A Computational Approach to Competition Impact Assessment

Emerson S. Bañez[*]

July 04, 2024

Abstract

Competition does not take place in a vacuum but is embedded within an existing legal and regulatory environment. Competition authorities are thus encouraged to evaluate existing laws to identify and remediate competition effects. To this end, organizations such as the OECD and the World Bank have released guidance on the conduct of competition impact assessments. Despite the importance and complexities of a competition impact assessment, the literature is sparse when it comes to implementation specifics. From selection of laws to be reviewed to the actual assessment of legal provisions, much is left to the subjective evaluation of assessors. This could mean that errors could compound in the course of analysis and lead to implausible results. For example: 1. There are no parameters for law selection that aligns with market definitions; 2. There is no consistent, granular unit of analysis; 3. There is a lack of provable basis for attributing specific competition effects to legal texts. The work aims to apply techniques of computational law to these problems. First: The work will encode the norms applicable to a specific sector (digital payments market in the Philippines) into forms that are amenable to computational processing – such as modelling market entities and interactions into a knowledge graph, and the normative constraints into inference rules. Second,

[*]Assistant Professor, University of the Philippines College of Law; LL.D. Student, Kyushu University Graduate School of Law.

these representations can then be subjected to automated reasoning in order to provide insights useful to competition analysis, such as: determination of other relevant laws, evaluation for consistency and compliance, and proving of specific competition effects.

Keywords: Computational Law, Competition, Impact Assessment, Artificial Intelligence

1 Introduction

1.1 Competition and the Legal Environment

Competition does not take place in a vacuum but is embedded within an existing legal and regulatory environment.[1] Barriers to entry and exit (which can cause failure to produce a competitive market) can be due not just to the structural features of a market, or the behavior of its actors, but also the policy environment maintained by the government.[2] The law can affect competition in a number of ways: It can openly favor some players, providing them with tax exemptions, and subsidies. It can also put other players at a disadvantage, by making it more expensive for them to operate in the industry (through barriers to entry and exit). In both of the above cases, the law works explicitly in limiting competition through advantages and constraints directly addressed to industry players. The enactment of a competition policy, specifically, anti-trust law, is explicitly directed at the competitive behavior of firms, and is designed to restrain monopoly and maintain market competition.

However, even the general legal environment outside of competition law can also work against competition through more subtle mechanisms. The law can control the flow of information between and amongst buyers and sellers, constraining their strategic choices. More importantly, the law can allow the state itself, with its size, economic power, and monopoly on regulatory powers, to be a direct player (as a buyer or seller) in any industry.[3]

[1] This evokes the vision of law not as a neutral, static stage, but as Laurence Tribe describes it - one possessed of a curvature, a shape, that can affect the movement of the actors on it. See Generally Laurence H. Tribe. "The Curvature of Constitutional Space: What Lawyers Can Learn from Modern Physics". In: *Harvard Law Review* (Nov. 1989), pp. 1–68.

[2] See Erlinda M Medalla. "Understanding the New Philippine Competition Act". In: *Philippine Institute for Development Studies (PIDS) Discussion Paper Series* (No. 2017-14 2017), pp. 1–24. URL: `http://hdl.handle.net/10419/173591` (visited on 01/08/2024), at 5.

[3] In some cases, the general legal environment is just us important as the competition law in the maintenance of a competitive market. See generally Iftekhar Hasan and Matej Marinč. "Should

1.2 Competition Impact Assessments

The law's impact on competition underscores the need for detailed studies on how the current legal and regulatory backdrop affects competition. This can be performed through a **competition impact assessment** of the laws that operate in specific sectors of economic activity. A competition impact assessment refers to the review of existing or proposed policies in order to determine their impact on competition.[4] This is with the view to formulating alternative policies that are more conducive to competition. The underlying logic is that while governments may pursue important policy goals through legislation - there are multiple pathways to these goals, and governments should pursue those paths that least impact competition. This in turn springs from the premise that more competition is beneficial, especially for the consumers.[5]

The Philippine Competition Commission has already conducted several such assessments of selected laws - either at its own instance or upon request by Congress or regulatory agencies. It has also worked with organizations such as the OECD, which has performed competitive impact assessments of certain economic sectors.[6] The OECD and other orga-

Competition Policy in Banking Be Amended during Crises? Lessons from the EU". in: *European Journal of Law and Economics* 42.2 (Oct. 2016), pp. 295–324. ISSN: 0929-1261, 1572-9990. DOI: 10.1007/s10657-013-9391-2. URL: http://link.springer.com/10.1007/s10657-013-9391-2 (visited on 01/08/2024), which suggests that competition policy in the financial sector can be inconsistent in times of crisis. Financial regulators, through prudential standards, bear the greater responsibility in ensuring against concentration. To the extent that this overlaps with market structure concerns of competition authorities, greater coordination is required. See also Tomaso Duso, Jo Seldeslachts, and Florian Szücs. "The Impact of Competition Policy Enforcement on the Functioning of EU Energy Markets". In: *The Energy Journal* 40.5 (Sept. 2019), pp. 97–120. ISSN: 0195-6574, 1944-9089. DOI: 10.5547/01956574.40.5.tdus. URL: http://journals.sagepub.com/doi/10.5547/01956574.40.5.tdus (visited on 01/08/2024), Competition policy may have significant impacts, but only to the lightly regulated sectors. On the other hand, highly-regulated firms are less likely to respond to competition policy.

[4] See OECD. *Competition Assessment Toolkit - Volume 1 (Principles)*. 2019. URL: https://www.oecd.org/daf/competition/46193173.pdf (visited on 10/10/2023), for Part 1 of the Organization for Economic Cooperation and Development's (OECD) 3-part guidelines for competition impact assessment. This and subsequent volumes will be referred to collectively as the "OECD Guidelines".

[5] Ibid., at 7.

[6] See for example OECD. *Competition Assessment Reviews: Logistics Sector in the Philippines*. 2020. URL: https://www.oecd.org/daf/competition/oecd-competition-assessment-reviews-philippines-2020.pdf (visited on 10/10/2023); See also OECD. *Competitive Neutrality Reviews: Small-Package Delivery Services in the Philippines*. 2020. URL: https://www.oecd.org/daf/competition/oecd-competitive-neutrality-reviews-philippines-2020.pdf (visited on

nizations interested in advocating for competition policy have also issued guidelines for the conduct of competition impact assessments.[7] The Philippine Competition Commission currently has unpublished draft guidelines[8] that it uses to guide its competition assessment exercises. The PCC Guidelines disclose that it is based on the OECD Guidelines as well as the World Bank's Markets and Competition policy Assessment Toolkit.[9] the full documentation of which is not publicly available. To the extent that these guidelines and instances of their implementation converge into common methodology, these guidelines will be idealized into a "canonical approach" to competition impact assessment, and represented by the OECD Guidelines as the focus of analysis.

1.3 The Canonical Approach to Competition Impact Assessment

The canonical approach to conducting competition assessments starts with identifying laws that are relevant to a sector, then proceeds to evaluating such laws for competition effects. As elaborated in the OECD Guidelines, the process of competition impact assessment involves the following steps:[10]

1. **Identify the laws to be assessed** - This can be straightforward in the case of assessing new or pending legislation or regulation. On the other hand, for situations where the impact of laws on an entire economic sector is required, discretion is involved in defining the boundaries of what will be reviewed. This is expected to result in a list of "relevant laws".

2. **Apply threshold tests** - The list of relevant laws can be narrowed down through a threshold test. This is based on a checklist of questions designed to identify potential

10/10/2023).

[7]See OECD Guidelines, supra. See also the International Competition Network's (ICN) recommended practices. Subsequently referred to as the "ICN Guidelines" ICN Advocacy Working Group. *Recommended Practices on Competition Assessments.* International Competition Network, 2014. URL: `https://www.internationalcompetitionnetwork.org/wp-content/uploads/2018/07/AWG_RP_English.pdf` (visited on 01/10/2024).

[8]"PCC Guidelines", on file with the author.

[9]See The World Bank. *Markets and Competition Policy.* World Bank. URL: `https://www.worldbank.org/en/topic/competition-policy` (visited on 01/16/2024), Subsequently, "the World Bank Assessment Toolkit".

[10]The enumerated steps are from OECD. *Competition Assessment Toolkit - Volume 3 (Operations Manual).* 2019. URL: `https://web-archive.oecd.org/2020-01-22/370055-COMP_Toolkit_Vol.3_ENG_2019.pdf` (visited on 10/10/2023), at 14-15.

restrictions to competition. This will result in a smaller set of flagged laws that can be subject to a more detailed review.

3. **Detailed review of flagged laws** - Performing a more detailed review to determine whether or not there are "actual and significant" restrictions on competition. Those with such restrictions form a set of "critical laws" for which the next stage of the process should be applied.

4. **Generate alternatives** - For those critical laws where restrictions are found, identify alternative measures that can achieve policy objectives while being less restrictive or competition.

5. **Selecting the best option** - Once policy alternatives have been generated, a judgment must be determined as to the "best" option. Once the "best" option has been identified, legislation must be drafted and passed that will implement this policy recommendation.

6. **Ex-post assessment** - Review and monitoring of the impacts of the law implementing the selected policy alternative.

The canonical approach requires a search methodology to enumerate the laws that can apply to the actors and transactions in a given market. The documentation assumes that the government will select or prioritize a sector to be assessed. The guidelines suggests a number of prioritization principles to aid in this determination, such as: 1. Selecting sectors with high economic impact (in terms of share of GDP, consumer expenditure, employment); 2. Or those have been the subject of frequent complaints or interventions; 3. The constraints of time, financial resources, and the availability of technical talent.[11] For the purpose of this work, it will be assumed that selection and prioritization can proceed independently, prior to the methodology to be outlined in this work. Aside from simplifying the scope of the work, the assumption is compatible with the notion that selection and prioritization of the sector is a matter of policy, to be made by accountable, human institutions.

Once a sector has been selected, the next step is to compile legislation that is relevant to the sector. This, in turn, is predicated on delineating a conceptual boundary for the sector. The guideline acknowledges that a boundary-setting exercise can be challenging. To

[11] OECD, *Competition Assessment Toolkit - Volume 3 (Operations Manual)*, at 18-19.

provide some structure into this exercise, the guidelines provide some suggestions on how to proceed: 1. Focusing on legislation relevant to one ministry. Using the correlation with the ministry concerned as a proxy for a relevant boundary, however, simply restates the problem especially where the ministry has a broad mandate. It can also risk missing laws that require inter-agency coordination; 2. Focusing on standard definitions - This can be done by referring to standard industry classifications, such as the United Nation's International Standard Industrial Classification of All Economic Activities[12], or the Statistical Classification of Economic Activities in the European Union[13]. The guideline, however, notes that these classification systems will often segregate economic activities in ways that are counter to both intuition as well as grounded knowledge as to how industries are actually run.

Assuming that the boundary of a market sector can be defined for purposes of the finding relevant laws - this process may still yield numerous laws for any modern regulatory environment.For this, the canonical approach suggests a process for filtering relevant laws in order to arrive at a set of critical laws.[14]

1.4 Problems with the Canonical Approach

Despite the importance of evaluating the competition impact of the legal environment, the methodological toolset for impact assessment has fallen behind (in terms of sophistication and rigor) those used in other areas of competition policy, such as: merger control, assessment of anticompetitive agreements and abuse of dominance.[15] According to the OECD Guidelines, Step 3 and the associated competition checklist lie at the heart of the competition impact assessment process. Despite the importance assigned to this section of the process, both the OECD Guidelines and the literature on competition impact assessment

[12]United Nations. *International Standard Industrial Classification of All Economic Activities (ISIC)*. Revision 4. United Nations, 2008. ISBN: 978-92-1-161518-0. URL: https://unstats.un.org/unsd/publication/seriesm/seriesm_4rev4e.pdf.

[13]European Commission. *Statistical Classification of Economic Activities in the European Union*. Rev. 2. 2008. URL: https://ec.europa.eu/eurostat/documents/3859598/5902521/KS-RA-07-015-EN.PDF (visited on 05/13/2024).

[14]OECD, *Competition Assessment Toolkit - Volume 3 (Operations Manual)*, at 17.

[15]See Nicole Robins and Hannes Geldof. "Ex Post Assessment of the Impact of State Aid on Competition". In: *European State Aid Law Quarterly* 17.4 (2018), pp. 494–508. ISSN: 16195272, 21908184. DOI: 10.21552/estal/2018/4/6. URL: http://estal.lexxion.eu/article/ESTAL/2018/4/6 (visited on 01/03/2024), which proposes a greater role for financial and economic analysis in evaluating the impact of state action on competition, at 494-495.

do not provide a detailed, rigorous, and consistent methodology for performing this step. The OECD Guidelines provide a checklist of questions that can be used to identify potential restrictions to competition. However, the OECD Guidelines do not provide a detailed methodology for applying the checklist. Much is left to the subjective evaluation of assessors. This could mean that errors could compound in the course of analysis and lead to implausible results. This rise to the following problems:

Law selection Assuming that the economic sector has been selected and its conceptual boundaries have been delineated, the assessor is expected to derive from this model "an exhaustive list of laws and regulations that influence the economic activities that take place in each of the sectors under examination".[16] There is no elaboration as to how the conceptual mapping in the previous step can translate into a search strategy that can be documented, refined, and shared. There are no parameters for law selection that aligns with market definitions. The literature require that laws relevant to a market be subject to competition analysis. Although there are well-established methods for defining a market, criteria for selecting laws that will be subject to analysis are not aligned with these market definitions. Assessors are likely to under- or over-select the laws.

