Abstract—To effectively manage cloud based services, organizations need to continuously monitor the performance metrics listed in the Cloud service contracts. However, these legal documents, like Service Level Agreements (SLA) or privacy policy documents, are currently managed as plain text files meant principally for human consumption. Additionally, providers often define their own performance metrics for their services. These factors hinder the automation of SLA management and require manual effort to monitor the cloud service performance. We have significantly automated the process of extracting, managing and monitoring cloud SLA using natural language processing techniques and Semantic Web technologies. In this paper, we describe our technical approach and the ontology that we have developed to describe, manage, and reason about cloud SLAs. We also describe the prototype system that we have developed to automatically extract information from legal Terms of Service that are available on cloud provider websites.

I. INTRODUCTION

Cloud based services are increasingly being adopted by organizations to fulfill their IT needs because of their promise of cost savings, high availability and device and platform independence. However consumers are still finding it difficult to continuously monitor the performance of their cloud based services, especially if they are using a plethora of services from various providers. The legal contracts between the cloud consumer and providers, like terms of service document, privacy policy, etc., that are used to determine performance measures/metrics are currently created as large text documents. So, the onus is on the cloud consumer to refer to these text documents, that are written in legal jargon, to determine if the cloud service is meeting their needs. At present, it is a very time consuming manual process and consumers often do not have the bandwidth to do so frequently. This makes consumers dependent on the provider provided performance reports to gauge the value the cloud service is providing them. Moreover, cloud contracts contain rules and policies that are not fully encapsulated by existing performance metrics that cloud providers track. Given lack of standard definitions for cloud metrics, different providers could have different definitions for the same measure. For instance, in our discussion with a large federal agency we found that one of their cloud service provider was tracking the service availability measure to only track system availability and not user access. So the provider insisted that their service was available and meeting SLA requirements given that it was up and running even if not every valid end user could access it at the same time. As a result, the federal agency had to update the service contract to clearly define all aspects of ‘service availability’ and the cloud provider had to change their business processes to track both system availability and user access statistics.

Due to the lack of standards for cloud service performance, providers often construct their own rules for the performance measures and metrics and define them as ‘clauses’ in the cloud legal documents like terms of service documents, privacy policy documents or Service Level Agreements (SLAs) that are part of the cloud contact. In addition, the regulatory and compliance bodies have also developed rules and policies that affect the way cloud services can be provided or consumed [1]. Reviewing all these cloud legal documents to ensure the cloud service is meeting the organizational requirements is a labor and time intensive endeavor for consumers and is often an after thought when a cloud service fails to live up to their expectations. Hence, continuous cloud service monitoring has been identified as a key open issue by consumers [2].

Continuous monitoring of cloud SLAs is essential to effectively manage cloud services. A critical step in automating cloud service management is to make the cloud SLAs machine readable so that monitoring tools can interpret the policy rules and metrics defined in the service contracts. We have created an integrated methodology to significantly automate the cloud service lifecycle [3] using Semantic Web technologies. We have also developed a semantically rich ontology to capture key elements of cloud SLAs [4]. Since majority of the cloud legal documents are created as text documents for human consumption, we have applied existing text mining techniques to extract performance rules, measures and metrics. These extracted terms are then reasoned over to automatically populate the instances of our Cloud SLA ontology, described on section III. In this paper we describe the techniques that we have developed to automatically extract SLA definitions and metrics from existing cloud SLA documents.

II. RELATED WORK

Researchers have applied Natural Language Processing (NLP) techniques to extract information from text documents. In Rusu et. al. [5] the authors suggest an approach to extract subject-predicate-object triplets. They generate Parse Trees from English sentences and extract triplets from the parse trees. Etzioni et. al. [6] developed the KNOWITALL system to automate the process of extracting large collections of facts from the Web in an unsupervised, domain-independent, and scalable manner. Etzioni et. al used Pattern Learning to
address this challenge. Another important natural language technique used for information extraction from unstructured text is ‘Noun Phrase Extraction’. Rusu et. al. in [5] show how to create triplets by considering ‘Noun Phrases’ obtained by using various part-of-speech taggers.

