Mining Contracts for Business Events and Temporal Constraints in Service Engagements

Xibin Gao and Munindar P. Singh, *Fellow, IEEE*

Abstract—Contracts are legally binding descriptions of business service engagements. In particular, we consider business events as elements of a service engagement. Business events such as purchase, delivery, bill payment, bank interest accrual not only correspond to essential processes but are also inherently temporally constrained. Identifying and understanding the events and their temporal relationships can help a business partner determine what to deliver and what to expect from others as it participates in the service engagement specified by a contract. However, contracts are expressed in unstructured text and their insights are buried therein.

Our contributions are threefold. We develop a novel approach employing a hybrid of surface patterns, parsing, and classification to extract (1) business events and (2) their temporal constraints from contract text. We use topic modeling to (3) automatically organize the event terms into clusters. An evaluation on a real-life contract dataset demonstrates the viability and promise of our hybrid approach, yielding an F-measure of 0.89 in event extraction and 0.90 in temporal constraints extraction. The topic model yields event term clusters with an average match of 85% between two independent human annotations and an expert-assigned set of class labels for the clusters.

Index Terms—Service engagements, Contract mining, Business events

1 Introduction

Modern business service engagements are becoming increasingly more numerous and more complex. We consider service engagements in the broad sense. Thus we include not just traditional examples of service engagements, such as customer relationship management or business process outsourcing, but also other business interactions, such as manufacturing and software licensing.

Because service engagements are specified via business contracts, the expansion of the importance of service engagements in modern business is seen in the increasing number of contracts. For example, InfoSys reports[1] that 60% to 80% of business transactions are governed by contracts and that an average Fortune 2000 company manages 20,000 to 40,000 active contracts at any given time.

The above business trend exposes some new broad challenges in service computing. The first challenge is how, during enactment, a contractual party can understand a contract so as to determine its actions (and design its IT systems) to support its participation in the service engagement. Specifically, would it be able to guide the development of its business processes and monitor its interactions? That is, would the party be able to deliver its part of a service engagement and determine what to expect from its partners in that service engagement? The second challenge is how, during negotiating a service engagement, a party can examine and draft contracts in a manner that incorporates the general practices of the relevant domain.

The problem of specifying, adopting, and enacting a service engagement is exacerbated by the fact that contracts are expressed in natural language. Further, often the people who negotiate and those who implement a contract have different skill sets. Accordingly, we are pursuing a research program that seeks to break the problem down into chunks that are amenable to computational analysis. In previous work [1], we tackled a part of the second of the above challenges by mining a repository of contracts to determine the possible business exceptions identified in different domains.

In this paper, we develop an approach that addresses both of the above challenges. This approach is based on the idea of *business events*—including business-related actions and activities such as purchase, delivery, bill payment, bank interest accrual, licensing, and dispute resolution. Business events indicate the essential processes involved in a service engagement as well as the risks and exceptions to consider. Moreover, the events are naturally temporally constrained, indicating the conditions on their occurrence. The violation of a temporal constraint is often an important factor in contractual breach and the resulting complications.

For these reasons, identifying and understanding business events and their temporal relationships in a service engagement can help a business partner in successfully enacting a contract: that is, determining both what to deliver (to others) and what to expect (from others). Understanding business events and their temporal relationships can also potentially help it decide whether to enter into an engagement in the first place. Note that real-life service engagements

• *X. Gao is with Microsoft, One Microsoft Way, Redmond, WA 98052-7329 and M. P. Singh is with the Department of Computer Science, North Carolina State University, Raleigh, NC 27695.*
E-mail: xibin.gao@gmail.com, m.singh@ieee.org

[1] http://www.infosys.com/consulting/knowledge-services/white-papers/Documents/contract-process-helping-hurting.pdf

are complex interactions with many nuances: we do not claim to have addressed all of the nuances just by identifying events and temporal constraints from contracts, though what we do identify provide a necessary underpinning for more elaborate future analyses.

Contributions

The broad problem tackled in this paper is to elicit requirements for service engagements. Since contracts are widely available in today's business practice and provide a ready basis for service requirements, it behooves us to try to mine contracts to determine such requirements. Despite the importance of extracting events and temporal constraints for service engagements, previous approaches have not tackled this task. This paper, first, formulates the problem of business events and temporal constraint extraction from contract text. Second, it shows how to solve the above problem by breaking it down into three subtasks. Further, it evaluates the methods applied to solve the three subtasks over a human-annotated gold standard dataset and obtains good results.

Organization

The rest of the paper is organized as follows. Section 2 formalizes business events and temporal constraints extraction problem and divides the problem into three subtasks. Sections 3, 4, and 5 describe our method and evaluation for each subtask. Section 6 surveys the relevant literature. Section 7 concludes with a discussion of remaining challenges and future work.

2 Problem and Approach Overview

Business events in contractual service engagements are distinct from other domains. First, their connotations are different. A contract usually is drafted and signed before the relevant business service engagement occurs; that is, a contract refers to future behaviors. In contrast, the events in other domains usually are descriptive of natural or social phenomena or scientific facts. Second, their scopes are different. Events from domains such as news [2] and biology typically focus on one narrow area and thus a tailored method may work well for each such specific task, whereas business events encompass many areas due to the diverse realms that service engagements deal with, e.g., manufacturing, licensing, supply, and employment.

Temporal information, which usually qualifies or provides details about events, may be expressed in various ways. Temporal relationships between events are indicated by either an explicit mention of date, time, frequency, or an implicit logical ordering of events. Researchers have annotated or extracted temporal information from different applications such as anchoring events, question answering, and timeline organization. However, due to the business nature of service engagements, temporal constraints typically have financial and legal ramifications. As a result, temporal constraints that qualify business events in service engagements are often explicit.

Below are some sample sentences from the Yahoo! Small Business Terms of Service.[2]

> All installation or setup fees and non-recurring charges, along with the first month's recurring charges, shall **be due and payable** <u>within ten (10) days of initiation of Service</u>.
>
> If You **cancel the Service** <u>before the end of the Initial or Renewal Term</u>, Your Service and access to the Service will be discontinued immediately, and no refund will be provided for any payments You have made.
>
> You agree that Yahoo! may **delete customer credit card information** from Yahoo! servers <u>14 days after You retrieve such information</u>, and may **delete all other Merchant Information** from Yahoo! servers <u>at the end of each calendar year</u>.

The bold text fragments—"be due and payable," "cancel the Service," "delete customer credit card information," and "delete all other Merchant Information"—express business events and are significant to the contracted service engagement. Such events are associated with the commitments, permissions, and prohibitions [3] of the contracting parties. The underlined text fragments—"within ten (10) days of initiation of Service," "before the end of the Initial or Renewal Term," "14 days after You retrieve such information," and "at the end of each calendar year"—place temporal constraints on the corresponding business events. The events may expire or become invalid when their temporal constraints do not hold. For instance, in the first example sentence above, the charges shall be due and payable within ten days of the initiation of Service; paying after ten days of the initiation of Service may breach the contract and potentially incur financial liability or lead to the cancellation of the service.

