Secret Algorithms, IP Rights, and the Public Interest

Meghan J. Ryan

Southern Methodist University, Dedman School of Law

Recommended Citation
SECRET ALGORITHMS, IP RIGHTS, AND THE PUBLIC INTEREST

Meghan J. Ryan*

The secrecy surrounding the algorithms that play a central role in American life today is proving to have alarming effects. Judges and juries are convicting defendants based on secret evidence. Major advertisers like Facebook are discriminating against minorities seeking housing. And Russians may very well be hacking our voting machines to change election outcomes. The algorithm secrecy underlying these results obscures whether such legal outcomes are actually accurate and fair or whether they were based on faulty evidence, affected by bias, or manipulated by outside influences. These are just a handful of the public-interest perils of algorithm secrecy. This Article explains that the pervasive secrecy surrounding algorithms is not entirely by accident. The Supreme Court’s recent overhaul of intellectual property (IP) law has driven algorithm developers toward secrecy. By limiting patent protection for software, the Court’s new IP regime pushes developers away from the required disclosure of patent law and toward the obscurity of trade secret law. In doing so, the new regime neglects to take into account the many negative effects that this heightened secrecy has on the public interest. Accuracy, fairness, and good policy require a more careful consideration of the tradeoffs between secrecy and transparency. This includes not only exploring how to minimize these swelling public-interest concerns but also reexamining the Court’s new IP rules with these negative effects in mind.

TABLE OF CONTENTS
INTRODUCTION ........................................................................................................................................... 62
I. THE NEW RULES OF INTELLECTUAL PROPERTY LAW .......................................................... 66
   A. Foundations ........................................................................................................................................ 67

* Associate Dean for Research, Robert G. Storey Distinguished Faculty Fellow, Gerald J. Ford Research Fellow, Altshuler Distinguished Teaching Professor, and Professor of Law, SMU Dedman School of Law. I thank Hillel Bavli, David Taylor, Keith Robinson, Orly Mazur, Xuan-Thao Nguyen, Sharon Sandeen, and Jenia Turner for their helpful comments on this project. I am also grateful for questions and comments I received on related projects from Rachel Barkow, Erin Murphy, Jeff Fagan, Tracey Meares, Rachel Harmon, Bernard Harcourt, Paul Butler, Sharon Dolovich, Dan Richman, David Sklansky, John Pfaff, Josh Kleinfeld, Kate Levine, Adi Leibovitch, Anna Lvovsky, Crystal Yang, Bennett Capers, Miriam Baer, Andrew Ferguson, Xuan-Thao Nguyen, and the participants at the IP/Life Science Symposium at Indiana University Robert H. McKinney School of Law, the 13th Annual Symposium on Emerging Intellectual Property Issues: Intellectual Property and Social Justice at the SMU Dedman School of Law, the ABA Criminal Justice Section Ninth Annual Fall Institute, the Southeastern Association of Law Schools Annual Meeting, and the Columbia Criminal Justice Roundtable.
INTRODUCTION

Judges and juries are convicting defendants based on secret evidence.1 Major advertisers such as Facebook are discriminating against minorities seeking housing.2 And Russians may be hacking our voting machines to change election outcomes.3 These are just some of the evils that have followed a burgeoning secrecy encompassing the algorithms that play such a central role in American life today.4 Transparency has an important role to play in ensuring the accuracy of these algorithms and the fairness of the outcomes they produce. But the U.S. Supreme Court’s recent change of course in intellectual property (IP) law now means that, where algorithm-based software is concerned, the law encourages trade secret protection over patent protection. This translates into encouraging algorithm secrecy over algorithm transparency. Considering the injustices that flow from this policy preference for trade secret over patent protection, however, courts really ought to consider the public-interest perils resulting from this decision.

The recent landmark case of Alice Corp. v. CLS Bank5 ushered in a new era of IP rights. This case, in which the Supreme Court invalidated a software pa-

---

1 See infra Section II.A.
2 See infra Section II.B.
3 See infra Section II.C. Please note that, although this Article does express concern about the security of our elections, is does not suggest that the 2020 presidential election was stolen, hacked, or produced fraudulent results.
tent on the ground that abstract ideas such as algorithms cannot be patented,6 left IP lawyers and their clients scrambling for how to deal with their proprietary technologies. The prospects for successfully patenting software, computerized algorithms, and other similar technologies are now dubious. And considering that the patent system is based on a tradeoff of surrendering secrecy for gaining a short-term monopoly on an idea,7 the loss of patent rights—or even just uncertainty about patent rights—has led to increased secrecy surrounding companies’ technologies. Understandably, if a company is unable to successfully patent its technology and obtain a short-term monopoly on the invention, the company will want to maintain the secrecy of the technology.8 Further, companies have an incentive to bolster that secrecy so that, even if they cannot obtain patent rights, they can secure trade secret rights on the technology.9

In this milieu of uncertainty, there has been concern that the American framework for innovation is collapsing.10 If companies are not disclosing their

6 See id. at 212 ("We hold that the claims at issue are drawn to the abstract idea of intermediate settlement, and that merely requiring generic computer implementation fails to transform that abstract idea into a patent-eligible invention.").

7 See Graham v. John Deere Co. of Kansas City, 383 U.S. 1, 9 (1966) (suggesting that the limited monopoly available under patent law is “a reward, an inducement, to bring forth new knowledge”); that “grant[ing] an exclusive right to an invention [is] the creation of society—at odds with the inherent free nature of disclosed ideas—and [i]s not to be freely given”; and that “[o]nly inventions and discoveries which further[] human knowledge, and [a]re new and useful, justify[] the special inducement of a limited private monopoly”); see also Clarisa Long, Patent Signals, 69 U. Chi. L. Rev. 625, 626–27 (2002) (referring to this—“fram[ing] intellectual property in general, and patents in particular, as an exchange of information for protection”—as “[t]he simple view”); infra note 25.

8 See Frank H. Easterbrook, Foreword: The Court and the Economic System, 98 Harv. L. Rev. 4, 21–22 (1984) (explaining that program developers generally only make their programs available to the public if they are compensated). But see generally Long, supra note 7, at 627 (urging readers to “relax the assumption that the private value of a patent is based solely on the ability to capture rents and exclude others from using the invention”).

9 See 1 ROGER M. MILGRIM & ERIC E. BENSON, MILGRIM ON TRADE SECRETS § 1.03 (2020) (“[F]ailure to use efforts reasonable under the circumstances to protect matter claimed to be a trade secret is necessarily fatal to the claim. . . . The value, then, of a trade secret rests in maintenance of secrecy.”); see also generally Defend Trade Secrets Act of 2016, Pub. L. No. 114–153, 130 Stat. 376 (creating a federal private cause of action for trade secret misappropriation); UNIF. TRADE SECRETS ACT (1979) (UNIF. L. COMM’N, amended 1985) (providing model rules for trade secret protection).

10 See, e.g., CLS Bank Int’l v. Alice Corp., 717 F.3d 1269, 1313 & n.1 (Fed. Cir. 2013), aff’d, 573 U.S. 208 (2014) (Moore, J., dissenting) (warning that “if all of these claims . . . are not patent-eligible, this case is the death of hundreds of thousands of patents, including all business method, financial system, and software patents as well as many computer implemented and telecommunications patents’’ and concluding that this “would decimate the electronics and software industries’’); Brief for Paul R. Michel as Amicus Curiae in Support of Neither Party at 9, Alice Corp. v. CLS Bank Int’l, 134 S. Ct. 2347 (2014) (No. 13-298) (arguing that the Court in Alice should “return to its seminal precedent in Diehr, a computer case’ and stating that following “the aberrational approach of Flook or the unworkable notion of relative abstractness of Bilski will complicate, confuse, and confound the patent law” and “also cripple, if not destroy, computer-related industries, of which there are many and which are vital to the future of the country in today’s highly competitive global economy.”).
technologies but are in fact strengthening the secrecy masking their technologies, then other inventors cannot build on the already-existing technologies. This slows innovation. The trajectory that we are set upon is troubling.

The incentive for secrecy that Alice created has fueled an additional troubling development in the domain of the public interest. As companies migrate from patent to trade secret protection and bolster the secrecy surrounding their new technologies, not only do these technologies become unavailable to the companies’ competitors, but they also become unavailable to the general public more broadly.\(^{11}\) This means that it is significantly more difficult to verify the important results these technologies produce by assessing the accuracy and fairness of their underlying algorithms.

This is already playing out in the criminal justice arena. For example, criminal defendants are often convicted by evidence such as breathalyzer results, fingerprint matches, and DNA analyses. These pieces of evidence are rooted in various computerized algorithms that have often been developed by independent technology companies and are ordinarily claimed as proprietary in nature.\(^{12}\)

From the perspective of the companies, this is perfectly understandable; they seek to profit off their investments in research and development. From the perspective of the criminal defendant, however, the proprietary—and thus secret—nature of these computerized algorithms is unfair. Although such a claim of unfairness has been asserted by only a handful of criminal defendants,\(^{13}\) this is not because the secret nature of these computerized algorithms on which convictions are based is not troubling; it is instead because, due to the sorry state of public funding for the assistance of counsel, many attorneys representing these defendants are not even aware that this is an issue.\(^{14}\) But it is an important issue. Under the Constitution, criminal defendants must “be afforded a meaningful opportunity to present a complete defense”\(^{15}\) and to “confront[] ... the wit-

---


\(^{11}\) *See infra* text accompanying notes 31–34.


\(^{13}\) *See id.* at 306, 313, 319–20.

\(^{14}\) *See id.* at 323–24 (suggesting that criminal justice actors lack the scientific and technological knowledge to understand the nuances of algorithms and source codes and noting that the sorry state of public defense in this country often allows for these issues to be overlooked).

nesses against [them].”16 Without access to the details of the computerized algorithms providing incriminating evidence against them, these defendants lack the opportunity to challenge this incriminating evidence that poses real questions of accuracy, not to mention bias.

These concerns of accuracy and bias, which stem from a lack of transparency and public access, are also popping up in areas such as housing and voting. For example, there are allegations that Facebook’s advertisement-targeting software discriminates against minorities in housing—a move that runs contrary to the Fair Housing Act and the Civil Rights Act of 1964.17 There is also evidence indicating that the algorithms used in voting machines are vulnerable to outside manipulation, which threatens to undermine the foundations of our democracy.18 In these contexts, too, the walls of secrecy shield from public scrutiny the independently developed technologies on which private companies and the government are regularly relying.

This troublesome clash between secrecy and transparency emerges from the increasing reliance on technology in legal decisionmaking and the competing interests of the technology developers, technology users, and general public. The dynamics among these different populations vary depending on the technology, making it difficult to find a universally applicable solution. In some circumstances, making the algorithm available under seal or allowing it to be viewed in camera might satisfy the users and public clamoring for access while simultaneously addressing the developers’ primary concerns. Under other conditions, pushing the developers and users into exclusive license agreements that would mitigate the developers’ disclosure anxieties could be feasible. But in some situations, as in the voting context, transparency poses concerns in addition to the developers’ IP rights, such as security issues. Because each scenario differs, attention must be paid to the various stakeholders and risks involved to find the appropriate balance of secrecy and transparency. A more universal approach might be possible, however, by reassessing the new rules of IP rights in play after Alice. If courts were to take into account these various public-interest perils that flow from secrecy when they sketch the shifting boundaries of IP rights, many of these risks and harms could be avoided.

This Article pinpoints the new regime of IP law as a significant contributor to the algorithm secrecy problem imperiling the public interest and advances a handful of approaches to minimize the negative effects flowing from the secrecy incentives created by Alice. Apart from this patchwork approach to mitigating the effects of algorithm secrecy, this Article suggests that courts should seriously consider the impacts on the public interest when charting the boundaries of IP protection, at least where software-related algorithm secrecy is involved.

---

16 U.S. CONST. amend. VI.
17 See infra Section II.B.
18 See infra Section II.C.
Part I of this Article lays the foundation, explaining the basic tenets of IP law and chronicling how the Supreme Court has recently narrowed subject matter eligibility for patents where computerized algorithms are concerned, thus pushing software developers in the direction of protecting their inventions through trade secret law instead. Naturally, this results in greater secrecy ensconcing these algorithms. Part II describes how many of these algorithms affect our everyday lives and how keeping them secret can have a real impact in various public-interest arenas. For example, judges and juries are convicting criminal defendants based on the results these secret algorithms produce even though questions have arisen about the accuracy of these results.\(^{19}\) Further, companies such as Facebook rely on secret algorithms in their advertisement targeting, which could discriminate against certain types of individuals in critical markets like housing.\(^{20}\) And we rely on these secret algorithms in assessing the outcomes of our elections even though, again, there are accuracy questions and, relatedly, questions about whether the algorithms have allowed for security breaches.\(^{21}\) Finally, Part III suggests that, in the interest of justice, we must strike a better balance between the needs for secrecy and transparency where the public interest is involved. The risks of secrecy and transparency differ depending on the context, so it is necessary to carefully assess these risks in determining the proper balance. Further, the problem of algorithm secrecy calls out for a broader solution that hearkens back to a significant contributor to the predicament: the new IP regime after Alice. In continuing to construct the boundaries of IP protections, courts ought to consider the public-interest impacts that stem from their decisions. These legal and policy determinations should not be made with blinders on but should instead consider how secrecy and transparency will impact important public-interest areas.

I. THE NEW RULES OF INTELLECTUAL PROPERTY LAW

Inventors have taken advantage of various IP protections for over a century.\(^{22}\) Patents, copyrights, trademarks, and, somewhat later, trade secrets have traditionally occupied different territories, but software is unique in that it possesses characteristics that lend themselves to various types of IP protection. This has led to some uncertainty about how developers can best protect their

\(^{19}\) See infra Section II.A.