Unit of analysis No consistent, granular unit of analysis. Competition authorities may look at individual laws and analyze these for competition impact. However, a statute may not be the appropriate unit of analysis, since the competition impacts operate through key provisions that work with other critical provisions found in other laws. Looking at more atomic levels of distinct rules within provisions can also enable more detailed forms of analysis.

Lack of proof Lack of provable, measurable basis for correlating textual provision with an anti-competitive effect. Even if a law is properly selected and studied at the appropriate level of description, the actual evaluation of competition impact is described on an intuitive, sometimes *ad hoc* basis. It is not encoded in a way that can be reliably communicated, proved, and further analyzed.

[16] The Guidelines acknowledge that this stage of the process is not trivial, since ensuring the inclusion of all relevant legislation requires casting a broad net. The guideline suggests an iterative process, which requires not just reference to texts initially found in electronic databases (as well as the laws they refer to, such as implementing rules), but also through consultation with stakeholders. OECD, *Competition Assessment Toolkit - Volume 3 (Operations Manual)*, at 21.

1.5 Specific Problems with Law Search and Selection

The first step in the canonical approach to competition impact assessment is to find the laws that are applicable to a sector. Given an industry or market sector under consideration - What law "covers" an i dustry or a market with all its actors and transactions? The goal is to arrive at either a complete or a heuristic but consistent mapping between actors (and their actions relevant to a market) - and the laws that would cover these actors and actions.

Risk of under-inclusion In evaluating a legal environment, assessors may overlook critical legal standards pertinent to the industry, due to a disconnect in the terms used by the industry and those employed in legal contexts. This phenomenon is particularly evident in emerging digital finance businesses. These entities often operate under novel designations or through distinct modalities, such as exchanges and applications. Consequently, there is a prevailing misconception that such entities fall outside the purview of traditional financial regulations. This overlooks the fact that, despite their innovative approaches, these entities perform analogous functions and are subject to similar risks as their conventional counterparts.

On the other hand, focusing on a single law nominated by a stakeholder may result in losing many critical signals. This is because competitive issues such as barriers to entry, disproportionate costs, and preferential treatment can arise not from a single law but from the interaction of key provisions spread across several legislative enactments.

The OECD framework cautions that "When performing this exercise, it is important to remember that, in addition to sector-specific regulation, there also exists horizontal, cross-sectoral, legislation (such as planning restrictions or environmental standards) that may have a considerable impact on the economic activities performed in that sector and may be a cause of additional competition restrictions."[17]

Risk of over-inclusion If we take a connected view of the law, all of the law can be relevant to a particular market. Every law has the potential to change the relative rights and obligations of parties involved in economic activity and thus result in some competitive impact. Considered this way, even laws that apply to everyone and only incidentally touch economic activities in a sector - can be interrogated for potential competitive impact, no

[17]See OECD, *Competition Assessment Toolkit - Volume 1 (Principles).*

matter how small or contingent. The Philippine Civil Code, for example, gives a preferential status for unsecured debt evidenced by a notarized instrument, over those that are embodied in a private document. Family and tax law provides rights and privileges that are accessible only to married heterosexuals. Nevertheless, these laws should usually not be the subject of competitive impact analysis. As they apply to everyone, in a large enough population their application to specific individuals can appear random and evenly distributed - any *de minimis* competitive impact is contingent and cancel each other out. More importantly, these laws do not directly relate to the sector under consideration, and their ultimate effects on the actors in the sector will only be coincidental.[18]

A lawyer's theoretical training and experience, and openness to economic thinking can perform the required analysis while compensating for these limitations. However, that lawyer may not always be available. It is also possible that the limited pool of legal talent will not be able to scale to the demands of extensive, industry-wide competition analysis, which could involve hundreds (if not thousands) of laws and regulations, all of which could interact with each other. Abstracting the problem into a computational form can allow parts of the analysis to be done by non-lawyers (i.e., the staff of a competition authority), or even by computers. The goal is not to supplant the human component of competition impact analysis but to augment it.

1.6 Computational Law in Aid of Competition Impact Assessment

These problems relating to scale, rigor, consistency, and predictability may be the appropriate setting for the application of computational law, which can look at legal rules as discrete units that can be evaluated. In this light, the question of competition impact can be structured as a computational problem. This work aims to apply computational techniques to: 1. The selection of laws for competition impact assessment, based on their

[18] "Plaintiffs stress that the LMRDA is a remedial measure and seek a liberal construction. This maxim is useless in deciding concrete cases. Every statute is remedial in the sense that it alters the law or favors one group over another... But after we determine that a law favors some group, the question becomes: How much does it favor them? Knowing that a law is remedial does not tell a court how far to go. Every statute has a stopping point, beyond which, Congress concluded, the costs of doing more are excessive — or beyond which the interest groups opposed to the law were able to block further progress." - Richard Stomper, et al., Plaintiffs-appellees, v. Amalgamated Transit Union, Local 241, Defendant-appellant, 27 F.3d 316 (7th Cir. 1994)

relevance to a market; 2. The representation of legal rules into discrete computable units; 3. The automated analysis and evaluation of legal rules for their competition impact. It hopes to introduce improvements to the problem of search and prioritization of laws: That is, the process of making an exhaustive mapping of concepts in a market in order to identify relevant laws, as well as applying the threshold tests in a consistent and rigorous manner.

Automated reasoning can also allow competition authorities and policy makers to make extensive evaluations efficiently and at scale. A responsive competition assessment system should not only evaluate retroactively, for existing laws, but also conduct the exercise for new or proposed legislation. While it is possible for each new law to carry not only prospective effects in a particular subject matter - it is also possible for it to interact with the existing legal environment in a way that would change the competition impact of prior laws. A new baseline understanding of the competition impact of the entire legal environment may have to be inspected with the passage of each new law, compounding the complexity of the task and adding to the burden of competition authorities. Part of this regression analysis can be automated through the computational approach.

It should be noted that the primary concern of this paper is in finding the relevant laws as well as evaluating them for competition effects into a computable problem. The work will explore formalizations for representing the above problems in a way that can be processed by computers. The emphasis is in developing tools for the facilitation of competition impact assessment. This assumes that an appropriate body is responsible for conducting such assessment. This and other features of a competition assessment regime, such as the location of the assessment in the larger policy development process, the involvement of the competition agency, will not be within the scope of this work. Although Computational Law is usually associated with automation of legal determinations though a computer - it is not the goal of this work to implement an automated counterpart for the sections of the competitive impact assessment process that will be encoded into a computational form. Some experimental code might be featured in order to demonstrate the feasibility of some proposals, but these are not production-quality implementations. It should be noted that the advantage of the computational approach goes beyond machine execution of routine legal tasks, but in helping develop notations through which we can understand and share problems of legal reasoning.

1.7 Next steps

Finding a constrained version of the problem The goal of the proposed work is to use the computational approach for competition analysis. Particularly, for the stages of legal search, selection, and threshold testing. The scope of the study is further limited to applications for the special case of the Philippine digital payments sector - where both the assumptions, definitions, and constraints are more explicit.

The study can be limited to well-defined standards in competition - i.e. those that are already extensively documented and tested in the economics literature, and so can be a source of explicit rules on how a competitive market ought to behave. As to the laws that will be evaluated, the plan is to focus on a segment that is already digitized and subject to very specific constraints. The digital payments sector is a good candidate. Out of necessity, the sector does not involve many entities and transactions with open-ended states. It is also a field characterized by extensive, semantically rich constraints from industry standards, government regulations, user contracts, and the functionality of the digital platforms themselves.

Miscellaneous considerations : The problem of competition impact analysis has the same shape as other bulk analysis problems of the Law Center, such as: 1. gap analysis; 2. impact analysis; 3. compliance analysis. They all involve some form of legal comparison and evaluation - old law against new law, n-level law versus n-1 level law, etc. So an advance in the solution of one problem can contribute to the other.

Why is all the effort towards abstraction preferable to the usual intuitive approach? In addition to gving us scale and automation, the computational approach can help us in two ways: 1. It can help us make our analysis more rigorous, and 2. It can help us make our analysis more transparent. Sharing legal knowledge through a formalized notation can help us build richer systems of legal knowledge.

It should be noted that regardless of the formalization, inference rules, defeasible deontic logic is already embedded in the practice. Just as a baker can make a cake without knowing the chemistry of baking, we can make legal determinations without knowing the formalisms of logic. But just as knowing the chemistry of baking can help us make better cakes, knowing the formalisms of logic can help us make better legal determinations.

Despite the initial wariness about the costs and consequences of large language models, their growing sophistication is compelling. Recent literature suggests that knowledge graphs can embedded into large language models, making the latter more efficient, more attuned to "ground truth", and therefore more reliable. Since both argumentation frameworks and proposition networks can be framed as extensions of the information contained in knowledge graphs, it may be possible to combine these approaches as well.

2 An Overview of Computational Law

2.1 Historical background

The general concern of applying computational techniques to the legal domain - e.g. encoding law into computational terms, and mechanically applying or analyzing these - was among the earliest directions of artificial intelligence research. Despite its early promise, however, the approach did not bear fruit.[19] There may have been some initial optimism (circa 1980's) about the prospect of computers performing automated legal reasoning. Grossman summarizes the research and programming activity towards this end. They note that while computers cannot replace lawyers, these machines can, in time nevertheless run "legal reasoning systems" that can assist attorneys.[20] Computerized legal reasoning offered speed, reliability, and the ability to carry out numerous,repetitive tasks. It could also provide a consistent application of the law.[21] It was also hoped that the availability of such systems could have knock-on effects on legal reasoning itself, molding the thought processes of legal professionals towards logical rigor, and force the field to be more explicit about its assumptions.[22]

2.2 Modern approaches

According to Genesereth:"Computational law is that branch of *legal informatics* concerned with codification of regulations in precise computable form."[23] In terms of practical applications - it can provide the basis for computer systems performing compliance checks, legal planning, the analysis of regulations, and related functions. Many computer applica-

[19]See Michael Genesereth and Nathaniel Love. "Computational Law". In: *Proceedings of the 10th International Conference on Artificial Intelligence and Law - ICAIL '05.* The 10th International Conference. Bologna, Italy: ACM Press, 2005. ISBN: 978-1-59593-081-1. DOI: 10.1145/1165485.1165517. URL: http://portal.acm.org/citation.cfm?doid=1165485.1165517 (visited on 09/18/2021), at 205.

[20]Garry S Grossman and Lewis D Solomon. "Computers and Legal Reasoning". In: *ABA Journal* 69 (1983), pp. 66–70, at 66: "Primarily, a legal reasoning system would serve as a repository of knowledge, outlining the general parameters of the law. In lieu of searching through a treatise or similar task, given a specific factual situation, the system could be relied on to present only the relevant law."

[21]Ibid.

[22]Ibid.

[23]See Michael Genesereth. *Computational Law: The Cop in the Back Seat.* CodeX: The Center for Legal Informatics Stanford University. 2015. URL: https://law.stanford.edu/publications/computational-law-the-cop-in-the-backseat/ (visited on 09/18/2021), at 2.

tions aid lawyers in their tasks, but these are not within the ambit of the term. Examples include legal databases to find the law, and office productivity suites to help the practitioner prepare briefs. In both cases, the legal reasoning is still performed by the human agent. The computer performs symbolic analysis for purposes of retrieval and presentation of data, without any recognition of the rules as such.[24]From a pragmatic perspective, Computational Law is important as the basis for computer systems capable of doing legal calculations, such as compliance checking, legal planning, regulatory analysis, and so forth".[25] Despite the recency of the term, its goal is shared by early projects in artificial intelligence, which saw the legal domain as a natural site for the application of computational techniques.[26]

The prevalent approach to meeting these goals has two components: 1. First, representing law (and surrounding facts) into a formal logical form and 2. Second, the ability to process those representations to assist in legal determinations. This means: That it would be possible through computational techniques, to arrive at consistent, correct (or at least plausible) legal conclusions from given set of premises and operations.

These determinations can be descriptive, recreating in computational form the law as it is, and guiding its users in evaluating whether certain actions or states of the world are in accordance with the encoded rules. It can also be prescriptive, meaning the rules

[24]See Genesereth, *Computational Law: The Cop in the Back Seat*, at 2-3 for a proposed example. According to Genesereth the Turbo Tax program is a computational law application. The user supplies values, and the program makes computations of the user's tax obligation. When prompted, it can explain its results by making references to the applicable tax law. Legal rules (whether or not taxable, the base, rate, and tax due) are encoded (however indirectly) as code, and the result of the processing is a legal determination - whether or not tax is due, and how much. But see Hans Andersson. "Computational Law: Law That Works Like Software". CodeX – The Stanford Center for Legal Informatics, Feb. 10, 2014. URL: `https://www.academia.edu/9286857/Computational_Law_Anderrson_and_Lee`, at 3-4. There is a tendency to invoke a Turing test analogue for computational law system: "Any system whose users inputting, through whatever interface such system might present, a legal query to obtain a legal response would find themselves unable, given only the response, to determine whether a legal professional...had provided the system's response." Andersson rejects this criteria because it would include systems that only outwardly appear to be computational law without actually solving its fundamental problems. Based on his rejection of the Turing or "imitation" principle of what constitutes computational law, Andersson argues that Love and Genesereth's inclusion of Turbo Tax within the definition is inaccurate. Although the program appears to replicate the behavior of a tax professional - it does not formally represent laws, or performs automated reasoning based on those representations.

