Digital Technology, Future Lawyers and the Computable Contract Designer of Tomorrow

Corrales Compagnucci, Marcelo; Fenwick, Mark; Haapio, Helena

Published in:
Research Handbook on Contract Design

Publication date:
2022

Document version
Early version, also known as pre-print

Document license:
CC BY-NC-ND

Citation for published version (APA):
Digital Technology, Future Lawyers and the Computable Contract Designer of Tomorrow

Marcelo Corrales Compagnucci, Mark Fenwick and Helena Haapio

Abstract This article explores the shifting character of contract design in the digital revolution, focusing on how emerging technologies are providing the contracting community with innovative new digital tools and resources that facilitate the creation of more user-friendly contracts. We identify emerging technologies that strive to deliver a smoother experience for both the creators and users of contracts. As a result of these technologies, we can imagine a future where contracts will not only be legally and operationally functional, but also code-based, interoperable across platforms, and built from library-stored modular components that utilize no- or low-code functionality. To remain relevant in this new environment, contract lawyers must operate as ‘relationship engineers’ who facilitate collaboration between different stakeholders with diverse expertise in multi-disciplinary teams comprising technology, business, and legal perspectives. The ‘computable contract designer’ of the future will be a key player leveraging the opportunities of digital technology to deliver the user-friendly contracts of tomorrow.

Keywords contract design, smart contracts, computable contracts, AI, GPT-3, future lawyers

1 Introduction

This article explores the shifting character of contract design in the context of the digital revolution, focusing on how emerging technologies provide the contracting community with innovative new tools and resources that facilitate the creation of more user-oriented contracts. The creative disruption that has been experienced in other fields of economic and social life because of the proliferation of digital technologies is beginning to affect the contracting process and its outcomes, and we can now imagine a better future in which a smooth experience of contracting is no longer the exception for users but the norm. We are interested in technology, but not those technologies that sustain or reproduce the overly legalistic and un-friendly contracts of the past. Instead, we are interested in existing and near-future technologies that build upon the insights of human-centred design and contribute to a genuine change in the practice of contracting.

Crucially, we don’t believe that such a development is necessarily bad news for lawyers. One consequence of technology-driven disruption is that the contracting ecosystem will expand

* This is a draft version. The final version will be available in ‘Research Handbook on Contract Design’ edited by Marcelo Corrales Compagnucci, Helena Haapio and Mark Fenwick, forthcoming 2022, Edward Elgar Publishing Ltd. The material cannot be used for any other purpose without further permission of the publisher, and is for private use only. The authors would like to thank Leila Hamhoum for her editorial assistance. Marcelo Corrales Compagnucci is an Associate Professor of Information Technology Law at the Center for Advanced Studies in Biomedical Innovation Law (CeBIL), Faculty of Law, University of Copenhagen, Denmark. He can be contacted at marcelo.corrales13@gmail.com; Mark Fenwick is a Professor of International Business Law, Graduate School of Law, Kyushu University, Fukuoka, Japan. He can be contacted at mark@law.kyushu-u.ac.jp; Helena Haapio is an Associate Professor of Business Law at the University of Vaasa School of Accounting and Finance and a Contract Strategist at Lexpert Ltd, Helsinki, Finland. She can be contacted at helena.haapio@lexpert.com
as new and more diverse actors with technology and design, and legal and business expertise become increasingly prominent. The shift towards more functional, user-friendly, technology-driven solutions does not mean that lawyers will become obsolete anytime soon. However, to remain relevant in this new environment, the contract lawyer of the future needs to function as a ‘relationship engineer’ that brings together and works with different stakeholders in flat, open multi-disciplinary teams. Lawyers do not need to become coders or designers themselves – coding and design will remain distinct skills and roles that take on ever-greater importance. Instead, the contract lawyers of the future will need to equip themselves with the necessary knowledge – or literacy – to seamlessly partner with and mediate between multiple perspectives and facilitate the successful interfacing of human and machine.

The article is structured as follows: Section 2 introduces some examples of incoming technologies that have the potential to contribute to this change in contracting, namely computer-coded smart contracts and GPT-3-enhanced, AI-powered contract reading and writing tools. This section argues that with these technologies, we can imagine a future where contracts will be not only legally and operationally functional, but also code-based, interoperable across platforms, and built from library-stored modular components that utilize no- or low-code functionality. When contracts and contract generators are seen as not just legal or technical tools but also as part of the commercial and social world where they impact and are impacted by organizational and human needs and behavior, design enters the picture: a framework with principles and methods that help respond to the human aspects of contracting, as well.

Section 3 considers the shifting role of legal professionals in this new world. Lawyers will need a combination of legal, design and technology skills and will operate as ‘relationship engineers’ who facilitate collaboration between different stakeholders. We envisage a world in which ‘computable contract designers’ will become key players leveraging the opportunities of new contract genres and patterns along with digital technologies. Section 4 concludes.

We concede that this line of argument might be considered somewhat speculative, based as it is on the possible future application of emerging technologies. Nevertheless, we believe that this is an exercise in responsible prediction and not pure fantasy. We intend to identify current trends that point to a better future in which business, legal, design and technology professionals work together in creating the legally compliant and operationally functional, user-friendly contracts of tomorrow, for the benefit of both contracting organizations and the people who work with contracts.

2 New tools for more user-centred contracts

Since the 1970s, with the invention of the first legal databases, information technology has supported lawyers in their work, reducing costs and improving performance.¹ The success of these earlier efforts and developments in information technology, more generally, attracted investment in the deployment of such legal technologies. First-generation technologies made law firms and lawyers more efficient in performing their traditional activities, without disrupting or undermining those activities. Examples of such efficiency gains include automated billing, document storage, practice management, and accounting software.² Most of these technologies were not designed to improve the experience of clients or the users of

¹ On information technology and the legal profession, see Richard Susskind, The End of Lawyers: Rethinking the Nature of Legal Services (Oxford University Press 2008).
lawyers’ work products, but to assist lawyers in their work. The ‘risk’ of such technologies was that by helping lawyers, information technologies merely perpetuated the status quo.

