# A Gray Box Conceptual Model for Accountability and Ethics in Business Contracts

**Munindar P. Singh**
NC State University

**Xibin Gao**
Amazon Alexa AI

***Abstract*—Current computational models are inadequate for the purposes of modeling interactions between autonomous parties in a way that highlights and supports their accountability. We propose a new conceptual model for business contracts based on norms motivated by a review of real-life business contracts. Our conception is of a gray box, reflecting the idea that a contract makes the participants accountable to one another and to outside entities, and therefore calls for the exposure of sufficient implementation details. The model consists of a recursively applicable taxonomy of clause types. In a preliminary study, we found that computer scientists are able to effectively identify the concepts introduced in this model, thereby indicating its potential for building Internet applications that support accountability.**

**Index Terms: Business contracts; Norms; Compliance**

## 1. Introduction

As Internet applications become evermore rooted in our social and business lives, it is important to capture their requirements in a manner that highlights expectations of good behavior.

We consider the challenge of developing Internet applications that involve cross-organizational interactions and tackle two crucial aspects: (1) preserving autonomy, i.e., ensuring that the participants have as much flexibility as possible in how they behave and (2) achieving accountability, i.e., each participant can call others to account for legitimate expectations of them and provide accounts to others.

We think of a system of two or more autonomous parties that interact via information technology as a *sociotechnical system* or *STS*. Importantly, the interactions of the participants in an STS are best understood in terms of their *contracts* with one another.

Specifically, a contract between two or more parties describes the ways in which they are accountable to one another, that is, what each party may legitimately expect from another. Accountability applies not only to the main deliverables of the interaction but to any ancillary expectations. For example, in e-commerce and manufacturing, the parties may expect sustainability and avoidance of child labor. In information settings, the parties may expect confidentiality and privacy.

Accountability arises when one party has the standing to hold another to account for some behavior or outcome. Accountability can be supported by explanations of decision making.

Whereas current computer science best practices rely on a black box view of components to promote substitutability, to talk of accountability properly, we adopt a *gray box* model that exposes internal implementation details that are subject to an accountability requirement. For example, capturing our motivating examples of accountability—labor laws, environmental sustainability, data confidentiality—requires access to internal details.

This article addresses a fundamental challenge in regards to contracts, namely, *how can we express accountability requirements of the concerned participants in a sociotechnical system?*

To this end, it presents a conceptual model for business contracts that identifies concepts needed for real-life business contracts. Our approach in developing this model was to consider (1) theoretical ideas in the understanding of business services from a conceptual architecture, in modeling sociotechnical systems, an application in a large-scale scientific collaboration setting, and (2) by a reading of real-life (natural language) business contracts. We synthesized these considerations to assemble a conceptual model that is simple yet comprehensive.

We conducted an informal empirical study involving human subjects to evaluate whether the concepts introduced in the proposed conceptual model would be effectively recognized in real-life contract by IT practitioners. Our results are promising and highlight the need for improved tool support for understanding and specifying contracts.

## 2. Norm Types

To support accountability, we underpin our conceptual model of contracts on a notion of *norm* inspired from the law [1, 2]. Work on social interactions investigates logical representations, such as of deontic logic and norms [3]. Formal representations for contracts are valuable because they provide a basis for (1) judging the compliance of the observed behaviors of the concerned parties, for example, as they enact a business process [4] and (2) designing agents via the requisite roles [5].

We consider the following main types of norms. The initial motivations for using this classification of norms were to capture two major kinds of scenarios: (1) scientific collaboration, understood based on discussions with stakeholders [2] and (2) business contracts, understood based on a review of real-life contract documents [6]. Additional backing for this classification arises from success in mining contract text to infer the norm types expressed in it [7]. Below each norm type is expressed as relating two agents or roles, subject and object, respectively.

A *commitment* means that its subject commits to its object to ensure the consequent if the antecedent holds [8]. We distinguished commitments of two subtypes [8]. A *dialectical* commitment represents a claim staked by its subject, i.e., that the consequent is true if the antecedent is. Representations and warranties made in a contract (e.g., the seller owns what she is selling) are dialectical commitments. A *practical* commitment represents a promise to ensure that the consequent will be brought about if the antecedent becomes true. For example, a seller's offer to a prospective buyer to provide a specified service is a practical commitment.