[25]Genesereth, *Computational Law: The Cop in the Back Seat*, at 2.

[26]Genesereth and Love, "Computational Law", at 205.

as encoded can be analyzed and evaluated against standards (such as efficiency), or their alignment with other rules, in order to arrive at more suitable rules.[27]

Computational law's philosophy contrasts with the notion of Legal Realism. In its stronger formulation, legal realism means that the text of the law doesn't matter, or at least does not matter as much as other considerations, in order to perform a balancing of interests (usually based on factors extraneous to law) on a case-by-case basis. Computational law may not be able provide this kind of normative flexibility.[28] Instead, it is more closely aligned with Legal Formalism. Thus it carries the notion that laws are definitive, and exhaustively account for all the normative calculations of the legislator.

Given its alignment and limitation, Genesereth suggests that computational law is most relevant to civil law jurisdictions - where the text of the law are interpreted literally or with very constrained space for interpretation . In contrast, it is least relevant in common law jurisdictions marked with on-the-fly legal innovation through judicial interpretation.[29] Although computational law has limits when applied to cases that require analogical or inductive reasoning (which often characterizes the reasoning in judge-made Laws), Genesereth suggests that the judicial process itself can generate categorical constraints from vaguely worded statutes. Judicial law can be a source of encoded rules. [30] Even in common law jurisdictions, however, there are categorical, codified statutes that may not be subject to significant judicial discretion. Examples include legislation on data privacy, securities, enterprise management, construction, electronic commerce, taxation. There is a growing tendency in these fields of law to move toward greater textual specification and codification. This makes them more amenable to the computational approach.[31]

[27]Andersson, "Computational Law: Law That Works Like Software", at 7.

[28]Genesereth, *Computational Law: The Cop in the Back Seat*, at 5.

[29]Ibid.

[30]Ibid. at 6

[31]To a certain extent, the end goal of the adjudicatory process is to come up with categorical interpretations of existing statutes. One can consider rules expressed in judicial decisions as expressed in judicial decisions as extensions of the legal text, and encode them computationally, as if they were part of the original statute. So to the extent that statutes are considered vague in a common law jurisdiction, judicial decisions can supplement them by coming up with interpretations which can be encoded. Genesereth is also convinced that as Computational Law becomes more useful, legislators and regulators will be encouraged to have more such categorical laws Genesereth, *Computational Law: The Cop in the Back Seat*, at 6.

2.3 Computational Law Examples

Let's imagine a business with the following configuration of employees and offices:[32]

John manages Ken	John is in office 22	John is male
John manages Kat	Kat is in office 24	Jill is female
Jill manages Mark	Ken is in office 22	Ken is male
Jill manages Mike	Kat is in office 24	Kat is female
		Mary is female
		Mike is male

Logic and programming constructs allow us to: First - use of variable to represent an arbitrary number of entities (X, Y, Z for employees and offices). Second - use of logical operators to express relationships between any of the above (not, and, or, if-then). These "representational extensions" allow us to define new relations in terms of existing relations:

> If X is in office Z and Y is in office Z and X and Y are *distinct*, then X is an *officemate* of Y.

In addition to merely describing entities and their relationships, we can encode rules and regulations through the use of these programmatic tools. We can ascribe the attribute of illegality to some facts or relationships:

> If X manages Y and X is an officemate of Y, then that is *illegal*.

Within a given set of facts (entities and relationships) and rules (deontic assertions) it may be possible to derive other conclusions:[33] These patterns of reasoning are called "inference rules" or rules of inference. Iterative use of inferential reasoning can generate all logical conclusion (facts and rules) from within a given set of premises (facts and rules).

Example of inferential rules discovery and compliance check:

- John is in office 22

[32] The following examples are from Genesereth, *Computational Law: The Cop in the Back Seat*, at pp. 3-5.

[33] According to Genesereth, 2015: "...by matching facts and conclusions of rules to the conditions of other rules and asserting their conclusions"

- Ken is in office 22
- John is an officemate of Ken
- John manages Ken and John is an officemate of Ken
- John is not Ken
- That is illegal

We can invert this reasoning working backwards and arranging facts to avoid illegalities

- John is in office 22
- Jill is in office 24
- Ken is in office 22
- Kat is in office 24

The inference rule discovery process can be extended to look for inconsistencies within a set of regulations. For example: We might require every project to have managers and subordinates, and no manage have a subordinate who is also an officemate. This might be inconsistent with a subsequent rule requiring special projects personnel be housed in a common work room. Compliance checking (through automated legal reasoning) can feed legal planning and regulatory analysis.

2.4 Applications of Computational Law

Based on the above definitions of computational law, and depending on the horizon of techological development considered, possible computational law implementations fall into two major categories:[34]

1. **Specific computational law** - Such as simply confirming the presence of necessary elements of a cause of action, as in a checklist.
2. **General computational law** - Capable of making nuanced determinations if presented with a complex fact pattern within a specific (or even several) legal regimes.

[34] Andersson, "Computational Law: Law That Works Like Software", at 6.

Genesereth sees potential in embedding computational law applications into software that supports workflows that are subject to legal, regulatory requirements - e-commerce, data privacy, etc. Genesereth points to Project Calc (A Stanford CodeX project under Harry Surden), which integrates into CAD software used by architects, routines for checking compliance with rules such as: building codes, environmental rules, accessibility laws.[35]

We can embed computational law applications in devices such as cellphones, car dashboards, smart glasses so that they can provide legal guidance at the point of decision. For example: An app that not only identifies the flower the picture of which you took, but also informs you that you should not pick it up. Compared to simply publishing an overwhelming mass of laws (often in a language inscrutable to the public) digitally-mediated legal determinations can help make the notice requirement of due process more meaningful.[36] For automobiles (whether manned and unmanned), in addition to basic functions such as navigation and collision avoidance, the system can help compliance with legal requirements such as: a. speed limits; b. whether or not a street is one way c. whether u-turns are allowed; d. what areas allow parking.[37]

The availability of rule detection and automated legal analysis can enable legal self-help - actors structuring/planning their activities (especially electronic transactions) to be legally valid/compliant. Similar to word processors reducing reliance on typesetters.[38] Logical representations can make it possible to derive common baseline rules, or discover bridging rules (or the exact points of divergence). This will then make it easier to have analyze cross-border contracts, or do comparative legal analysis.

Love and Genesereth maintains that such systems and self-help only extends to reducing transaction costs for legal compliance and does not mean that parties can appear *pro-se* in instances of conflict. The forum of computational law is within enterprises, and not courts.[39]

Other applications include: Enterprise-wide monitoring and automated compliance; simulation of impact of rule changes; automated rule changes based on specified end goals.

[35] Genesereth, *Computational Law: The Cop in the Back Seat*, at 6-7.
[36] Ibid. at 8.
[37] Ibid.
[38] Ibid.
[39] Genesereth and Love, "Computational Law", at 206.

Finally, there is a practical argument for the creation and use of computational systems. There are already many existing systems that we can leverage. We have computational systems with data on business transactions and code that embody business rules everywhere - for banking, and enterprise management. Computational law just extends this tendency by encoding public instead of private rules.[40]

2.5 Relationship with Current AI Implementations

Computational Law appears to overlap with artificial intelligence in terms of function. Explicitly encoding the rules of law and legal reasoning can be considered an application of declarative artificial intelligence, an approach to AI that focuses on the representation of knowledge and reasoning. Recent developments show great promise from the connectionist approach to AI, which focuses on the use of neural networks and deep learning. The latter approach has been used to develop large language models (LLMs) such as GPT, which can perform a variety of tasks, including generate text that reads like plausible legal reasoning. Given this state of affairs, will it not be better to develop LLMs to perform the tasks of legal analysis and evaluation? In some ways, the declarative approach is similar to the training of a machine learning model to perform the same function. The difference is that the former is a more explicit, transparent, and controllable process. The latter is more opaque, and the results are less predictable. Thus, for mission-critical domains like law, there is merit in explicitly encoding rules over LLMs and deep learning for the following reasons:

No ground truth In an ordinary conversation or task, humans rely on an internal theory of the world, a theory of mind for the entities that it is dealing with. We have a phenomenology. We are still not sure of this can be done for AI's. Certainly this is not what happens with LLMs. These models do not have ground truth. Without the capacity for ground truth, LLM's can spiral into delusions - making their applications unsafe for mission-critical applications.

Prohibitive costs LLM's are expensive - These systems are expensive to build, train, and maintain, even on a per-query basis once everything is set up. These are also expensive to retrain. If a Large Language Model gets "poisoned" by malicious input - what are the ways to mitigate it? How can one "nudge back" if the mechanism is hard to trace and hard

[40]Genesereth, *Computational Law: The Cop in the Back Seat*, at 7.

to explain. Fixing it will require a lot of resources and the solution might not be durable (It could also affect the accuracy and the responsiveness of the model)

3 Law as a Computable Structure

3.1 Computability of the Law

3.1.1 The nature of computability

The premise of computational law is that once we have both rigorous formal representations of law, and the appropriate logical methods to analyze them, law becomes computable. What is meant by a computable approach, or the computability of legal determinations? The formal meaning of a problem or a domain's computability relates to whether or not it can be solved through an algorithm. In other words, a problem is computable if there exists a step-by-step procedure (algorithm) that can be executed by a computer to solve the problem.[41]

Computability also means that once we have abstracted enough of the most important attributes of a thing into formalized model - we can map its behavior backward and forwards in time. We can access powerful shortcuts to the things behavior - to diagnose, analyze, and predict.[42] The modern world we have was achieved through computation - from bridges to bombs to games and deep space probes. These are possible because we could build models of the forces of nature, and predict their interactions through logic and math.

3.1.2 Can the law be computable?

Wolfram argues that the computability of law can flow from the computational character of nature, from which all phenomena (including humans and human institutions such as law are derived).[43]: The universe itself is built on a computational foundation, and our

[41]The notion of computability is derived from Alan Turing's description of problems that are amenable to an algorithmic solution (to be carried out by a computational model such as a Turing Machine). Alan M. Turing. "On Computable Numbers, with an Application to the Entscheidungsproblem". In: *Proceedings of the London Mathematical Society* 2.42 (1937), pp. 230–265. DOI: `10.1112/plms/s2-42.1.230`.

[42]Without computability, we are confined to recording descriptions of phenomena, and we are limited in our ability to draw insights and make predictions abut a system. Similar to the state of astronomy before Newton developed the formalisms of calculus - without a proper computational model for celestial mechanics, all that could be done was observation and recording.

[43]See Stephen Wolfram. *How to Think Computationally about AI, the Universe and Everything.* Stephen Wolfram Writings. Oct. 27, 2023. URL: `https://writings.stephenwolfram.com/2023/10/how-to-think-computationally-about-ai-the-universe-and-everything/` (visited on 12/14/2023).

current computational tools for representing and analyzing knowledge is is the latest (and perhaps ultimate) in a series of formalisms for representing and understanding reality.[44] It is not necessary to get into such a fundamental claim. As will be argued below - it should be enough that the computational approach capture what is essential of legal knowledge.

Of course, modeling the forces acting on a physical system is one thing, but trying to model the behavior of people and institutions under the constraint of law is a different category. Complexity and incompleteness conspire against us. The open-textured nature of legal concepts like "justice" means that our representations and analytical tools can only go to certain levels of description. Even if we can somehow develop a rich enough toolset to capture legal concepts, Gödel's incompleteness means that there will always be a gap in our formalization.[45]

This difficulty does not mean that the problem will be intractable. The physicist Stephen Wolfram states that in the teeth of complexity and incompleteness, even the hard sciences are beset by oceans of non-computability. Despite all their progress in theory-making and theory-testing, scientists still have to contend with a universe that largely resists mathematical certainty. And yet, they have found enough islands of computability amidst that ocean to lay the foundations of useful things like engineering, computer science, particle physics.[46]

Our models for law will likely be incomplete and thus inaccurate. But the incompleteness of a model does not mean it will be useless. A map will never be as detailed as the territory that it guides us through, but a good map should have enough information to be useful. The utility that can be derived from the computational approach is compelling enough to warrant its pursuit. The more obvious advantages come from the speed and reliability of computers, as well as their ability to retrieve relevant legal text from memory.[47]. However,

[44] See generally Stephen Wolfram. *A Project to Find the Fundamental Theory of Physics.* Champaign, Illinois: Stephen Wolfram, LLC, 2020. 770 pp. ISBN: 978-1-57955-035-6.

[45] Gödel's theorems on the fundamental incompleteness of any axiomatic system impacts mathematics and logic, and ultimately, the capacity of computational formalism to model reality Richard P. Feynman. *Feynman Lectures on Computation.* Ed. by Anthony J. G. Hey and Robin W. Allen. Boca Raton: CRC Press, 2018. 303 pp. ISBN: 978-0-7382-0296-9, at 52.

[46] See generally Stephen Wolfram. *A New Kind of Science.* Champaign, Illinois: Wolfram Media, 2002. 1197 pp. ISBN: 978-1-57955-008-0.

