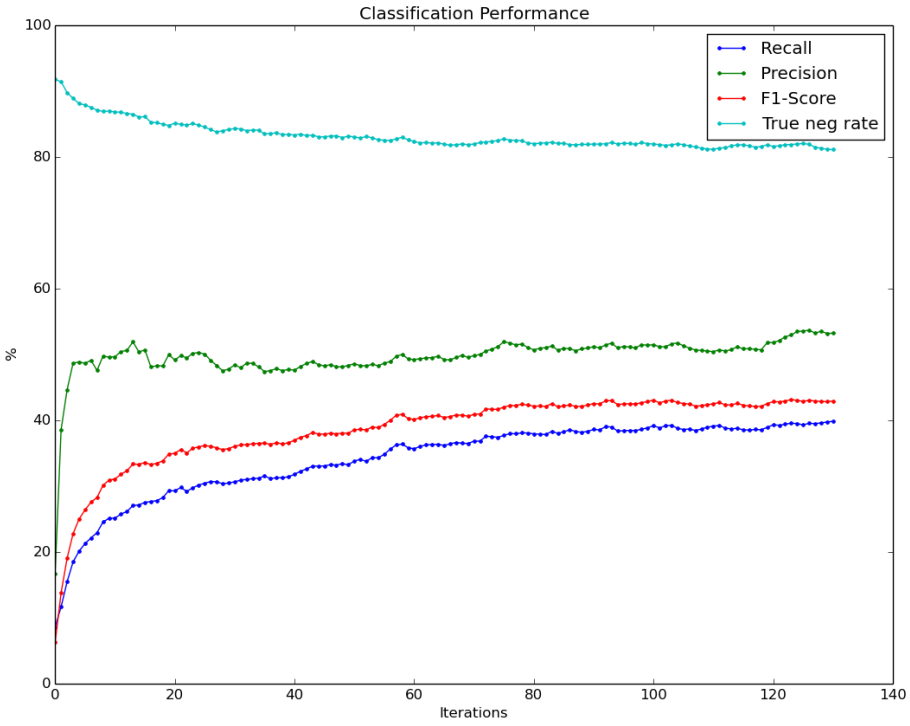



Figure 5.2: Classification performance averaged over 40 venue types. At each iteration, 25 posts are randomly selected from the test user's timeline, added to the truncated timeline and the truncated timeline is classified.

In Figure 5.4, we show the performance of classifiers for some venue types which are *learned slower*.

In Figure 5.5, we show the performance of classifiers for some venue types which are *hard to learn*.

## 5.4 Measuring information content

In this section, we present the results of the method described in Algorithm 1, but this time using the function which samples posts based on their information score. At each iteration, the remaining posts are re-ranked in terms of their information score and the top posts are added to the truncated timeline.

The setup of the experiment is the following:

- Iteration 1–10: Sample 1 post at a time.

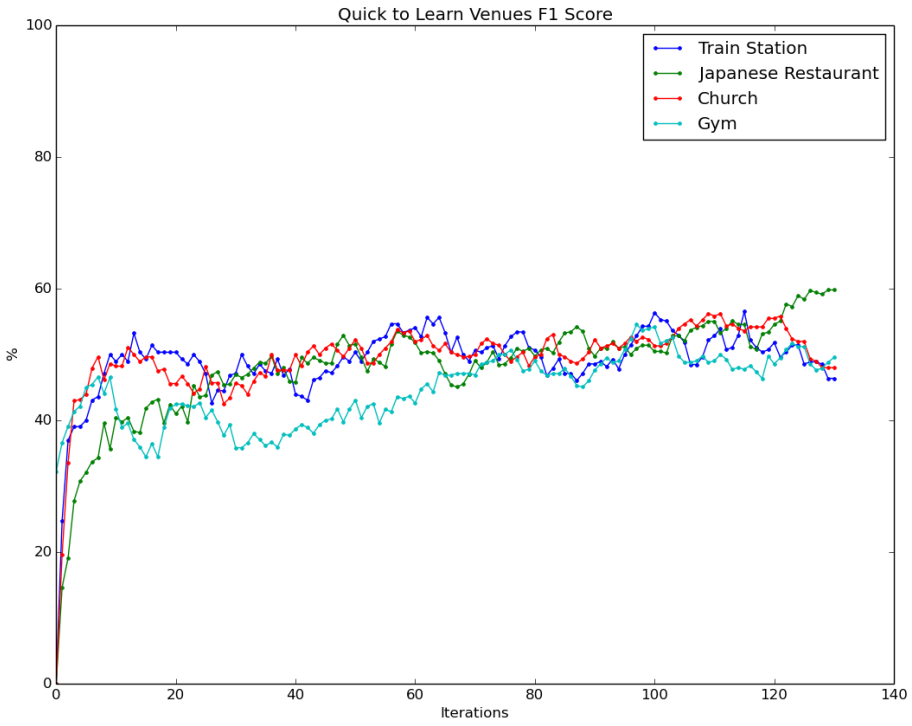



Figure 5.3: Example of *quickly learned* venue types. Y axis represents the $F_1$ score.

- Iteration 11–14: Sample 10 posts at a time.

In total, we add 50 tweets which are actively selected and we measure the classifiers' performance in this *reduced* test set. The setup is done in this way so that the task is computationally tractable for the given time. Furthermore, we restrict the analysis to a pre-selected list of venues, which are commercially interesting or can reflect a behaviour or a habit of a user.

The list of pre-selected venues is the following: *Church*, *Gym*, *Resort*, *Lounge*, *Gastropub*, *Sports Bar*, *Concert Hall*, *Automobile Shop*.

Table 5.4 summarizes the average performance of the classifiers. The individual classifiers can be found in the appendix in Table A.3 to Table A.10.

We can see that for different $\lambda$'s we can beat the baseline.