In a virtualized service-oriented environment, consumers and providers need to be able to exchange information, queries, and requests with some assurance that they share a common meaning. This is critical not only for the data but also for the policies followed by service consumers or providers. The handling of heterogeneous policies is usually not present in a closed and/or centralized environment, but is an issue in the open cloud. The interoperability requirement is not just for the data itself, but even for describing services, their service level agreements, quality related measures, and their policies for sharing data.

One possible approach to this issue is to employ Semantic Web techniques for modeling and reasoning about services related information. We have used this approach for automating Cloud Service Level Agreements. It enables data to be annotated with machine understandable meta-data, allowing automation. Semantic Web technologies include languages such as RDF [7] and OWL [8] for defining ontologies and describing meta-data using these ontologies as well as tools for reasoning over these descriptions.

III. ONTOLOGY FOR CLOUD SERVICE LEVEL AGREEMENT

As part of our previous work, we had developed a semantically rich ontology to capture key elements of cloud SLAs [4]. We referred to industry best practices, NIST use case 3.9 [1] and actual enterprise data from a large international financial organization to identify the cloud SLA components. Since majority of the cloud legal documents are created as text documents, we applied existing text mining techniques to extract rules and measures from publically available Terms of Service documents of four public cloud providers, as detailed in section IV. The document structure was different for each provider and included terms and definitions that were distinctly defined by the cloud providers. To capture these provider-defined terms and metrics, we had to add additional classes to our existing cloudSLA ontology.

Figure 1 illustrates our updated cloud SLA ontology. The main Service Level Agreement class consists of attributes that are common across all cloud applications. These include service fields like service delivery mode, service availability; and security related attributes like cloud location and data deletion. The class also includes additional properties, like hasEncryption, hasServiceCost, hasAvailability, hasEncryption, hasSecurity, hasSupportMetrics and hasDomainMetrics. In order to capture provider-specific clauses and metrics, we have defined a property hasProviderTerms whose domain is the Provider Terms class. In addition to the terms and definitions properties, this class also consists of hasPenalty and hasRebate properties.

IV. AUTOMATIC EXTRACTION OF SLA DEFINITIONS & MEASURES

Fig. 2. Architecture Diagram

We have developed a prototype system to automatically extract key SLA definitions and measures from the legal ‘Terms of Service’ documents. Figure 2 illustrates the architecture diagram for this prototype. We begin by retrieving publically available SLAs or customer agreement documents that are posted by cloud providers on their website. Next, we pass these documents to our Extractor module which uses Pattern based rules, Stanford PoS Tagger [9] and CMU Link Parser [10]; to automatically extract key term definitions and metrics from the document. The output generated from the Extractor module is passed to our Assessor module which evaluates it using our cloud SLA ontology discussed previously. Once we have identified the SLA definitions and terms, we save it as a RDF graph which is machine understandable and hence can

http://www.nist.gov/itl/cloud/3_9.cfm
be used to automate the monitoring of SLA compliance of the service. We used Semantic Web technologies like RDF and SPARQL to develop this prototype. We use Amazon EC2’s SLA as a running example in this section.

We have tested our system on SLAs from Google Apps, Microsoft Azure, Hewlett Packard compute cloud, and Amazon EC2. However, we believe that as our prototype is based on English rules and patterns, it can be applied to various other SLAs and legal documents.

**A. Extractor**

The first module in our system is ‘Extractor’ based on domain-specific English patterns and rules. Extractor is further divided into a rule-based Term Definition Extractor and a PoS Tagger based SLA Measure Extractor. We will next discuss both submodules in detail.

1) **Term Definition Extractor:** While drafting legal documents it is a common practice to define certain terms which are used multiple times. The main purpose of a definition in a legal document is to achieve clarity without needless repetition. Some generally accepted rules for stating legal terms and definitions are that they should not be defined if they are used only once, or they conflict with accepted usage. Also, they should be placed where they are most easily found, preferably ‘quoted’ before they are used.

SLAs contain definitions provided by the cloud service providers which are closely related to their operations. Our Term Definition Extractor aims to automatically find all such defined terms so as to make it easier for the decision-makers to understand them.

In order to extract definition terms from SLAs we use Pattern Learning. It involves learning a few extraction patterns which are then used to filter out term definitions from unstructured SLA documents.

We divide SLA text documents into different sentences and then pass each sentence to the CMU Link Parser. The link parser generates a parse tree which is then compared to patterns listed in Table I.