\(^{20}\) See infra Section II.B.

\(^{21}\) See infra Section II.C.

technologies. But this uncertainty was significantly magnified when the Supreme Court recently shook the foundations of IP law by severely limiting patent protection for software. This has pushed software developers to seek protection in trade secret law instead, which translates into a shift from disclosing the details of software to keeping them secret.

A. Foundations

The U.S. Constitution empowers Congress to “secure[e] for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries” for the purpose of “promot[ing] the Progress of Science and useful Arts.” Congress put this authority into action through a series of statutes preserving authors’ and inventors’ works via patent, copyright, trademark, and even trade secret protection. In large part, Congress has opted to promote scientific and artistic progress through providing inventors and authors with a limited monopoly on their work in exchange for sharing their work with the community. For example, Title 35 of the U.S. Code states that, if an inventor discloses a new, useful, and nonobvious invention with sufficient particularity, and if he also pays the relevant fees to the U.S. Patent and Trademark Office (PTO), then he is entitled to patent protection—a twenty-year term of exclusivity on his invention. Title 17 provides that, if an author creates an original

23 U.S. CONST. art. I, § 8, cl. 8 (“The Congress shall have Power . . . To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries.”).


25 See Graham v. John Deere Co. of Kansas City, 383 U.S. 1, 9 (1966) (suggesting that the limited monopoly available under patent law is “a reward, an inducement, to bring forth new knowledge”; that “grant[ing] . . . an exclusive right to an invention [is] the creation of society—at odds with the inherent free nature of disclosed ideas—and [is] not to be freely given”; and that “[o]nly inventions and discoveries which further[] human knowledge, and [are] new and useful, justify[] the special inducement of a limited private monopoly”); see also Long, supra note 7, at 626 (referring to this—“fram[ing] intellectual property in general, and patents in particular, as an exchange of information for protection”—as “[t]he simple view”). In Bonito Boats, Inc. v. Thunder Craft Boats, Inc., the Supreme Court explained:

The Patent Clause itself reflects a balance between the need to encourage innovation and the avoidance of monopolies which stifle competition without any concomitant advance in the “Progress of Science and useful Arts.” As we have noted in the past, the Clause contains both a grant of power and certain limitations upon the exercise of that power. Congress may not create patent monopolies of unlimited duration, nor may it “authorize the issuance of patents whose effects are to remove existent knowledge from the public domain, or to restrict free access to materials already available.”


26 See 35 U.S.C. § 101 (“Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to [certain] conditions . . . .”); id. § 103 (providing that, to obtain such a patent, though, the invention must be new, useful, and nonobvious “to a person having ordinary skill in the art to which the claimed invention pertains”); id. §§ 111–12
work fixed in a tangible form, he is entitled to copyright protection, which amounts to exclusivity of his work for the duration of his life plus seventy years thereafter.\textsuperscript{27} Moreover, if the author goes through the steps to register his copyright, he is entitled to additional protections amounting to the ability to sue for infringement, potentially obtain an injunction, and recover an array of damages and attorneys’ fees.\textsuperscript{28} Constitutional bargains like these benefit inventors and artists by allowing them to profit from exclusivity, and they simultaneously benefit society by allowing its members access to these inventions and arts.\textsuperscript{29} Society can simply consume this wealth of information or even build upon it, further fueling scientific and artistic progress.\textsuperscript{30}

Trade secret law complements these patent and copyright protections. But trade secret law is in some ways the exact opposite in that it protects secrets—disclosure is not required, and in fact disclosure inhibits the ability to protect the IP at issue.\textsuperscript{31} Pursuant to this corner of IP law, one may to a certain extent

(providing that a patent application should include a “specification . . . contain[ing] a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and . . . set[ting] forth the best mode contemplated by the inventor or joint inventor of carrying out the invention”); \textit{id.} § 154 (“Subject to the payment of fees . . . [the] grant [of a patent] shall be for a term beginning on the date on which the patent issues and ending 20 years from the date on which the application for the patent was filed . . .”).

\textsuperscript{27} \textit{See} 17 U.S.C. § 102 (“Copyright protection subsists, in accordance with this title, in original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.”); \textit{id.} § 302 (“Copyright in a work created on or after January 1, 1978, subsists from its creation and, except as provided by the following subsections, endures for a term consisting of the life of the author and 70 years after the author’s death.”).

\textsuperscript{28} \textit{See} Michael A. Epstein, Epstein on Intellectual Property § 401[C] (5th ed. Supp. 2020). An author may not file an infringement suit unless he first registers the copyright. \textit{See} 17 U.S.C. §§ 411–412; Epstein, supra, at § 401[C]. Note, however, that different rules apply if the work originated outside the United States. \textit{See} Epstein, supra, at § 401[C]. Although an author may bring claims based on infringements even prior to registration, such an author is not entitled to the full panoply of remedies that an author suing on infringements subsequent to registration is entitled. \textit{See id.} (“In addition to prohibiting the institution of an infringement suit until after registration, the [Copyright] Act conditions the remedies of statutory damages and attorneys’ fees on having registered the copyright before infringement.”) (footnotes omitted). The author of a work not registered at the time of infringement may be entitled to an injunction and actual damages, but the author of a registered work may also be entitled to statutory damages and attorneys’ fees. \textit{See id.} Where published works are concerned, the statute provides a three-month grace period. \textit{See id.}

\textsuperscript{29} \textit{See supra} note 25 and accompanying text.

\textsuperscript{30} \textit{See supra} note 25.

\textsuperscript{31} \textit{See} Orly Lobel, \textit{Filing for a Patent Versus Keeping Your Invention a Trade Secret}, HARV. BUS. REV. (Nov. 21, 2013), https://hbr.org/2013/11/filing-for-a-patent-versus-keeping-your-invention-a-trade-secret [https://perma.cc/3MNQ-QAUF] (“Patents and trade secrets present opposing choices. Trade secrets derive their legal protection from their inherently secret nature. Patents, by contrast, can only be protected through public disclosure.”); \textit{see also} 2 Milgrim & Bensen, supra note 9, at § 9.02[2][a] (“Trade secret law is pertinent to patent
SECRET ALGORITHMS

Fall 2020

69

protect his invention or special knowledge if it constitutes a “trade secret,” meaning that the individual has taken reasonable measures to maintain the secrecy of information that derives economic value from its secrecy.32 Although trade secret law does not provide as robust protection as patent or copyright law, it does provide some protection against the misappropriation of that trade secret. If another misappropriates the trade secret—discovering it directly or indirectly “by improper means,” or disclosing it in breach of the trade secret licensing because it provides an independent, complementary form of industrial property which in some instances offers alternative protective umbrellas for patentable technology and, in others, may together serve to protect subject matter that is the same or closely related to patented matter.”). Professors Sharon Sandeen and Ulla-Maija Mylly have explained:

As with . . . IP . . . laws more generally, the theory underlying trade secret protection is that society gets something that is of greater benefit than the advantages that flow from information diffusion and free competition. In the case of trade secrets, this includes the prevention of unfair competition and additional incentives for invention and creation over and above what is provided by patent and copyright laws.


[T]he term “trade secret” means all forms and types of financial, business, scientific, technical, economic, or engineering information, including patterns, plans, compilations, program devices, formulas, designs, prototypes, methods, techniques, processes, procedures, programs, or codes, whether tangible or intangible, and whether or how stored, compiled, or memorialized physically, electronically, graphically, photographically, or in writing if—

(A) the owner thereof has taken reasonable measures to keep such information secret; and

(B) the information derives independent economic value, actual or potential, from not being generally known to, and not being readily ascertainable through proper means by, another person who can obtain economic value from the disclosure or use of the information . . .

18 U.S.C. § 1839(3). The Uniform Trade Secrets Act, Restatement (First) of Torts, and Restatement (Third) of Unfair Competition define “trade secret” in much the same way. See UNIF. TRADE SECRETS ACT § 1(4) (1979) (UNIF. L. COMM’N 1979, amended 1985) (“‘Trade secret’ means information, including a formula, pattern, compilation, program, device, method, technique, or process, that . . . (i) derives independent economic value, actual or potential, from not being generally known to, and not being readily ascertainable by proper means by, others who can obtain economic value from its disclosure or use, and . . . (i) is the subject of efforts that are reasonable under the circumstances to maintain its secrecy.”); RESTATEMENT (FIRST) OF TORTS § 757 cmt. b (AM. L. INST. 1939) (“A trade secret may consist of any formula, pattern, device or compilation of information which is used in one’s business, and which gives him an opportunity to obtain an advantage over competitors who do not know or use it.”); RESTATEMENT (THIRD) OF UNFAIR COMPETITION § 39 (AM. L. INST. 1995) (“A trade secret is any information that can be used in the operation of a business or other enterprise and that is sufficiently valuable and secret to afford an actual or potential economic advantage over others.”).
holder’s confidence—trade secret law provides certain limited remedies.\footnote{See 18 U.S.C. § 1839; Unif. Trade Secrets Act § 1(2) (1979) (Unif. L. Comm’n, amended 1985); Restatement (Third) of Unfair Competition § 39 (Am. L. Inst. 1995); Restatement (First) of Torts § 757 (Am. L. Inst. 1939). The Defend Trade Secrets Act provides:

(a) Whoever, with intent to convert a trade secret, that is related to a product or service used in or intended for use in interstate or foreign commerce, to the economic benefit of anyone other than the owner thereof, and intending or knowing that the offense will, injure any owner of that trade secret, knowingly:

(1) steals, or without authorization appropriates, takes, carries away, or conceals, or by fraud, artifice, or deception obtains such information;
(2) without authorization copies, duplicates, sketches, draws, photographs, downloads, uploads, alters, destroys, photocopies, replicates, transmits, delivers, sends, mails, communicates, or conveys such information;
(3) receives, buys, or possesses such information, knowing the same to have been stolen or appropriated, obtained, or converted without authorization;
(4) attempts to commit any offense described in paragraphs (1) through (3); or
(5) conspires with one or more other persons to commit any offense described in paragraphs (1) through (3), and one or more of such persons do any act to effect the object of the conspiracy,

shall, except as provided in subsection (b), be fined under this title or imprisoned not more than 10 years, or both.

(b) Any organization that commits any offense described in subsection (a) shall be fined not more than the greater of $5,000,000 or 3 times the value of the stolen trade secret to the organization, including expenses for research and design and other costs of reproducing the trade secret that the organization has thereby avoided.

18 U.S.C. § 1832.}

34 FMC Corp. v. Taiwan Tainan Giant Indus. Co., 730 F.2d 61, 63 (2d Cir. 1984) (per curiam).

35 Orly Lobel has helpfully explained that “trade secret law and patent law can coexist,” though, and that, within a particular client’s portfolio, a patent might “protect the broad concept, while trade secret[] [might] protect the production details.” Lobel, supra note 31.
crets on any of their inventions. Some factors that one might consider in choosing the best category include the following:\footnote{36}{42\footnote{37}{[B]ecause this choice is frequently encountered in the real world, academics and practitioners have developed a well-established body of literature describing the considerations that should be taken into account when deciding between patent and trade secret protection.” Andrew A. Schwartz, The Corporate Preference for Trade Secret, 74 OHIO ST. L.J. 623, 637 (2013). Some additional considerations include “business concerns, antitrust issues, and industry-specific cultural views of patents.” David Hricik, Will Patenting Make As Much Sense in the New Regime of Weakened Patent Rights and Shorter Product Lifes? 20 VAND. J. ENT. & TECH. L. 457, 506 (2017). For a more robust discussion of relevant factors, see Andrew Beckerman-Rodau, The Choice Between Patent Protection and Trade Secret Protection: A Legal and Business Decision, 84 J. PAT. & TRADEMARK OFF. SOC’Y 371, 380–406 (2002).

37 See Schwartz, supra note 36, at 636 (“[B]ecause the scope of trade secret law is much broader than that of patent law, some valuable information—like a customer list or a marketing strategy—may qualify as a trade secret but be ineligible for patent protection.”).

38 See id. at 630 (“[O]nce the [trade] secret is disclosed, it is lost forever. On the other hand, if the secret is kept, trade secret law offers protection in perpetuity.” (footnotes omitted)). Often, “[t]he duration of an IP right is among its most important characteristics . . . .” Id. at 637. Accordingly, Andrew Schwartz has argued that corporations should generally prefer trade secrets over patents so that the corporations—“perpetual entities”—may “reap . . . perpetual returns that only a trade secret can offer.” Id. at 623, 649.


40 See Schwartz, supra note 36, at 644–46 (“The conventional analysis teaches that cost is a key concern when it comes to choosing between patent and trade secret protection. This includes both the cost of obtaining IP protection, and the cost of maintaining that protection, including through litigation.” (footnotes omitted)).

41 See FMC Corp. v. Taiwan Tainan Giant Indus. Co., 730 F.2d 61, 63 (2d Cir. 1984) (per curiam) (“A trade secret once lost is, of course, lost forever.”); see also David S. Levine & Ted Sichelman, Why Do Startups Use Trade Secrets?, 94 NOTRE DAME L. REV. 751, 780 (2018) (“It is notable that, in at least one study, the risk of departing employees misappropriating trade secrets did not appear to justify using patents instead of trade secrecy and other mechanisms.”).

42 See Schwartz, supra note 36, at 639–40. Contributors to this risk might include whether the product is “regularly observed in public settings” and whether “the invention is detectable and embedded in the product itself or is [I] part of an internal manufacturing process[.]”}
7. The “signal value” that patents provide to third parties, and
8. The extent to which patent protection will enhance the liquidity and alienability of the information.