In Figures 5.6, 5.7 and 5.8 and we can see the performance of different classifiers when we actively sample which posts to include in the test set next while varying the parameter $\lambda$ from the information score, which regulates

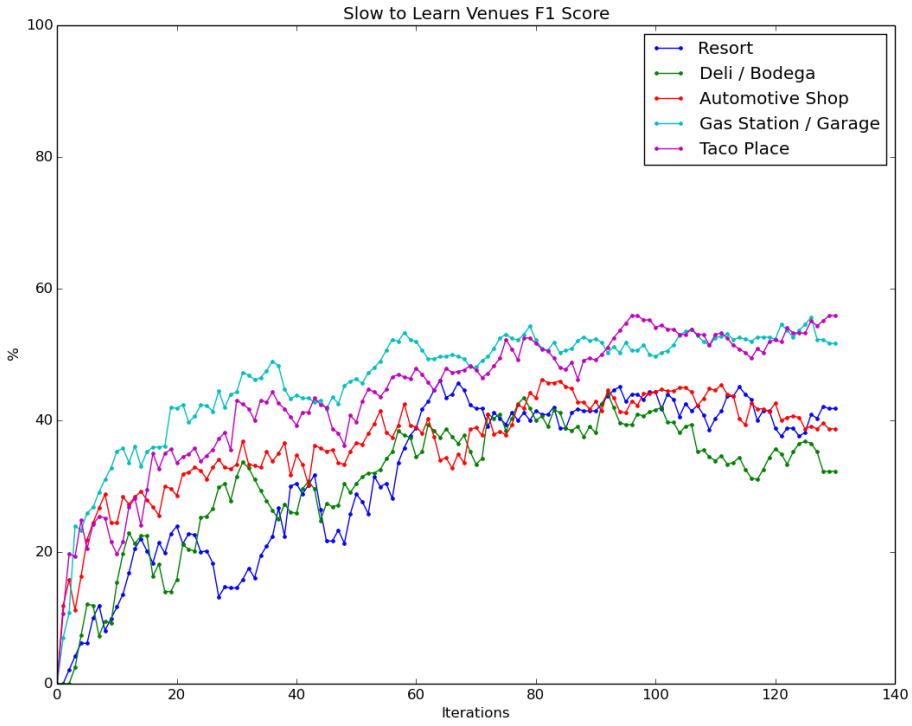



Figure 5.4: Example of *slowly learned* venue types. Y axis represents the $F_1$ score.

Table 5.4: Average classifier performance over the 8 venue types, for different $\lambda$. The first row expresses the beginning of the simulation, when the truncated timelines have only one post which was randomly sampled from the corresponding timelines. The last row represents baseline, which is when the posts are randomly sampled.

| Iteration | Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
|---|---|---|---|---|---|---|
| 0 | 65.8 | 0.0 | 0.0 | 99.6 | 0.0 | - |
| 50 | 69.3 | 46.8 | 41.7 | 80.5 | 41.9 | 0.0 |
| 50 | 69.1 | 48.6 | 41.2 | 80.7 | 42.0 | 0.2 |
| 50 | 70.5 | 50.7 | 41.9 | 82.4 | 43.8 | 0.4 |
| 50 | 69.1 | 49.1 | 41.5 | 80.6 | 43.1 | 0.5 |
| 50 | 70.0 | 53.6 | 42.1 | 81.8 | 44.9 | 0.6 |
| 50 | 69.0 | 47.0 | 38.1 | 82.2 | 40.2 | 0.8 |
| 50 | 68.1 | 29.2 | 8.7 | 96.3 | 12.6 | 1.0 |
| 50 | 71.8 | 81.6 | 22.1 | 97.4 | 34.8 | - |

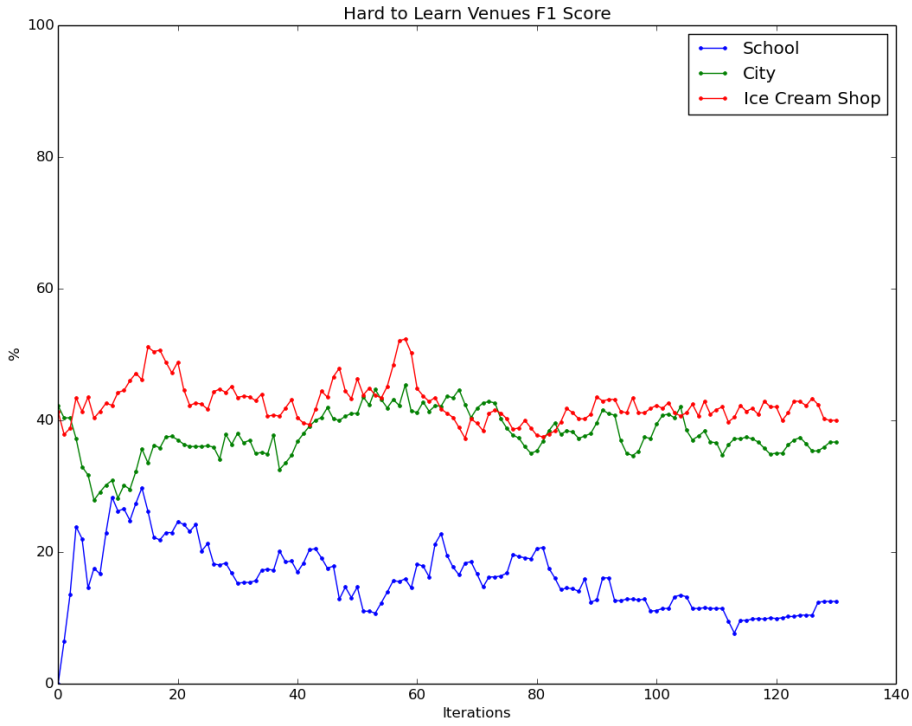



Figure 5.5: Example of *hard to learn* venue types. Y axis represents the $F_1$ score.

the preference towards novelty or relevancy. The *Baseline* label denotes the case where posts are randomly selected.