In poorly formulated contracts, business events such as payment and service delivery that bear implicit time requirements may lack temporal constraints. The resulting service engagement may fail. For example, disputes could occur when contracting parties default or fail to deliver services in a timely manner. Our tool, *Contract Miner*, captures the essential elements of a contract and thus provides a basis for future work on commitment-based contract analysis [3]. We now define business events and temporal constraints in the setting of text mining contracts for service engagements.

[2]http://smallbusiness.yahoo.com/tos/tos

Definition 1: Business event: an occurrence of significance to a service engagement, especially as indicated by subsentence-level text and often expressed with a subject and a corresponding verb phrase.

Definition 2: Temporal constraint: a constraint on the occurrence and ordering of business events, especially as indicated by a prepositional phrase.

Formally, our task is: given a corpus of contract text C, extract the business events E along with their subject and any associated temporal constraint T. Through this process, pairs (E, T) are extracted where T is optional.

Section 2.1 reviews information extraction methods to explain their inadequacy for our task. Section 2.2 introduces our approach and system flow.

2.1 Overview of Information Extraction Methods

Service engagements encompass diverse domains of knowledge ranging from manufacturing to employment to trade and their contracts exhibit similar diversity. Thus they pose special challenges to event extraction.

Event extraction methods rely heavily on patterns. Such methods typically work well in a specific area, for example, natural disaster events [2]. But they suffer from poor portability. For example, extraction patterns for genetic events cannot be applied for extracting financial events. Thus a purely pattern-based approach, which can work in a specific area, is inadequate for contracts. Some approaches use machine learning to fill in event slots as defined in a sentence context [4]. However, business events do not exhibit a well-defined structure so that slot-filling does not apply well.

Traditional temporal information extraction approaches prove inadequate for extracting temporal constraints from service engagement contracts. Unlike in domains such as news, where the challenge is figuring out temporal orderings [5], in service engagement contracts, time is often explicitly mentioned in prepositional phrases (PPs). However, a challenge is to tease apart the temporal constraints from the other kinds of information that PPs can express, such as space or the intention of an actor.

2.2 Overview of our Approach

Figure 1 illustrates the flow of our approach as a hybrid of surface patterns, linguistic parsing, and machine learning techniques. Contract Miner, first, takes raw online contracts as input, removes noise such as HTML tags and segments the contracts into sentence collections. Second, it filters out sentences such as definitions and postal addresses that obviously do not contain business events and temporal constraints. Third, it parses and prunes the remaining sentences to generate candidate events and temporal constraints. Fourth, it applies machine learning on local and contextual features to separately identify true events and temporal constraints from the candidates. Fifth, it applies topic modeling to extract hidden event topics. We divide our approach into three major tasks:

1) Business events extraction: Section 3.
2) Business event topics discovery: Section 4.
3) Temporal constraints extraction: Section 5.

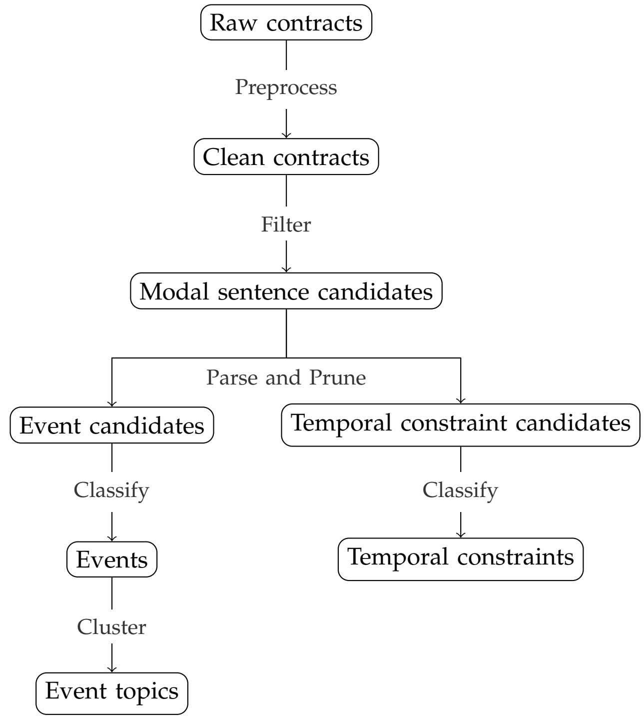


Fig. 1. Overall processing pipeline.

The three tasks are intimately related. Task 1 extracts the backbone of contracts—business events, and the extraction results from Task 1 are the prerequisite for Task 2, which uses the automatically extracted events as the input dataset for discovering event topics. Moreover, Task 3 is closely related to Task 1 because the temporal information constrains business events.

3 Task 1: Business Event Extraction

A typical service engagement contract contains parts such as header, definition, body, and sign off. At the core of a contract are the clauses specifying mutual expectations expressed as *normative relationships* such as commitments, powers, authorizations, prohibitions, and sanctions of the participating parties [3]. Normative relationships express business relationships among the parties to a service engagement and these normative relationships are built on top of business events. In English grammar, these normative expressions are often associated with modal verbs such as “shall,” “may,” and “must” [6]. We use modal verbs as signals to signify the occurrence of business events. Signal words are widely used in information extraction and serve as clues for locating the extraction context.

3.1 Approach

After the initial cleanup, Algorithm 1 selects contract sentences that include the signal words as event candidates, parses each candidate sentence to induce the grammar tree, then prunes the grammar tree, and finally builds a feature vector for each candidate using the features extracted from the grammar tree.

Algorithm 1 Business events extraction.

Require: Contract corpus C
for all contract c in C **do**
  for all sentence s in c that contains a signal word **do**
    Parse sentence s to induce grammar tree t
    Prune tree t to obtain event candidate e
    Build feature vector f for the event candidate e
  end for
end for
build classification model with the training data composed of entries in the form of $(e, f, \mathsf{Boolean})$

TABLE 1
Example signal words.

agree to	warrant	promise	sanction	obligate
prohibit	forbid	permit	authorize	

Using the Stanford Parser [7], we parse each event candidate sentence to produce its grammar tree that associates each token with a part-of-speech tag, and each phrase with a phrase label from the Penn Treebank [8].

Algorithm 2 Grammar tree pruning.

Require: Grammar tree t
Locate signal words in grammar tree t
Obtain the (tree-structured) verb phrase v where a signal word is located
for all children c in v **do**
  if the label of c appears in Table 2 **then**
    Prune c
  end if
end for

Consider this sentence from a supply agreement between two companies Baxter and IDEC[3]:

> CLIENT shall select and pay the freight forwarder who shall solely be CLIENT's agent.

[3]http://contracts.onecle.com/biogen/baxter.supply.2002.06.01.shtml

Algorithm 2 describes the steps to prune the grammar tree to obtain a concise representation of the event candidate. For example, from the above sentence we obtain "CLIENT shall select and pay freight forwarder who shall solely be CLIENT's agent" as the extracted event candidate because the signal word "shall" precedes this verb phrase. Within the event verb phrase, the clause "who shall solely be CLIENT's agent" with the SBAR chunk label qualifies "the freight forwarder." Our pruning algorithm removes the leaves of the grammar tree with the SBAR labels, so the event is abbreviated to "CLIENT shall select and pay the freight forwarder." Table 2 shows the complete list of phrasal chunks that are pruned from the event representation.