One challenge is that, because of various unknowns, it is often difficult to determine ex ante the type of protection that will prove best for any particular invention or client.

B. The Complexity of Software

The complexity of the question as to how best protect one’s inventions and valuable information has grown as rapidly advancing technology, especially in areas related to computing, has overtaken our world. During the software boom in the 1970s and 1980s, software developers generally protected their products as trade secrets. This may have resulted from the uncertainty of seeking other forms of protection. Over time, though, these developers sought other, more robust, forms of protection for their inventions and thus turned to copyright and patent law. This shift highlighted some tensions among the various forms of IP protection. Through the lens of IP law, computer software is a “hybrid” product: Software is expressed in a written form, similar to a manuscript deserving of copyright protection, but software is functional in nature, which is similar to an invention ripe for patent protection. This blend of characteristics has raised significant questions about the proper, or most successful, vehicle for

Lobel, supra note 31. Some commentators consider the risks of independent invention and reverse-engineering as part of the duration analysis. See Schwartz, supra note 36, at 637–40. Schwartz, supra note 36, at 641–42 (“The possession of a patent—any patent—conveys a strong positive ‘signal’ to outsiders, regardless of the technical merit of the actual invention.”). See generally Long, supra note 7 (emphasizing the signaling value of patents). Further, patents may be useful for building a portfolio of intellectual property assets, which firms can trade (e.g., by cross-licensing agreements) if a competitor asserts a patent against them.” Pamela Samuelson, The Uneasy Case for Software Copyrights Revisited, 79 GEO. WASH. L. REV. 1746, 1774 (2011).

See Schwartz, supra note 36, at 643–45. As Andrew Schwartz explained:

Both patents and trade secrets may be sold, licensed, or otherwise alienated. Patent, however, has a clear advantage over trade secrecy in that a patent is much easier, cheaper and less risky to alienate than a trade secret. This also makes a patent a more liquid asset than a trade secret.

Id. at 643.


See MARTIN CAMPBELL-Kelly, FROM AIRLINE RESERVATIONS TO SONIC THE HEDGEHOG: A HISTORY OF THE SOFTWARE INDUSTRY 107–08 (2003) (noting the uncertainty of patent protection and the usefulness of trade secret protection in the early days of software development); Menell, supra note 45, at 2652 (noting the early usefulness of trade secret law).

See infra Sections I.B.1, I.B.2.

Gregory J. Maier, Software Protection—Integrating Patent, Copyright and Trade Secret Law, 69 J. PAT. & TRADEMARK OFF. SOC’Y 151, 151 (1987) (“In intellectual property terms, software is a true hybrid... It is the hybrid nature of software that causes its failure to fit neatly into any one existing category of intellectual property, resulting in seemingly endless confusion as to how it may best be protected.”).
protection. Although there were a lot of bumps and bruises along the way, by the mid-1990s, software developers had generally found a way to “muddle[c] through” the difficulties of seeking patent and copyright protection for software.49

1. Copyrights

The Copyright Act provides protection for a “computer program”—“a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result.”50 Under § 102, copyright protection is reserved for expressive material, though.51 The statute specifically excludes from protection “any idea, procedure, process, system, method of operation, concept, principle, or discovery.”52 With respect to software, the legislative history of the Copyright Act makes clear that “the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law.”53

Despite the statute directly addressing computer programs, there has been some historical uncertainty about the scope of protection for software under the statute. Software does not fall squarely within traditional copyright subject matter like literary manuscripts; it is instead mostly functional in nature even though it is expressed in text through source and object code.54 Separating these distinct facets of software can be difficult, and this is exacerbated by the fact that judges ordinarily lack the technical expertise to clearly and competently separate the functional elements of software—which would be more appropri-

49 Menell, supra note 45, at 2652. Commenting on the state of IP protection for software in 1994, Peter Menell stated:

After many years of confusion, the existing legal regime appears to be muddling through. In the view of most commentators, the recent Altai and Sega decisions have correctly resolved two of the three major problems that have plagued copyright protection for software. The Lotus case, which presents the scope of copyright protection for user interfaces, could potentially resolve the last major difficulty inherent in the application of copyright law to computer software. Patent protection for software may prove to be problematic, although the extent of problems to date has been modest.

Id. at 2652–53 (footnotes omitted).

50 17 U.S.C. § 101; see also 17 U.S.C. § 117 (outlining some limitations on copyright protection where computer programs are at issue).

51 See 17 U.S.C. § 102(a) (“Copyright protection subsists, in accordance with this title, in original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.”).

52 Id. at § 102(b).


ately protected under patent law—from the expressive aspects of software—which would be more appropriately protected under copyright law.\(^{55}\)

It has generally been uncontroversial that copyright protects against the literal infringement of source and object code.\(^{56}\) This protection, however, is limited to the extent that computer programmers can reverse-engineer a program and author new code to achieve the same function with relative ease.\(^{57}\) Thus, the breadth of copyright protection in nonliteral infringement cases is important, and, perhaps not surprisingly, it has been more difficult to ascertain in these types of cases.\(^{58}\)

One area where there has been uncertainty about the scope of copyright protection is related to the “structure, sequence, and organization” (SSO) of computer programs, as well as programs’ “look and feel.”\(^{59}\) Initially, the Third Circuit determined that a program having an overall similar function and structure infringed the pre-existing program.\(^{60}\) Comparing software to other literary works, the court explained that “copyrights of other literary works can be infringed even when there is no substantial similarity between the works’ literal elements.”\(^{61}\) Six years later, the Second Circuit rejected this approach and significantly limited copyright protection for software.\(^{62}\) The court set forth a three-part test to determine whether a software copyright has been infringed:

The first step involves constructing a hierarchy of abstractions, from most abstract to most detailed, for the plaintiff’s program. The second step involves filtering out from the analysis various elements of the program that are beyond the scope of copyright protection. The third step involves comparing any remaining “golden nuggets” of expression in the plaintiff’s program with the defendant’s

---

\(^{55}\) *See id.*

\(^{56}\) *See id.* (“Over the past several decades, the only easily resolved software copyright cases have been those in which plaintiffs proved that defendants literally infringed source or object code or copied videogame audiovisuals or other expressive screen displays generated by programs.”); *Samuelson, supra* note 43, at 1748 (“[N]o one seriously questioned that source code forms of programs could be copyrighted as written texts . . . ”).

\(^{57}\) *See Samuelson, supra* note 43, at 1752 (noting that, “[i]f programmers could easily re-write code, and thereby avoid infringement of the first developer’s copyright, then little would be gained by the extension of copyright protection to programs”). *But cf.* *Samuelson, supra* note 54, at 295 (“Copyright provides very meaningful protection to these elements of programs.”).

\(^{58}\) *See Samuelson, supra* note 54, at 295 (“Nonliteral infringement claims have proven much more difficult for courts to resolve . . . ”).

\(^{59}\) *See Samuelson, supra* note 43, at 1765–71.

\(^{60}\) *See Whelan Assocs., Inc. v. Jaslow Dental Lab’y, Inc., 797 F.2d 1222, 1248 (3d Cir. 1986)* (“We hold that (1) copyright protection of computer programs may extend beyond the programs’ literal code to their structure, sequence, and organization, and (2) the district court’s finding of substantial similarity between the Dentalab and Dentcom programs was not clearly erroneous.”); *see also Samuelson, supra* note 43, at 1765–66 (summarizing the Whelan decision).

\(^{61}\) *Whelan, 797 F.2d* at 1234.

program to determine if the defendant copied substantial amounts of expression from the plaintiff’s program.\textsuperscript{63}

“Thin[ning]” out copyright protection in this way, the court suggested that Congress could provide additional copyright protection for software if desirable.\textsuperscript{64} Following the Second Circuit’s analysis in this case, courts have further constrained copyright protection for software.\textsuperscript{65}

Another area where there has been uncertainty about the scope of copyright protection is related to reverse-engineering.\textsuperscript{66} Does creating copies—either exact or very similar—of source or object code for the purpose of investigating the program’s function and recreating parts of the program using different code to achieve interoperability constitute fair use?\textsuperscript{67} In 1992, the Ninth Circuit determined that this was indeed fair use.\textsuperscript{68} In deciding this, the court built on the Second Circuit’s understanding of software as “utilitarian works that were eligible for only thin copyright protection.”\textsuperscript{69} It explained that, “[i]n order to enjoy a lawful monopoly over the idea or functional principle underlying a work, the creator of the work must satisfy the more stringent standards imposed by the patent laws.”\textsuperscript{70}

Although the uncertainty about the applicability of copyright law to software reigned until the early 1990s, after these cases, it became clear that copyright protection as applied to software would be quite limited.\textsuperscript{71} This pushed software developers to seek protection elsewhere, namely in the realm of patent law.\textsuperscript{72}

\textsuperscript{63} Samuelson, supra note 43, at 1770 (footnotes omitted); see also Altai, 982 F.2d at 706–10.

\textsuperscript{64} See Samuelson, supra note 43, at 1771; see also Altai, 982 F.2d at 712 (“[N]ow that more than 12 years have passed since CONTU issued its final report, the resolution of this specific issue could benefit from further legislative investigation—perhaps a CONTU II.”).

\textsuperscript{65} See Samuelson, supra note 43, at 1771 (“After Altai, courts became more openly skeptical about claims of copyright protection for the ‘look and feel’ of programs, as such claims typically sought to protect the now unprotected utilitarian aspects of programs.”).

\textsuperscript{66} See id. at 1772.

\textsuperscript{67} See id. (“Closely related to the SSO-in-interfaces controversy resolved in Altai was whether making copies of program code for the purpose of gaining access to information necessary to achieve interoperability was fair use.”).

\textsuperscript{68} See Sega Enters. v. Accolade, Inc., 977 F.2d 1510, 1520 (9th Cir. 1992) (“Where there is good reason for studying or examining the unprotected aspects of a copyrighted computer program, disassembly for purposes of such study or examination constitutes a fair use.”); see also Samuelson, supra note 43, at 1772.

\textsuperscript{69} Samuelson, supra note 43, at 1772; see also Sega Enters., 977 F.2d at 1520–27.

\textsuperscript{70} Sega Enters., 977 F.2d at 1526.

\textsuperscript{71} See Samuelson, supra note 43, at 1773.

\textsuperscript{72} See id. (“The ‘thinness’ of copyright protection for programs after Altai and Sega seems to have contributed to a shift among software developers away from heavy reliance on copyright protection for program SSO and toward a greater reliance on patents.”).
2. **Patents**

At the dawn of the software revolution, the PTO seemed open to the patentability of software.\(^{73}\) As software boomed, pressure mounted regarding the need to address whether software indeed constitutes patentable subject matter.\(^{74}\) In the 1960s, President Lyndon Johnson appointed a committee to study the matter, and the committee overwhelmingly concluded that software should not be patentable.\(^{75}\) Following this lead, the PTO issued a set of guidelines advising that the patentability of software would be severely limited.\(^{76}\) The PTO’s stated reason was that software is just the automation of “mental steps,” which are unpatentable in nature.\(^{77}\)

In the following decade, the Supreme Court decided a handful of cases explaining the limitations on patenting software. Section 101 of the Patent Act provides that “[w]hoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to [other] conditions and requirements of [t]he [A]ct.”\(^{78}\) Although this provision articulating the scope of patentable subject matter seems quite broad, the Court has historically held that there are three exceptions to this permissive language:\(^{79}\)

[L]aws of nature, natural phenomena, and abstract ideas are not patentable. . . . Such discoveries are “manifestations of . . . nature, free to all men and reserved exclusively to none.” . . . “Phenomena of nature, though just discovered, mental processes, and abstract intellectual concepts are not patentable, as they are the basic tools of scientific and technological work.” . . . And monopolization of those tools through the grant of a patent might tend to impede innovation more than it would tend to promote it.\(^{80}\)

The Court explained that, although “all inventions at some level embody, use, reflect, rest upon, or apply laws of nature, natural phenomena, or abstract ide-

---

\(^{73}\) See Steven G. Steger, *The Long and Winding Road to Greater Certainty in Software Patents*, CBA Rec., Apr. 2000, at 46, 47–48 (explaining that, “[i]n the early days of computer software development, the PTO had seemed receptive to the idea of patenting properly claimed software-related inventions”).

\(^{74}\) See id. at 48.

\(^{75}\) See id.

\(^{76}\) See id.

\(^{77}\) See id.