In Table 5.5, the highest ranked posts for different venue types are listed. These are randomly sampled from the top ranked tweets for each venue type classifier.

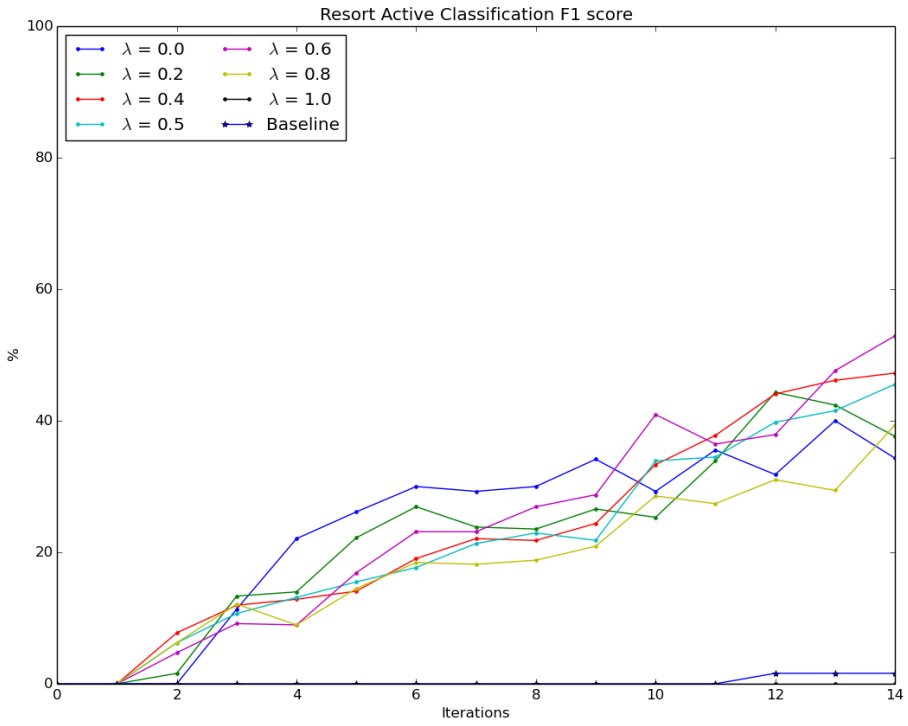



Figure 5.6: $F_1$ score of classifier *Resort* when actively choosing posts using information score functions with different $\lambda$.

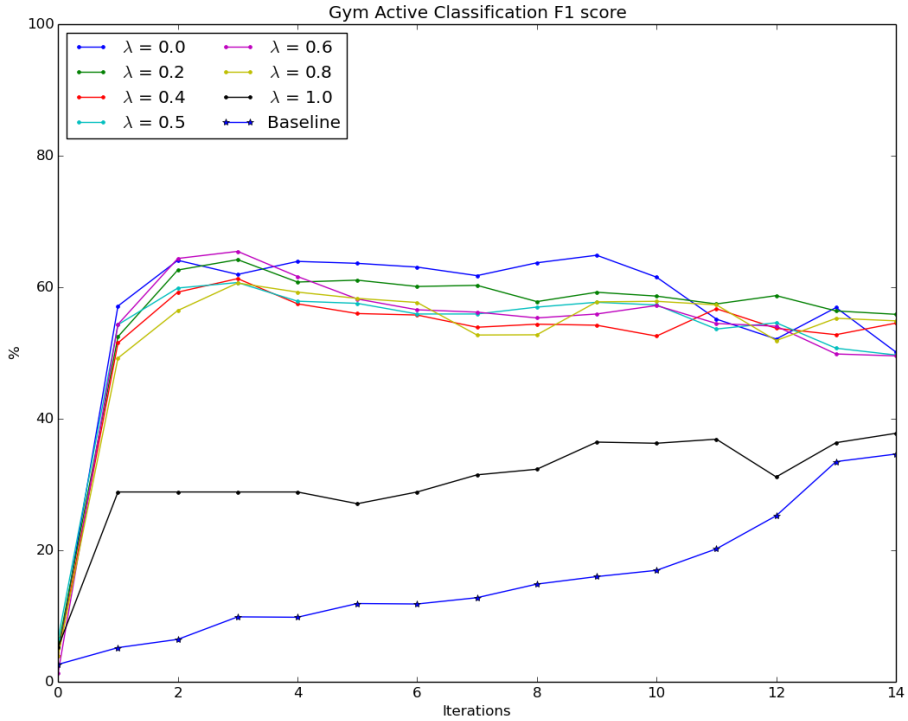



Figure 5.7: $F_1$ score of classifier *Gym* when actively choosing posts using information score functions with different $\lambda$.