A *prohibition* means that its subject is forbidden by its object from bringing about the consequent if the antecedent holds. For example, a collaborator may be prohibited from modifying the firmware of an instrument being shared by its owner.

Notice that, under these definitions, a prohibition is not the negation of an authorization, as in traditional deontic logic. Von Wright, who invented deontic logic, realized the limitations of the traditional thinking. His later thinking [1], of an authorization as a prohibition against its grantor, accords with our model.

An *authorization* means that its subject is authorized by its object for bringing about the consequent if the antecedent holds. The intuition is that an authorization concerns a "physical" action, i.e., a domain-level action as being conceptualized. For example, in scientific settings, an instrument owner may authorize a collaborator to control that instrument under specified conditions. If the collaborator fails to control the instrument despite the conditions being met, the authorization fails, and the instrument owner is accountable for that failure.

A *power* means that its subject is empowered by its object to bring about the consequent if the antecedent holds. A power refers to the ability to perform a "social" action, i.e., one that changes normative relationships [1, 9]. For example, a university may rescind the library privileges of a student who violates digital licenses.

A *sanction* specifies the penalties or rewards its subject faces from its object because of the state of another norm [10]. In healthcare, a physician who violates a prohibition against prescribing addictive pain killers to children may be sanctioned by having her board certification revoked. A sanction to be applied by a community [11] can be captured because we can treat a community as an Org and the Org can be the party that applies the sanction.

## 3. Conceptual Model for Contracts

Figure 1 presents our model for contracts. We model a contract recursively as a set of contracts bottoming out on a set of clauses. The recursion is unnecessary in that a contract could be represented without such recursion. However, the recursive formulation is natural since in real-life contracts the clauses are structured and often a contract exhibits a repeating structure. Each clause would map to one or more norms of the types introduced above and thereby enable formal reasoning [12]; we do not illustrate the norm syntax here.

Clauses in real-life contracts fall into the following major categories.

### 3.1. Service Clauses

These are the main clauses of the contract. They capture what the contract is about—i.e., what service each party provides to the others—and the main "business" reason for having a contract in the first place.

**A definitional clause** specifies the terms used in the remainder of the contract. The terms include descriptions of the parties involved and symbolic role names for them. Thus, typically, the definitional clauses encapsulate the binding between a principal and the role that the principal plays in the contract; the rest of the contract proceeds in terms of the role name.

**A provisioning clause** describes what services the roles in the contract are expected to provide each other or perform for each other during the contract. We can think of these services as capabilities to be exercised. Specifically, we understand these services in the broad sense that if one party carries out some complex task and the other merely pays cash, we would treat both the task and the payment as capabilities that the parties respectively bring to the table. In other words, the main service or business transaction described in the contract involves the provisioning of a suitable subset of the specified capabilities.

**A quality of service clause** describes one or more qualities associated with a capability being brought to bear in the contract. These would relate to the so-called nonfunctional requirements of software engineering, such as availability and latency [13].

### 3.2. Scoping Clauses

These clauses specify the purpose and scope of a contract. These clauses are crucial from the standpoint of accountability because of their potential effect on the legal standing of the parties involved. For example, the scope of an office lease contract may be limited to business use, which could provide the landlord a basis for evicting a tenant who runs a hotel in the office space or lets employees live in the space or begins to manufacture leather goods there. Also, the landlord would not need to satisfy the safety requirements of residential housing (where children may live) and be able to carry out repairs on work holidays without notifying the tenant.

**A purpose clause** specifies the kinds of services that are included, e.g., the noncommercial use of licensed software.

**A termination clause** would specify the trigger conditions under which the contract would end. For example, a student may use the campus network to access a university's digital library resources as long as the student's account is active. The student's account is terminated after one academic term of the student failing to register, whether upon graduation or otherwise.

**A duration clause** is a kind of termination clause; it states how long a contract would last. Such a clause is the easiest way in which termination can be specified by date and time or a duration from the date and time of initiating the contract.