\(^{79}\) See Alice Corp. v. CLS Bank Int’l, 573 U.S. 208, 216 (2014) (“‘We have long held that this provision contains an important implicit exception: Laws of nature, natural phenomena, and abstract ideas are not patentable.’ . . . We have interpreted § 101 and its predecessors in light of this exception for more than 150 years.” (internal citations omitted) (quoting Ass’n Molecular Pathology v. Myriad Genetics, Inc., 569 U.S. 576, 589 (2013)); Mayo Collaborative Servs. v. Prometheus Lab’ys, Inc., 566 U.S. 66, 70 (2012) (“The Court has long held that this § 101 contains an important implicit exception.”)).

as,” allowing patents on the most basic of concepts would wholly preempt progress.\(^{81}\)

The Court first applied the exceptions to computer software in *Gottschalk v. Benson*,\(^ {82}\) where the Court examined the patentability of “a method of programming a general-purpose digital computer to convert signals from binary-coded decimal form into pure binary form.”\(^ {83}\) The algorithm could “be carried out in existing computers long in use, no new machinery being necessary,” and it could “also be performed without a computer.”\(^ {84}\) In its analysis, the Court relied on its § 101 exception that “an idea of itself is not patentable”\(^ {85}\) and clarified that “[p]henomena of nature, though just discovered, mental processes, and abstract intellectual concepts are not patentable, as they are the basic tools of scientific and technological work.”\(^ {86}\) It explained that, in determining whether the invention amounts to something more than a simple, unpatentable algorithm, it is relevant whether the invention includes a particular machine or transformation.\(^ {87}\) Because the “mathematical formula involved here ha[d] no substantial practical application except in connection with a digital computer, . . . [a] patent would wholly pre-empt the mathematical formula and in practical effect would be a patent on the algorithm itself.”\(^ {88}\) Accordingly, the Court determined that the software could not be patented.\(^ {89}\)

Several years later, in *Parker v. Flook*,\(^ {90}\) the Court assessed the patentability of a “Method for Updating Alarm Limits.”\(^ {91}\) In determining that the software at issue could not be patented,\(^ {92}\) the Court reiterated that disembodied algorithms are unpatentable and that patents are warranted on algorithm-based

---

\(^{81}\) *Id.* at 71 (“The Court has recognized, however, that too broad an interpretation of this exclusionary principle could evincepatent law. For all inventions at some level embody, use, reflect, rest upon, or apply laws of nature, natural phenomena, or abstract ideas.”).

\(^{82}\) *Benson*, 409 U.S. at 64–65, 67 (1972).

\(^{83}\) *Id.* at 65.

\(^{84}\) *Id.* at 67.

\(^{85}\) *Id.* at 67 (alteration omitted) (quoting Rubber-Tip Pencil Co. v. Howard, 87 U.S. 498, 507 (1874)). The Court explained: "A principle, in the abstract, is a fundamental truth; an original cause; a motive; these cannot be patented, as no one can claim in either of them an exclusive right." *Id.* (quoting Le Roy v. Tatham, 14 How. 156, 175 (1852)).

\(^{86}\) *Id.*

\(^{87}\) See *id.* at 70 (“Transformation and reduction of an article ‘to a different state or thing’ is the clue to the patentability of a process claim that does not include particular machines.”). The Court noted, however, that it was “not hold[ing] that no process patent could ever qualify [for patentability] if it did not meet the requirements of [the machine-or-transformation test].” *Id.* at 71.

\(^{88}\) *Id.* at 71–72.

\(^{89}\) See *id.* at 72–73.


\(^{91}\) *Id.* at 585.

\(^{92}\) The Court explained that “all that [the patent application] provide[d] [was] a formula for computing an updated alarm limit.” *Id.* at 586, 594.
inventions only if those inventions, themselves, are “new and useful.”93 Contrasting his claim to that in Benson, the inventor explained that granting the patent here would not wholly preempt the mathematical formula and that certain “'post-solution' activity— the adjustment of the alarm limit to the figure computed according to the formula— distinguishes[ed] his case from Benson and makes his process patentable.”94 The Court disagreed, however, explaining that “[t]he notion that post-solution activity, no matter how conventional or obvious in itself, can transform an unpatentable principle into a patentable process exalts form over substance.”95

Then, in 1981, the U.S. Supreme Court recognized for the first time that computer software could be patented.96 In that case, Diamond v. Diehr97 the Court cut back somewhat on Benson and Flook. The Diehr Court found the software at issue patentable because, although “several steps of the process [employed] a mathematical equation and a programmed digital computer,” it “[could not] be disputed” that the patent “claims involve[d] the transformation of an article, in this case raw, uncured synthetic rubber, into a different state.”98 After Diehr, then, software that relied on underlying algorithms could be patented so long as it was paired with a physical step, even if that physical step was not new and was instead already known in the art.99 This case thus opened the door to the patentability of software.

93 Id. at 591 (“Mackay Radio and Funk Bros, point to the proper analysis for this case: the process itself, not merely the mathematical algorithm, must be new and useful.”). Interestingly, the Court made clear that its focus on the newness and usefulness of the invention were questions of patentable subject matter under § 101 rather than questions of novelty under § 102 and obviousness under § 103. See id. at 588 (“This case turns entirely on the proper construction of § 101 . . . which describes the subject matter that is eligible for patent protection. It does not involve the familiar issues of novelty and obviousness that routinely arise under §§ 102 and 103 when the validity of a patent is challenged.” (footnote omitted)).
94 Id. at 590.
95 Id.
98 Diehr, 450 U.S. at 184.
99 See id. at 187; Brieanna Dolmage, The Evolution of Patentable Subject Matter in the United States, 27 WHITTIER L. REV. 1023, 1029 (2006) (“The Court’s decision in Diehr distinguished cases that sought to patent computer software using mathematical algorithms in conjunction with an additional physical step, even where the step was previously known in the art.”). Although the Court had previously “resisted the patenting of software programs, primarily on the ground that they constituted ‘mathematical algorithms,’ ” in Diehr the
Following Diehr, the Supreme Court did not address the question of the extent that software is patentable until 2014, but, in the intervening thirty-three years, the Court of Appeals for the Federal Circuit gradually expanded the patentability of software. It first applied a test to assess whether software is patentable, examining whether the claim preempted the algorithm entirely or applied the algorithm to physical elements or processes that, themselves, constituted patentable subject matter. Then, in 1994, the Federal Circuit abandoned this test and narrowed the limitation on patenting software. In In re Alappat, the Federal Circuit examined the patentability of an invention that smoothed out an oscilloscope display by manipulating pixel intensities based on their locations in the display. It found that the Diehr Court did not intend to create a broad categorical exception to patentable subject matter for software. Instead, the Federal Circuit explained that “[t]he plain and unambiguous meaning of § 101 is that any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may be patented if it meets the [other] requirements” in the Patent Act. Indeed, in the 1980 case of Diamond v. Chakrabarty, the Supreme Court had stated that patentable subject matter includes “anything under the sun that is made by man.” Referring back to Benson, Flook, and Diehr, the Federal Circuit explained that these seemingly limiting decisions simply clarified

“Court signaled a slight weakening in this resolve, upholding the patent on a software program (embedded in a computer) that served to monitor continuously the temperature inside a synthetic rubber mold.” Fisher, supra note 22, at 270.

See Alice Corp. v. CLS Bank Int’l, 573 U.S. 208 (2014) (returning to the question of subject matter eligibility in a software case).

See Dolmage, supra note 99, at 1030–35; Fisher, supra note 22, at 270 (explaining that, after Diehr, “the Federal Circuit . . . adopted an increasingly receptive posture” toward considering software to be patentable).

See In re Abele, 684 F.2d 902, 905 (C.C.P.A. 1982); In re Walter, 618 F.2d 758, 767 (C.C.P.A. 1980); In re Freeman, 573 F.2d 1237, 1245–47 (C.C.P.A. 1978) (“Determination of whether a claim preempts nonstatutory subject matter as a whole . . . requires a two-step analysis. First, it must be determined whether the claim directly or indirectly recites an ‘algorithm’ . . . . Second, the claim must be further analyzed to ascertain whether in its entirety it wholly preempts that algorithm.”); see also In re Bilski, 545 F.3d 943, 959 (Fed. Cir. 2008) (“This test, in its final form, had two steps: (1) determining whether the claim recites an ‘algorithm’ within the meaning of Benson, then (2) determining whether that algorithm is ‘applied in any manner to physical elements or process steps.’” (quoting In re Abele, 684 F.2d at 95–107)), aff’d but criticized sub nom. Bilski v. Kappos, 561 U.S. 593 (2010).

See In re Alappat, 33 F.3d 1526, 1544–45 (Fed. Cir. 1994).

Id. at 1526.

Id. at 1537.

Id. at 1543, 1543 n.20 (“A close analysis of Diehr, Flook, and Benson reveals that the Supreme Court never intended to create an overly broad, fourth category of subject matter excluded from § 101.”).

Id. at 1542 (emphasis added).


Id. at 309 (quoting S. REP. No. 82-1979, at 5 (1952), and H.R. REP. No. 82-1923, at 6 (1952)); see In re Alappat, 33 F.3d at 1542 (quoting Chakrabarty, 447 U.S. at 309).
“a rather straightforward concept, namely, that certain types of mathematical subject matter, standing alone, represent nothing more than abstract ideas until reduced to some type of practical application, and thus that subject matter is not, in and of itself, entitled to patent protection.”

Although a claim consisting entirely of a mathematical algorithm is unpatentable, an invention is not unpatentable simply because it contains such an algorithm, the court explained. If a claim includes something more than just an algorithm or other vague mathematical concept, then it would not be subject to the exceptions outlined in Benson, Flook, and Diehr. With respect to the invention at issue in In re Alappat, the court explained that the combination of elements “form[ed] a machine for converting discrete waveform data samples into anti-aliased pixel illumination intensity data to be displayed on a display means.”

“This,” the court explained, was “not a disembodied mathematical concept which may be characterized as an ‘abstract idea,’ but rather a specific machine to produce a useful, concrete, and tangible result.” Accordingly, it constituted, patentable subject matter. In other words, when the inventor claimed the algorithm in conjunction with a computer so that it could be applied in a practical way, the invention constituted a patentable “machine” under § 101.

The Federal Circuit continued welcoming software into the world of patentability when it decided State Street Bank & Trust Co. v. Signature Financial Group, Inc. and AT&T Corp. v. Excel Communications, Inc. a

110 In re Alappat, 33 F.3d at 1543 (Fed. Cir. 1994).
111 See id. at 1543–44. The court stated:

It is . . . not necessary to determine whether a claim contains, as merely a part of the whole, any mathematical subject matter which standing alone would not be entitled to patent protection. Indeed, because the dispositive inquiry is whether the claim as a whole is directed to statutory subject matter, it is irrelevant that a claim may contain, as part of the whole, subject matter which would not be patentable by itself. “A claim drawn to subject matter otherwise statutory does not become nonstatutory simply because it uses a mathematical formula, mathematical equation, mathematical algorithm, computer program or digital computer:”

Id. (emphasis, alterations, and footnote omitted) (quoting Diamond v. Diehr, 450 U.S. 175, 187 (1981)).
112 See id. at 1544.
113 Id.
114 Id.
115 See id. at 1544–45.
116 See id. (stating that “a computer operating pursuant to software may represent patentable subject matter, provided, of course, that the claimed subject matter meets all of the other requirements of Title 35” and finding that the claimed invention did not constitute “a disembodied mathematical concept which may be characterized as an ‘abstract idea,’ but rather a specific machine to produce a useful, concrete, and tangible result”); Steger, supra note 73, at 50. As Steven Steger explained, “[a] software program running on a general purpose computer creates a new machine because the general purpose computer becomes a special purpose computer when performing according to instructions received from the program.”

few years later. In these cases, the Federal Circuit made clear that software does indeed constitute patentable subject matter. In State Street, the court explained that only abstract, “disembodied” mathematical algorithms fall into the algorithm exception to patentability. When a computer transforms data through mathematical formulas and techniques and produces “a useful, concrete and tangible result,” the exception does not apply. In AT&T, the court doubled down, reiterating that the “inquiry . . . focuses on whether the mathematical algorithm is applied in a practical manner to produce a useful result.” More explicitly, the court explained that it was “now clear that computer-based programming constitutes patentable subject matter so long as the basic requirements of § 101 are met.” The Federal Circuit could not have been clearer, and this line of cases established stable footing for the patentability of computer software. In fact, following these cases, it was understood that “virtually any software program (if novel, nonobvious, etc.) [was] patentable, so long as the applicant describe[d] it as being programmed into a general purpose computer.” Perhaps not unexpectedly, these decisions resulted in “an enormous surge in software patent applications.”

C. From Patents to Trade Secrets

Beginning in 2010, the Supreme Court decided a series of cases drastically shrinking the area of patentable subject matter. In Bilski v. Kappos, the Court assessed the patentability of a method for hedging risk—a common economic practice in industry. The Court’s confusing opinion first explained that the “machine-or-transformation” test as articulated in Benson cannot be the sole test for assessing whether something constitutes a “process” under § 101; in-
stead, it is just “a useful and important clue” as to whether the invention in question is a process.129 Still, the Court concluded that the method was not a patentable “process” under § 101 because it was an “abstract idea”—falling under the § 101 exception to patentable subject matter.130

Two years later, in Mayo Collaborative Services v. Prometheus Laboratories, Inc.,131 the Court again addressed the categorical exceptions to § 101—“[l]aws of nature, natural phenomena, and abstract ideas.”132 It examined the patentability of “[a] method of optimizing therapeutic efficacy for treatment of an immune-mediated gastrointestinal disorder,” which included “administering a drug[,] . . . determining the level of [the drug in the patient],” and assessing whether the drug level is too high or low based on the inventors’ research findings.133 The Court determined that the method “set forth laws of nature—namely, relationships between concentrations of certain metabolites in the blood and the likelihood that a dosage of a . . . drug will prove ineffective or cause harm.”134 And the claimed “steps [were] not sufficient to transform un-patentable natural correlations into patentable applications of those regularities.”135 In reaching this conclusion, the Court took a tour through the Court’s § 101 exception cases, including Benson, Flook, Diehr, and Bilski.136 Notably, the Prometheus Court explained that, in Benson, the Court had “held that simply implementing a mathematical principle on a physical machine, namely, a computer, was not a patentable application of that principle. For the mathematical formula had ‘no substantial practical application except in connection with a digital computer.’”137

129 See id. at 604 (“This Court’s precedents establish that the machine-or-transformation test is a useful and important clue, an investigative tool, for determining whether some claimed inventions are processes under § 101. The machine-or-transformation test is not the sole test for deciding whether an invention is a patent-eligible ‘process.’”). The Court further explained that business methods are not categorically unpatentable under § 101. See id. at 606 (“Section 101 . . . precludes the broad contention that the term ‘process’ categorically excludes business methods.”).