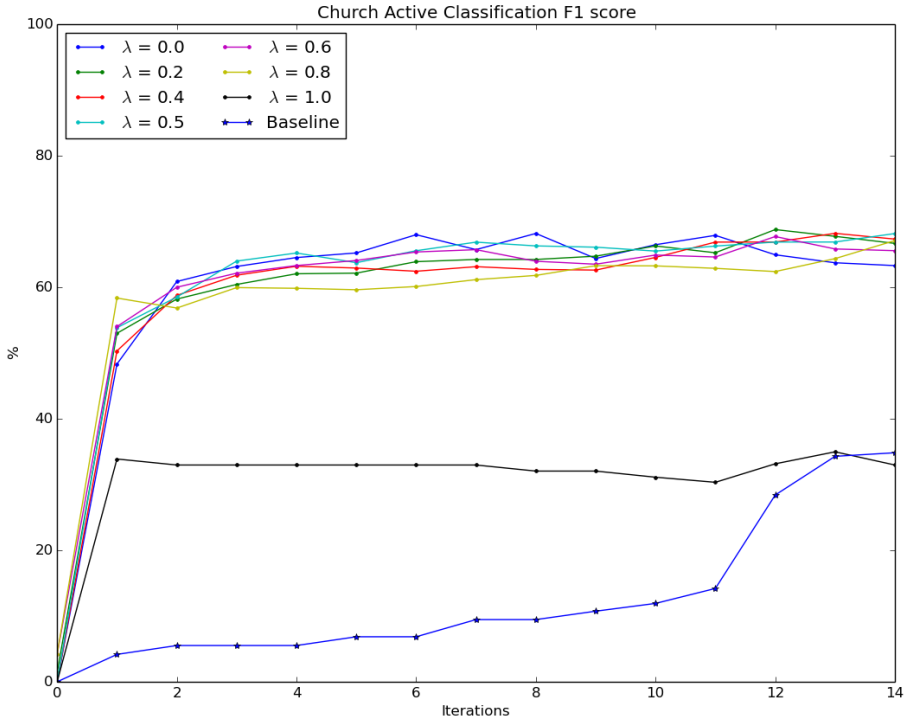



Figure 5.8: $F_1$ score of classifier *Church* when actively choosing posts using using information score functions with different $\lambda$.

Table 5.5: Top ranked posts using the designed information score function. The links and user mentions are omitted. Posts that contain links to Instagram and Foursquare are denoted as *[INSTAGRAMLINK]* and *[FOURSQUARELINK]*, respectively. All other links are replaced by *[LINK]*.

| Venue Type | Top Ranked Posts |
|---|---|
| Church | Saint Anthony (@ Cathedral of Saint Paul) *[FOURSQUARELINK]* |
| | #teamCatholic #justbecause #meanwhileatoctane #icecubephotoshoot #youheardme #perfecticecube *[INSTAGRAMLINK]* |
| | I'm at Church of Our Lady Of Perpetual Succour (Singapore) w/ 3 others *[FOURSQUARELINK]* |
| | Ready for a beautiful family day - first stop - church :) *[INSTAGRAMLINK]* |
| Gym | It's cool to love to win, but it's better to hate to lose. (@ Prairie Life Health & Fitness) |
| | When @USER sang happy birthday to me at the gym while on the treadmill. |
| | #outdoor #fitness better than #gym @USER |
| | How Low-Cost Gyms like Planet Fitness Psychologically Manipulate Members Into NOT Going To The Gym *[LINK]* |
| | @USER got a nice 45 min workout in gym this morning in prep for tomorrows tennis match..did light weights n cardio with stretching |
| Resort | HELLO Good morning (at @CaesarsPalace Hotel Casino in Las Vegas, NV) *[FOURSQUARELINK]* |
| | My view ...good night (@ The @Cheesecake Factory in Las Vegas, NV) *[FOURSQUARELINK] [INSTAGRAMLINK]* |
| | I'm at Luxor Hotel Casino (Las Vegas, NV) w/ 6 others *[FOURSQUARELINK]* |
| | STAYING OVERNIGHT AT RESORT WORLD HOTEL LTR YAY HEHEHEHEHE SHIOK |
| | I'm at Treasure Island - TI Hotel & Casino (Las Vegas, NV) w/ 6 others *[FOURSQUARELINK]* |

Chapter 6

# Conclusion

In this thesis, we investigate three fundamental questions. Firstly, we study cross OSN inference by fixing one OSN (Foursquare) as the inference's target and making the inference using information from two other OSNs (Twitter and Instagram). We can then understand the nature and information carried by different OSNs.

With the objective to better understand the information content contained in a single post within the timeline of a user and with respect to a particular inference task, we formulate an information score function. It attributes a score to a post based on its *relevance* towards a particular inference task and *novelty* to the user's previously seen timeline.

Finally, we seek to assign a tangible commercial value to a profile with respect to its hosting OSN. We use pricing models which are generally used to price an advertising campaign to price the user's potential to generate revenue for the OSN and take this as the profile's value. We use the information we inferred from users in the cross OSN inference task to model the users' interest in related ads and thus, the click probability on a related ad.

Instagram and Twitter have different information content. For example, for some venue types, Instagram has a richer vocabulary than Twitter, leading to an improvement in some classifiers performances, even though the Instagram data set is much smaller than the Twitter data set[1]. A hypothesis is that Instagram posts are, in general, more personal than tweets. Instagram posts are mostly pictures taken by the user, whereas tweets can serve various purposes: sharing personal content and opinions, reposting content and chatter.

We also found that some venue types are more predictable than others. For example, for venue types such as *Gym*, *Church*, *Sushi Restaurant* and Bars

[1]We are only able to recover up to 33 posts from Instagram per user, whereas we can recover up to 3200 posts per user in Twitter

(*Gastropub*, *Sports Bar*, *Cocktail Bars*) we were able to attain a reasonable performance with our classifier. For other venue types, such as: *Concert Hall*, *Furniture / Home Store* and *Plaza*, the classifier's performance was significantly worse. There might be several reasons for this, such as:

1. Not enough data;
2. Noisy labels (some venues - such as *Home*, should have nearly 100 % of positive labels and not the 36 % described in the data set);
3. Some venues can be expected to have a more distinct vocabulary than others.

Some venues are more prone to be broad-casted than others, for instance, it could be that going to the gym is perceived as being more interesting than going home. In other cases, the vocabulary might either be ambiguous e.g. *Resort* and *Hotel* share a similar vocabulary, or not specific enough e.g. vocabulary associated to *Plaza*.

