Clustering System Data using Aggregate Measures

Johnnie C-N. Chang
UC Santa Cruz
cchan60@ucsc.edu

Robert H-J. Chen
UC Santa Cruz
hchen118@ucsc.edu

Jay Pujara
Information Sciences Institute
jay@cs.umd.edu

Lise Getoor
UC Santa Cruz
getoor@soe.ucsc.edu

ABSTRACT

Many real-world systems generate a tremendous amount of data cataloging the actions, responses, and internal states. Prominent examples include user logs on web servers, instrumentation of source code, and performance statistics in large data centers. The magnitude of this data makes it impossible to log individual events, but instead requires capturing aggregate statistics at a coarser granularity, resulting in statistical distributions instead of discrete values. We survey several popular statistical distance measures and demonstrate how appropriate statistical distances can allow meaningful clustering of web log data.

ACM Reference Format:
Johnnie C-N. Chang, Robert H-J. Chen, Jay Pujara, and Lise Getoor. 2018. Clustering System Data using Aggregate Measures. In *Proceedings of SysML (SysML18).* ACM, New York, NY, USA, 2 pages.

1 INTRODUCTION

Understanding the behavior of complex computational systems is a prerequisite to building better systems that remove performance bottlenecks, enable customization, or adaptively scale. A common obstacle to understanding the behavior of systems is the magnitude of data involved – there can be billions or trillions of events, function calls, or sensor readings in the course of an hour. One solution to this problem is to aggregate these records, capturing the frequency of different states rather than treating them as individual events. Examples of aggregation include measuring the frequency of visits to a web page, profiling the amount of time spent in a function call or call stack, or determining the mean and variance of a particular sensor or resource. However, using aggregated values transforms simple event-based system data into a distribution over values, requiring a different analytical approach.

One of the key questions in understanding systems is determining how different system states or events are related. In the simplest case, where records correspond to the exact reading, well-studied distance functions can identify similar states and provide insight into how the system is behaving. In contrast, with aggregated records, features are often represented as sets or distributions over values, and the appropriate distance metrics must be defined over distributions. An added complexity is that these features often have hierarchical structure, such as IP addresses from the same subnet, function calls from the same method, or readings from sensors of the same resource class. To accurately capture similarities between system states, distance measures must use both the structure of the values as well as their distribution.

In this paper, we perform a case study of a particularly interesting web log dataset capturing the activities of mobile device users. The magnitude of the data makes it impossible to capture every action a user takes on the system, so aggregates are kept for each device identifier and browser cookie on an hourly granularity. The goal of the case study is to compare the efficacy of several statistical distance measures in determining the same or similar users based on ground truth data available from user login information. In the remainder of this paper, we introduce a formal problem definition and several popular statistical distance measures, then compare these approaches in identifying user clusters corresponding to ground truth labels.

2 PROBLEM DEFINITION

We define a set of n records $\mathcal{R} = \{R_1, R_2, \ldots, R_n\}$. We have access to ground truth of the form $R_j = R_k$, stating that records j and k denote the same or a similar system state. Each record, R_i consists of m features, $\{F_1, F_2, \ldots, F_m\}$. Each feature is defined as a distribution over values in a predefined domain D, e.g. $F_i \triangleq \{\forall x \in D_i : p(x)\}$. The ultimate goal is to learn some distance function $d^*(\cdot)$ such that when $d^*(R_j, R_k) > \tau$, $R_j = R_k$ in the ground truth.

3 STATISTICAL DISTANCE METHODS

We consider several different statistical difference methods, ranging from simple derived statistical measures to distances defined over probability distributions. These methods are summarized in Table 1. Methods such as mode and mean use a single statistic to compare the values of two distributions. The mode is useful for discrete values such as countries while the mean is useful for continuous values, like longitude. Jaccard distance measures discard the probability of each feature value, relying only on the overlap of the observed values with nonzero probability, but are simple to compute. More powerful statistical distance measures such as K-L divergence and Hellinger distance directly used the probabilities of each feature value.