[47] Grossman and Solomon, "Computers and Legal Reasoning", at 66.

some of the more fundamental advantages to the profession can be indirect: The rigorous structured approach of these systems may ”mold the thought processes of the lawyer” (and law students) into a more logical pattern, and the extended use and design of such systems will force legal scholars to confront and resolve the ambiguities of the law.[48]

A formal approach can allow smoother, more reliable collaboration and the building of higher ”towers of consequences”[49] - systems that will allow more detailed study of legal systems, as well as applications for real world problems that involve the law. For example - the ability to encode legal rules into a computer program may be the key to encoding firm, normative(”constitutional”) limits on artificial intelligence.[50]

3.2 Law and logic

Computational law requires some role for logic in legal reasoning. A significant goal of computational law is the production of a computer system capable of producing legal advice (as opposed to just textual information). This can only be possible if logic has a place in law. Because if anything, a computer system's only actual capability is demonstrating a logical system. Similarly, only a logical system can be computerized.

In legal theory (as well as AI research into the law domain), the formalized aspects of legal reasoning is divided into two principal approaches:[51]

First, the logical approach - where legal decision (e.g. the judge's justification) are thought to arrive through a formalized deductive process. Deductive reasoning draws conclusions from a set of general principles or premises that are given or established.

Second, the dialectic (or argument theory) approach, which views legal justification as arising from an adversarial process, where parties use discretion to evaluate between reasonable alternatives.

[48] *Ibid.*

[49] Wolfram, *How to Think Computationally about AI, the Universe and Everything.*

[50] *Ibid.*

[51] Henry Prakken and Giovanni Sartor, eds. *Logical Models of Legal Argumentation.* Netherlands: Kluwer Academic Publishers, 1997. ISBN: 0-7923-4413-8. DOI: 10.1007/978-94-011-5668-4, at 1.

The logical and dialectic approaches are seen as divergent, incompatible modes of legal reasoning, and for a long time have gone on separate tracks of development and application. The logical approach was seen as a tool for the legislative process, advancing the goal of representing laws as a set of consistent statements. Meanwhile, the dialectic approach was often applied to case-based problems that characterized litigation and judicial decision making - legal justifications derived from a process of presenting and evaluating pro and contra cases.[52] Nevertheless, Advancements in both legal theory and technology may allow for the unification of the divergent approaches (of logic and dialectics). Within the case-based reasoning that defines the dialectic approach, there is acknowledgement that consistent logical rules can be formalized. Within the logic approach, on the other hand, researchers have developed models that take into consideration the incomplete and defeasible nature of legal argumentation. [53]

The foregoing analysis will cover debates covering the first approach. Much of the work in the field has emphasized the deductive approach, due to its seeming ubiquity in legal reasoning. The deductive approach is viewed as essential to legal interpretation and application: Lawyers will analyze the text, structure (and history) of a statute to determine meaning and intent. These will then serve, along with a background of other established rules, as premises for determining applicability to specific cases.[54]

3.3 Confronting objections to logic in law

Computer scientists and philosophers have made many attempts to use logical tools to represent the intricacies of legal language and legal reasoning. This stream of work is based on the assumption that logic is a component of legal reasoning.[55]

At the same time, practicing lawyers have built a conceptual moat around the field of law, to distinguish it from the hard sciences, claiming that the law, unlike these fields, will

[52] *Ibid.*
[53] *Ibid.*
[54] Jaap Hage. “A Theory of Reasoning and a Logic to Match”. In: *Artificial Intelligence and Law* 4.3-4 (1996), pp. 199–273.
[55] See generally Matthias Armgardt, Patrice Canivez, and Sandrine Chassagnard-Pinet, eds. *Past and Present Interactions in Legal Reasoning and Logic.* Vol. 7. Logic, Argumentation & Reasoning. Cham: Springer International Publishing, 2015. ISBN: 978-3-319-16020-7 978-3-319-16021-4. DOI: 10.1007/978-3-319-16021-4. URL: https://link.springer.com/10.1007/978-3-319-16021-4 (visited on 03/16/2024).

always evade a reductionist, logical approach.[56] Law - its concepts and reasoning are not amenable to computation because these are largely not computationally reducible. Since legal concepts and rules are socially constructed and in flux, they cannot be fully represented into numbers and logical constructs.

3.3.1 Epistemiological divide between law and logic

Objections to logic often point to a fundamental difference not just in method (structured, formal versus discursive and intuitive), but also to their subjects. The basis of logic was the assumption that valid argument can be based upon the elemental form of the proposition, composed of a subject and a predicate linked by a connective. Any proposition, meanwhile has a truth value - either it is true or false. There is nothing in between (the law of the excluded middle).[57]. On the other hand, legal propositions are normative rather than fact-stating, and we only have an incomplete picture of the general logic of norms.[58] Synthesizing the arguments of A.G. Guest and other legal philosophers, Summers argues that most objections of this kind is often based on a misuse of the concept of logic. Upon closer inspection, even basic logical propositions do not refer to things in nature, but concepts that may not necessarily be subject to true-or-false evaluation. [59]

Summers adds that most likely, these statements are criticisms of the reasoning in particular cases, rather than general arguments against the use of logic in legal reasoning. [60]

[56] Jeffrey Goldsworthy. "The Limits of Judicial Fidelity to Law: The Coxford Lecture". In: *Canadian Journal of Law and Jurisprudence* 24.2 (July 2011), pp. 305–325. DOI: 10.1017/S084182090000518X, "The popular impression of legal thinking is that it is logically rigorous. But legal reasoning, whether of judges, advocates or legal scholars, rarely has the clarity and rigour of the best analytical philosophy. Often this is because the subject-matter is simply incapable of being treated as rigorously. But more importantly, legal reasoning in real cases leads to practical decisions that have drastic effects on individual's lives or the welfare of the community, for which judges properly feel some moral responsibility. Consequently, legal reasoning can have a tendentiousness—an almost palpable gravitation towards a desired conclusion—that is lacking in the work of analytical philosophers, pure mathematicians or nuclear physicists."

[57] Philip Leith. "Logic, Formal Models and Legal Reasoning". In: *Jurimetrics* 24.4 (1984), pp. 334–356, at 336.

[58] Robert S. Summers. "Logic in the Law". In: *Cornell Law Faculty Publications* (Paper 1133 1963), pp. 254–258. URL: http://scholarship.law.cornell.edu/facpub/1133, at 254.

[59] *Ibid.*

[60] *Ibid.* In cases where a decision is criticized for an "abuse of logic" (e.g. Whiteley v. Chapel), what may be at fault is the choice of legal premises, and not the (logical) manner in which the judge proceeds from premise to conclusion. Or, more often enough, it may be a problem with semantics.

More directly, the objection can be met by referring to legal pluralism, i.e. the notion that there are other forms of logic that can be used to represent legal reasoning.[61] This includes, as will be discussed below, deontic and defeasible logics.

Another aspect of the divide between law and logic is related to the necessary incompleteness of formal systems. The incompleteness of formal systems is a result of Gödel's incompleteness theorems, which states that any non-trivial formal system will contain statements that are true but cannot be proven within the system. This means that there will always be gaps in any formalization of the law, and that there will always be legal questions that cannot be answered through logical deduction.[62] This is a significant challenge to the idea of computational law, as it suggests that there will always be limits to what we can achieve through logical analysis. Without an overall general model for the world, representations in a formalism will always be incomplete. Wolfram asserts, however that an overall scheme is not necessary, and that it would be possible to capture concepts as needed.[63]

3.3.2 Historical convergence of logic and law

A profession as steeped in tradition and the weight of history as law may view embedding logic as an unnecessary modernist intrusion. However, the history of law is replete with examples of the convergence of logic and law. For Aristotle, law and logic were one and the same.[64] This was carried on through the scholastic tradition, which viewed law as a system of rules which can be logically deduced from immutable principles. These principles, in turn, can be discovered by man through a process of reasoning. Great jurists such as Thomas Aquinas, William Blackstone also proceeded along these lines.[65] For the longest time,

[61] Leith, "Logic, Formal Models and Legal Reasoning", at 340.

[62] Rebecca Goldstein. *Incompleteness: The Proof and Paradox of Kurt Godel.* New York, London: Atlas Books, 2005.

[63] See Stephen Wolfram. "Computational Law, Symbolic Discourse and the AI Constitution". In: *Data-Driven Law: Data Analytics and the New Legal Services.* Ed. by Ed Walters. Red. by Jay Liebowitz. Data Analytics Applications. Boca Raton, FL: CRC Press Taylor & Francis Group, 2019, pp. 144–174. ISBN: 13: 978-1-4987-6665-4. URL: `https://writings.stephenwolfram.com/2016/10/computational-law-symbolic-discourse-and-the-ai-constitution/` (visited on 01/14/2024), "At a foundational level, computational irreducibility implies that there will always be new concepts that could be introduced...[C]omputational irreducibility implies that none of them can ever be ultimately be complete".

[64] Lee Lovevinger. "An Introduction to Legal Logic". In: *Indiana Law Journal* 27.4 (Sum. 1952), pp. 471–522, at 471, citing A Treatise on Government, or The Politics of Aristotle, Book III, c. 16, Elli's translation, 1943.

[65] *Ibid.*

logic was Aristotelian logic. One of the Aristotelian logic's central theory of the judicial syllogism, where a judicial decision is justified through a form of syllogistic reasoning, i.e. as an inference from normative and factual premises.[66]

A more modern inspiration for the use of logic in law comes from the work of Gottfried Wilhelm Leibniz. Leibniz is commonly known a a leading figure in mathematics and philosophy but before his seminal work in those fields he was a legal scholar. His work combines law and philosophy, and proceeds from the premise that some of law's fundamental questions cannot be answered without philosophical thought.[67] Leibniz's forays into philosophy and law seems to be partially motivated by his numerous attempts at reconciling church doctrines (Protestants v. Catholics), conflicts over which led to the Thirty Years war that destroyed Germany. His works on legal reasoning have only been translated and published recently, indicating that he pursued a mathematical-logic approach similar to modern ideas in computational law.

The three underlying ideas of Leibniz's legal investigations are:[68]

1. Legal research and problem solving, particularly adjudication requires an interdisciplinary dialogue. The law needs to accept ideas from other disciplines such as philosophy, logic, theology, mathematics, and physics.

2. Law also needs to have an interdisciplinary dialogue, i.e., between the various schools of legal thinking.

3. Law requires a more diverse range of reasoning methods and cognitive tools. Practitioners can select the appropriate tool based on pragmatism, i.e. their effectivity in solving legal problems.

Although Leibniz's efforts to develop a logical formalism was not successful, these ideas continue to animate the field of computational law.

[66] Pablo E. Navarro and Jorge L. Rodríguez. *Deontic Logic and Legal Systems.* New York: Cambridge University Press, Sept. 29, 2014. ISBN: 978-0-521-76739-2. DOI: 10.1017/CBO9781139032711, at ix.

[67] prenote Matthias Armgardt. "Leibniz as a Legal Scholar". In: *Fundamina* (2014), pp. 27–38, at 28-29, citing *Specimen quaestionum philsophicarum ex jure collectarum, 1664.*

[68] Ibid., at 5.

3.3.3 Practicality of embedding logic

Another, more practical line of argument is that logic has no use for judges and lawyers, since their conclusion are arrived at intuitively. The reasoning is arrived at *post hoc*. One could point, however, to those fields of law (such as real property law) that are devoid of any moral or social ideals, through which intuitive generalizations can be derived.[69] There is also the argument that we should not privilege the default ways of thinking in the law. These intuitive, psychological processes are exactly the kind we need to scrutinize with logic for possible inconsistencies.

Although not couched in formalisms of modern symbolic logic, instances of both deductive and deductive thinking are inherent in legal reasoning:

> In his selection of competing propositions and in his consideration of the propriety of subsuming a particular case under a certain general rule, a judge is not, of course, guided by logic. He is guided by insight and experience. But in his application of the proposition selected, and in his testing of its implications before he adopts it, he uses a deductive form of reasoning in order to discover its potentialities. The directive force of the principle may be exercised along the line of logical progression, and a judge must always keep in mind the effect which his decision will have on the general structure of the law.[70]

Summers criticizes that this is incomplete, i.e., that logic can play a role even in the selection of premises necessary to decide particular cases. Guest also asserts that inductive logic is not applicable to law. However, Summers points out that when lawyers advise clients they often use a form of inductive logic when they make predictions and generalizations from individual cases.[71] In a way, a lawyer already treats legal questions as a computational problem, having his own estimation function based on past data such as the history of the controversy, the applicable law, and the court's past decisions.

[69]Summers, "Logic in the Law", at 255.

[70]Anthony G. Guest. "Logic in the Law". In: *Oxford Essays in Jurisprudence*. Oxford: Oxford University Press, 1961, at 188.

[71]Summers, "Logic in the Law", at 255-256.

3.3.4 The challenge of legal realism

The most potent challenge to the notion of identity between logic and law comes from Justice Oliver Wendell Holmes Jr.[72] Legal scholars continue to cite this epigram as an embodiment of the school of legal realism: "The life of the law has not been logic, it has been experience".[73]

Courts and advocates in the Philippines have cited this quote from Holmes, often without its full context to the point that it has become a slogan, a meme. It can be invoked to defeat a clear interpretation of the law on linguistic and rational grounds in order to introduce extraneous considerations. However, the reflexive invocation of this epigram in order to frustrate the application of logic is misleading.

If one were to read the rest of Holmes' work, one would realize that Holmes was not dismissing the role of logic and rational thinking in law. Instead, Holmes was urging us to include more inputs into what is still a logical process of making a legal determination.

In objecting to what he called "the fallacy of the logical form", Holmes:

1. Acknowledges that as a phenomena contained in the same universe as physical matter, law is ultimately subject to the same underlying rules, such as causation (otherwise, it would be a miracle);

2. Acknowledges that logic permeates through the practice: "The training of lawyers is a training in logic" - since it involves building familiarity with logical tools like analogy, discrimination, and deduction. Holmes also characterizes judicial decision as expressed in the language of logic.