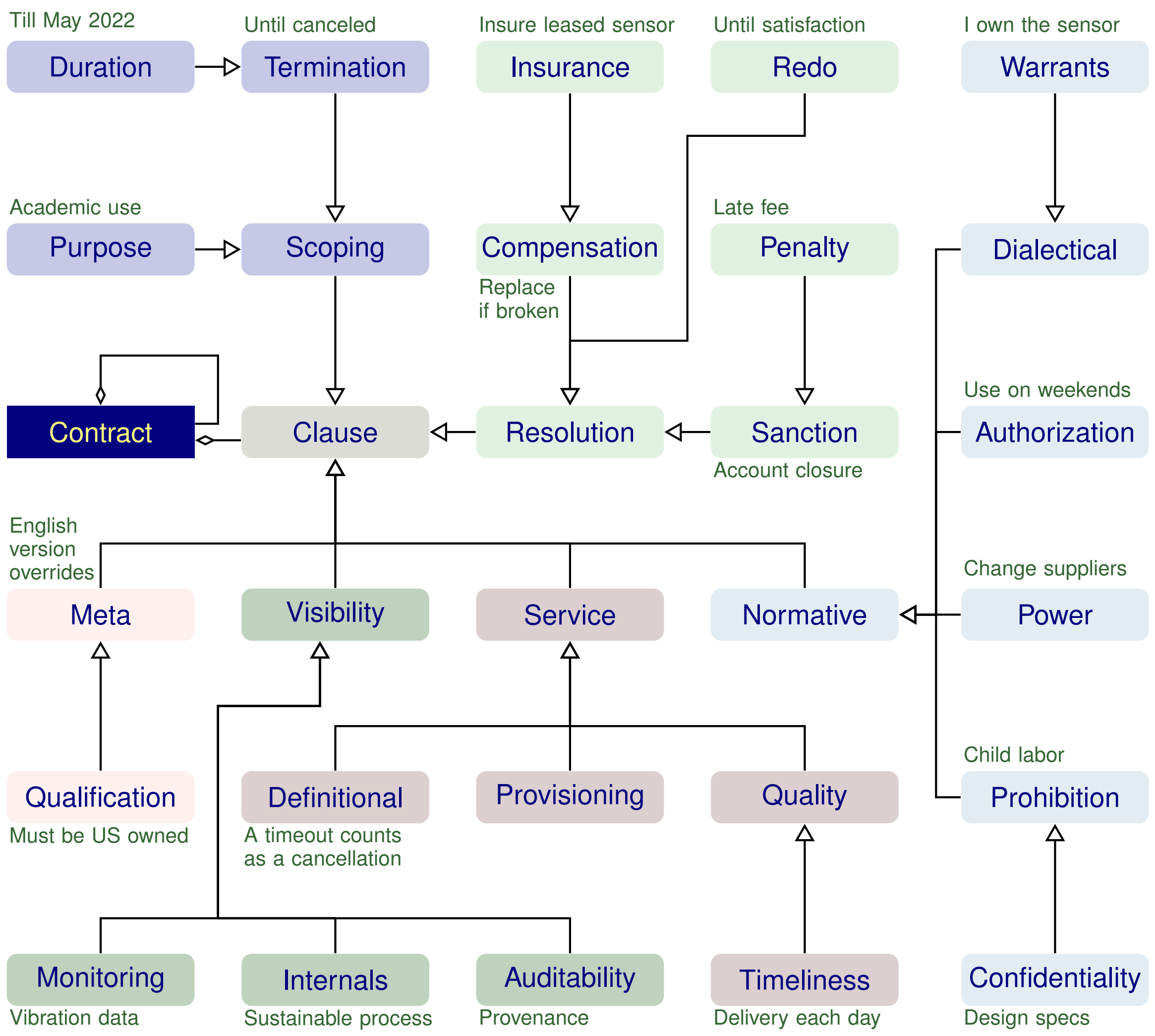

**Figure 1.** A conceptual model for business contracts geared toward services along with brief examples drawn from the domain of cyberinfrastructure for scientific collaboration. The model is rooted on Clause and its upper levels are deemed fixed; its lower levels, such as Timeliness and Confidentiality are merely illustrative of the kinds of clauses one might see in a contract. (A triangular arrowtip indicates subclass and a diamond arrowtip indicates aggregation.)

### 3.3. Resolution Clauses

These clauses deal with accountability violations, including the possibility of sanctions (of violators) and of compensations (by violators). In essence, these are ways to ensure coherence of the service engagement, possibly by restoring a good state of a service when it enters an undesirable state.