However, from the early 2010s, and in the broader context of the digital transformation, legal technologies have become more sophisticated and started to incorporate technology that not only assists legal professionals in their traditional activities, but also undermines, disrupts, and transforms the character of the underlying activity. Technological advances of this kind have already started to impact transactional lawyers and contract professionals. Contracts are increasingly becoming digitized and connected with enterprise systems. Incoming artificial intelligence (AI) tools promise to help anyone – lawyers and clients alike – to prepare, review and monitor contracts and legal documents. In the rest of the section, we explore the transformative impact of emerging technologies in contracting. By illustration, we focus on two recent developments, namely contracts as code and computer-coded contracts (looking at the example of smart contracts), and AI-powered generation of new contract genres that work for people and machines (looking at the example of Open AI’s GPT-3).

2.1 Smart contracts and contract generators

A smart contract can be defined as computer code that automatically executes all or parts of an agreement and is stored on a blockchain. The concept of smart contracts dates to 1994, when it was popularized by computer scientist Nick Szabo. He suggested that many kinds of contracts can be embedded in software and hardware architecture. According to Szabo, a smart contract is ‘a set of promises, specified in digital form including protocols within which the parties perform on these promises’. Szabo’s canonical example of a smart contract in its most simple form is the vending machine, which is designed to transfer the ownership of a good (eg, a can of soda) for the exchange of money. The vending machine is in control of the property – by being physically sealed – and anyone with enough money can engage in a transaction with the vendor.

---


5 See generally Marcelo Corrales, Mark Fenwick and Helena Haapio (eds), Legal Tech, Smart Contracts and Blockchain (Springer 2019); Marcelo Corrales Compagnucci, Mark Fenwick and Stefan Wrbka (eds), Smart Contracts: Technological, Business and Legal Perspectives (Hart-Bloomsbury 2021); Mayank Pratap, ‘Everything You Need to Know About Smart Contracts: A Beginners Guide’ (Hackernoon, 27 August 2018) <https://hackernoon.com/everything-you-need-to-know-about-smart-contracts-a-beginners-guide-3a133c138378a> accessed 16 August 2021.


8 Marcelo Corrales, Mark Fenwick and Helena Haapio, ‘Digital Technologies, Legal Design and the Future of the Legal Profession’ in Marcelo Corrales, Mark Fenwick and Helena Haapio (eds), Legal Tech, Smart
By extending the logic underlying mechanical devices such as a vending machine, Szabo suggested that the computer code could be used in place of vending machines. This idea could be implemented to structure more complex transactions. Instead of transferring the ownership of a can of soda, a smart contract could transfer ownership of shares, real estate, intellectual property rights, etc. More recently, smart contracts have become central to the development of cryptocurrencies and the decentralized finance movement.

Smart contracts are encoded in such a way that execution is guaranteed by a blockchain. A blockchain is essentially a distributed ledger that can be configured to be accessible publicly or privately, and the technology maintains a secure transaction history. This blockchain-based smart contract technology allows the involved parties to transfer, receive, and store value or information through a distributed peer-to-peer computer network. Various mechanisms exist to decide whether transactions can be validated. Crucially, each approved transaction is distributed across the entire network and stored in a block only when the rest of the network approves the validity of the transaction. This process is based on past events considering the previous block. Each block holds a unique fingerprint built on cryptographic hash code using techniques like those used to create digital certificates and electronic signatures to secure authentication. Furthermore, a transaction history is maintained and can be accessed to confirm the sequence of transactions up to the present.

To take another example, consider a smart contract in the case of a car loan. If the borrower misses a payment, then the contract/code would not allow the use and operation of the car, i.e., the contract would be ‘enforced’ via networked technologies that disable the car, rather than a ‘repo man’ coming to reclaim the car. In some contexts, such contracts can produce efficiency, timing, and performance gains because of the automation of contractual terms. This automation is achieved by computer code that controls the automated performance in an Internet of Things environment where digital devices are interconnected. For example, in our car loan example, a default on a loan payment triggers a disabling of the car, thus ensuring immediate enforcement of the contractual terms.

Smart contracts seem particularly well-suited to two types of transaction: (1) ensuring the payment of funds upon certain objectively verifiable triggering events, and (2) imposing financial penalties if certain objective triggering conditions are not satisfied. In both these possible use cases, computer code can easily represent the logic of the agreement in the form of ‘If X occurs, then execute Y’ statements. Human intervention is not required once the smart


10 See Mark Fenwick and Erik P M Vermeulen, ‘The Historical Significance of Blockchain and Smart Contracts’ in Marcelo Corrales Compagnucci, Mark Fenwick and Stefan Wrbka (eds), Smart Contracts: Technological, Business and Legal Perspectives (Hart-Bloomsbury 2021) 161.


12 Melanie Swan, Blockchain: Blueprint for a New Economy (O’Reilly 2015) 16.

13 See generally, Vincenzo Morabito, Business Innovation Through Blockchain: The B3 Perspective (Springer 2017).


15 Kost de Sevres (n 11).

contract is operational (assuming some technological mechanism is agreed upon for verifying the occurrence of the triggering conditions), greatly reducing the execution and enforcement risks of the contracting process.

A significant recent development is the emergence of tools to assist in the creation of smart contracts. Several such tools have emerged that facilitate coding smart contracts and deploying them to the blockchain. For example, the website Remix offers a web page-based ‘integrated development environment’ that allows the coding (using the Solidity language), debugging, and deployment of Ethereum smart contracts on a simulated blockchain.\(^\text{17}\)

In addition, there are so-called runtime environments – for example, Truffle and Hardhat that facilitate building Ethereum smart contracts by helping developers manage and automate many of the recurring tasks that are inherent to the process of building smart contracts and decentralized applications, as well as easily introducing more efficiency around workflow by using plug-ins that offer more specialized functionality.\(^\text{18}\)

Finally, several start-ups have emerged that promise a no-code or low-code, modular construction of smart contracts and applications via a simple user interface.\(^\text{19}\) Although these low-code services are still at an early stage of development, they point to a future in which a detailed knowledge of coding – eg, the Solidity language used in Ethereum smart contracts – is unnecessary.

Here, we would like to focus on a slightly different issue. When a smart contract automates the performance of a contract, how can the parties be sure that the code of the smart contract is faithful to their intent and meets their expectations? How can the parties know what the smart contract says? One might think of this as the human aspect of smart contracts, and this is where design enters the picture.