130 See id. at 609.
132 Id. at 70 (alterations omitted) (quoting Diamond v. Diehr, 450 U.S. 175, 185 (1981)).
133 Id. at 74–75.
134 Id. at 77.
135 Id. at 80.
136 See id. at 70–73, 78–82, 85.
137 Id. at 84–85 (quoting Gottschalk v. Benson, 409 U.S. 63, 71 (1972)). The next year, in Association for Molecular Pathology v. Myriad Genetics, Inc., 569 U.S. 576 (2013), the Court found that “a naturally occurring segment of deoxyribonucleic acid (DNA)” that was “isolated[ed] from the rest of the genome” was not patentable because it was a naturally occurring phenomenon but that “synthetically created . . . complementary DNA . . . which contains the same protein-coding information found in a segment of natural DNA but omits portions within the DNA segment that do not code for proteins” was patentable under § 101. Id. at 580.
Finally, in *Alice Corp. v. CLS Bank International*, the Court applied these revitalizations of the § 101 exception to computer software.\(^{138}\) As the Court explained, “[t]he patents at issue in th[e] case disclos[e][d] a computer-implemented scheme for mitigating ‘settlement risk’ *(i.e., the risk that only one party to a financial transaction will pay what it owes)* by using a third-party intermediary.”\(^{139}\) The *Alice* Court clarified the approach it set out earlier in *Prometheus*, examining, first, whether the inventor’s claim was “directed to one of th[e] patent-ineligible concepts” such as algorithms and, second, whether the combination of claims transformed the invention “into a patent-eligible application”—whether there was an “inventive concept.”\(^{140}\) This approach is again based on the concern of placing a monopoly on a basic principle and stunting innovation.\(^{141}\) Then, relying heavily on *Prometheus* and *Bilski*, the Court concluded that the claims were “drawn to the abstract idea of intermediated settlement, and that merely requiring generic computer implementation fail[ed] to transform that abstract idea into a patent-eligible invention.”\(^{142}\) More specifically, the Court stated that “[t]he introduction of a computer into the claims does not alter the analysis . . . .”\(^{143}\) Quoting *Prometheus*, it reiterated: “[S]imply implementing a mathematical principle on a physical machine, namely a computer, is not a patentable application of that principle.”\(^{144}\) And the Court went even further:

> [T]he mere recitation of a generic computer cannot transform a patent-ineligible abstract idea into a patent-eligible invention. Stating an abstract idea “while adding the words ‘apply it’” is not enough for patent eligibility. Nor is limiting the use of an abstract idea “to a particular technological environment.” Stating an abstract idea while adding the words “apply it with a computer” simply combines those two steps, with the same deficient result. Thus, if a patent’s recitation of a computer amounts to a mere instruction to “implement” an abstract idea “on a computer,” that addition cannot impart patent eligibility. This conclusion accords with the pre-emption concern that undergirds our § 101 jurisprudence. Given the ubiquity of computers, wholly generic computer implementation is not generally the sort of “additional feature” that provides any “practical assurance that the process is more than a drafting effort designed to monopolize the abstract idea itself.”\(^ {145}\)

In short, the application of algorithms through the use of computer software generally constitutes unpatentable subject matter. The claimed invention must

---


\(^{139}\) *Id.*

\(^{140}\) *Id.* at 217 (quoting *Prometheus Lab’ys, Inc.*, 566 U.S. at 72–73).

\(^{141}\) See *id.* at 216 (“We have described the concern that drives this exclusionary principle as one of pre-emption.”); *supra* text accompanying note 81.

\(^{142}\) *Alice*, 573 U.S. at 212, 218–19, 221.

\(^{143}\) *Id.* at 222.

\(^{144}\) *Id.* (alteration omitted) (quoting *Prometheus Lab’ys, Inc.*, 566 U.S. 66, 84 (2012)).

\(^{145}\) *Id.* at 223–24 (alterations and citations omitted) (quoting *Prometheus Lab’ys, Inc.*, 566 U.S. at 72, 77, 84, and *Bilski v. Kappos*, 561 U.S. 593, 610–11 (2010)).
“do more than simply . . . implement the abstract idea . . . on a generic computer.”

These decisions shook the patent law community and the software industry. They have made it tremendously more difficult for inventors, and in particular software developers, to obtain patents. Within two years, it was estimated that the PTO had rejected around 60,000 patent applications due to Alice, and prosecutors had abandoned about 8,400 applications.

The Court’s recent § 101 decisions have also made existing patents more vulnerable to invalidation, as decisions like Alice retroactively affect patents that have already issued. In fact, two years after the Court decided Alice, federal courts saw a 400% increase in the number of § 101 cases they decided, and about 70% of these cases resulted in patent invalidation. This has pushed software developers away from trying to protect their inventions through patents.

Even beyond shrinking patentable subject matter, the Court has decided other patent cases resulting in making patent protection less desirable for software developers. It has heightened the non-obviousness requirement by shifting

---

146 Id. at 225. Since Alice was decided, lower courts have limited the case’s reach, finding something of a middle ground between Alice and its predecessors. See, e.g., Cardionet, LLC v. Infobionic, Inc., 955 F.3d 1358, 1374 (Fed. Cir. 2020).

147 See Robert Sachs, Two Years After Alice: The Survey of the Impact of a “Minor Case” (Part 2), BILSKI BLOG (June 20, 2016), https://www.bilskiblog.com/2016/06/two-years-after-alice-a-survey-of-the-impact-of-a-minor-case-part-2/#_ftn1 [https://perma.cc/MRJ5-23ZS] (explaining published data shows that 36,000 applications had been rejected based on Alice and that 5,000 applications had been abandoned but that, because published applications ordinarily account for only 60% of applications, the numbers are likely much higher); see also ANDREW A. TOOLE & NICHOLAS A. PAIROLERI, ADJUSTING TO ALICE: USPTO PATENT EXAMINATION OUTCOMES AFTER ALICE CORP. v. CLS BANK INTERNATIONAL 3 (2020), https://www.uspto.gov/sites/default/files/documents/OCE-DH_AdjustingtoAlice.pdf [https://perma.cc/72SA-SBWJ (“For Alice-affected technologies, the chances of receiving a first office action rejection with a rejection for patent-ineligible subject matter increased by 31% in the 18 months following Alice.”)]. The PTO has worked to mitigate this “Alice Effect,” and, a year after issuing further guidance on subject matter eligibility in January of 2019, “the chances of receiving a first office action rejection for patent-ineligible subject matter [had decreased] by 25% for Alice-affected technologies.” See id. at 6.

148 According to one author: “It is difficult to understake the impact of the Supreme Court’s decisions involving [patentable subject matter under] Section 101. Thousands of patents likely became invalid after these decisions, essentially rendering them worthless at best.” Hricik, supra note 36, at 470–71 (footnote omitted).


its analysis from a focus on piecing together prior art to one focused instead on common sense, market demand, or design trends.151 It has also bolstered the definiteness requirement.152 The Court has made it more difficult for patent holders to establish infringement by narrowing the doctrine of equivalents,153 limiting the circumstances under which infringement can occur outside of U.S. borders,154 making it more difficult for a patent holder to establish liability for one who induced another party to infringe,155 and expanding the conditions under which a patentee has “exhausted” his patent rights such that he cannot establish liability against the relevant user.156 The Court has also made patent defense tougher by increasing the number of individuals and entities that can work to invalidate a patent,157 limiting venue shopping among patent holders,158 making it easier for patent challengers to be awarded attorneys’ fees from the patentee’s purse,159 and making it more difficult for patentees to obtain injunc-

151 See KSR Int’l Co. v. Teleflex Inc., 550 U.S. 398, 421 (2007) (“When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense.”); see also Hricik, supra note 36, at 471–72 (suggesting that this change deters inventors from seeking patent protection).

152 See Nautilus, Inc. v. Biosig Instruments, Inc., 134 S. Ct. 2120, 2124 (2014) (“In place of the ‘insolubly ambiguous’ standard, we hold that a patent is invalid for indefiniteness if its claims, read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention.”); see also Hricik, supra note 36, at 472 (indicating that this change makes it more difficult to patent inventions).

153 See Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co., 535 U.S. 722, 726–27, 741 (2002) (“While estoppel does not effect a complete bar, the question remains whether petitioner can demonstrate that the narrowing amendments did not surrender the particular equivalents at issue.”); see also Hricik, supra note 36, at 473–75.

154 See Life Techs. Corp. v. Promega Corp., 137 S. Ct. 734, 737 (2017); Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 441–42 (2007); see also Hricik, supra note 36, at 477–78.

155 See Limelight Networks, Inc. v. Akamai Techs., Inc., 572 U.S. 915, 917 (2014) (explaining that “a defendant may [not] be liable for inducing infringement of a patent under 35 U.S.C. § 271(b) when no one has directly infringed the patent under § 271(a) or any other statutory provision”); Global-Tech Appliances, Inc. v. SEB S.A., 563 U.S. 574, 766 (2011) (“Accordingly, we now hold that induced infringement under § 271(b) requires knowledge that the induced acts constitute patent infringement.”); see also Hricik, supra note 36, at 476. But see Commil USA, LLC v. Cisco Sys., Inc., 135 S. Ct. 1920, 1928 (2015) (“The question the Court confronts today concerns whether a defendant’s belief regarding patent validity is a defense to a claim of induced infringement. It is not.”).


159 See Octane Fitness, LLC v. Icon Health & Fitness, Inc., 134 S. Ct. 1749, 1755–56 (2014); see also Hricik, supra note 36, at 482–84 (explaining the significance of the Octane Fitness decision in dissuading parties from obtaining patents).
tive relief even after they have established infringement.\textsuperscript{160} Finally, the Court has made it easier to challenge patent validity by creating new, post-grant administrative proceedings within the PTO.\textsuperscript{161} Such proceedings are significantly less expensive and generally more effective in invalidating a patent than challenging validity in the courts.\textsuperscript{162}

While the Court has been making patent protection more onerous for software developers, Congress has been working to make trade secret protection somewhat easier. One historical disadvantage of trade secret law is that it was rooted in state law and thus varied from jurisdiction to jurisdiction.\textsuperscript{163} In 2016, though, Congress passed the Defend Trade Secrets Act, which for the first time created a federal private cause of action for trade secret misappropriation.\textsuperscript{164} The recent provision of federal protection makes trade secret protection more desirable. Further, the new federal law expanded trade secret protection beyond the limits that states were providing.\textsuperscript{165} This has made trade secret law an even more valuable source of information protection.

\textsuperscript{160} See eBay, Inc. v. MercExchange, LLC, 547 U.S. 388, 391 (2006); see also Hricik, supra note 36, at 485–86.

\textsuperscript{161} See Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) (amending various parts of Title 35); see also Hricik, supra note 36, at 489.

\textsuperscript{162} See Hricik, supra note 36, at 489–90 (“Post-grant proceedings are substantially less expensive than defending a patent suit. In addition, three features common to each proceeding, separately and together, make it demonstrably more likely that patents can be successfully challenged under these new post-grant proceedings than is the case in litigation.” (footnote omitted)). When challenging patent validity in the district courts, a presumption of validity (clear and convincing evidence) applies, whereas a determination of invalidity in a post-grant proceeding is based on just a preponderance of the evidence. See id. Additionally, claims are viewed more broadly in post-grant proceedings than in court, increasing the likelihood of an invalidity determination. See id. Finally, a court may stay a patentee’s infringement lawsuit if the defendant can convince the PTO that there is a reasonable likelihood that one of the patent claims is invalid. See id.

\textsuperscript{163} See S. REP. No. 114-220, at 2–3 (2016) (explaining that the differences between state laws can be significant, potentially affecting “which party has the burden of establishing that a trade secret is not readily ascertainable, whether the owner has any rights against a party that innocently acquires a trade secret, the scope of information protectable as trade secret, and what measures are necessary to satisfy the requirement that the owner employ ‘reasonable measures’ to maintain secrecy of the information”); Explaining the Defend Trade Secrets Act, BUS. L. TODAY (Sept. 20, 2016), https://www.americanbar.org/groups/business_law/publications/blt/2016/09/03_cohen/ [https://perma.cc/3PNV-39VZ] (“Before the enactment of the DTSA, . . . companies seeking redress for trade-secret misappropriation [often] had no choice but to sue in state court, where laws protecting against trade-secret misappropriation tend to differ from state to state both in the text of the laws themselves and in their application.”). But see S. REP. No. 114-220, at 2–3 (2016) (explaining that the UTSA “has been adopted (in its entirety or with some modifications) in 47 States and the District of Columbia” and noting that “the differences between State laws and the UTSA are generally relatively minor”).


\textsuperscript{165} Not only does the DTSA define “trade secret” more broadly than the UTSA, see Joseph D. Mornin, What You Need to Know About the Defend Trade Secrets Act, 28 INTELL. PROP. & TECH. L.J. 20, 21 (2016) (“The DTSA definition of ‘trade secret’ is broader than the UTSA definition.”), but it also, for example, “allows ex parte seizures of misappropriated trade se-
All of these changes in IP law have pushed software developers deeper into the world of trade secrets.\textsuperscript{166} If patents are more difficult to obtain and easier to invalidate, the trade-off of disclosing your inventions in exchange for a twenty-year much-more-limited patent is less worthwhile.