When we randomly sample posts from a user's timeline and make a prediction using this truncated timeline, we observe that for some venue types the performance curve is steeper than for others. We made an informal split of venue types which are *quick to learn*, *slow to learn* and *hard to learn*. The steepness is probably related to how frequently users visit these places. For instance, it is expected that a user goes to a gym more often than a resort. For the venues which are hard to learn, the lack of specific vocabulary might be one of the causes, as well as temporal artifacts from the data set (majority of the data is collected from Europe and North America during the months of January and February 2015, so the venue type *Ice Cream Shop* might not have a representative vocabulary which it would have had in the summer).

We show that the information score introduced in this study reflects the relative importance of posts with respect to the inference task we are performing. When actively selecting a subset of posts per user, we were able to beat the baseline (of randomly sampling posts) for the 8 venues chosen (*Church*, *Gym*, *Resort*, *Lounge*, *Gastropub*, *Sports Bar*, *Concert Hall*, *Automobile Shop*). In addition, for some venue types (e.g. for *Resort*, *Church* and *Sports Bar*), we can find a $\lambda \in [0, 1]$ that regulates the *novelty-relevance* trade-off, and attain a better classifier performance which outperforms using the users' full timelines.

## 6.1 Future work

There are several things that can be developed in the future.

Because one of the terms of this information score function, *relevance*, is dependent on the classifier, this informativeness score performance will de-

pend, to some degree, on whether the classifier is correctly identifying relevant words (features) or not. There is also a preference for longer posts in the beginning of the selection process, because longer posts are more likely to have a higher *novelty* score when the vocabulary that has been seen is small. Something to consider for future work would be to have an adaptive $\lambda$ which depends on how many posts have been seen. Functions other than linear ones can also be considered to better model the information score of a post.

The feature space we use for inference task is always given by a vector space model. There is a lot more information in a profile than just the user timeline's textual data, that could be included to better model the user. Furthermore, we do not model the conditional probability of user visiting venue type A, having already visited venue type B. For example, it would make sense to estimate the probability of user visiting a museum, knowing the user has been to a arts venue.

We make an attempt to attribute a value to a user's profile with respect to its hosting OSN. We use simple pricing models which are known from pricing advertising campaigns. However, one interesting aspect would be to include a privacy loss term. Authors of [14] design a market-place for private data exchange, where the user is compensated based on their privacy loss, calculated using differential privacy. One direction that could be explored would be to use the introduced information score function to approximate the privacy loss of the user.

Going towards the direction of building a stand-alone application which estimates the information leak in posts, a future task would be to validate the information score for predicting venue type visits on timelines which do not have Foursquare check-ins (or where the check-ins have been removed).

Appendix A

# Appendix A: Complete results

The full development of this thesis and results can be found in the GitHub repository [24]. The Wiki of the project reports all results, including the failed attempts, of this thesis [25].

| Venue | % Users visited |
|---|---|
| Spiritual Center | 36.73 |
| Home (private) | 36.57 |
| Other Great Outdoors | 36.57 |
| Gym | 36.08 |
| Neighborhood | 36.08 |
| Sports Bar | 35.92 |
| Lounge | 35.60 |
| Miscellaneous Shop | 35.44 |
| Salon / Barbershop | 34.95 |
| Performing Arts Venue | 34.79 |
| Supermarket | 34.14 |
| Sushi Restaurant | 33.66 |
| City | 33.66 |
| Nightclub | 33.01 |
| Cocktail Bar | 33.01 |
| Ice Cream Shop | 33.01 |
| Japanese Restaurant | 32.69 |
| Church | 31.88 |
| Dessert Shop | 31.55 |
| Gastropub | 31.39 |
| Brewery | 30.91 |
| School | 30.42 |
| Taco Place | 29.77 |
| Gas Station / Garage | 28.64 |
| Automotive Shop | 28.48 |
| Resort | 27.99 |
| Museum | 27.51 |
| Drugstore / Pharmacy | 27.51 |
| Deli / Bodega | 27.02 |
| Electronics Store | 26.21 |
| Concert Hall | 26.21 |
| Train Station | 26.05 |
| Wine Bar | 26.05 |
| Doctor's Office | 25.89 |
| Furniture / Home Store | 25.73 |
| New American Restaurant | 25.57 |
| Beach | 25.40 |
| Plaza | 25.08 |
| Residential Building (Apartment / Condo) | 24.92 |
| Beer Garden | 24.60 |

Table A.1: 40 selected venues and percentage of the positive class.

