Research Abstract for Semantic Anomaly Detection in Dynamic Data Feeds with Incomplete Specifications

Orna Raz
January 4, 2002

1 Motivation

Much of the software we use for everyday purposes incorporates elements developed and maintained by someone other than the developer. These elements include not only code and databases but also dynamic data feeds from online data sources. Although everyday software is not mission critical, it must be dependable enough for practical use. This is limited by the dependability of the incorporated elements.

It is particularly difficult to evaluate the dependability of dynamic data feeds, because they may be changed by their proprietors as they are used. Further, the specifications of these data feeds are often even sketchier than the specifications of software components.

We suggest a method of inferring invariants about the normal behavior of dynamic data feeds. We use these invariants as proxies for specifications to perform ongoing detection of anomalies in the data feed. We concentrate on semantic anomaly detection: identifying occasions when a dynamic data feed is delivering unreasonable values, even though its behavior may be superficially acceptable (i.e., it is delivering parsable results in a timely fashion). Initial results demonstrating the feasibility of our approach are presented in [12].

2 Thesis statement

We can infer characteristics of the normal behavior of a dynamic data feed and use these characteristics as proxies for missing specifications, to augment any existing specifications.

These augmented specifications suffice for practical semantic anomaly detection. Their inference can be done automatically, to a large extent.

3 Related work

Our work incorporates ideas from various areas, including software analysis, dependability, machine learning, and data mining. We concentrate on software for everyday usage (i.e., non-mission critical) and recognize incomplete specifications are a given. We briefly present related work by the part of our approach it is relevant to.

Motivation and setting  Berners-Lee et al. suggest the Semantic Web [3, 5] — a grand vision for a comprehensive solution to syntax/form and semantic failures, which requires many additions to the current Web. Even if the infrastructure is in place, someone will still need to supply the semantics and make sure they match the actual behavior. Our approach deals with the situation as it is today. It assumes the content supplier may not even be aware of its consumers.

Companies (IBM, Sun, Microsoft, Oracle) are providing various support for building Web Services, thus promoting a new development paradigm: building applications using as elements services available on the Web. The
emphasis is currently on discovery of services, automatic information exchange and integration, through standard-ization of protocols (XML-based) to support this. This is mainly related to connectivity and syntax/form failures, addressing the interface problem. Our work is concerned with semantic failures.

**Solutions to failures in online data sources** Various Tools addressing specific connectivity or syntax/form related failures exist. However, no general solutions that do not require domain specific knowledge exist for addressing semantic failures.

Our work can be viewed as concerned with data quality. However, most of the research related to data quality is concerned with measuring and increasing data quality by the producer of the data. Our emphasis is on measuring and increasing the quality of the data by the consumer of this data. Both approaches are necessary for improving data quality and are complementary.

**Approach: use behavior to characterize properties** Our approach of inferring the characteristics of a data feed from its behavior is similar to work in the areas of program analysis [7, 6, 4] and intrusion detection [10]. However, program analysis work naturally has a different domain, and often concentrates on a specific technique. The major difference between our work and intrusion detection is the fault model. Our model is semantic, unintentional faults, whereas intrusion detection assumes malicious faults. In addition, intrusion detection research often concentrates on specific techniques.

Our long term vision is to increase the dependability of software systems through self healing [14]. This is a similar vision to [9, 1]. [9] concentrate on comparing sequences of events as the basic detection method, and have demonstrated it for the security domain (intrusion detection). An existing IBM automatic computing [1] project is eLiza [2], which deals with self-managing servers.

## 4 Approach

Our long-term goal is to improve the dependability of everyday software systems. We are especially interested in systems that rely on elements which remain under the control of the elements’ proprietors. An example of such elements, which are also in particular need of increased dependability, is dynamic data feeds from online data sources. Examples of data sources include stock quotes, weather forecasts, airline ticket prices, and news reports. Examples of data feeds from these data sources include stock quotes for a specific company, the weather forecast for a specific city, airline ticket prices for specific origin and destination, and the front page news.

On the one hand, timely availability greatly increases the usefulness of dynamic data feeds and hence their value. On the other hand, it is hard to detect anomalies in such data feeds because of their incomplete specifications. Although anomalies in a data feed indicate anomalies in the data source providing this data feed, in our setting the system developer does not have control over the data source, so dependability enhancement can only be done by the client of the data. Our approach is a “black-box” approach: we assume no internal knowledge about the data source (e.g., no source code is available). For the client, the data source is represented by the data feeds the client uses. We, therefore, concentrate on data feeds (Figure 1). A data feed captures a particular usage of a data source, through a view. We further restrict our domain to data feeds that supply well structured content that consists of numeric and categorical attributes.

We are interested in semantic failures. Such failures are hard to detect without good (relevant and correct) specifications of the underlying semantics, since a system using the faulty data feed might superficially appear to behave well, yet the service this system delivers should not be relied upon. This is the availability facet of dependability [11], under a semantic fault model: the readiness for usage of a data feed that delivers parsable results, as indicated by whether or not it delivers reasonable results. To measure and assess this availability, the delivery

---

1Free text raises additional challenges, which are beyond the scope of this work
of semantically correct service needs to be estimated, with respect to the alternation of semantically correct and semantically incorrect service.

Detecting semantic failures would enable us to estimate the semantic correctness of a service, because we would be able to distinguish between correct and incorrect service. We concentrate on anomaly detection since the incomplete specifications of data feeds do not enable us to distinguish a failure from other anomalous behavior that does not constitute a failure. Fault tolerance approaches to detection often use a state-space method [15]. This requires specifications of normal, degraded, and abnormal states, as well as transition probabilities between states. Masking, which does not require detection, requires specifications of outputs and their selection.