TABLE 2
Types of phrasal chunks for pruning [8].

Label	Meaning
SBAR	Clause introduced by a subordinating conjunction
SBARQ	Direct question: wh-word or a wh-phrase
SINV	Inverted declarative sentence
SQ	Inverted yes/no or main clause of wh-question
S	Simple declarative clause
ADVP	Adverb phrase
PP	Prepositional phrase
WHADJP	Wh-adjective phrase
WHNP	Wh-noun phrase
WHAVP	Wh-adverbial phrase
WHPP	Wh-prepositional phrase

We glean appropriate features for the event candidates from the grammar tree. Table 3 summarizes and explains in greater detail the features we use.

TABLE 3
Features for event classification.

Feature name	Example
subject contains named entity	Motorola, Morgan Stanley
signal word	can, could, must
clause signal	if, unless, which
counterclause signal	if, unless, which

Named entities often bear a close association with the presence of business events and serve as the subjects of events. Company and organization names such as "Motorola" and "Samsung," contract-specific referral terms such as "parties," "client," and "buyer" are often the event subjects. The occurrence of such a term increases the chance of a candidate being a true event. We extract the subjects from the event sentence candidates and then detect if such indicative terms appear in the subject. Algorithm 3 shows our method for extracting the subject from an event candidate sentence.

Algorithm 3 additionally decomposes a complex event candidate into multiple event candidates. For example, a complex event candidate from the manufacturing agreement between Minnesota Mining and

Algorithm 3 Subject extraction.

Require: Event candidate sentence grammar tree: t
for all subtree sub in t with a signal word as root **do**
  if the preceding sibling of sub is ps AND ps is NP **then**
    Return ps as the subject of st
  else
    if the preceding uncle of sub is pu and pu is NP **then**
      Return pu as the subject of event candidate st
    end if
  end if
end for

Manufacturing Company and Sepracor Inc. expressed as[4]

> SEPRACOR shall have the right within 45 days to test batches on an audit basis prior to accepting the batch, however SEPRACOR shall have no right to delay payment.

is decomposed into two simpler event candidates:

> SEPRACOR shall have the right within 45 days to test batches on an audit basis prior to accepting the batch.

and

> SEPRACOR shall have no right to delay payment.

After we extract the subjects of events, we apply both dictionary and machine learning methods to detect if named entities are present. Terms such as company or organization names are detected with the Stanford Named Entity Recognizer, which is based on a machine-learned model [9]. Pronouns and words referring to the parties to a service engagement—such as "parties," "client,"' and "buyer"—are equivalent to named entities but they escape detection by Stanford NER. Therefore, we build a dictionary of such words and use it to supplement named-entity recognition to check the existence of subject terms.

The adverbial clauses of a condition bear some association with the occurrence of a business event. Clause subordinating conjunction words such as "if" and "unless" often indicate a business event, as showcased in the asset purchase agreement between Bausch & Lomb Inc. and Pharmos Corp. below[5]:

> If the actual, documented out-of-pocket costs of Buyer related to such clinical testing and filing exceed One Million Two Hundred Thousand Dollars, then Seller shall pay to Buyer fifty percent (50%) of the LE-T RD Costs in excess of such amount

and the supply agreement between Baxter and IDEC Pharmaceuticals below[6]:

> If BAXTER is unable to meet the specified Delivery Date, except when caused by CLIENT's delay in delivery of Bulk Conjugated Antibody or other CLIENT Supplied Components, BAXTER shall so notify CLIENT and provide to CLIENT an alternative Delivery Date which shall not be more than [. . .] later than the initial Delivery Date designated by CLIENT in its Purchase Order.

Conjunctions "unless" and "if" suggest business events because they indicate a conditional dependency, which is prevalent in events; however, subordinating conjunctions such as "which," "that," and "who" signify otherwise, because they often simply exhibit a modifier or auxiliary relation to the main subject. A clause signal is the conjunction of the current context, and a counterclause signal is the conjunction of the subordinating clause. We tap these clause connectives as features in deciding if a statement is a true event.

In summary, the features we use come from two sources: local and contextual. Local features are extracted from the grammar tree of the event statement. Event subject and signal words are local features and depend on the event representation itself. Contextual features such as clause and counterclause signals depend on the leading or subordinating clauses.

Upon generating the features, we use the Weka toolkit's [10] classification packages to identify the true events. After building the feature vectors for all the event candidates and annotating them by hand, we apply various machine learning methods. Previous studies indicate that Support Vector Machine (SVM) and Logistic Regression are effective in similar tasks.

3.2 Evaluation

We use the following well-known evaluation metrics: precision, recall, and F-measure [11]. Below, TP, FP, and FN, respectively, stand for true positive, false positive, and false negative. Precision measures the fraction of extracted instances that are relevant, while recall measures the fraction of relevant instances that are extracted. F-measure is the harmonic mean of precision and recall.

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

[4] http://contracts.onecle.com/sepracor/3m.mfg.2001.12.20.shtml

[5] http://contracts.onecle.com/pharmos/bausch.apa1.2001.12.28.shtml

[6] http://contracts.onecle.com/biogen/baxter.supply.2002.06.01.shtml

Our event extraction approach uses supervised classification. We (arbitrarily) select a set of 300 event candidates from the Onecle contract repository[7] and manually label true business events. We emphasize that this repository contains genuine contracts that were entered into by real-life businesses. For privacy, some details, such as the amounts involved are redacted in this repository and replaced with ⋆ characters—this deviation from the original contracts only makes our task harder because such redactions cause parsing to become harder than it would be in actual contracts. Gao et al. [1] provides some statistics regarding this repository including that the majority of contract sentences are shorter than 80 words.

We automatically generate the features as stated in Table 3 for all candidate events. Then we use SVM and Logistic Regression from Weka for evaluation. With ten-fold cross validation, we obtain results shown in Table 4. Here, each weighted average is calculated according to the proportion of instances in each class. Logistic Regression slightly outperforms SVM and achieves a weighted (according to the number of instances) F-measure of 0.89.

TABLE 4
Event extraction results.

Method	Class	Precision	Recall	F-Measure
SVM	Negative	0.86	0.82	0.84
SVM	Positive	0.87	0.90	0.88
SVM	Weighted average	0.87	0.87	0.87
LR	Negative	0.91	0.85	0.88
LR	Positive	0.89	0.93	0.90
LR	Weighted average	0.90	0.90	0.89

We compare different combinations of the features in terms of their predictiveness of the classes. Figure 2 shows the performance of the Logistic Regression classifier using different combinations of features. Here, CCS+MV+CS refers to the previous three features combined. A combination of all features yields the best predictiveness.

Using automatic event extraction, we build a repository of events from different service engagement domains. Table 5 shows the repository information. A total of 65,031 manufacturing, licensing, and lease events are classified from 229,996 candidates extracted from 1,821 contracts. An average of 38 events per contract highlights the abundance of events in contract text. Table 6 shows a sampling of the events in the repository.

The Contract Miner implementation uses a Perl module for preprocessing and filtering; a Java module for parsing, pruning, and generating features—and most of the processing; and Weka for model training. On a sample of 500 sentences from manufacturing service contracts, it takes Contract Miner 1,514 seconds—an average of three seconds per sentence—to process on a Toshiba Satellite L45-S7409 laptop with a 1.46GHz T2310 Intel CPU, 1.5GB memory, and running Windows 7. Most of the processing time is spent on part-of-speech tagging and dependency relations parsing.