Until recently, this human aspect of smart contracts has received less attention than the technical or legal aspects of such coded agreements. As noted by Karen EC Levy in her article ‘Book-Smart, Not Street-Smart’, contracts are deeply social tools, and they need to integrate the social world and human behaviour.\(^\text{20}\) Smart contracts can be viewed as communicative artefacts that intend to convey information for a purpose; they are not just technical or legal tools but also social tools that impact and are impacted by human behaviour.\(^\text{21}\) A balance needs to be found between the different stakeholders’ often conflicting needs and goals. Design practices and methods can be used to make the parties’ intentions and expectations tangible and visible and build a more user-friendly interface that facilitates greater transparency and understanding. Design methods can also bridge the language and other gaps that exist between different disciplines,\(^\text{22}\) help develop prototypes of contract generators that can handle text, images and code as their input and output,\(^\text{23}\) and pave the way towards the development of new tools for contracting.

---

\(^\text{19}\) See, eg, Bubble <https://bubble.io/> accessed 16 August 2021.
\(^\text{21}\) See eg, Arianna Rossi and Helena Haapio, ‘Proactive Legal Design for Health Data Sharing Based on Smart Contracts’ in Marcello Corrales Compagnucci, Mark Fenwick and Stefan Wrbka S (eds), Smart Contracts: Technological, Business and Legal Perspectives (Hart-Bloomsbury 2021).
\(^\text{22}\) Christopher D Clack has proposed language design to bridge the disciplines of law and computer science, as well as to prevent the potential for human error during conversion from agreement to code, making a large part of the code ‘valid by design’. See Christopher D Clack, ‘Languages for Smart and Computable Contracts’ (8 April 2021) <https://arxiv.org/abs/2104.03764> accessed 3 August 2021.
Simplification and visualization seem particularly important in this regard. User interface design is obviously at the centre of integrating design into technology-based solutions aimed at improving transparency and understanding. Supplementing text with explanatory diagrams and navigation tools makes it easier for everyone to find the information they need, understand the information they find, and use and act upon it. With the help of design, smart contract builders can create better interfaces and output, supplementing text or code with layers of summaries, explanatory diagrams, examples, plain language translations, audio, or video. Merging technology with design helps switch between the different versions of the same content and facilitates the making of better contracts.

Of particular significance are UI layer tools for smart contracts. A smart contract is code, and it needs a UI layer, in real-world applications, most simply in the form of a web page or app. The emergence of tools and libraries has hugely simplified the process of building a front end of this type. For architects, interaction designers, and software engineers, design patterns and pattern libraries are a standard way to share and develop reusable solutions to commonly occurring problems. Borrowing this concept and tools from these more established design-oriented professions seems effective in the context of contracts in general and smart contracts in particular.

As regards smart contracts, React is a JavaScript library for building user interfaces. Simple views for each state in the application can be set up and React will efficiently update and render the right components when the data changes. There is also a growing design pattern library for smart contract code – GitHub – which contains numerous examples or templates of code for various types of contractual terms. These tools are particularly important for smart contracts, where the choices the builders and users make can have a huge impact on both the processes and systems as well as their outputs. The emergence of start-ups offering no- and low-code smart contract services mentioned above are also evidence of this trend. Reducing friction through the incorporation of design principles and methods is central to the service that such firms are looking to offer.

Although smart contracts can be applied in multiple situations, they are still in a nascent phase. Some of the most successful current applications are in the field of Fintech and cryptocurrency-driven gaming. However, as the adoption of decentralized blockchain-based technologies increases, the need for well-designed interfaces and outputs also becomes more pressing. As smart contracts become more sophisticated, so too must the tools and patterns used in their development.


26 Corrales, Fenwick and Haapio (n 8) 7.

27 See Fenwick and Vermeulen (n 10).
platforms and applications accelerate – along with the associated smart contracts that power these platforms and applications – they will become increasingly capable of handling a much more diverse range of business transactions.

Nevertheless, several challenges remain before smart contracts can be deployed on a large scale. Some of these challenges are purely technological relating to the security and stability of the blockchain platforms – most obviously Ethereum – on which smart contracts run. Here we mention some other challenges where design may offer a potential solution.

First, if a contract is expressed in code, then the understanding of that code by the parties becomes crucially important. Parties to an agreement (particularly non-experts, such as consumers) may not be bound by those terms which they do not understand or there may not be sufficient mutual understanding of the terms of the agreement to form a contract at all. This seems to be a particularly pressing concern in the B2C deployment of smart contracts but is a more general issue. Therefore, a key challenge in the widespread adoption of smart contracts is that parties may need to rely on a third party to either capture the parties’ agreement in code or confirm that code written by a third party is accurate. This can be compared to the hiring of a lawyer to explain the legalese of paper-based contracts.

A more sceptical view would be to argue that while non-lawyers can typically understand key provisions, especially those setting forth business terms, a non-programmer would not understand any aspect of even the most basic smart contract. Consequently, parties relying on smart contracts may need to draft a separate ‘term sheet’ stating what and how the smart contract should do, which could be provided to the programmer. The parties may also want a representation from the programmer that the code will perform as stated. In the absence of widely used templates, the parties might want to enter into a separate agreement with the smart contract programmer.

Second, many smart contracts require information that is not stored on the blockchain itself – so-called ‘off-chain’ resources. For example, imagine an insurance contract that pays out if a flight is delayed by more than two hours and that information is not on the blockchain. In such a case, the parties will need a so-called ‘oracle’. Oracles are trusted third parties that gather off-chain information and then ‘push’ that information to the blockchain. In the previous example, the oracle would monitor flight delays, add information about any such delay to the blockchain, and the contract would then automatically pay-out. Oracles are a solution to utilising off-chain resources, but this process adds another party with whom the parties may need to contract. This can detract from the benefits of smart contracts and creates additional risks. For example, an oracle might experience a system flaw and be unable to push out the necessary information, provide erroneous data, or go out of business. Smart contracts will need to account for and mitigate these additional risks.

Third, smart contracts inevitably raise complex legal issues in related fields of information law, most obviously data protection and privacy. Moreover, given the diverse range of use cases of smart contracts in areas like company law (eg, decentralized autonomous organizations) and financial law (decentralized finance), these collateral legal effects are further complicated, raising the need for greater coordination and harmonization between fields of law that might in the past have operated more independently. In addition, the transaction costs of designing smart contracts are significantly increased by the need to incorporate more fields of law.