II. PRIVATE ALGORITHMS, PUBLIC INTERESTS

In the wake of these movements in IP law, trade secret law has emerged as a useful means of software protection. It is debatable whether the resulting broad secrecy surrounding valuable software could spur or quell competition within the industry and serve as a boon or bust to consumers. Within particular pockets of public law, though, this secrecy has created significant public policy, statutory, and even constitutional concerns. These serious issues arise from several potential shortcomings of algorithms that secrecy can mask. First is the problem of inaccuracy. This could result in travesties such as defendants being convicted and deprived of their liberty or lives based on erroneous machine outputs.\textsuperscript{167} Another potential problem of algorithms is bias. This could mean that an individual is deprived of fair treatment in housing, employment, or even services because of algorithmic injustice.\textsuperscript{168} A third concern is security. In the realm of voting, for example, secrecy masking the security or insecurity of electronic voting machines could result in undermining democracy.\textsuperscript{169}

Overall, the broad secrecy resulting in part from recent changes in IP law has created significant concerns. Secret algorithms used in areas that affect the public interest are particularly problematic. Issues are already arising in a whole host of contexts, including criminal justice, housing, and voting.

A. Criminal Justice

Secret algorithms are starting to play a major role in the world of criminal justice. Police officers, judges, lawyers, and forensic scientists are making use of new science– and technology–based algorithms to measure blood-alcohol levels, match fingerprints, predict recidivism for bail and sentencing purposes, improve policing, and gather and act on other data.\textsuperscript{170} Some advantages of relying on algorithms in these contexts is that they offer the potential to provide greater fairness to defendants across cases and remove judicial and other con-

\begin{footnotesize}
\begin{itemize}
\item See infra Section II.A.
\item See infra Section II.B.
\item See infra Section II.C.
\end{itemize}
\end{footnotesize}
scopic and unconscious biases from criminal justice decisionmaking.\footnote{See \textit{id.} at 281–87.} They also may lend greater efficiency to the criminal justice system.\footnote{See \textit{id.} at 286.}

Despite these advantages of the purported evidence-based approach that these algorithms offer, there are real questions about the accuracy of these algorithms. For example, in a 2016 study of COMPAS—an algorithm-based program used to predict recidivism for criminal justice decisionmaking—ProPublica found the program to be only “somewhat more accurate than a coin flip.”\footnote{See Julia Angwin et al., \textit{Machine Bias}, PROPUBLICA (May 23, 2016), https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing [https://perma.cc/8GDM-X55C].} And even more concerning than the accuracy of prediction programs like COMPAS are the questions about the accuracy of “conviction programs”\footnote{Ryan, \textit{supra} note 12, at 270.}—programs that criminal justice actors use to actually determine the guilt or innocence of a criminal defendant.\footnote{See \textit{id.} at 293 (explaining that “more is simply at stake when the question is one of innocence rather than one of temporary pretrial confinement or punishment for those already determined guilty”).} Prosecutors regularly present evidence in court based on the secret algorithms embedded in breathalyzers, automated fingerprint identification systems, and probabilistic genotyping systems.\footnote{See \textit{id.} at 292–93.} Judges ordinarily readily admit these computer outputs into evidence, and judges and jurors often eagerly convict defendants based on this evidence.\footnote{See \textit{id.} at 272 (“Yet even more is at stake—a defendant’s liberty or even his life—where conviction programs are involved.”).} All of this could lead to a long term of imprisonment or even death for the defendant.\footnote{See \textit{id.} at 303 (“Although much is at stake where convictions are involved, and although prosecutors regularly present evidence produced by conviction programs as nearly indisputable, there are real questions about the accuracy of the outputs that these conviction programs produce and that prosecutors rely on so heavily today.”); cf. PASQUALE, \textit{supra} note 31, at 14–15 (“Although internet giants say their algorithms are scientific and neutral tools, it is very difficult to verify those claims. And while they have become critical economic infrastructure, trade secrecy law permits managers to hide their methodologies, and business practices, deflecting scrutiny.” (footnote omitted)).} Yet, the true accuracy of these tools often remains unknown.\footnote{See \textit{id.} at 305–23.} When defendants have requested access to the underlying source codes and algorithms powering these programs and producing their outputs, judges have ordinarily denied their requests.\footnote{See \textit{id.} at 310, 316.} Yet, on occasion, the public has gotten a glimpse into the potential inaccuracies propagated by these secret programs.\footnote{See \textit{id.} at 310, 316.} For example, in one case out of New York, two rival probabilistic genotyping systems—TrueAllele and STRmix—were used to determine
whether the DNA found at a murder scene matched the defendant’s DNA.\textsuperscript{182} The two programs produced contradictory results, indicating that one of them had to be incorrect.\textsuperscript{183} And errors are not limited to new technologies like probabilistic genotyping systems. When experts examined breathalyzer source code that a judge (anomalously) ordered released in New Jersey, they discovered algorithmic errors leading to erroneous blood-alcohol concentration outputs that had been regularly used in criminal cases.\textsuperscript{184}

Beyond this striking inaccuracy concern, algorithm-based programs also raise questions of bias—that these algorithms unjustly discriminate against certain groups of people.\textsuperscript{185} For example, the ProPublica study noting the unreliability of COMPAS’s recidivism predictions also exposed some of the biases embedded in the algorithm’s outcomes.\textsuperscript{186} Specifically, ProPublica explained that the program “turned up significant racial disparities.”\textsuperscript{187} In predicting recidivism, “[t]he formula was particularly likely to falsely flag black defendants as future criminals, wrongly labeling them this way at almost twice the rate as white defendants. . . . White defendants were mislabeled as low risk more often than black defendants.”\textsuperscript{188} Even after accounting for criminal history, type of


\textsuperscript{183} See id.

\textsuperscript{184} See State v. Chun, 943 A.2d 114, 157 (N.J. 2008) (explaining that one “expert . . . identified a significant flaw in the [breathalyzer] program’s source code that, in limited circumstances, can lead to an inaccurate reported BAC test result” and that, although an opposing expert “disputed many of the conclusions proffered by defendants’ experts, . . . he acknowledged and explained the buffer overflow defect, admitting that he was responsible for the inclusion of this error in the code”); see also State v. Underdahl, 767 N.W.2d 677, 685 (Minn. 2009) (noting the appellant’s report, which “analyzed the New Jersey machine’s computer source code and uncovered a variety of defects that could impact the test result”).

\textsuperscript{185} One of the draws of relying on algorithms is that they have a reputation of being more objective than humans, and in that sense they provide promise for shedding bias in important decisionmaking. \textit{See supra} text accompanying note 171. But algorithms can also be biased. Sometimes the bias is baked in through biased computer programmers or bad data. \textit{See} Rumman Chowdhury & Narendra Mulani, \textit{Auditing Algorithms for Bias, HARV. BUS. REV.} (Oct. 24, 2018), https://hbr.org/2018/10/auditing-algorithms-for-bias [https://perma.cc/QDQ8-ESRV]. Rumman Chowdhury and Narendra Mulani have questioned:

\begin{quote}
Can AI provide the veil of ignorance that would lead us to objective and ideal outcomes? . . . The answer so far has been disappointing. However objective we may intend our technology to be, it is ultimately influenced by the people who build it and the data that feeds it. Technologists do not define the objective functions behind AI independent of social context. Data is not objective, is it [sic] reflective of pre-existing social and cultural biases. In practice, AI can be a method of perpetuating bias, leading to unintended negative consequences and inequitable outcomes.
\end{quote}

\textit{Id.} Complementing this baked-in bias, sometimes algorithm results create a disparate impact on minorities simply because of the particular factors used to assess the issue in question.

\textsuperscript{186} Angwin et al., \textit{supra} note 173.

\textsuperscript{187} \textit{Id.}

\textsuperscript{188} \textit{Id.}
crime, race, and gender, ProPublica found that “[b]lack defendants were still 77 percent more likely to be pegged as at higher risk of committing a future violent crime and 45 percent more likely to be predicted to commit a future crime of any kind.”189 In other words, the algorithm was biased.190

Another prediction algorithm used within the criminal justice system is PredPol—software developed for police departments that allows police officers to concentrate their limited resources in areas that are probabilistically most likely to see future crime.191 This software also comes with concerns about biases. Some commentators have characterized it as discriminatory in nature, because PredPol’s outputs generally direct law enforcement to police poor, minority neighborhoods more closely than other neighborhoods.192

Despite this evidence of inaccuracies and biases in both prediction and conviction programs, the true extent of these issues remains veiled by the secrecy enveloping these programs.193 Often the developers of these algorithms are private, for-profit entities, and they understandably want to maintain their monopoly over the programs to amass the greatest returns on their investments.194 As such, these developers claim trade secret protection over their programs, and judges have generally upheld these claims.195 Perhaps surprisingly, even the police officers, forensic scientists, and other experts who use and rely

189 Id.
190 The picture is actually more complicated, though. Northpointe, “the company that developed COMPAS,” and several criminologists disputed this conclusion. See Matthias Spielkamp, Inspecting Algorithms for Bias, MIT TECH. REV. (June 12, 2017), https://www.technologyreview.com/2017/06/12/105804/inspecting-algorithms-for-bias/ [https://perma.cc/FA53-ZJPA]. And, in fact, whether the algorithm is fair depends upon how you measure fairness. See id. Cathy O’Neil has asserted that, although “the COMPAS scoring system is well-calibrated,” it was unfair. CATHY O’NEIL, WEAPONS OF MATH DESTRUCTION 224 (2016). “[T]he percentage of high-risk scores among blacks and among whites matches the actual rate of recidivism among blacks and among whites[,]” but ProPublica “found there were twice as many false positives for blacks as for whites, and twice as many false negatives for whites as for blacks.” Id.
191 See PREDPOL, https://www.predpol.com/ [https://perma.cc/9XZN-VM2A]; see also O’NEIL, supra note 190, at 85 (“Predictive programs like PredPol are all the rage in budget-strapped police departments across the country.”).
193 See Ryan, supra note 12, at 304–05.
194 See id. at 324 (“One of the reasons that these algorithms and source codes are kept secret is that outside companies have created them and rely on this secrecy to make profits; the algorithms and source codes are proprietary in nature.”).
195 See id. at 304–23.
on these programs are not privy to their algorithms.\textsuperscript{196} The government may have asked for validation information upon purchasing or licensing the programs, but there often has been no thorough, independent review of the instruments producing outputs that could affect the fates of criminal defendants.\textsuperscript{197} This is just astounding.

The secrecy surrounding these algorithms that affect the dispositions of criminal cases and criminal defendants’ futures goes beyond shocking and may even amount to constitutional violations.\textsuperscript{198} Under the Supreme Court’s revitalization of the Confrontation Clause in \textit{Crawford v. Washington}\textsuperscript{199} and its progeny,\textsuperscript{200} denying criminal defendants access to evidence that the prosecution is relying on for convictions could constitute a Sixth Amendment violation.\textsuperscript{201} Even if a forensic scientist conducts a probabilistic genotyping system test on a DNA sample and the defense has the opportunity to cross-examine the forensic scientist about the report, because the forensic scientist has little to no knowledge about how the probabilistic genotyping system functions, this cross-examination may not be very useful to the defense.\textsuperscript{202} Instead, one could say that the Confrontation Clause requires that the defense has the opportunity to cross-examine the developer of the algorithm or have access to the algorithm itself.\textsuperscript{203} Intertwined with this Sixth Amendment Confrontation Clause argument is the concern that allowing judges and juries to base convictions on the prosecution’s presentation of secret evidence amounts to a Due Process violation as well.\textsuperscript{204} Among many other things, the Due Process Clause requires that criminal defendants have a “fair opportunity to defend against the State’s accusations.”\textsuperscript{205} This right even includes providing defendants with the necessary

\textsuperscript{196} \textit{See id.} at 294, 297–98.

\textsuperscript{197} \textit{See id.} at 295 (“In reality, though, there may be insufficient research or validation studies propelling many of these programs or the algorithms and source codes on which they are built.”).

\textsuperscript{198} \textit{See id.} at 329–41. In addition to the Confrontation Clause and Due Process arguments I discuss here, convicting criminal defendants based on secret evidence also raises concerns about constitutional violations under the disclosure requirements of \textit{Brady v. Maryland}, 373 U.S. 83 (1963), and its progeny. \textit{See Ryan, supra} note 12, at 339–40.


\textsuperscript{200} \textit{See, e.g., Bullcoming v. New Mexico}, 564 U.S. 647, 657–58 (2011) (requiring that the forensic examiner conducting forensic tests later presented in court must be the same individual that the defendant has the opportunity to cross-examine about the report); \textit{Melendez-Diaz v. Massachusetts}, 557 U.S. 305, 310–11 (2009) (determining that a forensic report is “testimonial” such that the Confrontation Clause requirements apply and thus requiring that the defendant have the opportunity to cross-examine the expert about the report).

\textsuperscript{201} \textit{See Ryan, supra} note 12, at 334–38 (“Defendants’ inabilitys to gain access to information that may be essential to presenting a complete defense thus poses a significant constitutional concern under both the Due Process Clauses and the Confrontation Clause.”).

\textsuperscript{202} \textit{See id.} at 336–37.

\textsuperscript{203} \textit{See id.}

\textsuperscript{204} \textit{See id.} at 329–41.