[7] http://contracts.onecle.com

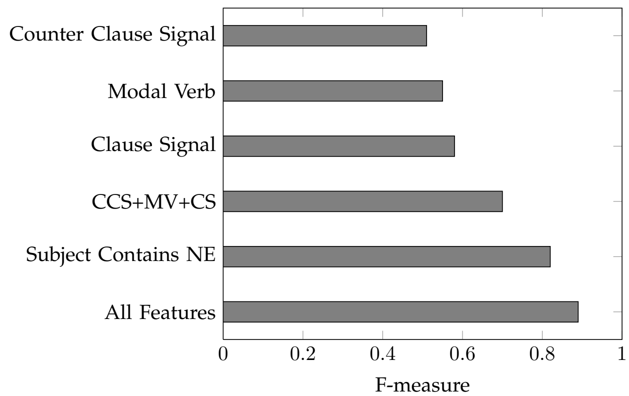


Fig. 2. Comparing the effectiveness of different features.

TABLE 5
Event repository summary.

Domain	Events	Event Candidates	Contracts
Manufacturing	8,833	25,590	208
Licensing	51,831	179,840	1,363
Lease	4,367	24,566	250
Total	*65,031*	*229,996*	*1,821*

4 Task 2: Event Term Clustering

Business events in service engagements naturally fall into categories such as product delivery, payment, and natural hazards. Automatically discovering the event categories can help us better organize events in different service engagement domains. Further, it would help complete the full knowledge discovery cycle by beginning from raw text and ending with automatically discovered event categories.

Classification and clustering are widely applied to categorize text. Classification methods [12] are supervised, so a training dataset needs to be built manually beforehand that predefines the categories. However,

TABLE 6
Sample business events.

Sample Business Events
Each party shall be licensed under those rights of the other party
3M shall notify SEPRACOR without undue delay
3M will perform release testing of all batches to agreed upon specifications
SEPRACOR shall reimburse 3M the reasonable cost of travel
3M may increase the supply price for licensed product
3M shall have the exclusive right and license to supply SEPRACOR
The customer shall have the right to withhold the monthly investment grant specified in I

business events found in contracts cut across numerous service engagement domains, with potentially different categories across domains. For example, in licensing contracts, the event categories may be of patent infringement, financial payment, and product licensing. And, in leasing contracts, the event categories may be of property management, rent payment, and eviction.

We seek a method that can apply to the services domain where the categories may have not been seen, so classification would not be applicable here. Clustering methods [13] do not need predefined classes and are unsupervised.

4.1 Approach

We adopt topic modeling, a method to discover abstract topics from document collections. In contrast with clustering, topic models can extract the hidden topics of the events, and identify the vocabularies describing these topics. Topic models serve our purposes of automatically discovering the event categories and extracting the representative words for different events. Latent Dirichlet Allocation (LDA) [14] is a popular method for topic modeling that assumes both the order of words in the document and the order of documents in the corpus are irrelevant and that the number of topics is fixed and known.

For our purpose of categorizing events by discovering event topics (or themes) and their corresponding descriptive vocabularies, we apply topic modeling in event categorization. In abstract terms, each event is regarded as a *document*; each *document* is a distribution of event topics; and each event topic is a distribution of event terms. Specifically, using the R implementation[8] of LDA, we extract prominent business event topics and representative vocabularies for each topic.

4.2 Evaluation

We evaluate LDA as applied in extracting business event clusters in two ways: *centrality* and *clarity*. First, we evaluate the ability of LDA to discover terms that are centered on a meaningful business event topic. We do so beginning with a human annotator assigning meaningful class labels to the automatically discovered terms groups. If the annotator is able to come up a descriptive label that covers the theme of a group of terms, it shows good centrality of the cluster. Second, we evaluate the separation of the terms clusters. We do so by using two independent human annotators matching a given list of class labels assigned by one of the authors to the term clusters. Terms of different themes should fall under different clusters.

In our study, we apply LDA to automatically extract groups of terms describing themes of events. Here we treat the 8,833 business events from 208 manufacturing contracts from Table 5 as documents and apply LDA to discover topics and the terms describing each topic. We set the number of topics to ten. Table 7 shows the top representative terms for each of the ten topics.

In the centrality evaluation, a human annotator reads the terms for each topic and manually assigns a class label that best describes the theme of the term collection. For example, terms such as "write," "notify," and "notice" are assigned the *communication* class label; terms such as "cost," "pay," and "purchase" are assigned the *payment* class label. Table 7 shows that manually assigned class labels as concepts cover the central theme of a term cluster, demonstrating the effectiveness of LDA in discovering event classes from contract text. Note that the top topics also reflect the subdomain of manufacturing. In manufacturing service engagements, the business events are often related with financial payment, manufacturing process, and product orders, and they are evident in the top topics.

In the clarity evaluation, we asked two independent human annotators to assign the predefined labels as shown in Table 9 to term clusters in three domains: manufacturing, licensing, and leasing. The Kappa coefficients for the two human annotators and the provided annotations are 87% and 83%, with an average of 85% over all domains. The high agreement demonstrates the extracted topics are clearly separated, demonstrating the effectiveness of our approach LDA in distinguishing meaningful event term clusters.

TABLE 8
Topic evaluation.

	Manufacturing	Licensing	Lease	Overall
Annotator 1	100%	80%	80%	87%
Annotator 2	100%	70%	80%	83%
Average	100%	75%	80%	**85%**

As one can see from the class labels arising in the various domains, some class labels are common across all service engagement domains. For example, contracts generally include events relating to *communication*, such as "notify" and "write." Other class labels are domain specific. For example, manufacturing contracts refer to events related to *service provisioning* and *quality control and testing*. Likewise, licensing agreements refer to "patent infringement" and "software licensing." And, leasing agreements refer to "property management" and "repair and maintenance."

5 Task 3: Temporal Constraints Extraction

Service contracts involve temporal information of various forms (Table 10). The temporal expression format

[8] http://cran.r-project.org/web/packages/lda/index.html

TABLE 7
Event topics from 208 manufacturing contracts, automatically extracted with LDA.

Class Labels (manually assigned)	Top Vocabularies (automatically extracted)
Product processes and facility	provid product document record tercica cbsb maintain insur batch includ relat complet execut data properti support substanc manufactur liabil approv
Cost, expense, and liability	cost expens indemnifi liabil claim damag oblig incur reimburs respons pai insur loss defens connect liabl fee respect limit
Service provisioning	provid servic perform provis waiver oblig deem hereund insur respect right joint breach compani set review ventur certif requir default
Quality control and testing	agre applic mutual time perform test process respons requir accept write product procedur law manufactur report approv regulatori compli qualiti
Product order and delivery	product order purchas materi servic forecast packag deliveri provid requir hitachi quantiti accord deliv novost date suppli time ship nanogen
Communication	notic termin dai written notifi effect date write receipt provid period chang give advanc event year forc expir busi
Confidentiality and records	inform provid confidenti tercica cbsb gen document pd manufactur drug batch substanc record copi accord data specif approv ferment
Materials supply	reason effort commerci request materi time make chang manufactur execut raw servic act order meet requir resolv disput provid suppli
Asset and property transfer	assign product project right transfer provid written consent properti manag sole determin equip brew sequenom abi own affili interest technolog
Payment	pai amount cost credit invoic part iae price purchas payment engin account year rate date air appli oper charg base

TABLE 9
Cluster labels assigned in different domains indicating common and domain-specific labels.