In short, smart contracts have the potential to revolutionize contracting, particularly in commercial settings (both B2B and B2C), and current contract law seems robust enough to deal with many of the issues surrounding the enforceability of such agreements. However, there are still some important issues that need to be resolved. In the following, we want to suggest that resolving these difficulties will require a more seamlessly integrated or inter-disciplinary approach to such contracts. Addressing the many challenges created by smart contracts requires
going beyond any one disciplinary perspective or frame of reference. This is the point at which design can bring a fresh and important perspective to contracts, whether smart or otherwise, most obviously by highlighting the human side of things, building powerful interfaces, and facilitating the integration of different approaches aiming at the same goal: better contracts and better outcomes.

### 2.2 AI-powered contract reading and writing; towards new contract genres

AI has the potential to impact the contracting process and its outcomes in many ways. It can disrupt the way contracts are drafted, negotiated, reviewed, and performed, and it has in fact already done so. In her 2018 Harvard Business Review article, Beverly Rich described how AI contracting software can change the tools and processes of contracting and increase the productivity and efficiency of developing, reviewing, and executing contracts.

A good example of AI developments offering great promise for the contract space is GPT-3, Open AI’s Generative Pre-trained Transformer 3. Released in May 2020, it is a state-of-the-art autoregressive language model which uses deep learning to produce human-like text with a capacity of 175 billion machine learning parameters. This is ten times more powerful than any previous non-sparse language model. GPT-3 can be used to generate summaries, explanations, language translations, and answer questions. In addition to text, it can also generate images and code. It is claimed that the quality of the text generated by GPT-3 is so high that it can be difficult to distinguish from text written by a human.

---


GPT-3 has impressed people with its ability to generate prose as well as poetry. Elon Musk is one of the cofounders of OpenAI and GPT-3 created several poems about him. One of them attracted much social media attention in the AI community for including a reference to a 2018 tweet that landed him in trouble with the U.S. Securities and Exchange Commission (SEC), resulting in his exit as Tesla chairman and a $10 million fine.35 The process of generating these kinds of poems can take several attempts of deleting and retrying lines that match and rhyme.36

While general GPT-3 AI technology is not specifically trained on contracts or legal language, it can still be useful in this context. Owing to its ability to gather and analyze vast amounts of information, the model can quickly learn to identify problematic or missing clauses, flag risk and suggest edits and redlines to mitigate those risks.37 The model is also able to translate legalese to simpler text,38 offering many new opportunities, including reducing deliberation costs during contract negotiation. It remains to be seen how far AI technology can go. It may not be long until we have access to machines that can automate the reading and writing of contracts.

As regards reading, Yonathan A Arbel and Samuel Becher have written an eye-opening article on what they call ‘smart readers.’39 These are AI-powered tools that can read, analyze, and assess contracts, disclosures, and privacy policies and are capable of simplifying, personalizing, constructing, and even benchmarking them. Smart readers can read complex text and explain the contents in plain language to people who do not speak legalese or are unfamiliar with the law. Using a smart reader, anyone can take her phone, scan any contractual clause, and click ‘explain.’ The smart reader will then respond by providing a summary that is succinct, personalized, and direct.40 In their article, Arbel and Becher offer several examples of contract clauses with summaries and explanations produced by GPT-3.

One of the authors of this paper has been playing with the idea of taking GPT-3 generated summaries and explanations and merging them with the layering approach presented by Robert Waller in ‘Designing contracts for human readers.’41 Figure 1 shows the result: a sample clause extract from a contract displaying three different layers for different reader needs: (1) the action

---

36 Ibid.
37 For examples of commercial AI software tools already available for contracting, see note 4.
layer for quick skim reading; (2) the explanation layer for a deeper understanding, containing clear text written from the user’s perspective; and (3) the full text.\(^\text{42}\)

<table>
<thead>
<tr>
<th>Action Layer</th>
<th>Explanation Layer (GPT-3)</th>
<th>Full Text (Original)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check the car carefully!</td>
<td>The dealership is not responsible for anything that happens to the car. They don’t have to fix anything on the car, and they don’t have to do anything for you if there is a problem with the car. They are not going to do anything to make sure the car is okay or safe.</td>
<td>Notwithstanding the above, the Buyer hereby attests that Buyer is buying Vehicle as-is condition and the Dealership makes no further warranties or representations regarding the condition of Vehicle.</td>
</tr>
</tbody>
</table>

Figure 1 Contracting for buying a car ‘as is’: a sample clause from a GPT-3-powered new contract genre consisting of three layers. © Helena Haapio. Used with permission.

In the example, the buyer might not know what it means to buy a car ‘as is’. But the buyer would not need to get a smart reader: with the help of GPT-3, the writer-designer could take the existing original text and create a first draft of a layered layout, a summary, and an explanation, and make these available in the contract itself. The reader can then skim read the document’s action layer and, if the reader is interested in reading more, the explanation and the full text are available as well. The layering could be done with hyperlinks, but as shown in our example, layering can also be done in a pdf or MS Word paper document. And given a prompt and the full text, GPT-3 can quickly provide first drafts for the shorter text versions. In our example, there is always a ‘human in the loop’, checking the drafts and deciding, for example, what role each layer should play in contract interpretation. As GPT-3 learns, the human writer may not have much to adjust in later rounds.

In this way, computable language models such as GPT-3 can potentially empower contract readers and writers alike. Up until now, (re)designing contract content has been time-consuming and has required extensive manual work. With the help of new AI tools, such as GPT-3, the process can be streamlined and at least the preparation of the first draft text can be partially automated. And not only text: GPT-3 can help with adding a code layer, too, making appropriate parts of the contract machine-readable and smart.\(^\text{43}\) GPT-3’s ability to translate

\(^{42}\) The text in the explanation layer is prepared using Open AI’s GPT-3 and is shown here with the kind permission of Yonathan A Arbel and Samuel Becher, the authors of ‘Contracts in the Age of Smart Readers’ (n 39). For the layered information architecture, see Waller (n 41). For more on the layering design pattern in contracts, see World Commerce and Contracting WorldCC, Stefania Passera and Helena Haapio, ‘Layering’ (Contract Design Pattern Library) <https://contract-design.worldcc.com/library/layering> accessed 4 August 2021.