![Pattern based rules for our Term Definition Extractor](http://www.smart-words.org/linking-words/)

<table>
<thead>
<tr>
<th>Patterns</th>
<th>Keywords</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>X is defined Y</td>
<td>‘is defined’</td>
<td>X is a quoted, bold, underlined or italicised text.</td>
</tr>
<tr>
<td>X means Y</td>
<td>‘means’</td>
<td></td>
</tr>
<tr>
<td>X is calculated Y</td>
<td>‘is calculated’</td>
<td></td>
</tr>
<tr>
<td>X is Y</td>
<td>‘is’</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE I**

Each pattern consists of X a ‘Noun Phrase’, Keywords, and Y a ‘Verb Phrase’. Keywords are like ‘is defined’, ‘means’, ‘is calculated’, ‘is’. The verb phrase, Y can itself be a complex sentence. To decrease our search space we add a constraint as per the rules listed in [16].

As a final part of the rule based approach, in order to extract other terms that we may have missed, we add a regular expression: ‘(’.*?’)’. This is done so as to handle text for example in Amazon’s SLA: “...policy governing the use of Amazon Elastic Compute Cloud (“Amazon EC2”) and Amazon Elastic Block Store (“Amazon EBS”)...” We need this rule to extract terms like Amazon EC2 and Amazon EBS where the definition precedes the term. Here the definition is present between a set of linking words and the term. For example, in the sentence “... and Amazon Elastic Block Store (“Amazon EBS”)...” the definition for the term ‘Amazon EBS’ is between the term and a connecting word ‘and’.

If a sentence matches a pattern our Term Definition Extractor breaks the sentence ‘S’ into X, a keyword, and Y. We can then add S to our RDF store as the relation definition(X,Y).

![Iteration on one sentence](http://www.smart-words.org/linking-words/)

2) **SLA Measures Extractor:** The next part of our Extractor is the SLA Measures Extractor which we use to populate our SLA Cloud Ontology. For further details regarding ontology please refer to Section II. We employ the technique of ‘Noun Phrase Extraction’ to retrieve relevant information from SLA documents.

In our system we iterate over all sentences and each sentence is passed to the Stanford POS Tagger and CMU Link Parser to generate a Parse Tree. Once we have generated the Parse Tree we create triples (Subject-Predicate-Object) where important keywords from the ontology are found.

In order to create triples we use the technique of ‘Noun Phrase Extraction’ [5], [6]. In this technique we look...
at the ‘Noun Phrase’ part of the sentence and match them to keywords in our Cloud SLA Ontology in order to generate triplets.

**B. Assessor (Evaluator)**

In our system, after extracting information from the SLA text we need a method of evaluating the ‘quality’ of the Knowledge Base created. Currently, we are using an Ontology based value type assessor. For example, we believe that a cloud provider’s uptime will have a numerical data type. Hence, we discard all non-numeric information extracted for the field uptime. These data types are listed in the SLA ontology (discussed in section II). Work on evaluation techniques is in progress.

**C. Populating the Knowledge Base**

The main goal of our system is to create a Knowledge Base of all facts present in various SLA documents. We envision a Knowledge Base with SLA term definitions and measures for all cloud service providers operating in the domain. The end user will have access to our Knowledge Base and will be able to query it using a SPARQL interface which is quite similar to SQL. To give an example we show snippets of RDF statements generated for Amazon EC2:

```rdfs
:uptime a owl:DatatypeProperty;
rdfs:domain :Service_Availability;
rdfs:range [ a owl:DataRange; owl:oneOf (‘at least 99.95%’) ].
```

**V. CONCLUSION AND FUTURE WORK**

Currently cloud related legal documents, like terms of service or customer agreement documents are managed as text files. As a result extensive manual effort is required to monitor the metrics and measures agreed upon in these SLAs. We have worked on significantly automating this process using semantic web technologies like RDF. In this paper we have described our system that we have developed to illustrate how the SLA term definitions and measures can be automatically extracted from legal Terms of Service or customer agreement documents that are available in the public domain.

In the future we would like to add functionality which will allow users to compare and contrast SLAs of different service providers. SLA and other legal documents include obligations, permissions, claims, etc. we aim to add these to our system. We would also like to extend our system to include other legal documents and agreements.

**REFERENCES**