**A compensation clause** describes how, and how perfectly, a correct (or acceptable) state is restored [14]. For example, if the goods delivered are damaged, then the purchaser may obtain a refund upon returning the goods, with or without shipping and handling costs. In general, the restored state may not be perfect from the perspective of each party.

**An insurance clause** describes an external protection, usually via a third-party, against certain kinds of exceptions. The third-party thus provides a means to mitigate the risk that the parties to the contract face, especially regarding each other's potential unreliability in different circumstances.

Insurance arguably relieves the concerned parties of their accountability because a third-party would cover for their misbehavior or negligence.

**A redo clause** describes how and how often a service provisioning may be redone. For example, if shoes received the wrong size, the merchant will replace with another size once.

**A sanction clause** describes how to respond to accountability violations. Sanctions could be punitive or educational [10]. Where sanctions are not customary, a dialectical commitment (expressed in a dialectical clause, as below) would make their existence clear. Common forms of sanctioning are through the somewhat amorphous means of reputation and via escalation of complaints [14]. An example is when a library complains about a student to a university board, which may suspend the student.

**A penalty clause** specifies sanctions applied in monetary terms.

### 3.4. Visibility Clauses: The Gray Box

Naively one might treat a contract as applying between parties viewed as black boxes, analogous to Web services in computing. However, support for accountability requires the parties to have access to suitable internal details of each other. Each party would rely upon visibility clauses to make sure (1) that the deliverable is of an adequate quality; (2) that the effort is robust; and (3) that the provisioning of the service does not violate any laws or regulations to which one of the parties might be subject, and which might be a source of potential legal or business liability.

Visibility may be accorded to designated parties. For example, a consumer may rely on a Fairtrade [15] label, but the Fairtrade organization would examine a vendor's trade practices to certify it; likewise, an organic certification may review the provenance of the ingredients used.

**A monitoring clause** specifies how progress on the service will be reported, usually on an ongoing basis until the engagement is complete. For example, the manufacturer will notify the purchaser weekly of how many of the goods are ready and being shipped. A mortgage broker will notify a loan applicant of progress on the loan approval and underwriting every day. A shipper will provide information every time a shipped package moves a step closer to delivery.

**An implementation clause** specifies how the provisioning may be realized. For example, the manufacturer should use only new parts or may use refurbished parts for the case but not for the main electronics. An outsourcing service may only use workers of legal age as determined in the country of the outsourcer, not just the outsourcing service.

**An auditability clause** describes the level of access each of the parties has on the operations of the other party. For example, a vendor may conduct spot checks on the manufacturing processes of its supplier. A customer of a cloud provider may ask to see disk and network failure logs.

### 3.5. Normative Clauses

As stated above, each contract clause maps to one or more interrelated norms. For example, definitional clauses map to dialectical commitments; provisioning and quality clauses to practical commitments; scoping clauses to dialectical and practical commitments, authorizations, and prohibitions; resolution clauses to practical commitments and powers; visibility clauses to practical commitments, authorizations, and powers.

Real-life contracts contain additional clauses, which characterize the regulations that apply on the interactions among the participants and help delineate more subtle expectations from an interaction. We term such accountability-focused clauses *normative clauses* and include them in the present category.

**A dialectical clause** specifies the dialectical commitments that feature in a contract. A typical usage of such clauses is with the representations and warranties that each party gives the others.

Another variant of the dialectical clauses consists of acknowledgments by one party to another, for example, that it recognizes that it is only obtaining a sensor as a loan and is not becoming an owner of the sensor. One might see that such clauses are superfluous because if the contract does not transfer ownership, there should be no need to say that ownership is transferred. In general, all such negative assertions cannot be effectively enumerated. However, for some important cases, especially when there is a history of prior encounters or there are some accepted industry practices, it may be worth explicitly over-turning them through an acknowledgment.

Interestingly, practical commitments are common as the bases for contractual clauses and don't need to be treated separately under normative clauses.

**A warrants clause** is a kind of dialectical clause that specifies what representations a party makes, e.g., about being licensed to operate.

**An authorization clause** specifies what domain actions a party may carry out.

**A power clause** specifies what a party is empowered to do. For example, in international trade, to clear customs, a purchasing enterprise may select who its designated receiving agent will be; the seller will contact that agent when appropriate. Or, a vendor may place a lien on the buyer's property: exercising that power would be one of the sanctions that the vendor may impose for failure to pay.