\(^{43}\) See, eg, Vincent Tabora, ‘GPT-3 and Code Generation – AI-enabled Instant Software Development’ (Medium, 12 January 2021) <https://becominghuman.ai/gpt-3-and-code-generation-ai-enabled-instant-software-development-270795077cbd> accessed 16 August 2021 – An early example of a layered approach that includes code can be found in Creative Commons licenses, Their license layers are the Legal Code, the Commons Deed (also known as the ‘human readable’ version of the license), and the Code layer. ‘In order to make it easy for the Web to know when a work is available under a Creative Commons license, we provide a “machine readable”
language into code opens up new possibilities beyond outputting text or code, but also in terms of generating explanatory diagrams, layered layouts, and other design patterns.

An example of the use of GPT-3 empowering lay people as readers and writers is offered by Augrented, a start-up helping the renters of apartment deal with a variety of situations, including eviction prevention and writing letters to a landlord for negotiating a rent discount. According to their website, their tool will help people understand if they might qualify for rental assistance funds and then help write, for example, a confidential offer of settlement to the landlord. ‘Thanks to OpenAI’s GPT-3 and our proprietary legal notice processing technology, everyone can write a professional, formal letter to their landlord in minutes. … To explain why you need to request a discount, you only have to enter a few words and our AI will write a formal explanation for you.’ Augrented makes it clear that the app does not replace the need for an attorney in some situations, seeking to steer away from alleged unauthorized practice of law.

GPT-3 is not the only pre-trained language model that is currently available. And these language models are not just for laypeople, of course. It is highly unlikely that they will replace lawyers or contracts experts anytime soon, but they can certainly help these professionals, too, in the process of contract drafting and analysis. For example, tools offered by companies such as LegalSifter, Akorda, LawGeex, and Maigon prove the point, helping to review contracts faster and identify frequently missed clauses, and Skritswap promises to simplify contracts and other complex documents into plain and clear language, with the help of augmented intelligence.

Computable language models bring many benefits, but they also introduce new concerns and risks, for example contractual bias and discrimination. A study on consumer contracts published by Noam Kolt showed several potential pitfalls that may arise in the use of GPT-3. The case study utilized a question set relating to the terms of service of popular US websites. The questions related to issues such as eligibility to access the services, payment for the services, limitations of liability, intellectual property rights, and dispute resolution procedures. While the model can successfully exploit subtle informational cues embedded in questions thanks to the vast training data, the study revealed that the model is also fragile in unexpected ways. It performed poorly on consumer contractual provisions, implying a possible anti-

---

49 LegalSifter (n 4).
50 Akorda (n 4).
51 LawGeex (n 4).
52 Maigon (n 4). This Swedish company’s contract review applications, according to their website, support more than 55 languages, including English, Ukrainian, Russian, and others.
54 See Arbel and Becher (n 39).
consumer bias. Performance was also very sensitive to the wording of questions, but staggeringly indifferent to the subtleties and nuances of contractual language.55

This case study highlights the importance of awareness of potential biases on AI and mitigating them from the outset to prevent misleading advice. Many other types of biases may be present not only in the algorithm but also in the data used to train the algorithm. This could be compared to the study published in Science in October 2019,56 where researchers found significant racial bias in the AI decision-making software used by U.S. hospitals. The algorithm used healthcare costs data accrued in one year instead of illness data as a proxy for health needs. The U.S. healthcare system, however, spent less money on minorities and the data revealed that the average black person was significantly more likely to exhibit poor health than the average white person, especially in chronic health conditions such as diabetes, anemia, kidney failure and high blood pressure.57

Regarding GPT-3 technology, ‘contextual bias’ should also be mentioned. Contextual bias or ‘context effects’ occur when ‘well-intentioned experts are vulnerable to making erroneous decisions by extraneous influences.’58 Objectivity is hindered as the extraneous influences that can cause the examiner to subconsciously develop expectations about the outcome of an examination. This bias occurs due to the natural human tendency to see what is expected, rather than what is presented.59

This is where merging the expertise and competencies of lawyers, designers and technologists can play an important role. Based on an awareness of the potential prejudices, these professionals can act as an interface to monitor and translate the technical and legal terms generated by AI systems. Working together, using design tools and methods, they can aid in the implementation of more flexible and user-friendly automated contracts, finding the best criteria for their users and adding clarity and efficiency in negotiations. They can also steer or nudge organizations and users – contractual parties and people representing them in negotiation and implementation – to make better decisions regarding the content of any agreement.

Despite the limitations, we envision a future where computable language models such as GPT-3 will enable and empower both contract readers and writers, allowing writers to provide their readers – whether humans or machines – with a new genre of operationally and legally functional and understandable contracts.

The recent technological developments discussed here, as well as in other chapters of this volume – from text-only legal rules to multimodal tools and code-based and AI-powered contract readers and generators – offer exciting new opportunities for the practice of contracting. This potential can be enhanced by exploiting the synergy effects of multiple technologies together to produce contracts that work for both people and machines. We envisage a future where many contracts will be digitalized, modular, interoperable across platforms, and built from library-stored components that utilize no- or low-code functionality. When design enters the picture, it creates a framework with principles and methods that help respond to the human aspects of contracts and contracting tools.

3 Future lawyers and the contracting ecosystem of the future

In various ways, technology is reconfiguring the practice of contracting and the contracting community, and these changes are challenging traditional thinking and role of contracts and legal professionals. This does not mean, however, that these traditional professionals will cease to be relevant. Crucially, they will continue to play a key role in establishing trust in business transactions and relationships. Well-designed contracts will still need to establish confidence in the validity of the transaction and successfully realize the benefits of the deal for the contracting parties. Important legal matters, such as the truth about ownership and control, the transfer of property, and the allocation of risk and control, will still need to be covered in a contract. Deal-making, matchmaking, gatekeeping, and enforcing roles will increasingly be performed with the assistance of technology, but lawyers will remain centrally involved in all these tasks as well as in their automation. In this section, we first review the role of lawyers in the contracting ecosystem of the future and then speculate as to how new roles will emerge to support building the computable contracts of the future.