| Venue | $F_1$ Score | Accuracy | Precision | Recall | Specificity |
|---|---|---|---|---|---|
| City | 47.33 \| 50.98 | 67.31 \| 69.10 | 51.76 \| 54.59 | 43.59 \| 47.81 | 79.35 \| 79.77 |
| Automotive Shop | 45.14 \| 40.96 | 72.32 \| 70.38 | 51.48 \| 45.79 | 40.19 \| 37.04 | 85.14 \| 83.73 |
| Concert Hall | 27.21 \| 23.69 | 68.45 \| 66.84 | 34.05 \| 30.75 | 22.65 \| 19.27 | 84.88 \| 83.84 |
| Cocktail Bar | 46.22 \| 51.90 | 67.99 \| 71.06 | 51.52 \| 58.30 | 41.91 \| 46.77 | 80.38 \| 83.15 |
| Gastropub | 50.76 \| 42.96 | 71.03 \| 65.53 | 53.91 \| 45.28 | 47.95 \| 40.86 | 81.46 \| 77.16 |
| Museum | 43.44 \| 38.33 | 71.85 \| 70.22 | 48.34 \| 44.13 | 39.44 \| 33.87 | 84.11 \| 84.00 |
| Wine Bar | 36.57 \| 31.63 | 71.21 \| 68.94 | 44.41 \| 35.41 | 31.09 \| 28.59 | 85.73 \| 83.12 |
| Spiritual Center | 57.25 \| 54.21 | 69.74 \| 68.78 | 60.11 \| 58.87 | 54.65 \| 50.23 | 78.82 \| 79.87 |
| Lounge | 50.10 \| 53.18 | 68.29 \| 68.94 | 57.16 \| 58.18 | 44.59 \| 48.97 | 81.18 \| 80.53 |
| Performing Arts Venue | 45.56 \| 40.65 | 63.74 \| 61.32 | 48.15 \| 44.78 | 43.23 \| 37.21 | 75.29 \| 74.88 |
| Residential Building (Apartment / Condo) | 32.34 \| 32.09 | 71.37 \| 72.49 | 41.50 \| 43.14 | 26.49 \| 25.55 | 86.74 \| 88.37 |
| Dessert Shop | 39.47 \| 43.07 | 66.36 \| 66.67 | 45.63 \| 46.25 | 34.77 \| 40.29 | 80.61 \| 79.10 |
| Beach | 31.60 \| 31.01 | 70.39 \| 69.75 | 37.87 \| 37.13 | 27.12 \| 26.63 | 85.46 \| 84.75 |
| Brewery | 51.08 \| 46.92 | 72.96 \| 69.58 | 59.63 \| 51.50 | 44.68 \| 43.09 | 86.30 \| 81.77 |
| Home (private) | 50.09 \| 51.75 | 64.88 \| 66.03 | 51.26 \| 53.81 | 48.98 \| 49.85 | 73.98 \| 74.78 |
| Gym | 49.48 \| 56.18 | 65.67 \| 70.70 | 53.04 \| 61.62 | 46.36 \| 51.62 | 76.64 \| 81.84 |
| Furniture / Home Store | 33.27 \| 32.08 | 70.22 \| 69.75 | 40.41 \| 38.10 | 28.27 \| 27.71 | 84.97 \| 84.51 |
| Supermarket | 56.69 \| 55.74 | 71.86 \| 73.00 | 60.21 \| 62.64 | 53.55 \| 50.21 | 81.51 \| 84.87 |
| Train Station | 51.74 \| 51.56 | 76.55 \| 78.17 | 56.15 \| 62.77 | 47.97 \| 43.75 | 86.89 \| 90.34 |
| Miscellaneous Shop | 44.10 \| 48.31 | 64.08 \| 65.53 | 49.14 \| 52.15 | 39.99 \| 45.00 | 77.19 \| 76.75 |
| Japanese Restaurant | 49.39 \| 48.40 | 67.81 \| 68.46 | 51.34 \| 53.17 | 47.58 \| 44.42 | 77.49 \| 80.19 |
| Deli / Bodega | 38.19 \| 41.73 | 71.67 \| 71.83 | 49.42 \| 47.86 | 31.12 \| 36.98 | 87.08 \| 84.99 |
| Sushi Restaurant | 50.37 \| 55.33 | 68.44 \| 71.21 | 52.60 \| 57.36 | 48.33 \| 53.44 | 78.58 \| 80.30 |
| Other Great Outdoors | 42.33 \| 41.20 | 61.00 \| 60.70 | 46.82 \| 45.03 | 38.62 \| 37.97 | 75.55 \| 73.78 |
| Drugstore / Pharmacy | 54.00 \| 55.12 | 77.04 \| 77.67 | 61.62 \| 61.04 | 48.06 \| 50.25 | 88.22 \| 87.93 |
| School | 42.46 \| 43.43 | 66.16 \| 68.43 | 45.36 \| 48.96 | 39.91 \| 39.01 | 77.31 \| 81.20 |
| Gas Station / Garage | 52.26 \| 54.76 | 75.41 \| 77.54 | 57.27 \| 62.12 | 48.06 \| 48.96 | 86.87 \| 88.64 |
| Plaza | 36.40 \| 35.18 | 73.15 \| 72.19 | 42.61 \| 44.18 | 31.77 \| 29.23 | 86.71 \| 86.78 |
| New American Restaurant | 40.97 \| 39.12 | 72.96 \| 72.65 | 46.57 \| 45.18 | 36.57 \| 34.50 | 85.67 \| 85.87 |
| Nightclub | 45.64 \| 48.58 | 66.65 \| 69.24 | 48.91 \| 54.66 | 42.79 \| 43.72 | 78.72 \| 82.24 |
| Beer Garden | 26.60 \| 28.18 | 70.71 \| 69.73 | 34.37 \| 34.53 | 21.69 \| 23.81 | 86.77 \| 85.05 |
| Electronics Store | 40.74 \| 35.01 | 73.14 \| 72.50 | 48.05 \| 46.25 | 35.36 \| 28.17 | 86.69 \| 87.68 |
| Ice Cream Shop | 38.07 \| 43.01 | 62.47 \| 65.21 | 42.05 \| 46.97 | 34.77 \| 39.67 | 76.00 \| 77.91 |
| Sports Bar | 58.12 \| 54.67 | 70.25 \| 68.79 | 59.62 \| 58.11 | 56.71 \| 51.61 | 78.23 \| 78.71 |
| Neighborhood | 48.07 \| 48.56 | 63.44 \| 65.71 | 49.28 \| 54.32 | 46.92 \| 43.91 | 72.63 \| 77.85 |
| Doctor's Office | 35.24 \| 37.10 | 70.84 \| 73.77 | 41.55 \| 46.50 | 30.60 \| 30.86 | 84.94 \| 88.85 |
| Resort | 53.17 \| 49.86 | 74.92 \| 76.06 | 54.53 \| 57.22 | 51.88 \| 44.17 | 83.31 \| 87.66 |
| Taco Place | 45.85 \| 47.78 | 69.75 \| 71.20 | 49.46 \| 51.67 | 42.72 \| 44.44 | 81.55 \| 82.98 |
| Church | 54.60 \| 57.18 | 74.92 \| 75.26 | 64.05 \| 63.96 | 47.57 \| 51.71 | 87.96 \| 87.07 |
| Salon / Barbershop | 45.14 \| 44.76 | 65.85 \| 64.73 | 50.57 \| 50.27 | 40.77 \| 40.33 | 78.98 \| 78.04 |

Table A.2: Classifier's performance on the 40 selected venues, averaged across 10 cross-validation folds. The results correspond to the performance using Twitter and Twitter+Instagram data, in the following form Twitter | Twitter+Instagram.