Thus, Holmes objection, and the actual divide between "natural law" and legal realism is not whether or not logic should be applied at all, but to what materials logical processes

[72]See Lovevinger, "An Introduction to Legal Logic", at 472.

[73]Holmes, The Common Law, p. 1, 1881.:"The life of the law has not been logic: it has been experience. The felt necessities of the time, the prevalent moral and political theories, intuitions of public policy, avowed or unconscious, even the prejudices which judges share with their fellow-men, have had a good deal more to do than the syllogism in determining the rules by which men should be governed. The law embodies the story of a nation's development through many centuries, and it cannot be dealt with as if it contained only the axioms and corollaries of a book of mathematics..."

should work with. For the “natural law” school, they believe that there are transcendent basic principles which can be grasped intuitively, or derived through deduction. On the other hand, “legal realists” reject *a priori* transcendent rules and emphasize an inductive approach from empirical data (or experience).

It should also be noted that logic has evolved from Holmes’ schoolboy days, when most likely education would only cover classical propositional logic (or syllogistic logic as originally systematized by Aristotle)It can be conceded that classical propositional logic, as formulated during Holmes’ time, the logic that most of us are aware of (and the one usually employed in programming) is not the most appropriate tool for representing legal rules. Chapter 6 will discuss the more appropriate logical systems for representing legal rules, such as deontic logic and defeasible logic.

4 Overview of encoding and analysis approaches

The proposed work is based on restating the problem of competition impact analysis in computational terms:

1. *The Relevance Problem* - Given a law, is it **relevant** to the sector for which the assessment is being made?

2. *The Threshold Testing Problem* - Given a rule within a relevant law, is the rule **compliant** with the norms laid out by the threshold test?

From a computational point of view, the problem of competition impact assessment is a problem of logical comparison and evaluation. It involves comparing the provisions of the law that cover a sector with a set of standards, and then evaluating the extent to which the law complies with the standards. The standards can refer to the OECD threshold tests (and are further elaborated in the economics literature, usually based on models of a competitive market). In order to proceed with automating this evaluation, a computational law system will require: 1. A system for encoding the content of legal text, as well as 2. Algorithms that can process these encodings.

Table 1: Encoding and Analysis Approaches

Problem	**Encoding**	**Analysis**
Relevance Testing: Does the law map with the industry being assessed? (Actors, transactions)	Ontologies (Ontology Web Language)	Reasoning engines to determine relationships: - No mapping? - Identity? - Classification? - Mereological? - Inference?
Threshold Testing: Given a specific rule within a relevant law - How does this rule relate to the norm of the threshold test?	Inference rules (Prakken, Sartor) - LegalRuleML	Argumentation Frameworks Propositional networks

Based on the previous chapter, we are proceeding from the notion that law and questions of law are largely computable problems.[74] Facilitating computation of law requires encoding systems for both problems: First to represent, then to analyze these representations(determine relevance, and evaluate for compliance). These appear to be distinct

[74] A computable question is one that can be computed by a sufficiently powerful "Turing machine".

problems and require different encoding systems. The encoding methodology for this study uses two divergent approaches, each applicable to a different aspect of the law. The first approach aims to capture the semantic content of the law through ontologies, which are used to model the entities and relationships in a domain. The second approach is concerned with representing the normative constraints contained in the law as a set of defeasible inferential statements in deontic logic.[75] This chapter provides an overview of both approaches, with a focus on how they can be applied to the domain of competition law.

Since every modern computer language is Turing complete (i.e. it can fully implement a Turing machine), these programming languages are capable of computing legal questions. The only constraints will be time, memory, and computing power. Andersson (2014) asserts that most software tools (general purposes, modern languages) are overkill for implementing the requirements of a computational law system. It would be more efficient (cost-benefit wise) to develop and use domain-specific languages for computational law.[76] However, it is very difficult to come up with domain specific languages specific to law - this may be a function of few lawyers knowing how to program, and few programmers understanding law.

4.1 Ontological representation of legal semantics

4.1.1 Definition and benefits

Law provides a description of the world - which can be made legible as a configuration of entities and relationships. The entities are the actors, transactions, and objects that are the subjects of the law. The relationships are the connections between these entities, and the attributes that describe them. This aspect of the law can be encoded as an ontology. An **ontology** is a formal, explicit description of concepts that are part of a domain.[77]. It

[75] It may be possible to combine both the semantic and normative aspects. Both ontologies and inference statements are based on logic and can be arranged into network structures. In the future, machine learning may be used to automatically translate rules into logical formalisms. Meanwhile, the exercise will be undertaken by humans.

[76] See Andersson, "Computational Law: Law That Works Like Software", at 21.

[77] Natalya F Noy and Deborah L McGuinness. "Ontology Development 101: A Guide to Creating Your First Ontology". In: *Stanford Medical Informatics Technical Report* (SMI-2001-0880 Mar. 2001). URL: `http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html`, at 3. The term ontology originally referred to a branch of philosophy concerned with the study of being. It was borrowed by computer science to refer to the formal definition of objects in a domain, and the relationships between these objects. See Lamy Jean-Baptiste. *Ontologies with Python: Programming OWL 2.0 Ontologies with Python and Owlready2.*

consists of: 1. **classes** that represent concepts; 2. **properties** that describe features of these concepts, including their relationship with each other; and 3. **restrictions** to the way these classes and attributes are defined.[78] An ontology of classes, along with specific instances of these classes, constitute a **knowledge base**, although as a practical matter there can be little to distinguish this from an ontology.[79] Ontologies can be used to make web pages (or other electronic resources) more "understandable" to electronic agents. Many disciplines are developing standardized ontologies used by experts to encode, annotate, and share knowledge in their respective fields,providing a common vocabulary researchers and a source of machine-readable definitions.[80]. Noy (2001) suggests that for extensive domains of knowledge, ontologies can provide the following benefits:

1. *Sharing and collaboration* Experts and practitioners can represent their shared understanding.

2. *Enabling reuse* Users can build on existing ontologies - extending or refining them as needed.

3. *Making assumptions explicit* Assumptions can become explicit in the design of an ontology, making it easier to question and resolve them as necessary.

4. *Separating domain knowledge from operational knowledge* We can analyze a class of concepts in the abstract, independent of particular instances.

5. *Analyzing domain knowledge* Once a representation is available, it can be subjected to formal analysis.

An ontology can be formally expressed in a computer language. This work will use the Web Ontology Language (OWL) to express ontologies. The choice is largely based on the OWL's broad adoption, and the availability of supporting software and documentation.

Berkeley, CA: Apress, 2021. ISBN: 978-1-4842-6551-2 978-1-4842-6552-9. DOI: `10.1007/978-1-4842-6552-9`. URL: `http://link.springer.com/10.1007/978-1-4842-6552-9` (visited on 04/03/2024), at §3, p. 61.

[78] Michael De Bellis. *A Practical Guide to Building OWL Ontologies*. Oct. 8, 2021. URL: `https://www.michaeldebellis.com/post/new-protege-pizza-tutorial` (visited on 01/31/2024), at 6.

[79] Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology".

[80] For medicine, for example, there is SNOMED (Price and Spackman, 2000) and the Unified Medical Language System (Humphrey and Lindberg, 1993); For describing products and services for the purpose of trade regulation, see the United Nations Standard Products and Services Code(UNSPC), at https://www.unspsc.org/

OWL is a language that is based on Description Logic, a subset of first-order logic that is used to represent knowledge in a structured and formal way.[81][82] For protoyping and visualization purposes, the author will use the Protégé ontology editor, which is a widely used tool for creating and editing ontologies in OWL.[83]

Managing and retrieving data from ontologies is more efficient and cost-effective compared to Large Language Models (LLMs). To make corrections, one simply needs to identify and modify the specific entity and attribute. This approach is more appropriate for making precise factual determinations where accuracy is prioritized over expressiveness. The use of ontologies is also more transparent and interpretable compared to LLMs. The structure of the ontology can be visualized and understood by humans, and the reasoning process can be traced and explained. This is important for legal applications, where the reasoning process must be transparent and understandable to the parties involved.

4.1.2 Ontology components:classes and properties

Classes are the primary focus and building blocks of an ontology. These describe concepts in a domain.[84] Since we are concerned with modelling entities that interact with each other and the law, our ontology can have a `Person` class that represents the legal definition of a person - an individual or entity that has the capacity to enter into legal relations. A class can have **subclasses** that represent more specific concepts.[85] For example, the `Person`

[81]See Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 3.

[82]OWL is a standard that is maintained by the World Wide Web Consortium (W3C), the organization that sets standards for the web. It is used to represent knowledge in a way that is machine-readable and can be processed by computers. OWL is based on the Resource Description Framework (RDF), a standard for representing information on the web. RDF is used to represent information in the form of triples, which consist of a subject, a predicate, and an object. OWL extends RDF by providing a way to represent classes, properties, and relationships between classes and properties. It also provides a way to represent restrictions on classes and properties, such as cardinality constraints and value constraints. OWL is used in a wide range of applications, including the Semantic Web, data integration, and knowledge representation. It is a powerful language that can be used to represent complex knowledge in a structured and formal way.

[83]See Mark A. Musen. "The protégé project: a look back and a look forward". In: *AI Matters* 1.4 (2015), pp. 4–12. DOI: `10.1145/2757001.2757003`. URL: `https://doi.org/10.1145/2757001.2757003`.

[84]Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 3.

[85]Ibid., at 3.

class can have subclasses such as `Natural_Person` to represent a human individual and `Juridical_Entity`, such as a corporation. Individuals (or *instances* of these classes) are the actual objects in the domain of interest.[86] For example, the `Natural_Person` class can have instances such as `Alice` and `Bob`.

Properties and inheritance describe the attributes of and relationships among classes and instances. The class definition of the `Person` class can have `has_name` property that describes the name of a person, which can be provided for an instance of that class. Properties can also be used to describe the relationships between classes. For example, the `Person` class can have a `has_child` property that describes the relationship between a parent and a child. The `has_child` property can be used to connect a `Natural_Person` instance to another `Natural_Person` instance that is their child. Properties can also have restrictions that define the cardinality of the property, the value of the property, or the relationship between the property and other properties. For example, the `has_child` property can have a restriction that specifies that a child can have at most two parents. Subclasses inherit the properties of their parent classes, and can have additional properties that are specific to them.[87]

4.2 Ontology construction

There is no one "right" methodology for constructing an ontology. Noy(2001) proposes an iterative approach: With a rough, initial pass, filling details along the way. It is a question of what is most appropriate for the applications in mind and the developments anticipated for the ontology. There should at least be a sense of isomorphism, or closeness, between an ontology and the common understanding of the domain.[88] This can be achieved by reflecting on the statements that describe the domain. The nouns correspond to the classes/instances, while the verbs and adjectives correspond to the attributes.

The ontology to be used for this work shall be designed based on the following process outline in Noy(2001), with some details provided by DeBellis (2021):

[86] See De Bellis, *A Practical Guide to Building OWL Ontologies*, at 7. At the same time, classes can be thought of as sets that contain individuals.

[87] See ibid., at 7.

[88] Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 4.

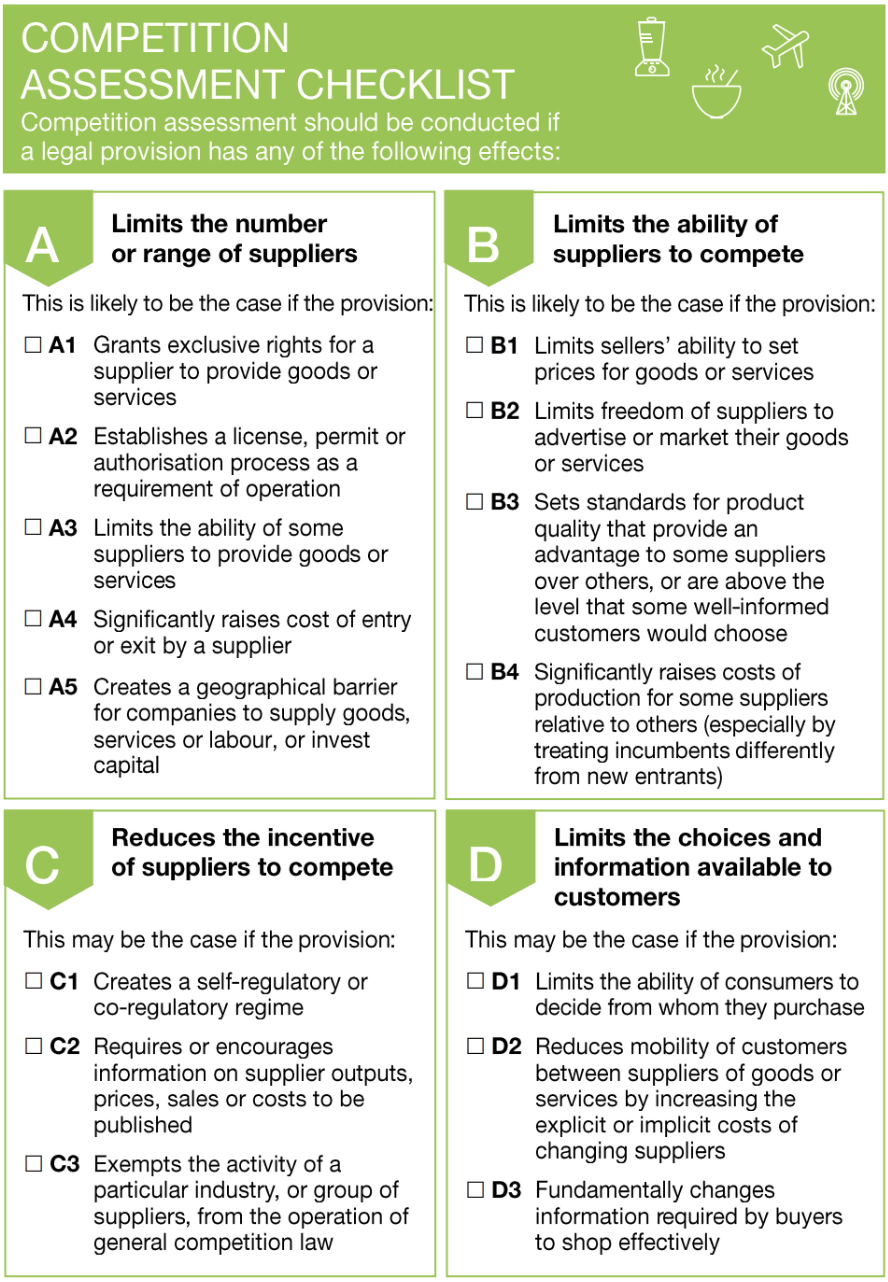
1. Determine the domain and scope of the ontology
2. Consider reusing existing ontologies
3. Enumerate important terms
4. Define the classes and class hierarchy
5. Define the internal structure of classes
6. Define the restrictions of attributes

This short tour of the design process will also serve as an opportunity to describe how ontologies can model the semantic content of the relevant competition law, as well as some of the early design decisions taken.