However, a particular problem in the domain of dynamic data feeds is that their specifications are, if they even exist, sketchy and incomplete. Such specifications are not good enough to model the semantics of the data feeds and, therefore, not good enough to support anomaly detection that relies on specifications. In [13] we noted state-space models are difficult to work with when the specifications are inaccurate and suggested an alternative incremental-improvement, gradient view.

Our approach to anomaly detection follows the gradient view, overcoming the limitation of requiring precise definitions of states and transitions. This is possible since anomalies can be viewed not only as behavior that violates the specifications but also as behavior that does not happen often. We, therefore, suggest to infer characteristics of the data feed, using and adapting existing statistical and machine learning techniques for the inference. Notice we assume the data feed usually behaves normally. The inferred characteristics, which we use as proxies for missing specifications, can augment existing specifications and domain knowledge. These characteristics are captured in the form of invariants. We concentrate on boolean and statistical invariants. Anomalies are detected when an invariant does not hold when it is evaluated over fresh observations (observations that were not used for the inference of the invariant). For cost-effectiveness reasons, we want to provide as much automation support as possible for our approach (in the form of semi-automated (e.g., heuristics) and fully automated tools).

The main research challenge is to infer invariants that are strong enough to separate normal behavior from abnormal behavior. This entails additional research challenges, related to using and adapting existing techniques for invariant inference: model selection for a specific technique and data feed (tuning parameters of a techniques and preprocessing the data), choosing the size and period of time of the training data, and choosing a collection of techniques that is good-enough for inferring invariants for cost-effective semantic anomaly detection.

5 Expected contribution

We expect to

- provide a method and tools for inferring characteristics of normal behavior of a dynamic data feed that can be used as proxies for missing specifications
- demonstrate the usefulness of these characteristics for semantic anomaly detection
An additional possible contribution would be to define a model for characterizing techniques and data feeds in a way that is useful for capturing their semantics.

Possible usages of invariants that characterize usual behavior We use the inferred invariants for semantic anomaly detection, as it is a first step towards increasing the dependability of dynamic data feeds. However, the inferred invariants may have additional usages, such as: assessing independence of data feeds, helping to deduce likely specifications, and detecting mismatches between specifications and actual behavior. Moreover, our approach for inferring invariants may be useful not only at the consumer’s side, but also at the producers’ side. For example, historical behavior within a context may be used to automatically establish sanity checks for entering data into a database.

6 Validation

To validate our approach we need to address three major criteria: breadth, strength, and cost-effectiveness.

Breadth Our experimental plan aims to make sure this research explores an interesting and wide-enough subset of dynamic data feeds. This subset includes examples of the major classes of data feeds that are especially vulnerable to semantic failures (to low data quality): data feeds that report sensor data and data feeds that rely on humans to enter the data. For example, truck weights is sensor data, book prices are entered by a human, and stock quotes resemble sensor data.

Cost-effectiveness Evaluating the cost-effectiveness of our approach is a difficult task, since different users will incur different benefits.

Formal accounting theories exist for a quantitative estimation of the value of information (e.g., [8]). The value of information is related to the outcomes of user actions that are based on this information. However, a lot of information, some of which is difficult to estimate, needs to be provided. Furthermore, many simplifying assumptions are often required. In addition, our approach is more general than a specific data feed and usage scenario.

We, therefore, concentrate on a qualitative evaluation. Since the computation cost appears to be reasonable, and it can be borne by a third party, we do not take it into account. We expect the main cost factor to be human attention. This includes the amount of human intervention and guidance that is needed, their nature, and the amount of false positives.

For each step in our approach, we will make sure the frequency of using this step matches the level of human attention required in this step. Steps that are infrequent may require intensive human intervention, whereas more frequent steps should be less demanding of human attention. For example, adapting and adding an existing technique to the invariant inference tool-kit is done once. Therefore, is will be satisfactory if we provide a procedure (that may be human attention intensive) for doing so. Inferring invariants is done once every training widow size, therefore, it should be made routine. Detecting anomalies over newly observed data is done very frequently (as frequently as every observation), therefore, it should be automated.

Strength Anomaly detection is not only a first step towards increasing dependability but also the means we use to validate the strength of the inferred invariants: we measure how effective these invariants are for semantic anomaly detection. We use classification accuracy to quantify the effectiveness for anomaly detection.

For each invariant inference technique we want to validate that the model we selected is effective. We do this by measuring the classification accuracy of the technique, where the base line of all anomalies to detect consists of only these anomalies that could be detected using the kind of invariants the technique is able to infer. In general, we do not expect to get perfect detection, as it is often required to tune parameters for a technique and to pre-process the data.
For a collection of invariant inference techniques we want to validate its effectiveness in detecting anomalies over all the anomalies in the data. Producing a base line is a difficult task. The lack of complete specifications is inherent to our domain, and so is the lack of an oracle for what constitutes an anomaly. Therefore, we cannot determine precisely what is the set of all anomalies in the data.

However, we recognize that prefect anomaly detection is not practical in our setting. Rather, we settle for anomaly detection that is comparable to what an “ideal” human would find. “Ideal” means the human is concentrating and is able to pay attention to the data; we are interested in the kind of anomalies humans are good at detecting, not in the error rate of a human. This gives us a reasonable way to establish a base line for the anomalies that exist in the data. We are surveying the human factors literature to establish a set of base line invariants, based on human capabilities. If the techniques we use find additional semantic anomalies we will extend this base line.

References