\textsuperscript{205} \textit{Chambers v. Mississippi}, 410 U.S. 284, 294 (1973); \textit{see also} Holmes v. South Carolina, 547 U.S. 319, 324 (2006) (“Whether rooted directly in the Due Process Clause of the Four-
financial resources to do so.\textsuperscript{206} Denying defendants access to the details of the evidence presented against them robs them of this fair opportunity.\textsuperscript{207}

Overall, in the world of criminal justice, relying on algorithms to dole out more uniform justice certainly has its benefits—and its proponents—but it also creates some concerns that are tied to the lack of transparency involved in relying on these algorithms. When the price of relying on algorithms in this context is a potential violation of important constitutional rights, it raises the questions of whether we should really be using algorithms in this way and, if we do, whether we can make uses of them more transparent to avoid these constitutional concerns.

B. Housing

There are also secret algorithms at play in the housing context that similarly raise serious concerns—particularly with respect to bias. Algorithm-created bias is a phenomenon across fields, but algorithmic bias in the context of the basic human right to housing\textsuperscript{208} is especially troubling.

Discrimination in housing has a long history in this country.\textsuperscript{209} Landlords historically turned away Blacks who sought to lease their properties, neighborhood associations refused to let Blacks move into the area, banks refused to offer mortgages to minorities, and the federal government refused to insure any

\textsuperscript{206} See Ake v. Oklahoma, 470 U.S. 68, 76–77 (1985) (reiterating that “mere access to the courthouse doors does not by itself assure a proper functioning of the adversary process” and explaining that “a criminal trial is fundamentally unfair if the State proceeds against an indigent defendant without making certain that he has access to the raw materials integral to the building of an effective defense”).

\textsuperscript{207} See Ryan, supra note 12, at 335 (“Defendants’ inability to gain access to information that may be essential to presenting a complete defense thus poses a significant constitutional concern under both the Due Process Clause and the Confrontation Clause.”).

\textsuperscript{208} See Carlie Armstrong, Slow Progress: New Federal Rules Only Begin to Address Housing Discrimination Based on Sexual Orientation and Gender Identity, 9 MOD. AM. SUMMER 2013, at 2, 3 (explaining that Congress has expanded the Fair Housing Act because of “a growing recognition of housing as a fundamental human right”); Miles Walser, Note, Putting the Brakes on Rent Increases: How the United States Could Implement German Anti-Gentrification Laws Without Running Afoul of the Takings Clause, 36 WIS. INT’L L.J. 186, 209–10 (2019) (“In the United States, the human right to housing is, thanks in part to housing rights activists, arguably a part of these baseline values, even if not yet formally recognized.”).

minors who were successful in obtaining mortgages.210 In 1968, Congress passed the Fair Housing Act, which prohibits discrimination in housing on the basis of “race, color, religion, sex, familial status, or national origin.”211 Despite this action, discrimination in housing remains a persistent, pernicious problem. It may not be as widespread or obvious as it once was, but evidence suggests that individuals and corporations continue to discriminate against minorities—whether intentionally or unintentionally—on a regular basis.

Today, discrimination in housing is often masked by secret algorithms protected by company trade secrets. Still, some limited investigations have found evidence of such discrimination. For example, following an in-depth study of algorithmic discrimination,212 in March 2019, the U.S. Department of Housing and Urban Development (HUD) filed suit against Facebook for allowing advertisers to discriminate in their housing advertisements on the Facebook platform and also for Facebook’s own discrimination in using an algorithm that shows advertisements to only particular users based upon the users’ characteristics, including proxies for race.213

212 See Julia Angwin et al., Facebook (Still) Letting Housing Advertisers Exclude Users by Race, PROPUBLICA (Nov. 21, 2017, 1:23 PM), https://www.propublica.org/article/facebook-advertising-discrimination-housing-race-sex-national-origin [https://perma.cc/K3WQ-33F8]; Julia Angwin & Terry Parris, Jr., Facebook Lets Advertisers Exclude Users by Race, PROPUBLICA (Oct. 28, 2016, 1:00 PM), https://www.propublica.org/article/facebook-lets-advertisers-exclude-users-by-race [https://perma.cc/8J43-B4CK]; see also Ariana Tobin, HUD Sues Facebook Over Housing Discrimination and Says the Company’s Algorithms Have Made the Problem Worse, PROPUBLICA (Mar. 28, 2019, 1:18 PM), https://www.propublica.org/article/hud-sues-facebook-housing-discrimination-advertising-algorithms [https://perma.cc/U5M6-WYNJ] (“ProPublica first reported in 2016 that Facebook allowed housing advertisers to exclude users by race. Then in 2017, ProPublica found that—despite Facebook’s promised changes—the company was still letting advertisers exclude users by race, gender, ethnicity, family status, ability and other characteristics.”).
With respect to allowing the advertisers, themselves, to discriminate on the Facebook platform, HUD alleged the following:

Respondent [Facebook] has provided a toggle button that enables advertisers to exclude men or women from seeing an ad, a search-box to exclude people who do not speak a specific language from seeing an ad, and a map tool to exclude people who live in a specified area from seeing an ad by drawing a red line around that area. Respondent also provides drop-down menus and search boxes to exclude or include (i.e., limit the audience of an ad exclusively to) people who share specified attributes. Respondent has offered advertisers hundreds of thousands of attributes from which to choose, for example to exclude “women in the workforce,” “moms of grade school kids,” “foreigners,” “Puerto Rico Islanders,” or people interested in “parenting,” “accessibility,” “service animal,” “Hijab Fashion,” or “Hispanic Culture.” Respondent also has offered advertisers the ability to limit the audience of an ad by selecting to include only those classified as, for example, “Christian” or “Childfree.”

Beyond that, Facebook also allows advertisers to select “Custom Audiences” and “Lookalike Audiences,” which empowers the advertiser to target a specific list of users to view the advertisement, choose the viewers based on users’ interactions with the advertisers’ other content, or select viewers based on the characteristics they share with the advertisers’ existing customers.

Even aside from discrimination resulting from advertisers’ selections of audiences, HUD alleged that Facebook’s proprietary algorithm, itself, discriminated in violation of the Fair Housing Act. According to HUD:

During . . . the ad delivery phase,Respondent selects from among the users eligible to see an ad which users will actually see it. Respondent bases this decision in large part on the inferences and predictions it draws about each user’s likelihood to respond to an ad based on the data it has about that user, the data it has about other users whom it considers to resemble that user, and the data it has about “friends” and other associates of that user. To decide which users will see an ad, Respondent considers sex and close proxys for the other protected classes. Such proxies can include which pages a user visits, which apps a user has,

to the company’s user base, a request the social media giant denied because it would have set a dangerous precedent.” Jan & Dwoskin, supra. The Washington Post Editorial Board has cautioned that we ought to be wary of the motives behind the lawsuit. See Ed. Bd., Discrimination on Facebook Is a Real Problem, But Carson’s Motives Warrant Skepticism, Wash. Post (Mar. 31, 2019, 1:47 PM), https://www.washingtonpost.com/opinions/discrimination-on-facebook-is-a-real-problem-but-carsons-motives-warrant-skepticism/2019/03/31/0c28fd6e-524f-11e9-8d28-f5149e5a2fda_story.html [https://perma.cc/6CSM-3SUE] (“[S]ome skepticism is warranted.”). It notes that, “as a matter of policy, the housing department under this administration has seemed more interested in dismantling housing protections than ensuring they are respected” and suggests that Secretary Carson’s actions likely stem from the general sentiment by Republicans and the White House that tech companies are biased against right-wing politicians and ideas. Id. The Board suggests that this could be just one more attack on tech companies, and that a crack-down on discrimination in housing “may be happening for the wrong reason.” Id.


215 See id. at para. 15–16.

216 See id. at para. 17.
where a user goes during the day, and the purchases a user makes on and offline. Respondent alone, not the advertiser, determines which users will constitute the “actual audience” for each ad.\textsuperscript{217}

HUD Secretary Ben Carson explained that “[u]sing a computer to limit a person’s housing choices can be just as discriminatory as slamming a door in someone’s face.”\textsuperscript{218} Commentators have characterized Facebook’s practices as “massively illegal [and] . . . as blatant a violation of the federal Fair Housing Act as one can find.”\textsuperscript{219} They have also characterized these practices as a blatant violation of the Civil Rights Act of 1964.\textsuperscript{220}

Facebook’s tactics are not unique. Twitter and Google, for example, use similar algorithms to target advertisements, and their conduct has also been under review by HUD.\textsuperscript{221} As one reporter noted, “[t]he [Facebook] case is likely to have ripple effects throughout the tech industry, which considers targeting advertising to be standard practice and has historically enjoyed immunity from prosecution when third parties commit abuses on their platforms.”\textsuperscript{222}

Targeted advertising is not the only area in which reliance on algorithms has contributed to the already existing problem of discrimination in housing. For example, a 2019 study uncovered biases against minorities in the algorithms mortgage companies rely on in determining how much interest to charge mortgagees.\textsuperscript{223} According to the study, mortgage companies relying on algorithms regularly set mortgage purchase prices higher for African American and Latinx mortgagees than Caucasian mortgagees to the tune of more than five additional basis points.\textsuperscript{224} This costs these minority groups about an extra $250–$500 million each year.\textsuperscript{225} This sort of discrimination related to housing could

\textsuperscript{217} Id. at para. 17.
\textsuperscript{218} Tobin, supra note 212.
\textsuperscript{219} Angwin & Parris, supra note 212 (quoting “prominent civil rights lawyer John Relman”).
\textsuperscript{220} See id.
\textsuperscript{221} See Jan & Dwoskin, supra note 213.
\textsuperscript{222} Id.
\textsuperscript{224} See Bartlett et al., supra note 223 at 6 (“Latinx and African-American pay 5.3 basis points more in interest for purchase mortgages and 2.0 basis points for refinance mortgages originated on FinTech platforms.”). Despite this discrimination, the algorithms do seem to cause significantly less discrimination than face-to-face encounters. See id. (“FinTech algorithms discriminate 40% less than face-to-face lenders . . . .”).
\textsuperscript{225} See id. at 5 (“Averaging across the distribution of these products in the U.S., lending discrimination currently costs African-Americans and Latinx borrowers $765 million in extra interest per year.”).
very well also be a violation of the Fair Housing Act, as it shows a disparate impact on minorities for a reason unrelated to a legitimate business necessity.\textsuperscript{226}

Despite this evidence of pernicious biases within the housing context and across public-interest sectors, the full extent of such biases resulting from reliance on algorithms is uncertain. Again, algorithm developers usually refuse to reveal their algorithms and instead operate under the protection of trade secret law. Thus, while biases can sometimes be detected through obvious disparate impacts, researchers generally engage in these analyses in only a relatively small number of instances. Moreover, algorithm secrecy makes it exceedingly difficult to assess biases baked into the algorithms themselves.

\section*{C. Voting}

The increasing secrecy surrounding software has created somewhat different concerns in the world of election law. Voting in elections has long raised a number of difficulties. Two particular concerns that stand out are the fears of inaccurate and fraudulent results.\textsuperscript{227} How can we trust election officials to accurately count millions of votes and, further, what prevents these election officials—and others—from surreptitiously flipping or discarding votes to change election outcomes? During the infamous 2000 U.S. presidential election, Florida was mired in confusion about how punch ballots—with “hanging chads”—should be counted.\textsuperscript{228} Following federal legislation, as well as some litigation, many jurisdictions shifted toward electronic voting.\textsuperscript{229} Indeed, most states today conduct at least some of their voting by electronic voting machine.\textsuperscript{230} Casting

\begin{footnotesize}
\begin{enumerate}
\item See 42 U.S.C. § 3605; Bartlett et al., \textit{supra} note 223, at 25 n.20 (noting “that a lender that intentionally treated applicants differently based on a protected characteristic would be liable under the \textit{disparate treatment} theory of discrimination, for which there is no legitimate-business-necessity defense”).
\item See Kristen E. Eichensehr, \textit{Giving Up on Cybersecurity}, 64 UCLA L. REV. DISCOURSE 320, 328 (2016) (“\textquotesingle In the wake of the controversy about \textquotesingle hanging chads\textquotesingle in the 2000 presidential election, jurisdictions across the United States moved to modernize their voting equipment, including by adopting electronic voting machines or direct record electronic machines.\textquotesingle”; Daniel P. Tokaji, \textit{The Paperless Chase: Electronic Voting and Democratic Values}, 73 FORDHAM L. REV. 1711, 1713 (2005) (“Spurred by both legislation and litigation, states from Florida to California have taken steps to replace the infamous ‘hanging chad’ punch card with more modern—and supposedly more reliable—voting technology.”)).
\item See Voting Methods and Equipment by State, BALLOTENDB, https://ballotpedia.org/Voting\_methods\_and\_equipment\_by\_state [https://perma.cc/2K3T-J63Z] (providing information on voting equipment in use in each state as of November 2018). These machines are also
\end{enumerate}
\end{footnotesize}
voting electronically offers the potential to quickly and accurately count votes and present electoral outcomes, and electronic voting machines are also meant to make voting easier and reduce the state costs of voting. But along with these benefits, automation brings new risks. Although electronic voting machines mitigate the concern that officials counting individual votes might miscount or that they could intentionally change election outcomes, electronic voting machines create new risks of inaccuracy and fraud. There is the possibility that the algorithms and source codes on which the machines rely could contain errors. And there is the possibility that someone could hack the system to flip, discard, or manufacture votes, changing electoral outcomes. Unfortunately, these are not just hypothetical situations. They have already occurred to a surprising extent.

There are several stories where the inaccuracies of electronic voting machines are apparent:

In Fairfax County, VA, in 2003, a programming error in the electronic voting machines caused them to mysteriously subtract 100 votes from one particular candidate’s totals.

In San Bernardino County, CA in 2001, a programming error caused the computer to look for votes in the wrong portion of the ballot in 33 local elections, which meant that no votes registered on those ballots for that election. A recount was done by hand.