SysML18, Feb 2018, Stanford, CA USA
2018.

mode (aggregate discrete)	exact match of most probable value in distribution	$\begin{cases} 1 \text{ if mode}(\forall x : p_i(x))) = \text{mode}(\forall x : p_j(x)) \\ 0 \text{ otherwise} \end{cases}$
mean (aggregate continuous)	L_2 measure of mean values of each feature	$\|\sum_x x p_i(x) - \sum_x x p_j(x)\|_2$
jaccard (set)	omit frequency information and compute set overlap	$\{x_i : p_i(x) > 0\}, \{x_j : p_j(x) > 0\}, \frac{x_i \cap x_j}{x_i \cup x_j}$
K-L divergence (probabilistic)	distance between distributions, non-symmetric	$\sum_x p_i(x) \log \frac{p_i(x)}{p_j(x)}$
Hellinger distance (probabilistic)	distance between distributions, symmetric	$\frac{1}{\sqrt{2}} \sqrt{\sum_x \sqrt{p_i(x)} - \sqrt{p_j(x)}}$

Table 1: A comparison of several distance measures useful for comparing aggregated values

4 DATASET DESCRIPTION

We evaluate the applicability of distance metrics for aggregate measures by testing on an extremely common type of device data: web log records. These records correspond to web requests by users to a web server, and are associated with several types of metadata. We categorize attributes into four types based on their properties as follows:

Type 1. Categorical values: This type of attribute has a set of discrete values, often represented as unique strings
- **Type** is the category of the domain, as defined by the Interactive Advertising Bureau (IAB) (e.g., IAB3)
- **URL** is the hashed toplevel domain of the website (e.g., *facebook.com*).
- **Manufacturer** is the string of the manufacturer name (e.g., *HTC*).
- **Model** and **Device** are identifiers for the device model and device name,

Type 2. Hierarchical tuples of numerical values: These attributes contain a set of numerical values where a hierarchy is established based on the order of values.
- **IP** is the IP address of the visitor, consisting of four octets. Note that due to the hierarchical nature of the attributes, two IP addresses differing only in the last octet, e..g, $(127, 23, 118, \mathit{47})$ and $(127, 23, 118, \mathit{48})$ are still quite similar while a difference in the first octet, e.g., $(\mathit{127}, 23, 118, 47)$ and $(\mathit{128}, 23, 118, 47)$, results in a much lower similarity.

Type 3. Hierarchical complex tuples: These are tuples with a more complicated structure, such as a categorical value and a tuple of numerical values. In these features, there is generally a hierarchy in both the categorical and numerical values.
- **Operating system (OS)** is the tuple of an OS string and an OS version number of the device, which also include hierarchical structure: (Android, 5.1.1) and (iOS, 5.1.1) are very different, but (Android, 5.1.1) and (Android, 4.1.1) are similar.

Type 4. Tuples in $\mathcal{R}^2$: These are 2-dimensional points.
- **Geolocation** is the tuple of latitude and longitude where the device is identified. Each value is a point in $\mathcal{R}^2$, e.g., $(36.96, -122.30)$.

5 EVALUATION

We use a corpus of web logs from mobile devices, where ground truth of browsing behavior is provided by several label providers. Our goal is to capture similarities or biases in the labels provided by each label provider. We perform a k-means clustering of these device logs using three different statistical distance measures: aggregate statistics (mode/mean), set distance (Jaccard), and a probabilistic distance (K-L divergence). Furthermore, we provide a modification of K-L divergence which uses hierarchical structure in attribute values, which implements weighted distances based on the structure of attributes. Weights exponentially decay from most to least significant features. To evaluate these clusterings, we report cluster purity, which measures the percentage of records that come from the majority provider in each cluster. These results are showin in Figure 1. We observe that clusterings generally improve as k increases and more clusters are used. The best aggregate distance measure is clearly a hierarchical distance using the probability distribution over features.

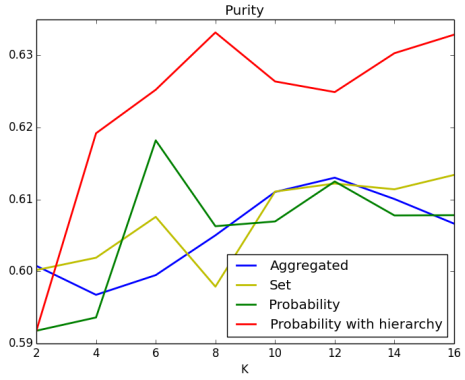

Figure 1: Using a probability distribution with hierarchical, decaying weights has superior results

6 CONCLUSION

In this paper, we describe a common problem setting where system states are compared on the basis of aggregate measures. We evaluate several different measures of statistical distance, and conclude that using probabilistic measures that respect the hierarchical structure in attributes perform best. In future work, we plan to evaluate on a larger corpus of system datasets and articulate general principles for state clustering.