Table A.3: *Gym* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Gym* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 63.35 | 48.19 | 27.03 | 83.71 | 34.63 | - |
| 57.52 | 43.35 | 59.46 | 56.44 | 50.14 | 0.0 |
| 63.59 | 49.48 | 64.19 | 63.26 | 55.88 | 0.2 |
| 63.59 | 49.45 | 60.81 | 65.15 | 54.55 | 0.4 |
| 60.19 | 45.51 | 54.73 | 63.26 | 49.69 | 0.5 |
| 58.01 | 43.59 | 57.43 | 58.33 | 49.56 | 0.6 |
| 65.29 | 51.48 | 58.78 | 68.94 | 54.89 | 0.8 |
| 67.23 | 59.42 | 27.70 | 89.39 | 37.79 | 1.0 |

Table A.4: *Resort* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Resort* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 70.15 | 100.00 | 0.81 | 100.00 | 1.60 | - |
| 73.06 | 64.44 | 23.39 | 94.44 | 34.32 | 0.0 |
| 74.27 | 69.57 | 25.81 | 95.14 | 37.65 | 0.2 |
| 76.70 | 74.14 | 34.68 | 94.79 | 47.25 | 0.4 |
| 76.21 | 73.21 | 33.06 | 94.79 | 45.56 | 0.5 |
| 78.40 | 76.92 | 40.32 | 94.79 | 52.91 | 0.6 |
| 73.06 | 61.02 | 29.03 | 92.01 | 39.34 | 0.8 |
| 69.90 | 0.00 | 0.00 | 100.00 | 0.00 | 1.0 |

Table A.5: *Church* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Church* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 71.84 | 81.58 | 22.14 | 97.43 | 34.83 | - |
| 71.84 | 56.82 | 71.43 | 72.06 | 63.29 | 0.0 |
| 74.27 | 69.57 | 25.81 | 95.14 | 37.65 | 0.2 |
| 75.73 | 62.05 | 73.57 | 76.84 | 67.32 | 0.4 |
| 75.97 | 61.99 | 75.71 | 76.10 | 68.17 | 0.5 |
| 74.76 | 61.11 | 70.71 | 76.84 | 65.56 | 0.6 |
| 75.24 | 61.18 | 74.29 | 75.74 | 67.10 | 0.8 |
| 70.39 | 71.43 | 21.43 | 95.59 | 32.97 | 1.0 |

Table A.6: *Lounge* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Lounge* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 62.38 | 80.00 | 7.32 | 98.79 | 13.41 | - |
| 61.89 | 52.12 | 52.44 | 68.15 | 52.28 | 0.0 |
| 62.38 | 52.69 | 53.66 | 68.15 | 53.17 | 0.2 |
| 65.29 | 57.34 | 50.00 | 75.40 | 53.42 | 0.4 |
| 61.89 | 52.29 | 48.78 | 70.56 | 50.47 | 0.5 |
| 63.83 | 55.47 | 46.34 | 75.40 | 50.50 | 0.6 |
| 60.68 | 50.86 | 35.98 | 77.02 | 42.14 | 0.8 |
| 59.95 | 0.00 | 0.00 | 99.60 | 0.00 | 1.0 |

Table A.7: *Gastropub* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Gastropub* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 68.20 | 33.33 | 2.34 | 97.89 | 4.38 | - |
| 65.78 | 45.70 | 53.91 | 71.13 | 49.46 | 0.0 |
| 66.99 | 47.22 | 53.12 | 73.24 | 50.00 | 0.2 |
| 68.93 | 50.00 | 52.34 | 76.41 | 51.15 | 0.4 |
| 70.39 | 52.21 | 55.47 | 77.11 | 53.79 | 0.5 |
| 68.45 | 49.22 | 49.22 | 77.11 | 49.22 | 0.6 |
| 66.02 | 45.16 | 43.75 | 76.06 | 44.44 | 0.8 |
| 69.42 | 66.67 | 3.12 | 99.30 | 5.97 | 1.0 |

Table A.8: *Sports Bar* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Sports Bar* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 64.81 | 42.86 | 2.08 | 98.51 | 3.97 | - |
| 71.84 | 61.86 | 50.69 | 83.21 | 55.73 | 0.0 |
| 70.39 | 59.82 | 46.53 | 83.21 | 52.34 | 0.2 |
| 71.84 | 62.07 | 50.00 | 83.58 | 55.38 | 0.4 |
| 69.90 | 58.62 | 47.22 | 82.09 | 52.31 | 0.5 |
| 72.33 | 63.64 | 48.61 | 85.07 | 55.12 | 0.6 |
| 70.39 | 59.02 | 50.00 | 81.34 | 54.14 | 0.8 |
| 64.32 | 0.00 | 0.00 | 98.88 | 0.00 | 1.0 |

Table A.9: *Automotive Shop* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Automotive Shop* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 72.33 | 46.88 | 13.39 | 94.33 | 20.83 | - |
| 67.48 | 13.33 | 3.57 | 91.33 | 5.63 | 0.0 |
| 68.93 | 25.00 | 7.14 | 92.00 | 11.11 | 0.2 |
| 70.87 | 37.50 | 10.71 | 93.33 | 16.67 | 0.4 |
| 66.02 | 24.07 | 11.61 | 86.33 | 15.66 | 0.5 |
| 70.87 | 40.00 | 14.29 | 92.00 | 21.05 | 0.6 |
| 69.42 | 30.56 | 9.82 | 91.67 | 14.86 | 0.8 |
| 69.17 | 36.36 | 17.86 | 88.33 | 23.95 | 1.0 |