We understand indemnification as a power clause. The party who indemnifies another empowers the latter to demand restitution in case of loss.

**A prohibition clause** specifies what a party is forbidden from doing. We adopt the design pattern proposed by Singh [2] for IT settings, namely, that authorizations are imposed architecturally or computationally whereas prohibitions are external to the computational system. Notice that in general the above design pattern may not hold.

**A confidentiality clause** is a kind of prohibition that applies to information sharing. These clauses are common in contracts. They can apply both to the information produced while enacting the contract and to information about the contract itself. In the latter case, they would feature as meta clauses, which are described below.

### 3.6. Meta Clauses

This clause type captures contractual requirements about the contract itself, such as whether one of the parties to the contract can pass on or sell off the contract to a third party. Meta clauses can apply to the other clause types. For example, a contract may be renewed as long as both parties agree. Or, the contract can be terminated any time with a two-week notice by either party (as common in employment or consulting agreements).

Meta clauses can provide a more natural way than duration to model some situations that are suggestive of termination. For example, "if you don't use your account for six months, the account will expire and you will lose any accumulated credits" does not have a clear duration but can be captured as a meta clause.

Meta clauses are subject to the normative clauses potentially. They generally define powers of the specified participants but they can also involve the other norms.

**A qualification clause** describes who might participate in the contract. For example, the seller of a car or of a house ought to be its title holder (owner). In current US practice, this qualification is captured reliably for a car, wherein the owner of a car provides a document establishing its title in his or her name. For a house (or land), in contrast, there is greater perceived risk of any such documents being invalid (because of a possible flaw in some prior transaction) and thus it is customary to use an external title insurance provider—thereby demonstrating an insurance clause.

## 4. Informal Evaluation and Results

We conducted an informal empirical evaluation to gauge the effectiveness of our contract model in capturing real-life contracts. To this end, we selected five representative real-life contracts from Onecle [6] that addressed business needs in disparate domains.

Our respondents were 32 graduate students enrolled in Computer Science. (We have obtained a waiver of consent from the NCSU Institutional Review Board.) This sample is representative of early-stage developers.

We divided the respondents into five groups and assigned a contract to each group; each to work independently of others. The task was to identify clauses of the different types in the assigned contract.

Since we performed the evaluation over naturally occurring contracts without constraining the respondents, we gained some confidence in the quality of the evaluation. However, we also faced some challenges, which could have weakened our conclusions. In particular, respondents marked different parts of a contract segment; annotated a clause, especially one that is long, with more than one clause type; or, didn't precisely state what part of a contract segment corresponds to what clause type.

We provide illustrative results for three contracts below.

**Asset purchase.** The purchase agreement between ARN Tellem and SFX Sports Group[1] was judged by seven respondents. We randomly selected the judgments provided by two respondents. Out of 78 annotations, 53 clauses were identical, yielding an interannotator agreement of 68%. Among the 53 agreed clauses, 26 are definitional clauses composing 49% of the total.

**Licensing.** The licensing agreement between Yahoo and Microsoft[2] was annotated by eight respondents. Two randomly selected respondents agreed on 83 out of 107 annotations, yielding an interannotator agreement of 78%.

**Corporate reorganization.** The reorganization agreement between CTI Corporation and PETNet Pharmaceutical Services[3] was annotated by three respondents. Two randomly selected respondents agreed upon 36 out of 94 annotations, yielding an interannotator agreement of 38%. Among these 36 clauses, two are definitional clauses, i.e., 6% of the total.

The agreement between the annotators demonstrates the effectiveness of our contract model in guiding them to formalize and categorize the contract clause types. There are clauses that are often confused in the labeling process, which include power and authorization, termination and sanction, and prohibition and penalty.

## 5. Conclusions and Directions

Contracts are a time-honored means to capture the interactions of autonomous parties and identify their mutual accountability. Our domain-independent conceptual model provides a natural basis for incorporating accountability in Internet applications. To build agents who support accountability not only leads us to modeling legal norms but to adopting a gray box model that exposes the relevant internal information so that an agent can demand and provide accounts, and any regulators involved can observe internal actions as well as public outcomes.

A formal representation as norms enables logical reasoning and compliance verification without curtailing the autonomy of the participants—a key ingredient of accountability [12]. This conceptual model seeks to streamline the creation of such formal representations. A key challenge is to develop tools to facilitate specifying contracts that reflect stakeholder requirements, including ways to extract formal representations from existing textual contracts and regulations.