STEP 1: Determine domain and scope of the ontology The first step requires us to specify the domain of interest, as well as the contemplated uses of the ontology.This study is concerned with several domains, each of which can be modelled through separate ontologies: 1. The entities and transactions in the digital payments market in the Philippines, as described by the relevant laws; and 2. The idealized configuration of entities and transactions in a competitive market, as described by the OECD threshold tests. This work will focus on the OECD Guidelines since it has become the *ad hoc* basis of the Philippine's competition impact assessment regime. It is also the most comprehensive and most updated resource of this type available to the public. Other similar guidelines, such as those issued by the International Competition Network, the Asia Pacific Economic Cooperation, and the UK's Competition and Markets Authority will be used to supplement our understanding of the norms applicable to competition impact assessment. The primary purpose of the ontology is to enable the evaluation of laws governing a particular sector for competition effects. For this chapter, we will use the first OECD threshold test standard as an example. The OECD tests for the following competition concerns:

For this demonstration of the design process we are only concerned with A1, which flags a law as having competition issues if it "Grants exclusive rights for a supplier to provide goods or services". Note that although the header for Section A by itself is not a threshold test, and its general normative requirement (i.e., that it not "limits the number or range of

Figure 1: The OECD Threshold Test Checklist



suppliers"), it is considered part of the domain since it may still provide information as to required classes and properties.

STEP 2: Consider reusing existing ontologies The knowledge base can be also be based on existing ontologies that have already been developed for some knowledge domains or specific activities. For example, the financial sector is already covered by the Financial Industry Business Ontology (FIBO), a knowledge graph that models the entities and transactions in the financial sector.[89]It is a standard that is already being used by financial institutions, regulators, and other stakeholders. For concepts related to law, we may derive from the design of LegalRuleML[90].Finally, the Wikidata project is a knowledge base that models data that can be found in the open web.[91] Whenever appropriate, we can use these ontologies directly, or design our ontology to be compatible with them.

STEP 3: Enumerate important terms We next proceed to listing the important terms that the ontology needs to describe and explain, as well as their relevant properties - **property attributes** can qualify classes (i.e. what they are "like"), while **functional attributes** can describe what the classes can do, or what can be done to them.[92]. The rule of thumb is to consider the nouns of statements as the classes of the ontology, while adjectives and verbs can be considered as the properties. For the competition impact assessment ontology, we can start with the following terms (with implied terms in parentheses):

The ontology designer should also take note of any term that may be in the statement being modelled, but are nevertheless implied by the other terms. For example, since the standards mention a `Supplier`, it can be inferred even at this point that we need to model the ultimate recipient of the goods and services supplied - a `Consumer`. Both `Supplier` and `Consumer` are legal `Persons`, which we will also need to define and elaborate later on. Finally, since the standards in the threshold test are meant to apply to laws - hence the need for a `Law` class. The `State` class is also implied, as the standards assume that there is a state that is enacting and enforcing the law.

[89]See EDM Council. *The Financial Industry Business Ontology.* FIBO. URL: `https://spec.edmcouncil.org/fibo/` (visited on 01/18/2024).

[90]Oasis Open. *LegalRuleML Core Specification Version 1.0.* Aug. 30, 2021. URL: `http://docs.oasis-open.org/legalruleml/legalruleml-core-spec/v1.0/legalruleml-core-spec-v1.0.html` (visited on 10/06/2023), See also See https://www.gecad.isep.ipp.pt/ieso/contract/v1.0.0/#description for a basic contract ontology.

[91]See Wikimedia Foundation. *Wikidata.* URL: `https://www.wikidata.org/wiki/Wikidata:Main_Page` (visited on 01/18/2024).

[92]Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 6.

Table 2: Example terms for the ontology

Nouns (Classes)	**Verbs or Qualifiers (Attributes)**
Right	
(Person)	limit
Supplier	number
(Consumer)	range
Goods	grant
Services	provides
(State)	exclusive
(Law)	

STEP 4: Define the classes and class hierarchy Several approaches are open to determining the classes and their place in the hierarchy (i.e. the subclass-superclass relationship). There is the **top-down approach** which is to start with the most general concepts, and then proceed to the more specific cases. Alternatively, one can also take a **bottom-up approach**, which means to start with defining the most specific classes, then determine if these can be grouped into general concepts (i.e. generate common superclasses). The more realistic approach is a combination of both, i.e. define the salient concepts and then generalize or specialize as needed. No method is best - it would depend on the circumstances surrounding the modeling, i.e. if a general view is available, if data is granular enough to describe specific cases.[93] To determine which terms can be classes or subclasses, a good rule of thumb is that objects that are capable of independent existence (rather than descriptions of other objects) can be the principle classes in a class hierarchy.Once classes are identified and defined, arrange them hierarchically into a taxonomy. This can be done by asking for each class, whether it can be an instance of the same class.[94]

For the competition impact assessment ontology, we can start with the following configuration of classes and subclasses:

- `Person` - An individual or entity with legal capacity. This can have the following subclasses:
  - `Natural_Person` - A human individual, to which the class of `Consumer` belongs.

[93]Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 6-7.
[94]Ibid., at 7-8.

- Juridical_Entity - A legal entity, which can include a Corporation - which in turn is the superclass of any Supplier object (an entity that provides Goods or Services).[95]

- Right - A legal entitlement (or permission, in deontic terms) that can be granted or limited by the State through a Law. The right concerns the ability to offer and enter into a contract concerning a Provision, which can have the following subject matters:

  - Goods - Physical objects that can be supplied by a Supplier. Can refer to any tangible object that can be bought or sold.
  - Services - Intangible objects that can be supplied by a Supplier. Can refer to any contractual performance.[96]

Class hierarchies show how concepts are related. They use terms like "is-a" or "kind-of" to show these connections. When one class is a subclass of another, it means the subclass represents a more specific type of the general concept represented by the main class.[97]The class hierarchy, as constructed in Protégé can be visualized as shown in the following figure:

Note that in OWL, all classes are subclasses of a root class called owl:Thing, the class that represents the set containing all individuals. All empty ontologies still contain one class called owl:Thing.[98]

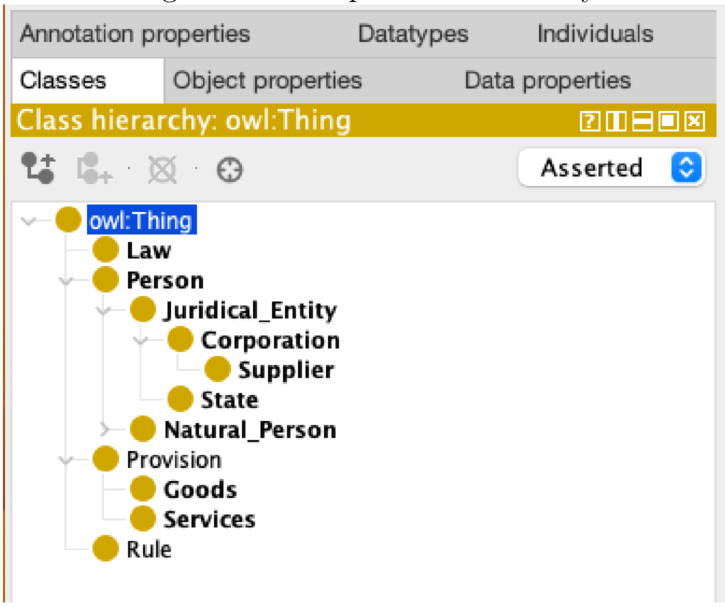
[95]Note the simplifying assumptions that we are holding for now in order to facilitate the design of the ontology. In the real world, a corporation can be a consumer, and a natural person can be a supplier. The artificial distinction however, may be "true enough" for the purposes of our ontology.

[96]The classes of Goods and Services can be bound by reference to another ontology, such as the United Nations Standard Products and Services Code (UNSPC).

[97]Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 12.

[98]De Bellis, *A Practical Guide to Building OWL Ontologies.*

Figure 2: Example class hierarchy



STEP 5: Define the internal structure of classes The internal structure of the classes can be defined through its properties or attributes.[99]. For every class in our ontology, we are concerned with both intrinsic and extrinsic properties:[100]

- **Intrinsic properties** - There are the essential, or inherent to the class itself. These properties are essential to the identity and nature of the class, independent of external factors or contexts. They are characteristics that an instance of the class possesses purely by being an instance of that class. For the class `Person`, intrinsic properties might include a `has_name` - since each legal person, whether an individual human being or a corporation, has a name.

- **Extrinsic properties** - These are context-dependent, relational attributes of a class. Extrinsic properties are those that depend on external factors or the context in which an instance of the class exists. These properties are not essential to the identity of the class and can change depending on the environment, relationships, or interactions with other entities. For the class `Person`, extrinsic properties might include the person's

[99] Also called slots in earlier documentation

[100] Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 8.

current location, occupation, marital status, or the clothes they are wearing.[101]

Subclasses inherit the properties of their parent classes, and can have additional properties that are specific to them.[102] For example, the `Natural_Person` class can inherit the `has_name` property from the `Person` class, and can have additional properties such as `has_age` and `has_address`. The `Juridical_Entity` class can inherit the `has_name` property from the `Person` class, and can have additional properties such as `has_registration_number` and `has_legal_address`.

STEP 6: Define the attribute restrictions The properties of a class can have restrictions that define the cardinality of the property, the value of the property, or the relationship between the property and other properties.[103] We can define the cardinality of an attribute - how many values a property can have. The `has_name` for a person can have a single cardinality - that is, a person is allowed to only have one legal name. Other properties can have multiple cardinality. For example, the `has_child` property of a `Natural_Person` class can have a restriction that specifies that a child can have at most two parents, or several friends. We can also define restrictions for acceptable values that can be entered for each property: The `has_name` property can have a restriction that specifies that the value of the property must be a string(i.e. a series of text characters), or that the `has_age` property can have a restriction that specifies that the value of the property must be a positive integer. By specifying the domain and **range** of an attribute, we can place restrictions on the relationships of classes. The **domain** of a property refers to the set of all objects that can have that property asserted about it.[104] The **range** of a property, on the other hand, the set of all objects that can be the value of the property.[105] For example, the fact that a `Law` can contain many `Rules` can be modelled by the `has_rule` attribute. The `has_child` property can also have a restriction that specifies that a child must be a `Natural_Person` instance.

[101] A form of extrinsic properties that relate the class to other classes are mereological properties, i.e. a class can also have can have physical and abstract parts (e.g. the parts of an engine or the courses of a meal)

[102] Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 9.

[103] Ibid., at 9.

[104] De Bellis, *A Practical Guide to Building OWL Ontologies*, at 26.

[105] Ibid.

When defining a domain or range of an attribute, Noy(2001) recommends finding the most general classes or class that can serve the purpose. Nevertheless the domain or the range should not be too general, i.e. the classes in the domain of an attribute should be described by the attribute, and the instances of all the classes in the range of an attribute should be potential values for the attribute.[106]

4.3 Representation of normative constraints

4.3.1 Inference rules

The threshold test of competition impact assessment can be stated more formally as follows: Given a set of rules (i.e., the rules that cover an industry) - does it comply with or diverge from the idealized norm of the threshold test? In previous assessment exercises, to make things manageable logistically, the author has proposed making individual provisions the unit of analysis. However, even a provision can still express several rules, each of which can be independently evaluated. Therefore rules will serve as our unit of analysis.

Knowledge graphs only give us a part of the picture. Besides the entities, their attributes and interactions - all these are subject to constraints and transformations based on law. These only provide static data about the semantics of entities and their interactions - but these do not reflect the legal constraints that act upon those objects, and how the semantics could be qualified, transformed, or annulled by such constraints. Another way of putting it is that knowledge graphs reflect only the whats and the who's, not the oughts and ought nots that contained in legal knowledge.