3.1 Lawyers as engineers, facilitating transactions and relationships

Historically, lawyers have been useful in a business context when they have operated as ‘transaction engineers’ that facilitate and sustain new types of transactions or other economic relationships. This engineer function can be contrasted with what might be termed a compliance function that focuses on mitigating legal risk. Both functions overlap, and both are necessary for a business context, but recognizing the different goals involved (making a deal happen as opposed to reducing legal exposure) and ensuring an appropriate balance between these two roles is vital for any discussion of the future of the legal profession – not least, because it highlights the different audiences that each of these two functions target. While compliance solutions are primarily for other lawyers, regulators, and judges, the transaction engineer is speaking to the users of contracts, namely the parties to the transaction and the people representing them, including the people in charge of implementing those contracts. This shift becomes important when considering the legal solution: traditional, legalistic solutions may be less appropriate when compliance is not the primary focus or goal.

The development and growth of Silicon Valley as a hub of digital technologies in the early 1970s, for example, is often presented as an example of lawyers successfully performing their role as transaction engineers in a way that added genuine value. Although the model of clustering similar (in this case, technology) businesses was innovative, there is a consensus that lawyers were crucial in both facilitating technology firms and in sustaining the ecosystem that emerged around these new businesses. For example, lawyers drafted the contractual provisions that provided appropriate protection for high-risk investors – ie, angel investors and early venture capitalists – from the various risks associated with investing in start-up companies engaged in blue-sky projects. Moreover, lawyers were involved in various other activities, such as deal-making, matchmaking, and gatekeeping, to support and protect founders in starting and scaling their new businesses. Finally, various new contractual mechanisms

---

60 David Howarth, Law as Engineering. Thinking about What Lawyers Do (Edward Elgar Publishing 2013) 67; Mark Fenwick and Erik P M Vermeulen, ‘The Lawyer of the Future as Transaction Engineer: Digital Technologies and Disruption of the Legal Profession’ in Marcelo Corrales, Mark Fenwick and Helena Haapio (eds), Legal Tech, Smart Contracts and Blockchain (Springer 2019) 253.


62 Ibid.
helped reduce information asymmetries between the entrepreneurs and investors and, as such, were necessary to bring the demand and supply-side of venture capital together in a way that was effective and advantageous to all sides.

From the Silicon Valley example, we can see how lawyers functioned as effective transaction engineers, ie, important intermediaries that brought together in a relatively safe environment multiple parties with different but mutually compatible interests. The contribution of local law firms helps explain, in some part, the success of Silicon Valley.63

The problem, however, is that lawyers often fail to perform this function of being proactive transaction engineers that add value. Instead, lawyers became a hindrance to such creative transactions and deal-making. This happens for many reasons, but the habit of lawyers to proceduralize solutions and employ standard form templates that are excessively complicated and irrelevant is a significant factor. Over the last three decades, lawyers have developed a reputation as one of the least trusted professions.64 The complaints are familiar: lawyers are verbose, they don’t listen, they are unresponsive, they charge too much, they don’t care about clients, they spend too much time on trivial issues, they don’t keep clients informed, they constantly ‘over-lawyer,’ and they don’t communicate clearly or concisely.65

Of course, lawyers would justify much of the above-listed behaviour because it is necessary to manage the multi-layered legal risk that all businesses – and particularly younger businesses – now confront. A process of net-widening (more regulation) has greatly impacted business and created powerful incentives to invest in the management of that risk – what has come to be known as compliance. However, in a highly competitive global economy, lawyers need to focus on their function in assisting parties in facilitating interactions and reducing costs. There are multiple costs in a modern business environment that need to be cut, most obviously agency costs and transaction costs. In addition to cost-cutting, the scope for lawyers to add value to a growing business is enormous.66

This leads us to the thought that the future lawyer will need to operate as a relationship engineer. Moreover, ‘[l]ike engineers, transactional and legislative lawyers want to make something useful that works for their clients.’67 Here, we prefer to speak of relationship engineers – in the contracting community, ‘transactional’ may be understood to describe low-value, commoditized work – in contrast to the idea of relational contracts, which is seen as being more strategic and value-adding.68

3.2 Lawyers as project team members working for proactive contract design

Crucially, developments in technology benefit from the integration of design thinking. Many contracts and legal professionals will do things in ‘designerly’ ways, with an ‘emphasis on

63 ibid.
65 Fenwick and Vermeulen, ‘The Lawyer of the Future as Transaction Engineer’ (n 60) 268. ibid 268–69.
66 ibid 268–69. 67 Howarth (n 60) 67.
communication, experimentation, and making things visible and tangible.\textsuperscript{69} It is no longer just about knowing the law, it is also about communicating, capturing agreements and objectives, and making the meaning of contracts and law transparent and visible. Future lawyers need knowledge, skills, and tools to help others understand and act upon their advice. They need to be prepared for contracts and law in action, not just in books.\textsuperscript{70}

The digital transformation does not mean the end of lawyers, not least because of the complex regulatory environment and associated legal risk that all firms must now navigate. Nevertheless, we do think that if the lawyers of the future are to add value as they did in the Silicon Valley example, they need to operate as effective relationship engineers and that the legal profession is going to need to adapt to this rapidly evolving operating environment.\textsuperscript{71}

Most obviously, lawyers of the future will need to be able to assume the role of project team members or leaders in complex multi-disciplinary teams that create and implement the new solutions of the future. A capacity to operate in such a complex environment will be vital, as communicating with a more diverse set of actors is increasingly required. In the digital world, lawyers will have to work closely with traditional partners such as accountants or financial service advisors and engineers, designers, architects, and other experts.