## Acknowledgments

The preparation of this paper was partially supported by the NSF (grant IIS-1908374).

## REFERENCES

1. G. H. Von Wright, "Deontic logic: A personal view," *Ratio Juris*, vol. 12, no. 1, pp. 26–38, Mar. 1999.
2. M. P. Singh, "Norms as a basis for governing sociotechnical systems," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 5, no. 1, pp. 21:1–21:23, Dec. 2013.
3. A. K. Chopra and M. P. Singh, "From social machines to social protocols: Software engineering foundations for sociotechnical systems," in *Proceedings of the 25th International World Wide Web Conference*. Montréal: ACM, Apr. 2016, pp. 903–914.
4. Ö. Kafalı and P. Torroni, "COMODO: Collaborative monitoring of commitment delegations," *Expert Systems with Applications*, vol. 105, pp. 144–158, Sep. 2018.

[1] http://contracts.onecle.com/live-nation/san-vicente-purchase-2006-01-26.shtml

[2] http://contracts.onecle.com/yahoo/microsoft-license-2009-12-04.shtml

[3] http://contracts.onecle.com/cti-molecular/petnet.mer.2000.10.10.shtml

5. Ö. Kafali and P. Yolum, “PISAGOR: A proactive software agent for monitoring interactions,” *Knowledge and Information Systems*, vol. 47, no. 1, pp. 215–239, Apr. 2016.
6. Onecle, “Business contracts from SEC filings,” 2021, last accessed: 2021-04-15. [Online]. Available: http://www.onecle.com
7. X. Gao and M. P. Singh, “Extracting normative relationships from business contracts,” in *Proceedings of the 13th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. Paris: IFAAMAS, May 2014, pp. 101–108.
8. M. P. Singh, “Semantical considerations on dialectical and practical commitments,” in *Proceedings of the 23rd Conference on Artificial Intelligence (AAAI)*. Chicago: AAAI Press, Jul. 2008, pp. 176–181.
9. A. J. I. Jones and M. J. Sergot, “A formal characterisation of institutionalised power,” *Logic Journal of the IGPL*, vol. 4, no. 3, pp. 427–443, Jun. 1996.
10. L. G. Nardin, T. Balke-Visser, N. Ajmeri, A. K. Kalia, J. S. Sichman, and M. P. Singh, “Classifying sanctions and designing a conceptual sanctioning process model for socio-technical systems,” *The Knowledge Engineering Review (KER)*, vol. 31, no. 2, pp. 142–166, Mar. 2016.
11. C. Bicchieri, *The Grammar of Society: The Nature and Dynamics of Social Norms*. Cambridge, United Kingdom: Cambridge University Press, 2006.
12. S. H. Christie V, A. K. Chopra, and M. P. Singh, “Hercule: Representing and reasoning about norms as a foundation for declarative contracts over blockchain,” *IEEE Internet Computing (IC)*, vol. 25, no. 5, pp. 1–9, Sep. 2021, in press.
13. F.-L. Li, J. Horkoff, J. Mylopoulos, R. S. S. Guizzardi, G. Guizzardi, A. Borgida, and L. Liu, “Non-functional requirements as qualities, with a spice of ontology,” in *Proceedings of the 22nd IEEE International Requirements Engineering Conference (RE)*. Karlskrona, Sweden: IEEE Computer Society, Aug. 2014, pp. 293–302.
14. 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.
15. Fairtrade, “Fairtrade America,” 2021, last accessed: 2021-05-11. [Online]. Available: https://www.fairtradeamerica.org

**Munindar P. Singh** is a Professor in Computer Science and a co-director of the Science of Security Lablet at NC State University. His research interests include the engineering and governance of sociotechnical systems, and in AI ethics. Singh is an IEEE Fellow, a AAAI fellow, a AAAS fellow and a former Editor-in-Chief of *IEEE Internet Computing* and *ACM Transactions on Internet Technology*. Contact him at singh@ncsu.edu.

**Xibin Gao** is an applied scientist in the Alexa AI team at Amazon in Seattle, Washington. Previously, he obtained his Ph.D. in Computer Science at North Carolina State University. His research interests include natural language processing, dialogue systems, and machine learning. Contact him at gxibin@amazon.com.