The prevalent approach is to state the text of legal provisions into atomic inference rules, which have the structure - If P then Q. It is also possible to state a rule categorically as simply Q, but this should be rare in operation.[107]

Take for example the simple rule "If a lane is designated as a bus lane, then only buses can drive through it". This can be broken down to several inference rules:

---

[106] Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", at 10.

[107] See Giovanni Sartor. "A Formal Model of Legal Argumentation". In: *Ratio Juris* 7.2 (July 1994), pp. 177–211. ISSN: 0952-1917, 1467-9337. DOI: 10.1111/j.1467-9337.1994.tb00175.x. URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9337.1994.tb00175.x (visited on 06/20/2023).

- If [Lane has Bus Only Markings] then [Lane is Designated]
- If [Lane is Designated] then ¬[Driver Enters]
- If [Driver Enters] then [Violation]

Once we have formal representations, the next step would be to apply analytical methods grounded in logic. We can trace chain of inferences (via *modus ponens*), discover other rules, even look for potential inconsistencies.

4.3.2 Deontic Logic

Legal statements are for the most part, not composed of factual statements. They do not describe the state of the world as it is, but how it ought to be. They can't be assessed for truth values. Furthermore, legal conclusions are arrived at under an informational environment marked by incompleteness, uncertainty, and inconsistency.[108] Logicians have since developed a a form of logic, called Deontic Logic, which is not concerned with True or False, but oughtness: Whether certain acts or states of the world are: Obligatory, Prohibited, or merely Permitted.[109]

Deontic Logic was influenced by modal logic (which concerns modalities, or expressions that qualify the truth of propositions, i.e., necessity and probability) Although notions of Deontic Logic have been explored in fourteenth century Europe as well as Islamic thought (in the 10th century), its modern version grounded in symbolic logic is based on the work of Von Wright (1951).[110]

Instead of the binary values of True or False, Deontic Logic accommodates six normative states:

1. It is obligatory that (OB)

[108] See Kathleen Freeman and Arthur M. Farley. "A Model of Argumentation and Its Application to Legal Reasoning". In: *Artificial Intelligence and Law* 4.3-4 (1996), pp. 163–197, at 165.

[109] See G.H. von Wright. "Deontic Logic". In: *Mind* 60.237 (Jan. 1951), pp. 1–15, for the original use of the term and the first modern systemization of the field; See also Paul McNamara and Frederik Van De Putte. "Deontic Logic". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2022. Metaphysics Research Lab, Stanford University, 2022. URL: `https://plato.stanford.edu/archives/spr2022/entries/logic-deontic/` (visited on 06/08/2022), for an updated overview.

[110] McNamara and Van De Putte, "Deontic Logic", at § 1.

2. It is permissible that (PE)
3. It is impermissible that (IM)
4. It is omissible that (OM)
5. It is optional that (OP)
6. It is is non-optional that (NO)

Recasting the earlier example under the Deontic mode:

- If [Lane has Bus Only Markings] then [Lane is Designated] - No changes because this is actually a factual statement.
- If [Lane is Designated] then ¬[Driver Enter] becomes: O([Lane is Designated] $\rightarrow \neg$ [Driver Enters]) - The inference is neither true nor false, but has the deontic modality of Obligation (O).
- If [Driver Enters] and [Lane is Designated] then [Violates] becomes O([Driver Enters] $\wedge$ [Lane is Designated] $\rightarrow$ [Violation]) - That is, if the car enters the lane when the lane is designated as a bus lane, then we must find a violation

4.3.3 Defeasibility and argumentation

Another attribute of legal propositions is that they are **defeasible**. This means that they are tentative - accepted until some other proposition - a new fact that activates an exception, better evidence, or even a higher law - defeats our original proposition.[111]

Legal conclusions are arrived at based on knowledge that is incomplete, uncertain, and inconsistent. Despite this, an adequate theory of legal reasoning should provide a sound basis of what to believe. Argumentation theory is suited to the problem because it takes into consideration contrasting claims under an environment of uncertainty and inconsistency.[112]

[111] See generally Giovanni Sartor. "Defeasibility in Legal Reasoning". In: *Rechtstheorie* 24.3 (1993), pp. 281–316.
[112] Freeman and Farley, "A Model of Argumentation and Its Application to Legal Reasoning", at 163-164.

The model proposed by Freeman views argument in the following ways:

1. As a structure for supporting explanation - It consists of discrete units of arguments that connect claims with data
2. As a dialectical process - It consists of a series of moves by opposing parties that either support or attack a given claim[113]

Freeman's model integrates the notion of burden of proof - the level of support necessary for any one party to "win" the argument. This serves as filter, turntaking mechanism, and termination criteria. The process enables the generation of decisions that could fall anywhere within the continuum of skeptical and credulous.[114]

4.4 Automated analysis and evaluation

The canonical approach requires evaluation of the relevant laws for features that match a predetermined list of factors (usually based on the economics literature). It relies on both a reading of the text, and the lawyer's training on how the text is most likely interpreted and enforced. What usually happens, based on the recommendations of these guides, is an appeal to the lawyer's intuition as to the intent and consequences of the legal text. Some of these guides suggest, to balance out the inherent subjectivities in that determination: Consulting other stakeholders (regulators and industry stakeholders). While this cross analysis might go a long way towards making the conclusions less stilted, there is still no proof of work that can be shared and independently studied, changed, and evaluated. We should be able to rely on a transparent chain of reasoning proceeding from plausible assumptions into consistent propositions, that can be shared, analyzed, built on top of each other.

Once we have the rules encoded, the goal is to perform automated evaluations. We can look for internal inconsistencies, or gaps in the coverage of industry entities and transactions. Then we can compare one set of rules - such as the legislation under competition impact assessment, with the standards set by the economic literature, or the competition authority, or international organizations. Once law is reduced to a formalized structure,

[113] Freeman and Farley, "A Model of Argumentation and Its Application to Legal Reasoning", at 167.
[114] *Ibid.*

then it becomes amenable to direct comparison - for finding difference and inconsistency. Unlike intuitive assessments, though, the reasoning process is exposed from the start - the assumptions are provided (or at least very easy to look up), and each step towards the conclusion is available for proof.

Knowledge graphs and inference rules can be combined into network structures, and it is possible to compare network structures - i.e. to what extent these structures are similar or different. But beyond some of the more obvious methods, this work will explore two pathways that will enable computers to compare and evaluate the encoded rules: 1. Argumentation frameworks and 2. Propositional networks.

The first takes into account the dialectic nature of arriving at a legal determination. Conclusions about law are often only arrived at after an argument - one side presents a plausible reading of the law, another counters with a supposedly better reading of the law, or evidence of factual circumstances that would make the law inapplicable, or a higher law.[115] The initial proponent could counter, and on and on until the arguments are exhausted and a decision has to be made by some process and standard. In the computational law field, there are so called argumentation frameworks. These are tools for modeling both rules and facts into arguments. Normative claims can be encoded just like rules, while the facts embodied in knowledge graphs can serve as evidence, or a warrant that either supports or undercuts a claims. In order to be processed an argumentation framework, we need to add information as to how all the claims and warrants relate to each other - either supporting or attacking. A reviewer can set the burden of proof, the weight of different kinds of evidence, and the standard required for an argument to prevail over the other.

Another method to be explored is through propositional networks. Propositional networks are an extension of game theory.[116] It is used in artificial intelligence, used for playing games and programming logic. Under this approach, entities and transactions can be modeled as they are in a knowledge graph - related to each other through states, attributes, and

[115]See generally Frans H. Van Eemeren et al. *Handbook of Argumentation Theory.* Dordrecht: Springer Netherlands, 2014. ISBN: 978-90-481-9472-8 978-90-481-9473-5. DOI: `10.1007/978-90-481-9473-5`. URL: `https://link.springer.com/10.1007/978-90-481-9473-5` (visited on 06/20/2023).

[116]See Michael Genesereth and Michael Thielscher. *General Game Playing.* Red. by Ronald J. Brachman, William W. Cohen, and Peter Stone. Synthesis Lectures on Artificial Intelligence and Machine Learning 24. Morgan & Claypool Publishers, 2014.

transactions. Unlike the static representation of knowledge graphs, however, propositional nets allow us to model transitions in both entities and relationships that can be caused either by constraints or actions - which can be provided by law. Propositional networks can be used to model the behavior of entities and transactions over time, and how they interact with each other.

The approach should combine the norms in our deontic propositions with the structured information in a knowledge graph, such that the norms can interact with the semantic information. Because the law can assume that the [Driver] is an adult and is licensed, and if neither of those are true, then a different set of norms apply. At the same time, a state of [Violation] would mean that the status of [Driver] could be modified i.e., suspended or annulled.

5 A computational solution to relevance testing

The canonical approach does not provide a search strategy, often assuming that the applicable law would be obvious, or determined by a motivated party. The field of computation has developed a toolset for approaching search - it begins with understanding the search space by having a model for the entire set of entities to be searched through (and what you're searching for), and then having a strategy for either an exhaustive or a heuristic search.

The understanding of the search space will be based on a model of the industry or market under analysis. One such model is based on the value chain - The model of a sector's value chain can reveal its important features - the industry's actors, and transactions, any of which may be the subject of a legal provision.[117] Most competitive assessment frameworks move from modeling the value chain directly to conducting keyword searches for relevant laws. While a value chain model can provide an overall view of the sector - it still lacks resolution as to how the actors and transactions relate to each other, as well as the underlying reasons for these interactions. In addition to surfacing concepts that are distinct from or related to the usual labels for the components of a value chain, a concept map can generate additional keywords and ensure that the search can be more comprehensive.

Building an ontology means representing concepts (entities, transactions, problems, and solutions) as nodes, and the relationships between these concepts (sameness, belonging) as lines of a network structure.[118] Visualizing the sector this way can help surface other concepts and other legal concerns. For example, "programmers" are usually subsumed into software and platform companies in most models of the technology sector. Building a concept map can encourage the explicit declaration that the Software Company entity

[117]See Ramonette B. Serafica et al. "Issues Paper on the Philippine Digital Commerce Market". In: *PCC Issues Paper.* 2020th ser. 3 (2020). URL: `https://www.phcc.gov.ph/wp-content/uploads/2020/07/PCC-Issues-Paper-2020-03-Issues-Paper-on-the-Philippine-Digital-Commerce-Market.pdf`, for an example of a value chain approach to mapping an industry, and its challenges.

[118]See Noy and McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology", for an overview of the ontology building process; See also Giovanni Sartor. "Legal Concepts as Inferential Nodes and Ontological Categories". In: *Artificial Intelligence and Law* 17.3 (Aug. 21, 2009), pp. 217–251. ISSN: 0924-8463, 1572-8382. DOI: `10.1007/s10506-009-9079-7`. URL: `http://link.springer.com/10.1007/s10506-009-9079-7` (visited on 06/20/2023), for an application of the approach in the legal domain.

employs the Programmer entity, which develops the Software Product entity. This could prompt the researcher to consider a possible competitive dimension beyond price effects on the end product, i.e., a large enough company can hire all the top programmers of a particular programming specialization, in order to deny potential rivals a development pathway. This in turn can lead a researcher to pursue laws that govern the hiring of programmers (or other people with technical skills), or industry-wide software standards - both of which could have potential competitive impacts. In addition to generating new keywords, a concept map can identify synonyms that can be used for database searches. Consequently, it may also reveal different meanings of the same word. This also helps determine the limits of the sector as a bounded context, i.e., the appearance of the same word to describe a different entity or transaction could signal to the researcher that the concept actually belongs to another sector.

This also helps address the semantic gap - the relevance of a concept is not wholly determined by the particular label attached to it, but also shaped by its relationship to other entities. Even if an actor or its actions are designated by different words, if they are related to the same entities and transactions, then they are part of the same industry.

Keywords can be generated by traversing each branch and then each leaf of the concept map iteratively, looking for combinations of concepts and relationships that evoke competitive dimensions and their coordinate legal issues. Even though it might be challenging for a human to explore all possible paths, a computer can perform this operation with ease.

6 Bibliography

Andersson, Hans. “Computational Law: Law That Works Like Software”. CodeX – The Stanford Center for Legal Informatics, Feb. 10, 2014. URL: https://www.academia.edu/9286857/Computational_Law_Anderrson_and_Lee.

Armgardt, Matthias. “Leibniz as a Legal Scholar”. In: *Fundamina* (2014), pp. 27–38.

Armgardt, Matthias, Patrice Canivez, and Sandrine Chassagnard-Pinet, eds. *Past and Present Interactions in Legal Reasoning and Logic*. Vol. 7. Logic, Argumentation & Reasoning. Cham: Springer International Publishing, 2015. ISBN: 978-3-319-16020-7 978-3-319-16021-4. DOI: 10.1007/978-3-319-16021-4. URL: https://link.springer.com/10.1007/978-3-319-16021-4 (visited on 03/16/2024).

Artosi, Alberto and Giovanni Sartor. “Leibniz as Jurist”. In: *The Oxford Handbook of Leibniz*. Ed. by Maria Rosa Antognazza. Oxford University Press, Dec. 27, 2018, pp. 640–663. ISBN: 978-0-19-974472-5. DOI: 10.1093/oxfordhb/9780199744725.013.38. URL: https://academic.oup.com/edited-volume/34667/chapter/295400716 (visited on 10/10/2023).

Competition & Markets Authority. *Competition Assessment Guidelines Part 2: Guidelines*. UK Government, 2023. URL: https://www.gov.uk/government/publications/competition-impact-assessment-guidelines-for-policymakers (visited on 01/10/2024).