For proactive contract designers, the contract is not the goal: its successful implementation is the goal.\textsuperscript{72} Still, many organizations and lawyers representing them adopt a transactional rather than relational approach, focusing on deal-making and getting to contract rather than implementation. In the words of Danny Ertel, ‘[a]n organization that embraces the deal maker approach … tends to structure its business development teams in a way that drives an ever-growing stream of new deals. … But they also become detached from implementation and are likely to focus more on the agreement than on its business impact.’\textsuperscript{73}

Regarding contract design, we want to highlight the importance of implementation and successful relationships, not just deal-making or transactions. That is why we speak of relationship engineers. For the future lawyer, the main objective is not to get a contract made or a deal signed. As Ertel notes, ‘[t]o be successful, negotiators must recognize that signing a contract is just the beginning of the process of creating value.’\textsuperscript{74}

In pursuing these new solutions, some lawyers will work with established organizations while others will be confronted with a very different type of client. Fast-growth technology companies with limited assets and a small number of employees are central to the digital world. The most successful companies have used networked digital technologies to create new business models. Trust and value are created through platforms and networks instead of by the management of workers or assets. The lawyer of the future will need to understand the opportunities and challenges and help firms to re-invent their governance structures to be more open and inclusive. They will need to work with business and subject matter experts to assess when to focus on the deal and when on the relationship and ensure that the most appropriate contracting model – whether relational or transactional – is selected for each situation. Furthermore, they will need to choose the best fitting contract type and model, ensure that their


\textsuperscript{70} Helena Haapio, ‘Visualisation in Contract Education and Practice: The First 25 Years’ in Emily Allbon and Amanda Perry-Kessaris (eds), Design and Visualisation in Legal Education (Routledge forthcoming)

\textsuperscript{71} Fenwick and Vermeulen, ‘The Lawyer of the Future as Transaction Engineer’ (n 60) 269.


\textsuperscript{73} ibid 62.

\textsuperscript{74} ibid 62.
contract terms are fit for purpose, functional and understandable by the people negotiating, signing, and implementing those contracts.75

In this new model, corporate legal departments and law firms will need to operate as dynamic and flexible legal platforms, connecting legal and other experts and managing the collaboration, transactions, and relationships. Matchmaking and project-based temporary partnerships will mean that legal advisors will need to be aware of the way network technology and other code-based technologies operate. After all, many of the solutions that the lawyer of the future will be expected to help design will be technology-based and dependent on computer code.

A better understanding of technology will be crucial for the lawyer of the future to perform this relationship engineer function effectively. More and more businesses and industries will revolve around code-based products or services, so facilitating transactions and relationships – being an active and effective relationship engineer – will inevitably involve coding to some degree. Such coding will require multi-disciplinary teams working in collaboration, and the capacity of lawyers to participate in these teams actively will be crucial to their success as legal professionals.

Finally, the future lawyer will need to acquire literacy across multiple fields and, crucially, the capacity to plan and structure relationships based on this polycentric skillset. There must be a shift from passively using templates to actively creating customized solutions based on the current ‘state of the art’ and best practices. In this context, traditional legal skills need re-evaluating and reconstructing. There needs to be a move away from the lawyer as a ‘sage’ with authoritative knowledge of the mystical ‘texts’ of law to a vision of the lawyer as partner, problem-solver, and designer. The future lawyer will take on a key role in translating and mediating between multiple partners – a crucial interface at the ‘edges’ in building and managing the complex communities necessary for meaningful co-creation.

This account of the future lawyer seems particularly pertinent in a volume on contract design, especially if we understand contract design widely, as a team activity: writing (or coding) and designing operationally functional and legally compliant contracts that work for their makers and users. This involves engineering the framework within which deals, transactions, and relationships are carried out, and, crucially, both business and legal objectives must be achieved and balanced in this effort. This project involves leveraging technology in designing and building contracting systems or platforms. We need mechanisms to generate greater cross-disciplinary ‘buy in’ where lawyers contribute to the seamless interfacing of human (design principles) and the machine (digital technologies) in the promotion of organizational and user interests.

3.3 The emergence of new professionals: computable contract designers

In thinking about the future, it is possible to imagine the emergence of a new and crucial role, which John Cummins has referred to as a ‘computable contract designer.’76 To understand this new role (or that of an automated computable contract builder), an analogy can be made with the function and responsibility of a web-site creator, in the sense that the task of web-site design has evolved into a multi-disciplinary task that integrates design with the use of diverse technological tools (both software and libraries) that facilitate this work.77 In the same way that

---

75 For ‘formal relational contracting’, see Vitasek, Winn and Nickel (n 68) and Frydlinger and others (n 68); for aligning expectations and interests and architecting the deal points, see Frydlinger and others (n 68) 213-37.
77 ibid 28: ‘What was once the domain of the computer programmer is now the domain of the designer.’
no one builds a web page from scratch anymore but instead uses a diverse range of tools and services to expedite the process, the computable contract designer of the future will employ no- or low-code software solutions and libraries to design and build a customized digital contract relevant to a specific situation and the needs of specific users. Following John Cummins’ line of thought, we anticipate that

in the not-too-distant future, several start-ups (or even more established technology companies) will emerge that successfully integrate the concepts associated with website building with those of automated document assembly systems to create a wholly new concept: automated computable contract builders...

These computable contract builders will have access to libraries of data objects eg, clauses with metadata or maybe a ‘dispute resolution object’ that could be used for certain dispute types. Just as a website can have an embedded e-commerce functionality, computable contracts may have one or more embedded dispute resolution functionalities.78

In addition to dispute resolution functionalities, we would also anticipate dispute prevention and ‘de-escalation’ oriented functionalities as well as relational and possibility-driven promotive clauses and functionalities. In any case, we expect the digital contracts of the future to include a broader range of code-powered functionalities than those operating on a blockchain (ie, smart contracts).79

Contracts intended for both human and computer readers have been given different names to describe what is the same phenomenon. Hazard and Haapio, for example, refer to them as ‘wise contracts’; 80 while Cummins and Clack, along with other researchers at the UCL Financial Computing and Analytics research group, have used the term computable contracts.81 This vision involves the transformation of contracts from static documents created in a natural language form and with only limited direct linkage to business operations to contracts readily understood by humans and computers and with a significantly greater level of accessibility, usability and digital functionality. Cummins and Clack predict this transformation will facilitate improvements in productivity and lower costs ‘through reducing contract drafting times, increasing the scope for the automation of contract execution, and speeding up (or even avoiding) dispute resolution.’82 Again, the analogy is with web pages and web page design, in the sense that these computable contracts are code-based, as well as easily accessible and understood.