Manufacturing	Licensing	Leasing
Communication	Communication	Communication
Confidentiality	Confidentiality	Contractual matters
Expense and liability	Cost and liability	Liability
Asset and property transfer	Right and transfer	Lease termination
Payment	Royalty and payment	Payment
Product order and delivery	Patent infringement	Expense
Material supply	Software licensing	Obligation
Product process and facility	Research and development	Property management
Quality control and testing	Time	Property facility
Service provisioning	Law	Repair and maintenance

also varies. Some temporal information is expressed explicitly as dates, for example, "Feb. 3th, 2010" and "10-01-1949."

TABLE 10
Varieties of temporal information in service contracts.

Classification	Example
Time point	on Friday
Frequency	at the beginning of every month
Constraint	before the next payment date

In service engagements, the most relevant temporal information pertains to the constraints that the participants need to observe. For example, a business workflow usually follows a temporal order, and the successful fulfillment of a service engagement greatly depends on the timely completion of those business processes. Such temporal relations among the business events are usually expressed explicitly for the purpose of clarity and emphasis. Temporal constraints in contracts are mostly expressed in *prepositional phrases* (PP).

Definition 3: A prepositional phrase comprises a preposition and noun phrases or clauses.

Prepositional phrases function as adverbs in a sentence, and express "where," "how," and "when." Some prepositions indicate temporal boundaries for the completion of a task. For example, "before," "after," "within," "during," "upon," "at," "until," and "between" generally convey the temporal constraints on business events.

In our approach, as illustrated in Algorithm 4, we apply similar early steps as in event extraction: clean up the contract text, filter with signal words, and parse the sentences using linguistic tools. We extract the prepositional phrases labeled as "PP" by the Stanford Parser [7]. Because a PP may express a wide range of meanings such as "when," "where," "how," and "why," we treat prepositional phrases as temporal constraint candidates, and employ a classification model to decide if each candidate is a true temporal constraint.

Prepositional phrases serve multiple functions in a sentence. For example, prepositional phrases below followed by "at" may indicate "when", "where," or "how" and only the first expresses a temporal constraint.

Algorithm 4 Temporal constraints extraction.

Require: Contract corpus C
for all contract c_i in C **do**
  for all sentence s in c_i that contains signal word **do**
    Parse sentence s to induce grammar tree
    Extract the PPs from the grammar tree as temporal constraint candidates
    Build a feature vector for each temporal constraint candidate
  end for
end for
Build a classification model with the training data composed of entries in the form of (PP, Boolean)

> The registration is for the full period of years selected and paid for at the time of Application or renewal.[9]
>
> Furthermore, you acknowledge and agree that this Style Guide is subject to modification by the Registry with any such changes appearing at the previously designated URL.[10]
>
> Yahoo! reserves the right, at its own expense, to assume the exclusive defense and control of any matter otherwise subject to indemnification by You, but doing so shall not excuse Your indemnity obligations.[11]

5.1 Approach

We formulate the problem as a text classification task: given a prepositional phrase p, we assign either class label t (temporal constraint) or n (not a temporal constraint) to p. The above problem faces unusual challenges. Traditional text classification tasks generally consider passages from news articles and technical papers that are long enough to build a useful feature vector. Our task is classifying short phrases not exceeding twenty words in most cases. The temporal property of prepositional phrases has been studied in extracting temporal information [15]. However, the ambiguity of prepositional phrases has not been explored. We disambiguate a whether prepositional phrase signifies a temporal or another kind of property.

In our task, we apply well-known classification techniques—KNN, Naïve Bayes, and Logistic Regression—to classify the PPs into two classes: temporal and not temporal. In summary, the temporal constraint extraction task is decomposed into two stages: finding PPs and classifying PPs. Linguistic parsing using the Stanford Parser produces PPs and the classification methods detect the temporal PPs.

[9]http://smallbusiness.yahoo.com/tos/domain-reg-agreement
[10]http://smallbusiness.yahoo.com/tos/domain-reg-agreement
[11]http://smallbusiness.yahoo.com/tos/smallbiz-tos

TABLE 11
Training set sample instances.

Temporal Constraint
on the effective date set forth above
on the first to occur of the date
within thirty (30) days of receipt of notification
not less than two (2) working days notice of such disclosure
within 45 days
after signing of this Agreement on a date as agreed to by the parties
during development to be agreed upon by the parties
within [**] days of the Effective Date
after the first [**] hours of work for 2002
within [**] days of the date of 3M's invoice
until the 31st December 2001
since the last increase in the hourly rate
no later than the initiation of Phase III clinical studies
no later than the NDA filing
at the time it incurs such increase
during the term of this Agreement
for up to 24 months after notice of termination
after the date of notice of termination
at the time of shipment
after a period often (10) years from the date of this Agreement or five (5) years from the date of termination of this Agreement
within thirty (30) days of receipt of notification
within twenty (20) days of notification of a dispute
not less than two (2) working days notice of such disclosure

5.2 Evaluation

Since our temporal extraction approach is supervised classification, we manually annotated 1,000 prepositional phrases from manufacturing contracts from the Onecle contract repository—the same one we used above for business events. The annotated prepositional phrases serve as the ground truth. Examples of the positive training set are shown in Table 11. We adopt the bag-of-words model for the features of PPs. For each classification approach, we perform a ten-fold cross validation. We compare the temporal constraints extracted by our system with the ground truth to compute the true and false positives and negatives. With such data, we calculate the precision, recall, and F-measure averaged over ten folds. Using Lingpipe,[12] we build a classification model on the training set and evaluate its performance. We detail each classification method's output below.

KNN

The K-nearest neighbor (KNN) approach labels an instance with the class that is the majority of all its neighbors [10]. Two important factors in KNN are the number of neighbors, k, and the distance function. We adopt the commonly used Euclidean distance to measure the proximity of trained instances. With different neighbor thresholds, we obtain the results shown in Table 12, where $k = 5$ yields the best results.

[12]http://alias-i.com/lingpipe

TABLE 12
Results using KNN.

Neighbors	Precision	Recall	F-Measure
3	0.86	0.86	0.86
5	**0.87**	**0.86**	**0.86**
7	0.84	0.82	0.82
9	0.84	0.81	0.82

Naïve Bayes

As a probabilistic text classification approach, Naïve Bayes assumes that the words in the text are mutually independent [10]. Our experiment involves three settings: no preprocessing, removing stop words only, and removing stop words and stem tokens. The results are shown in Table 13. The first setting produces the best results.

TABLE 13
Results using Naïve Bayes over a unigram model of words.