— *Competition Impact Assessment Part 1: Overview*. UK Government, Sept. 15, 2015. URL: https://www.gov.uk/government/publications/competition-impact-assessment-guidelines-for-policymakers (visited on 01/10/2024).

De Bellis, Michael. *A Practical Guide to Building OWL Ontologies*. Oct. 8, 2021. URL: https://www.michaeldebellis.com/post/new-protege-pizza-tutorial (visited on 01/31/2024).

Duso, Tomaso, Jo Seldeslachts, and Florian Szücs. “The Impact of Competition Policy Enforcement on the Functioning of EU Energy Markets”. In: *The Energy Journal* 40.5 (Sept. 2019), pp. 97–120. ISSN: 0195-6574, 1944-9089. DOI: 10.5547/01956574.40.5.tdus. URL: http://journals.sagepub.com/doi/10.5547/01956574.40.5.tdus (visited on 01/08/2024).

EDM Council. *The Financial Industry Business Ontology*. FIBO. URL: https://spec.edmcouncil.org/fibo/ (visited on 01/18/2024).

European Bioinformatics Institute (EMBL-EBI). *Ontology Search*. Ontology Lookup Service (OLS). URL: https://www.ebi.ac.uk/ols4/ontologies (visited on 01/18/2024).

European Commission. *Statistical Classification of Economic Activities in the European Union*. Rev. 2. 2008. URL: https://ec.europa.eu/eurostat/documents/3859598/5902521/KS-RA-07-015-EN.PDF (visited on 05/13/2024).

Feynman, Richard P. *Feynman Lectures on Computation*. Ed. by Anthony J. G. Hey and Robin W. Allen. Boca Raton: CRC Press, 2018. 303 pp. ISBN: 978-0-7382-0296-9.

Freeman, Kathleen and Arthur M. Farley. "A Model of Argumentation and Its Application to Legal Reasoning". In: *Artificial Intelligence and Law* 4.3-4 (1996), pp. 163–197.

Genesereth, Michael. *Computational Law: The Cop in the Back Seat*. CodeX: The Center for Legal Informatics Stanford University. 2015. URL: https://law.stanford.edu/publications/computational-law-the-cop-in-the-backseat/ (visited on 09/18/2021).

Genesereth, Michael and Nathaniel Love. "Computational Law". In: *Proceedings of the 10th International Conference on Artificial Intelligence and Law - ICAIL '05*. The 10th International Conference. Bologna, Italy: ACM Press, 2005. ISBN: 978-1-59593-081-1. DOI: 10.1145/1165485.1165517. URL: http://portal.acm.org/citation.cfm?doid=1165485.1165517 (visited on 09/18/2021).

Genesereth, Michael and Michael Thielscher. *General Game Playing*. Red. by Ronald J. Brachman, William W. Cohen, and Peter Stone. Synthesis Lectures on Artificial Intelligence and Machine Learning 24. Morgan & Claypool Publishers, 2014.

Goldstein, Rebecca. *Incompleteness: The Proof and Paradox of Kurt Godel*. New York, London: Atlas Books, 2005.

Goldsworthy, Jeffrey. "The Limits of Judicial Fidelity to Law: The Coxford Lecture". In: *Canadian Journal of Law and Jurisprudence* 24.2 (July 2011), pp. 305–325. DOI: 10.1017/S084182090000518X.

Grossman, Garry S and Lewis D Solomon. "Computers and Legal Reasoning". In: *ABA Journal* 69 (1983), pp. 66–70.

Guest, Anthony G. "Logic in the Law". In: *Oxford Essays in Jurisprudence*. Oxford: Oxford University Press, 1961.

Hage, Jaap. "A Theory of Reasoning and a Logic to Match". In: *Artificial Intelligence and Law* 4.3-4 (1996), pp. 199–273.

Hasan, Iftekhar and Matej Marinč. "Should Competition Policy in Banking Be Amended during Crises? Lessons from the EU". In: *European Journal of Law and Economics* 42.2 (Oct. 2016), pp. 295–324. ISSN: 0929-1261, 1572-9990. DOI: 10.1007/s10657-013-9391-2. URL: http://link.springer.com/10.1007/s10657-013-9391-2 (visited on 01/08/2024).

ICN Advocacy Working Group. "Framework of Competition Assessment Regimes". In: ICN 14th Annual Conference. Sydney, Apr. 2015. URL: https://www.internationalcompetit org/wp-content/uploads/2018/09/AWG_FrameworkCompetitionAssessmentRegimes.pdf (visited on 10/10/2023).

— *Recommended Practices on Competition Assessments*. International Competition Network, 2014. URL: https://www.internationalcompetitionnetwork.org/wp-content/uploads/2018/07/AWG_RP_English.pdf (visited on 01/10/2024).

Jean-Baptiste, Lamy. *Ontologies with Python: Programming OWL 2.0 Ontologies with Python and Owlready2*. Berkeley, CA: Apress, 2021. ISBN: 978-1-4842-6551-2 978-1-4842-6552-9. DOI: 10.1007/978-1-4842-6552-9. URL: http://link.springer.com/10.1007/978-1-4842-6552-9 (visited on 04/03/2024).

Leith, Philip. "Logic, Formal Models and Legal Reasoning". In: *Jurimetrics* 24.4 (1984), pp. 334–356.

Lovevinger, Lee. "An Introduction to Legal Logic". In: *Indiana Law Journal* 27.4 (Sum. 1952), pp. 471–522.

Ma, Megan. "The Law's New Language". In: *Harvard International Law Journal Frontiers* 61 (2020), pp. 1–9.

McNamara, Paul and Frederik Van De Putte. "Deontic Logic". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2022. Metaphysics Research Lab, Stanford University, 2022. URL: https://plato.stanford.edu/archives/spr2022/entries/logic-deontic/ (visited on 06/08/2022).

Medalla, Erlinda M. "Understanding the New Philippine Competition Act". In: *Philippine Institute for Development Studies (PIDS) Discussion Paper Series* (No. 2017-14 2017), pp. 1–24. URL: http://hdl.handle.net/10419/173591 (visited on 01/08/2024).

Musen, Mark A. "The protégé project: a look back and a look forward". In: *AI Matters* 1.4 (2015), pp. 4–12. DOI: 10.1145/2757001.2757003. URL: https://doi.org/10.1145/2757001.2757003.

Navarro, Pablo E. and Jorge L. Rodríguez. *Deontic Logic and Legal Systems*. New York: Cambridge University Press, Sept. 29, 2014. ISBN: 978-0-521-76739-2. DOI: 10.1017/CBO9781139032711.

Noy, Natalya F and Deborah L McGuinness. "Ontology Development 101: A Guide to Creating Your First Ontology". In: *Stanford Medical Informatics Technical Report* (SMI-2001-0880 Mar. 2001). URL: http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html.

Oasis Open. *LegalRuleML Core Specification Version 1.0*. Aug. 30, 2021. URL: http://docs.oasis-open.org/legalruleml/legalruleml-core-spec/v1.0/legalruleml-core-spec-v1.0.html (visited on 10/06/2023).

OECD. *Competition Assessment Reviews: Logistics Sector in the Philippines*. 2020. URL: https://www.oecd.org/daf/competition/oecd-competition-assessment-reviews-philippines-2020.pdf (visited on 10/10/2023).

— *Competition Assessment Toolkit - Volume 1 (Principles)*. 2019. URL: https://www.oecd.org/daf/competition/46193173.pdf (visited on 10/10/2023).

— *Competition Assessment Toolkit - Volume 2 (Guidance)*. 2019. URL: https://www.oecd.org/daf/competition/45544507.pdf (visited on 10/10/2023).

— *Competition Assessment Toolkit - Volume 3 (Operations Manual)*. 2019. URL: https://web-archive.oecd.org/2020-01-22/370055-COMP_Toolkit_Vol.3_ENG_2019.pdf (visited on 10/10/2023).

— *Competitive Neutrality Reviews: Small-Package Delivery Services in the Philippines*. 2020. URL: https://www.oecd.org/daf/competition/oecd-competitive-neutrality-reviews-philippines-2020.pdf (visited on 10/10/2023).

Prakken, Henry and Giovanni Sartor, eds. *Logical Models of Legal Argumentation*. Netherlands: Kluwer Academic Publishers, 1997. ISBN: 0-7923-4413-8. DOI: 10.1007/978-94-011-5668-4.

Robins, Nicole and Hannes Geldof. "Ex Post Assessment of the Impact of State Aid on Competition". In: *European State Aid Law Quarterly* 17.4 (2018), pp. 494–508. ISSN: 16195272, 21908184. DOI: 10.21552/estal/2018/4/6. URL: http://estal.lexxion.eu/article/ESTAL/2018/4/6 (visited on 01/03/2024).

Sartor, Giovanni. "A Formal Model of Legal Argumentation". In: *Ratio Juris* 7.2 (July 1994), pp. 177–211. ISSN: 0952-1917, 1467-9337. DOI: 10.1111/j.1467-9337.1994.tb00175.x. URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9337.1994.tb00175.x (visited on 06/20/2023).

— "Defeasibility in Legal Reasoning". In: *Rechtstheorie* 24.3 (1993), pp. 281–316.

— "Legal Concepts as Inferential Nodes and Ontological Categories". In: *Artificial Intelligence and Law* 17.3 (Aug. 21, 2009), pp. 217–251. ISSN: 0924-8463, 1572-8382. DOI: 10.1007/s10506-009-9079-7. URL: http://link.springer.com/10.1007/s10506-009-9079-7 (visited on 06/20/2023).

Serafica, Ramonette B. et al. "Issues Paper on the Philippine Digital Commerce Market". In: *PCC Issues Paper*. 2020th ser. 3 (2020). URL: https://www.phcc.gov.ph/wp-content/uploads/2020/07/PCC-Issues-Paper-2020-03-Issues-Paper-on-the-Philippine-Digital-Commerce-Market.pdf.

Summers, Robert S. "Logic in the Law". In: *Cornell Law Faculty Publications* (Paper 1133 1963), pp. 254–258. URL: http://scholarship.law.cornell.edu/facpub/1133.

The World Bank. *Markets and Competition Policy*. World Bank. URL: https://www.worldbank.org/en/topic/competition-policy (visited on 01/16/2024).

Tribe, Laurence H. "The Curvature of Constitutional Space: What Lawyers Can Learn from Modern Physics". In: *Harvard Law Review* (Nov. 1989), pp. 1–68.

Turing, Alan M. "On Computable Numbers, with an Application to the Entscheidungsproblem". In: *Proceedings of the London Mathematical Society* 2.42 (1937), pp. 230–265. DOI: 10.1112/plms/s2-42.1.230.

United Nations. *International Standard Industrial Classification of All Economic Activities (ISIC)*. Revision 4. United Nations, 2008. ISBN: 978-92-1-161518-0. URL: https://unstats.un.org/unsd/publication/seriesm/seriesm_4rev4e.pdf.

Van Eemeren, Frans H. et al. *Handbook of Argumentation Theory*. Dordrecht: Springer Netherlands, 2014. ISBN: 978-90-481-9472-8 978-90-481-9473-5. DOI: 10.1007/978-90-481-9473-5. URL: https://link.springer.com/10.1007/978-90-481-9473-5 (visited on 06/20/2023).

Von Wright, G.H. "Deontic Logic". In: *Mind* 60.237 (Jan. 1951), pp. 1–15.

Wikimedia Foundation. *Wikidata*. URL: https://www.wikidata.org/wiki/Wikidata:Main_Page (visited on 01/18/2024).

Wolfram, Stephen. *A New Kind of Science*. Champaign, Illinois: Wolfram Media, 2002. 1197 pp. ISBN: 978-1-57955-008-0.

— *A Project to Find the Fundamental Theory of Physics*. Champaign, Illinois: Stephen Wolfram, LLC, 2020. 770 pp. ISBN: 978-1-57955-035-6.

— "AI Law and Computational Irreducibility". FutureLaw 2023, Stanford Law School. Apr. 25, 2023. URL: https://www.youtube.com/watch?v=8oG1FidVE2o (visited on 01/14/2024).

— *Computational Law, Symbolic Discourse and the AI Constitution*. Stephen Wolfram Writings. Oct. 12, 2016. URL: https://writings.stephenwolfram.com/2016/10/computational-law-symbolic-discourse-and-the-ai-constitution/ (visited on 01/14/2024).

— "Computational Law, Symbolic Discourse and the AI Constitution". In: *Data-Driven Law: Data Analytics and the New Legal Services*. Ed. by Ed Walters. Red. by Jay Liebowitz. Data Analytics Applications. Boca Raton, FL: CRC Press Taylor & Francis Group, 2019, pp. 144–174. ISBN: 13: 978-1-4987-6665-4. URL: https://writings.stephenwolfram.com/2016/10/computational-law-symbolic-discourse-and-the-ai-constitution/ (visited on 01/14/2024).

— *How to Think Computationally about AI, the Universe and Everything*. Stephen Wolfram Writings. Oct. 27, 2023. URL: https://writings.stephenwolfram.com/2023/10/how-to-think-computationally-about-ai-the-universe-and-everything/ (visited on 12/14/2023).

World Bank Group. *Enhancing Competition Conditions and Competitiveness of Philippine Domestic Shipping*. World Bank, Sept. 2014. DOI: 10.1596/24800. URL: http://elibrary.worldbank.org/doi/book/10.1596/24800 (visited on 08/18/2021).