Table A.10: *Concert Hall* average classifier performance, for different $\lambda$ in the information score function. The first row represents the simulation that uses a sampling function which is random.

| *Concert Hall* | | | | | |
|---|---|---|---|---|---|
| Accuracy | Precision | Recall | Specificity | $F_1$ Score | $\lambda$ |
| 74.51 | 0.00 | 0.00 | 99.68 | 0.00 | - |
| 71.84 | 16.67 | 2.88 | 95.13 | 4.92 | 0.0 |
| 71.60 | 24.00 | 5.77 | 93.83 | 9.30 | 0.2 |
| 70.63 | 13.04 | 2.88 | 93.51 | 4.72 | 0.4 |
| 71.84 | 25.00 | 5.77 | 94.16 | 9.38 | 0.5 |
| 73.30 | 38.46 | 9.62 | 94.81 | 15.38 | 0.6 |
| 71.84 | 16.67 | 2.88 | 95.13 | 4.92 | 0.8 |
| 74.27 | 0.00 | 0.00 | 99.35 | 0.00 | 1.0 |

# Bibliography


[1] GustavoE.A.P.A. Batista, RonaldoC. Prati, and MariaC. Monard. Balancing strategies and class overlapping. In A. Fazel Famili, Joost N. Kok, José M. Peña, Arno Siebes, and Ad Feelders, editors, *Advances in Intelligent Data Analysis VI*, Lecture Notes in Computer Science, pages 24–35. Springer Berlin Heidelberg, 2005.

[2] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[3] Leo Breiman. Arcing classifier (with discussion and a rejoinder by the author). *Ann. Statist.*, 26:801–849, 1998.

[4] eBiz — MBA. *Top 15 Most Popular Social Networking Sites — April 2015*, 2015 (accessed April 21, 2015). http://www.ebizmba.com/articles/social-networking-websites.

[5] Raphael Ottoni et al. Of Pins and Tweets: Investigating how users behave across image- and text-based social networks. In *Proceedings of the 8th International AAAI Conference on Weblogs and Social Media (ICWSM'14)*.

[6] Foursquare. *About Foursquare*, 2015 (accessed April 21, 2015). https://foursquare.com/about.

[7] Foursquare. *Foursquare API overview*, 2015 (accessed May 2, 2015). https://developer.foursquare.com/overview/.

[8] J. Golbeck, C. Robles, M. Edmondson, and K. Turner. Predicting personality from twitter. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 149–156, Oct 2011.

[9] Yuheng Hu, Lydia Manikonda, and Subbarao Kambhampati. What we instagram: A first analysis of instagram photo content and user types, 2014.

[10] Nathalie Japkowicz. The class imbalance problem: Significance and strategies. In *In Proceedings of the 2000 International Conference on Artificial Intelligence (ICAI*, pages 111–117, 2000.

[11] Michal Kosinski, David Stillwell, and Thore Graepel. Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, February 2013. Available at http://www.pnas.org/content/early/2013/03/06/1218772110.

[12] Ravi Kumar and Andrew Tomkins. A characterization of online browsing behavior. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 561–570. ACM, 2010.

[13] David Lazer, Alex Pentland, and et al. Computational social science. *Science*, 323(5915):721–723, 2009.

[14] Chao Li, Daniel Yang Li, Gerome Miklau, and Dan Suciu. A theory of pricing private data. *ACM Trans. Database Syst.*, 39, 2014.

[15] Wen Li, Carsten Eickhoff, and Arjen P. de Vries. Want a coffee?: predicting users' trails. In William R. Hersh, Jamie Callan, Yoelle Maarek, and Mark Sanderson, editors, *SIGIR*, pages 1171–1172. ACM, 2012.

[16] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[17] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht. A Comparison of Random Forest and its Gini Importance with Standard Chemometric Methods for the Feature Selection and Classification of Spectral Data. *BMC Bioinformatics*, 10:213, 2009.

[18] D. Quercia, M. Kosinski, D. Stillwell, and J. Crowcroft. Our twitter profiles, our selves: Predicting personality with twitter. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 180–185, Oct 2011.

[19] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: Estimating the click-through rate for new ads. In *Proceedings of the 16th International World Wide Web Conference(WWW-2007)*, 2007.

[20] Prof. Dr. Benno Stein. *Decision Trees Lecture*, 2015 (accessed May 6, 2015). http://www.uni-weimar.de/medien/webis/teaching/lecturenotes/machine-learning/unit-en-decision-trees-impurity.pdf.

[21] Twitter. *Twitter Ads overview*, 2015 (accessed May 2, 2015). https://biz.twitter.com/ad-products.

[22] Twitter. *Twitter Rest API overview*, 2015 (accessed May 2, 2015). https://dev.twitter.com/rest/public.

[23] Princeton University. *About WordNet*, 2015 (accessed May 9, 2015). http://wordnet.princeton.edu.

[24] Maria Veiga. *Master thesis repository*, 2015 (accessed May 10, 2015). https://github.com/hanveiga/master-thesis.

[25] Maria Veiga. *Master thesis wiki*, 2015 (accessed May 10, 2015). https://github.com/hanveiga/master-thesis/wiki.

[26] Kohki Yamaguchi. *Pay Per What? Choosing Pricing Models In Digital Advertising*, 2014 (accessed April 22, 2015). http://marketingland.com/pay-per-pricing-models-digital-advertising-97913.

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

# Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

---

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

**Title of work** (in block letters):

**Authored by** (in block letters):
*For papers written by groups the names of all authors are required.*

| **Name(s):** | **First name(s):** |
|---|---|
| | |
| | |
| | |
| | |

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

| **Place, date** | **Signature(s)** |
|---|---|
| | |
| | |
| | |
| | |

*For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.*