Setting	Precision	Recall	F-Measure
Unigram (as is)	**0.90**	**0.90**	**0.90**
Remove stopwords	0.88	0.88	0.88
Stem, Remove stopwords	0.87	0.88	0.87

Logistic Regression

Assuming a parametric form for the distribution $P(Y|X)$, Logistic Regression learns a mapping from an input vector to a continuous output [10]. Using Logistic Regression, we obtain four sets of results with different selections of features as shown in Table 14. We also obtain the coefficients associated with each stemmed token. Table 15 shows the two ends of the spectrum of the token's association with temporal properties. As expected, tokens such as "time," "date," "duration", and "day" fall near the highly temporal end, whereas tokens such as "behalf," "purpose," "expense," and "exhibition" fall near the nontemporal end. In temporal expressions, prepositions are often used together with tokens from the temporal end such as "by year xxx" and "before the month of xxx." However, in nontemporal expressions, prepositions are often used in the expression such as "on behalf of," "for the purpose of," and "at the expense of" to convey nontemporal properties.

TABLE 14
Result using Logistic Regression over a unigram model of words.

Setting	Precision	Recall	F-Measure
Unigram (as is)	0.90	0.88	0.89
Stem	**0.90**	**0.89**	**0.89**
Remove stopwords	0.89	0.87	0.87
Stem, Remove stopwords	0.88	0.86	0.86

TABLE 15
Logistic Regression: token and associated coefficients. The coefficient corresponds to the predictiveness of a token to be a nontemporal constraint. The lower the coefficient, the higher the association of a token with a temporal property.

Token	Coefficient	Token	Coefficient
time	−2.77	behalf	1.04
date	−2.49	purpos	1.00
dure	−2.41	expens	0.94
dai	−2.15	exhibit	0.76
period	−1.67	option	0.68
year	−1.24	cost	0.66
month	−1.02	ari	0.63
within	−0.95	own	0.55
termin	−0.88	portion	0.48
expir	−0.86	roxio	0.46
monthli	−0.85	sale	0.39
quarterli	−0.83	product	0.39
upon	−0.79	parti	0.39

In summary, Naïve Bayes and Logistic Regression perform the best among the four methods whereas the KNN performs the worst. In our experiment, prepositional phrase disambiguation achieves an F-measure of 0.90 using classification methods. This result demonstrates the feasibility of text classification as applied in extracting temporal constraints for service engagements.

Annotator

The text classification tasks we consider are not time critical. Applications such as annotator can process the documents offline and then provide users with highlighted information.

To illustrate the use of our trained model, we built a temporal annotator using the model we trained on top of the GATE framework [16]. The quoted text below illustrates the annotation result on a purchasing agreement between Redhook Ale Brewery Incorporated ("Redhook") and Anheuser-Busch Incorporated.[13] The underlined text is the business event and the italic text is the temporal constraint discovered by our model.

> (c) In the event that the orders and deliveries of Packaging Materials made by Supplier to Redhook have failed in respects material to Redhook's Portsmouth operations to comply with the terms of the Supply Agreement and Redhook determines (such determination to be made in good faith and on a commercially reasonable basis) that such failures are likely to continue, Redhook may terminate the purchase and sale obligations of Redhook and ABI under this Agreement *upon 30 days written notice* to ABI and Supplier.

We ran the annotator on a Lenovo T520 laptop with 8G memory and Intel i5-2520M 2.50GHz CPU

[13] http://contracts.onecle.com/redhook/anheuser.supply.2002.11.21.shtml

running Windows 8. Table 16 shows the running time of sample manufacturing services contracts. The time varies with the contract length, sentence complexity, sentence structure, and computing environment. The preprocessing and annotation can be done offline. After the document processing is done, a user can easily find the events and temporal constraints in a contract.

The text below is from the contract body of a manufacturing agreement between B&W and Star Scientific, Inc.[14] The underlined text shows events and the italic text shows temporal constraints extracted using our model. Our model finds most of the events and temporal constraints, but it misses an event—"it will open the Chase City facility" (in item 7 below)—as a false negative.

1) B&W will buy 15 million green weight pounds of Stars contracted flue-cured tobacco for delivery *in 2003* at the price set forth in the Restated Master Agreement, namely, at a purchase price determined pursuant to B&Ws Contract Price Schedule plus a $0.10/pound (green weight) premium. B&W will make such payments directly to the growers and will provide whatever additional personnel at Chase City is needed for such payments.
2) B&W will also buy Stars excess 2003 flue-cured tobacco up to a maximum of 3.7 million green weight lbs. (no more than a maximum of 20,000 green weight lbs. per barn) for delivery *in 2003* at a purchase price determined pursuant to B&Ws Contract Price Schedule. B&W will also pay the growers on Stars behalf the $0.10/pound (green weight) premium contracted by Star for such tobaccos. B&W will pay directly to Golden Leaf Tobacco the $0.07/pound (green weight) processing fee for the first 15 million pounds set forth in the Chase City License and Services Agreement and $0.05 for any amount above 15 million pounds, but B&W will have no further obligations to either Star or Golden Leaf Tobacco for services rendered at Chase City. Star will reimburse B&W weekly for all such premium payments made on Stars behalf, *within one day* of B&Ws billing of such costs to Star.
3) B&W and Star agree that B&W will have no obligations to buy leaf from Star *after 2003*, and B&W will not be required to reduce Stars Obligations by the $0.80/pounds amount for unpurchased tobaccos as set forth in Section 3.04 of the Restated Master Agreement.
4) B&W will reduce Stars $7,161,005.42 liability to pay B&W for 1.9 million lbs. of processed leaf purchased for Stars account. B&W will retain possession of such leaf and reduce Stars obligation for such leaf from $7,161,005.42 to $3,700,000.
5) B&W will extend the repayment schedule on the notes payable by Star to B&W so that the balance is payable in equal consecutive monthly installments over 96 months, rather than the current 60 month term. This extension applies to both principal and interest.
6) B&W will retain possession of unshipped cut tobacco prepared for Star under the Supply Agreement for Star Scientific Blend and Star will pay B&W a disposal fee of $60,000.
7) Star agrees that it will open the Chase City facility no later *than August 18, 2003* for operation in accordance with the Chase City License and Services Agreement except as the terms of such Agreement have been modified by the terms of this Letter Agreement.

[14]http://contracts.onecle.com/star-scientific/brown.mfg.2003.08.14.shtml

Challenges and Prospects

Our evaluation demonstrates the effectiveness of machine learning methods for mining business events and temporal constraints. Supervised information extraction from service contracts faces unusual challenges. First, a contract is a legal artifact, and often exhibits more complicated nested structure and longer sentences than ordinary English text. Section and clause headings often cause the sentence boundary detector to break. The length of the sentences challenges the Stanford Parser to output the grammar tree. Second, an event is a subtle semantic unit that challenges automatic extraction. We define events as activities that capture essential business processes. Whereas other event extraction settings involve sentence selection, our events occur at the subsentence level. Pruning helps reduce redundancy in a long legal sentence to capture the most important phrase that expresses an event. The extra processing enhances clarity but may lose information in some cases. Third, building a gold standard dataset is time consuming. Due to the lack of benchmark datasets relating to contracts, we built our own training corpus for event and temporal classification. Evaluation of the event topics is time consuming because there is no gold standard data available.

TABLE 16
Annotator running time.