For the moment, these developments are still at an early stage. And yet, there are important shifts occurring in certain sectors of the contract ecosystem. For example, there are a number of developments in the insurance and reinsurance industries around computable contracts, in

---

78 ibid 29.
79 This tracks developments in a public law context, where the discussion focuses on ‘rules as code.’ See James Mohun and Alex Roberts, ‘Cracking the Code: Rulemaking for Humans and Machines’ (OECD Working Papers on Public Governance, no. 42, OECD Publishing 2020) <https://doi.org/10.1787/3afe6ba5-en> accessed 4 August 2021. Much of the impetus for this idea of ‘code is law’ can be traced back to the work of Larry Lessig, see Larry Lessig, Code, and Other Laws of Cyberspace (Harvard University Press 2006).
80 Hazard and Haapio (n 23).
82 See Cummins and Clack (n 23) 3.
which insurers seek to design digital contracts based on a modular approach; ensure interoperability across platforms and systems; design and build repositories (or libraries) for contract components, and create (no-code) contract building applications. It seems likely that other industries will follow this model, and future lawyers will be needed in various new roles to partner with the computable contract designer of the future. Equipped with new tools such as those discussed in Section 2 it may not be long until next generation contract designers provides us with access to machines that make it easy to automate the generation of user-friendly contracts that contain explanatory diagrams, layered layouts and other design patterns that work for both human and machine readers.

4 Conclusion

Contracts are not just technical or legal tools; they are also social tools that impact and are impacted by human behaviour. They are communication tools that need to be designed with their users in mind, and not just drafted to meet the requirements and expectations of the law and lawyers. The development and deployment of technology-driven solutions across contracting do not make lawyers obsolete. Future lawyers – understood as active relationship engineers – need to be able to navigate the operational, technical, and legal issues involved in modern contracting, but also the design of human-machine and human-contract interactions and successful interfacing between all stakeholders. Working as relationship engineers, the lawyers of the future will benefit from learning to do things in ‘designerly’ ways, with an emphasis on communication and making the meaning of contracts and law transparent and actionable.

Since many of these new contracting technologies are code-based or code-generated, lawyers need to be able to work with technologists, code, and data to participate in a meaningful way in the construction and design of next-gen contracts as well as the on-going development of the technological infrastructure that supports their creation and execution. Although many of these technologies are currently in a nascent phase, they will become increasingly relevant and capable of successfully handling a more diverse range of business transactions and relationships.

New contract genres and AI-powered solutions, such as the recent GPT-3 developments, raise important new opportunities and challenges for the contracting community and the legal profession. New skills are required to develop user-friendly systems that best capture the true intent of contracting organizations and – crucially – deliver a smooth and meaningful experience to the users. The emergence of new professionals – what we have called the computable contract designer of the future – will emerge to help in this task. The article has suggested that combining the worlds of law and technology with design represents the best way forward. The result? A better future in which business, legal, technology, and design professionals work together in creating the legally and operationally functional and user-friendly contracts of tomorrow.

5 Bibliography

– ‘Computable Contracts’ (University College London) <https://www.ucl.ac.uk/computer-science/research/research-groups/financial-computing-and-analytics/computable-contracts> accessed 4 August 2021
Akorda <www.akorda.com> accessed 14 July 2021

83 See, eg, Cummins (n 76).


Augrented <www.augrented.com> accessed 4 August 2021


Bubble <https://bubble.io/> accessed 16 August 2021


Corrales M, Fenwick M and Haapio H (eds), *Legal Tech, Smart Contracts and Blockchain* (Springer 2019)

Corrales M, Fenwick M and Wrubka S (eds), *Smart Contracts: Technological, Buisness and Legal Perspectives* (Hart-Bloomsbury 2021)


Fenwick M and Vermeulen EPM, ‘The Historical Significance of Blockchain and Smart Contracts’ in Corrales Compagnucci M, Fenwick M and Wrak S (eds), Smart Contracts: Technological, Business and Legal Perspectives (Hart-Bloomsbury 2021)

Fenwick M and Vermeulen EPM, ‘The Lawyer of the Future as Transaction Engineer: Digital Technologies and Disruption of the Legal Profession’ in Corrales M, Fenwick M and Haapio H (eds), Legal Tech, Smart Contracts and Blockchain (Springer 2019)

Fink M, Blockchain Regulation and Governance in Europe (Cambridge University Press 2019)


Haapio H, Next Generation Contracts: A Paradigm Shift (Lexpert 2013)


Hardhat <https://hardhat.org/> accessed 16 August 2021


Howarth D, Law as Engineering. Thinking about What Lawyers Do (Edward Elgar Publishing 2013)


Kronblad C, Digital Transformation of the Legal Field - A Bubble in Trouble (Doctoral thesis Chalmers University of Technology 2021)


LegalSifter <www.legalsifter.com> accessed 14 July 2021

Legalzoom <https://www.legalzoom.com> accessed 14 July 2021


Lessig L, Code, and Other Laws of Cyberspace (Harvard University Press 2006)

Levy KEC, ‘Book-Smart, Not Street-Smart: Blockchain-Based Smart Contracts and the Social Workings of Law’ (2017) 3 Engaging Science, Technology, and Society 1


Morabito V, Business Innovation Through Blockchain: The B3 Perspective (Springer 2017)


OpenAI <https://openai.com/about/> accessed 4 August 2021


Remix <https://remix.ethereum.org/> accessed 16 August 2021


Rossi A and Haapio H, ‘Proactive Legal Design for Health Data Sharing Based on Smart Contracts’ in Corrales Compagnucci M, Fenwick M and Wrba S (eds), *Smart Contracts: Technological, Business and Legal Perspectives* (Hart-Bloomsbury 2021)


Swan M, *Blockchain: Blueprint for a New Economy* (O’Reilly 2015)


Tefula M [@michaeltefula], ‘Just taught GPT-3 how to turn legalese into simple plain English. All I gave it were 2 examples [Exploding head] Might build a term sheet and investment document interpreter out of this [Nerd Face]’ (Twitter, 21 July 2020) <https://twitter.com/michaeltefula/status/1285505897108832257?lang=en> accessed 9 August 2021


Tidwell J, Designing Interfaces (2nd edn, O'Reilly Media 2014)

Truffle <https://www.trufflesuite.com/> accessed 16 August 2021


Waisberg N and Hudek A, AI for Lawyers: How Artificial Intelligence is Adding Value, Amplifying Expertise and Transforming Careers (Wiley 2021)

Waller R, ‘Designing Contracts for Human Readers’ in Marcelo Corrales Compagnucci, Helena Haapio and Mark Fenwick (eds), Research Handbook on Contract Design (Edward Elgar 2022)