Contract Parties	Contract Source	Time (in seconds)
Kopin and Xybernaut	http://contracts.onecle.com/xybernaut/kopin.interim.1996.05.13.shtml	48
Nephros and Medica	http://contracts.onecle.com/nephros/medica.mfg.2003.05.12.shtml	414
Net Cash Services and Mark Deering	http://contracts.onecle.com/silver-star/deering.supply.2002.12.01.shtml	186
JingAo and Powerlight	http://contracts.onecle.com/ja-solar/powerlight-mfg-2007-01-12.shtml	59
DSM and Cubist	http://contracts.onecle.com/cubist/dsm.mou.2000.01.26.shtml	92
B&W and Star Scientific	http://contracts.onecle.com/star-scientific/brown.mfg.2003.08.14.shtml	24

6 RELATED WORK

We focus our comparisons on service computing.

6.1 Contract Analysis

Traditional studies on contracts have focused on their representation, abstraction, execution, monitoring, and model-checking [17], [18]. In general, our approach does not address the challenges these studies pursue but would support such studies by helping identify the relevant events and temporal constraints.

Milosevic et al. [19] present a contract monitoring facility. Their approach involves the Business Contract Language (BCL) as a way to represent and monitor contracts. Their focus is on the technical aspects of representing and monitoring contracts. However, since BCL is includes the notions of events and temporal constraints, one can conceivably use an approach such as ours to help create a BCL specification based on a contract describing a service engagement.

Vidyasankar et al. [20], [21] studied activities in contracts with an focus on payments. Business events, which we extract here, are a broader conception than just payments. We observe that payments are an important family of business events in practical contracts. Indeed, Table 9 shows that payment and related events show up in different domains. And, Table 7 shows how the extracted vocabularies map to payment events. Vidyasankar et al.'s main focus is on modeling and executing contracts, whereas our interest is in extracting the relevant business events by mining contracts.

Molina-Jimenez et al. [22] provide an approach for checking the compliance of monitored business interactions with respect to a formally specified contract. The above approaches perform their enactment, monitoring, and analysis based upon a formal model. Our contribution in this work is complementary in that we show how to extract the elements of such a formal model in terms of the business events and temporal constraints involved in a service engagement.

Van der Aalst [23] studies service and business process mining from execution logs. Service and process mining seeks to discover process models expressed in execution logs at an operational level, e.g., to determine control flow models describing the order in which certain messages tend to (or need to) occur. In contrast, our interest is in business events, whether or not they correspond to individual messages. Further, we extract business events from contracts. Since cross-organizational processes are created and maintained to support contracts among business partners, potentially our approach could be used to seed the mining of execution logs.

Despite its great potential, information extraction from unstructured contract text to aid in the elicitation of service engagement requirements has not received much attention in the research community. A few notable efforts apply classification to study contract clauses and structures. Indukuri and Krishna [24] adopt SVM to classify clauses in contracts as being either payment related or not so. They obtain the best results from an n-gram model when $n = 4$. Curtotti and McCreath [25] study the segmentation of Australian contracts with a combination of rules and machine learning. They use 40 features including structural and statistical information to classify a sentence into one of 32 classes.

In prior work, we pointed out the importance of discovering knowledge and insights from contract text, and motivated the problem of bridging the gap between executable electronic contracts and difficult-to-analyze textual contracts [1]. The specific task we addressed was contractual exception extraction. We found that contracts often use routinized expressions to convey the service exceptions and that patterns can be an effective method for their extraction.

Khandekar et al. [26] propose MTDC (Methodology and Toolkit for Deploying Contracts) system based on ER^{EC} data model. MTDC supports visualizing and enacting contracts as bases for deployable and executable electronic contracts. Linguistic and statistical features along with domain and contract specific keywords are used in the contract management system.

6.2 Service Engagement Modeling

Recognizing that service engagements pervade the modern economy, Purvis and Long [27] take an *interactionist* rather than an *objectivist* perspective as the underlying principle for modeling real-world businesses. They place multiagent concepts such as norms and institutions at the center of service modeling. Purvis and Long's ideas are naturally cohesive with our approach because business events are the fundamental elements of normative relationships. Therefore, extracting events helps ground the relationships that characterize service engagements.

Our work accords well with conceptual models for service-oriented applications in open environment, e.g., [28], [29]. In these settings, contracts provide a natural basis for capturing how a service engagement is constructed and enacted. Chopra et al. [30] present an approach for modeling service engagements via commitment protocols to improve the flexibility and expressiveness of engagements. Our approach can help elicit the business events and constraints that ground such protocols.

Kohlborn et al. [31] study 30 extant service identification approaches and propose a consolidated approach to identify and analyze business services. However, in this work, the process of abstracting and identifying service engagements is manual. Therefore, significant human effort is needed to build the abstract representations of a service engagement. Our supervised approach for extracting business events and temporal constraints facilitates service engagement analysis and provides the necessary foundations for automated service engagement identification, and addresses challenges posed in a open contractual environment.

Service components analysis facilitates service requirements analysis in business domains. Vitharana et al. [32] propose the knowledge-based component repository (KBCR) to aid service requirement analysis. Similar to their approach, Contract Miner studies a repository of contracts describing service engagements. In contrast to KBCR, which focuses on formally represented services, Contract Miner studies a contracts repository represented in unstructured text. Further, Contract Miner discover topics of different contract domains in an unsupervised fashion, thereby potentially facilitating the creation of a repository such as KBCR.

7 Discussion

We studied contracts as specifications of service engagement. Business events and temporal constraints are crucial to enacting a service engagement, therefore extracting them is essential for each party to an engagement to ensure it is being enacted correctly. Business events and constraints can be automatically analyzed to determine whether a potential service engagement is well-formed. Moreover, each party can check if the engagement is acceptable given its individual goals.

Importantly, our techniques work on real-life contracts and can thus facilitate service engagements that arise in practice. Our classification-based extraction yields F-measures in the high 80% range and vocabulary clustering yields a 85% match with the gold standard.

We plan to extend our tool suite. It would be interesting to discover the dependency relationships across business events, e.g., if one event is a prerequisite of another. In the case of manufacturing, a down payment may be a prerequisite for product delivery and installment payments for continued product supply. Interlocked events form a network of business activities and lay the foundation for effective service engagements as a basis for successful commerce.

It is also worth studying the *types* of dependencies because these are associated with different (normative) business relationships. In particular, these relationships can be categorized as normative relationships, such as commitments, permissions, and prohibitions. Events relate intimately to the antecedents and consequents in such normative relationships [3]. Enriching the models in this manner can lead to improved requirements elicitation for service engagements as well as a principled basis for automating the service engagement life cycle from the perspective of a business partner.

Acknowledgments

We thank Pankaj Mehra and Chung-Wei Hang for helpful discussions on topic modeling. Xibin Gao would like to thank Edward Curry and Sean O'Riain for hosting his visit to Digital Enterprise Research Institute, National University of Ireland, Galway.

References

[1] X. Gao, M. P. Singh, and P. Mehra, "Mining business contracts for service exceptions," *IEEE Transactions on Services Computing*, vol. 5, no. 3, pp. 333–344, Jul. 2012.

[2] H. Tanev, J. Piskorski, and M. Atkinson, "Real-time news event extraction for global crisis monitoring," in *Proceedings of the 13th International Conference on Natural Language and Information Systems: Applications of Natural Language to Information Systems*, ser. NLDB. London: Springer-Verlag, 2008, pp. 207–218.

[3] M. P. Singh, "Norms as a basis for governing sociotechnical systems," *ACM Transactions on Intelligent Systems and Technology (TIST)*, pp. 1–21, 2013, to appear; available at http://www.csc.ncsu.edu/faculty/mpsingh/papers.

[4] H. H. Malik, V. S. Bhardwaj, and H. Fiorletta, "Accurate information extraction for quantitative financial events," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. Glasgow: ACM, 2011, pp. 2497–2500.

[5] M. Pasca, "Answering definition questions via temporally-anchored text snippets," in *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, Hyderabad, January 2008, pp. 411–417.

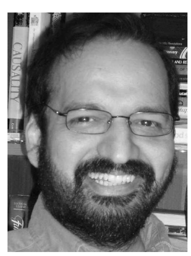
[6] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik, *A Grammar of Contemporary English*. Harlow, Essex: Longman, 1984.

[7] M. De Marneffe, B. MacCartney, and C. Manning, "Generating typed dependency parses from phrase structure parses," in *Proceedings of the 5th International Conference on Language Resources and Evaluation*. Genoa: European Language Resources Association (ELRA), 2006, pp. 449–454.

[8] M. Marcus, M. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of English: The Penn Treebank," *Computational Linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[9] J. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by Gibbs sampling," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Ann Arbor, Michigan: Association for Computational Linguistics, 2005, pp. 363–370.
[10] I. Witten, E. Frank, and M. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2011.
[11] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel, "Performance measures for information extraction," in *Proceedings of the DARPA Broadcast News Workshop*, 1999, pp. 249–252.
[12] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, March 2002.
[13] P. Berkhin, "A survey of clustering data mining techniques," *Grouping Multidimensional Data*, pp. 25–71, 2006.
[14] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet allocation," *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, January 2003.
[15] F. Schilder and C. Habel, "From temporal expressions to temporal information: Semantic tagging of news messages," in *Proceedings of the ACL Workshop on Temporal and Spatial Information Processing*. Toulouse: Association for Computational Linguistics, 2001, pp. 65–72.
[16] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "A framework and graphical development environment for robust NLP tools and applications," in *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, ser. ACL. Philadelphia: Association for Computational Linguistics, 2002, pp. 168–175.
[17] P. Radha Krishna and K. Karlapalem, "Electronic contracts," *IEEE Internet Computing*, vol. 12, no. 4, pp. 60–68, 2008.
[18] F. Meneguzzi, S. Miles, M. Luck, C. Holt, and M. Smith, "Electronic contracting in aircraft aftercare: A case study," in *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS) Industry Track*. Estoril, Portugal: IFAAMAS, May 2008, pp. 63–70.
[19] Z. Milosevic, S. Gibson, P. Linington, J. Cole, and S. Kulkarni, "On design and implementation of a contract monitoring facility," in *Proceedings of the 1st IEEE International Workshop on Electronic Contracting*. San Diego, California: IEEE, 2004, pp. 62–70.
[20] K. Vidyasankar, P. Radha Krishna, and K. Karlapalem, "Study of execution centric payment issues in e-contracts," in *IEEE International Conference on Services Computing*, vol. 2. IEEE, 2008, pp. 135–142.
[21] K. Vidyasankar, P. R. Krishna, and K. Karlapalem, "Study of dependencies in executions of e-contract activities," in *Proceedings of the 13th East European Conference on Advances in Databases and Information Systems*, ser. LNCS, vol. 5739. Berlin: Springer, 2009, pp. 301–313.
[22] C. Molina-Jimenez, S. Shrivastava, and M. Strano, "A model for checking contractual compliance of business interactions," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 276–289, 2012.
[23] W. van der Aalst, "Service mining: Using process mining to discover, check, and improve service behavior," *IEEE Transactions on Services Computing*, p. 1, 2012.
[24] K. V. Indukuri and P. Radha Krishna, "Mining e-contract documents to classify clauses," in *Proceedings of the 3rd Annual ACM Bangalore Conference (Compute)*. Bangalore, India: ACM, 2010, pp. 1–5.
[25] M. Curtotti and E. McCreath, "Corpus based classification of text in Australian contracts," in *Proceedings of the Australasian Language Technology Association Workshop*, Melbourne, Australia, 2010, pp. 18–26.
[26] A. Khandekar, P. R. Krishna, and K. Karlapalem, "A methodology and toolkit for deploying contract documents as e-contracts," in *Tutorials, posters, panels and industrial contributions at the 26th international conference on Conceptual modeling*. Auckland, New Zealand: Australian Computer Society, Inc., 2007, pp. 91–96.
[27] M. Purvis and A. Long, "Affinities between multi-agent systems and service-dominant logic: Interactionist implications for business marketing practice," *Industrial Marketing Management*, vol. 40, no. 2, pp. 248–254, 2011.
[28] M. Verdicchio and M. Colombetti, "Commitments for agent-based supply chain management," *ACM SIGecom Exchanges*, vol. 3, no. 1, pp. 13–23, 2002.
[29] M. P. Singh, A. K. Chopra, and N. Desai, "Commitment-based service-oriented architecture," *IEEE Computer*, vol. 42, no. 11, pp. 72–79, Nov. 2009.
[30] A. K. Chopra, F. Dalpiaz, P. Giorgini, and J. Mylopoulos, "Modeling and reasoning about service-oriented applications via goals and commitments," in *Proceedings of the 22nd International Conference on Advanced Information Systems Engineering (CAiSE)*, 2010, pp. 417–421.
[31] T. Kohlborn, A. Korthaus, T. Chan, and M. Rosemann, "Identification and analysis of business and software services—a consolidated approach," *IEEE Transactions on Services Computing*, vol. 2, no. 1, pp. 50–64, Jan. 2009.
[32] P. Vitharana, H. Jain, and F. M. Zahedi, "A knowledge based component/service repository to enhance analysts' domain knowledge for requirements analysis," *Information & Management*, vol. 49, no. 1, pp. 24–35, Jan. 2012.

Xibin Gao Xibin Gao is a research software design engineer at Microsoft working on document understanding. Xibin Gao received the PhD degree in Computer Science from North Carolina State University, Raleigh. Previously, he also received the BS degree from Sichuan University and the MS degree from Chinese Academy of Sciences. His research interests include natural language processing, information extraction, and machine learning.



Munindar P. Singh Munindar P. Singh is a Professor in the Department of Computer Science, North Carolina State University, Raleigh. His research interests include multiagent systems and service-oriented computing with a special interest in the challenges of trust, service selection, and business processes in large-scale open environments. His books include the coauthored *Service-Oriented Computing* (Wiley, 2005). Singh is the Editor-in-Chief of the *ACM Transactions on Internet Technology*; he was the Editor-in-Chief of *IEEE Internet Computing* from 1999 to 2002. Singh is a member of the editorial boards of *IEEE Internet Computing*, *Autonomous Agents and Multiagent Systems*, *Journal of Artificial Intelligence Research*, the *IEEE Transactions on Services Computing*, and the *ACM Transactions on Intelligent Systems and Technology*.