In Volusia County, FL in 2000, an electronic voting machine gave Al Gore a final vote count of negative 16,022 votes.

The 2003 election in Boone County, IA, had the electronic vote-counting equipment showing that more than 140,000 votes had been cast in the Nov. 4 municipal elections. The county has only 50,000 residents and less than half of them were eligible to vote in this election.

There are literally hundreds of similar stories.

Other inaccuracies may be more subtle. In recent years, there have been reports of vote-flipping in Georgia, Nevada, North Carolina, Pennsylvania, Tennessee, and Texas. Vote flipping can result from calibration or degradation of

known as “Direct Recording Electronic (DRE) Systems.” See id. Although most states use these machines, eighteen states still rely exclusively on paper ballots, and three states conduct voting through mail. See id.

See Blanc, supra note 227, at 12–14.

See id.


the touch screen components that voters use to record their votes.\textsuperscript{235} This is a growing problem as states’ electronic voting machines are being used past their expiration dates.\textsuperscript{236} In 2018, the Brennan Center reported that forty-one states used electronic voting machines that were at least ten years old—“dangerously close to or past the end of the expected lifespan for the core components of [the machines].”\textsuperscript{237} Further, many of the electronic voting machines currently in use are no longer manufactured, which has left election officials combing through Craigslist or eBay for spare parts.\textsuperscript{238} In addition to age and maintenance concerns, there is little verification that the machines properly record votes in ordinary course. Although some voting machines provide paper receipts to voters who then can ensure that their votes were correctly recorded, several states’ machines do not provide this information, and, in some states, only a fraction of the machines in service provide these receipts.\textsuperscript{239} Further, even when machines

\textsuperscript{235} See Halpern, supra note 234 (“Typically machines flip votes because they aren’t properly calibrated. This can happen, and does happen, to candidates from any party.”); Lau, supra note 234.

\textsuperscript{236} See Halpern, supra note 234 (“The voting machines purchased back in the early two-thousands were never meant to last this long. They have a shelf life of ten, maybe fifteen years. Many are no longer made, or the companies that manufactured them have gone out of business, or both.”); Lau, supra note 234; see also Jonathan Lai, N.J. Was Going to Have Paper-Based Voting Machines More than a Decade Ago. Will It Happen by 2020?, Phila. Inquirer (Mar. 10, 2019), https://www.inquirer.com/politics/new-jersey/nj-voting-machines-paper-trail-20190310.html [https://perma.cc/Y22H-WEYV] (stating that New Jersey’s “current machines are nearing death”).

\textsuperscript{237} Lau, supra note 234.

\textsuperscript{238} See Halpern, supra note 234 (“To get spare parts, election officials have had to scour eBay and Craigslist, looking for old machines that other municipalities have discarded.”); see also Lau, supra note 234 (“As of 2018, 43 states and the District of Columbia use polling machine models that are no longer manufactured. The use of these machines is troubling because in the event of a technical malfunction, it can be difficult to find replacement parts and technicians with the ability to repair them.”).

\textsuperscript{239} See Voting Methods and Equipment by State, supra note 230; see also Voting System Paper Trail Requirements, Nat'l Conf. State Legislators (June 27, 2019), http://www.ncsl.org/research/elections-and-campaigns/voting-system-paper-trail-requirements.aspx [https://perma.cc/Q4A2-PYQ6]; see also Castro, supra note 229, at 7 (suggesting “voter-verified paper audit trails” would allow the voter to verify that her vote was correctly recorded but explaining “[r]equesting that voter-verified paper audit trails be generated by DRE voting machines would increase the cost and complexity of elections”). Importantly, many states simply do not have the funds to upgrade their electronic voting machines to ones that provide paper trails. See Voting System Paper Trail Requirements, supra (explaining “that some states . . . may have [a statutory] requirement for a voter-verifiable paper record but have not had the funding to replace voting equipment in recent years, so in practice may have machines without a paper trail”).
produce these paper records for voters, there seems to be no data about how many voters review their receipts, verify the accuracy of the recorded votes, and lodge a complaint if the machine erroneously recorded the votes.240

Beyond the accuracy concern is the alarming question of whether hackers can physically or virtually break into these machines to alter individual votes and election outcomes. Although recent evidence that the Russians have been meddling in U.S. elections has heightened security concerns,241 the security of electronic voting machines has long been in question. For decades, computer scientists have been sounding the alarm about the vulnerabilities of these machines.242 Ever since states adopted electronic voting machines, computer scientists have repeatedly demonstrated that almost every type of machine is vulnerable to hacking.243 “The systems were not initially designed with robust security in mind, and even where security features were included, experts have found them to be poorly implemented with glaring holes.”244

There is an overwhelming number of examples of successful attempts to hack electronic voting machines. In 2005, an election official in Florida’s Leon County hired experts to test the security of the county’s voting machines.245 In

240 Provided that a voter reviews her vote to determine that it was correctly recorded, the paper trail would allow for a back-up manual counting mechanism.


242 See Halpern, supra note 234 (stating that electronic voting machines’ “vulnerabilities have been known for nearly two decades”); Bruce Schneier, By November, Russian Hackers Could Target Voting Machines, WASH. POST (July 27, 2016, 12:10 PM), https://www.washingtonpost.com/posteverything/wp/2016/07/27/by-november-russian-hackers-could-target-voting-machines/ [https://perma.cc/G87R-HCEQ] (“While computer security experts like me have sounded the alarm for many years, states have largely ignored the threat, and the machine manufactures have thrown up enough obfuscating babble that election officials are largely mollified.”); Zetter, supra note 241 (explaining that a computer scientist rang the alarm before the Help America Vote Act was passed).


244 Id.

each of the four tests, experts were able to break into the machines and alter the (fictitious) election outcomes by manipulating the machines’ memory cards.246 In 2016, one individual—luckily a security researcher—purchased old voting machines on eBay with the purpose of examining the machines to observe their vulnerabilities.247 Not only did he discover that “the tamper-proof screws didn’t work, . . . the hard drives had not been wiped . . . [and] [t]he information [he] found on the drives, including candidates, precincts, and the number of votes cast on the machine, were not encrypted,” but he was easily able to reverse-engineer the machines to manipulate them.248 Further, the researcher bought additional “next generation” machines two years later and found them even easier to hack.249

In 2018, at DEF CON 26, an annual conference that brings together some of the best hackers in the country, hackers explored some of the vulnerabilities of U.S. electronic voting equipment.250 In a stunning report, California’s Secretary of State, Alex Padilla, chronicled some of the findings from the conference’s “Voting Village”:

- A voting tabulator that is currently used in 23 states is vulnerable to be remotely hacked via a network attack. Because the device in question is a high-speed unit designed to process a high volume of ballots for an entire county, hacking just one of these machines could enable an attacker to flip the Electoral College and determine the outcome of a presidential election.
- A second critical vulnerability in the same machine was disclosed to the vendor a decade ago, yet that machine, which was used into 2016, still contains the flaw.
- Another machine used in 18 states was able to be hacked in only two minutes, while it takes the average voter six minutes to vote. This indicates one could realistically hack a voting machine in the polling place on Election Day within the time it takes to vote.
- Hackers had the ability to wirelessly reprogram, via mobile phone, a type of electronic card used by millions of Americans to activate the

246 See id.
248 Id.
249 See id. (“To my dismay, I discovered that the newer model machines—those that were used in the 2016 election—are running Windows CE and have USB ports, along with other components, that make them even easier to exploit than the older ones.”).
voting terminal to cast their ballots. This vulnerability could be exploited to take over the voting machine on which they vote and cast as many votes as the voter wanted.\footnote{BLAZE ET AL., supra note 250, at 5 (emphasis omitted). The full report by Matt Blaze et al. goes into significant detail about numerous vulnerabilities of various electronic voting machines. See generally id. (detailing machine vulnerabilities and laying out some steps that states may take in improving the security of their elections).}

The full report, which contains expositions by various experts, sets forth an astounding number of vulnerabilities embedded in our electronic voting machines.\footnote{See generally id. (detailing machine vulnerabilities and laying out some steps that states may take in improving the security of their elections).} This devastating report and the smattering of anecdotes above are just a small window into the immense tangle of electronic voting machine vulnerability issues that plague our voting systems across the country.

Manufacturers of the electronic voting machines emphatically deny that security risks exist and that their machines are susceptible to hacking. Many of the hacking successes, they explain, required direct access to the machine, which they suggest is not a realistic scenario.\footnote{See Goldfarb, supra note 245 (explaining that Diebold said the hacking experiment in question was ‘‘analogous to if I gave you the keys to my house and told you when I was gone’, . . . giving hackers ‘complete unfettered access’ to the equipment, something a responsible elections administrator would never allow’).} Instead, the machines are air-gapped—they are not connected to the internet.\footnote{See Zetter, supra note 243.}

Jeanette Manfra, the now former Assistant Secretary at the Department of Homeland Security’s Office of Cybersecurity and Communications, has recently said that “voting machines are physically secure, we’ve got thousands of jurisdictions across the country that all use different things. And so while you may be able to get into a few voting machines, you can’t really affect that at scale without detection, and it would be really hard.”\footnote{See Zetter, supra note 243.} Accordingly, Manfra concluded that “it’s actually really, really difficult to manipulate the actual vote count itself.”\footnote{Id.}

The overwhelming number of examples of successful voting machine hacking exploits makes these denials dubious. One information technology expert who is also a former White House liaison on cybersecurity literally said

\footnote{Alex Hern, Kids at Hacking Conference Show How Easily US Elections Could Be Sabotaged, GUARDIAN (Aug. 22, 2018, 5:00 AM), https://www.theguardian.com/technology/2018/aug/22/us-elections-hacking-voting-machines-def-con [https://perma.cc/M54R-N9AB]. Professor Alex Halderman explained that this “diversity” argument does not eliminate the security threat. See id. In fact, although:}

\begin{quote}
the diversity of US election technology poses a challenge for an attacker, . . . “that helps in some ways and hurts in some ways.” A real threat, he points out, doesn’t need to steal every vote in every county in every state in the country. The bad actor just needs to steal enough votes in a few counties in America’s battleground states – just enough to swing a close election. “So rather than diversity protecting us, we have a diversity of strength and weakness, and that’s a weakness for everybody.”
\end{quote}

\footnote{Id.}
Manfra’s conclusion is “bullshit.” He explained that the greatest risk to the security of voting machines is that the machines’ components are manufactured outside of the United States, primarily in China. “[O]nce you’re in the chips,” he said, “you can hack whole classes of machines, nationwide, from the fucking Kremlin.” Further, the coding on these machines is quite centralized—“[t]he large vendor codes the system for 2,000 jurisdictions across 31 states”—and if a hacker targets the coders, this could affect the functioning of the electronic voting machines, making sabotage a real possibility. In addition to these concerns, even air-gapped machines that do not originally have compromised chips or coding remain exposed. It was recently discovered that an unknown number of machines across the country have remote-access software installed on them—probably for easy remote maintenance of the machines by the manufacturer—meaning that they could be remotely hacked. Further, election officials connect the machines to the internet to electronically transfer vote counts on election night, making them vulnerable to hacking when they transmit data to a central counting server.

It is difficult to know the true extent of the accuracy and security issues that electronic voting machines raise because, as with the criminal justice and housing algorithms, their underlying algorithms are closely held trade secrets. Some states have sought to examine these questionable algorithms by requiring electronic voting machine manufacturers to disclose their algorithms to the

257 Id.
258 See id.
259 Id.
260 Id. Other states use local small businesses. As Halderman explained, if a hacker were to target those small businesses, he could potentially “reprogram thousands of election machines at once.” Id.
262 See Dan Patterson, Why Voting Machines in the U.S. Are Easy Targets for Hackers, CBS NEWS (Sept. 19, 2018, 11:52 AM), https://www.cbsnews.com/news/why-voting-machines-in-the-u-s-are-easy-targets-for-hackers/ [https://perma.cc/27GM-89LB] (“The other primary vulnerability is data transmission. In 2016 Symantec Security Response director Kevin Haley told CBS News, ‘The results go from [the voting machine] into a piece of electronics that takes it to the central counting place. That data is not encrypted and that’s vulnerable for manipulation.’”); Zetter, supra note 241 (“[M]any voting machines that elections officials insist are disconnected from the internet—and therefore beyond the reach of hackers—are in fact accessible by way of the modems they use to transmit vote totals on election night.”). Moreover, military personnel and citizens overseas often vote via the internet. See id. (“Although most American voters cast ballots in person or by mail, 31 states and the District of Columbia offer some form of internet voting to military personnel and citizens living overseas.”).
state before the state would do business with them. For example, North Carolina has a statute requiring manufacturers to turn over their algorithms and source codes so that state officials can review them and verify their accuracy and security, thereby "restore[ing] public confidence in the election process."263 Efforts like North Carolina’s have mostly been ineffective, though. In 2005, North Carolina’s supplier of electronic voting machines—Diebold Election Systems, Inc.—refused to turn over its software and code, claiming that they were secrets subject to confidentiality agreements with third parties.264 Despite a state court in effect concluding that Diebold must comply with the state legal requirements if it desired to sell its machines to North Carolina, the State Board of Elections ignored this directive and authorized Diebold as an electronic voting machine vendor anyway (although Diebold ultimately withdrew as a possible vendor).265 Accordingly, the true accuracy and security of North Carolina’s electronic voting machines remain unknown—just like the accuracy and security of machines across the country.

* * * *

These serious questions of accuracy, bias, and security are pervasive. For example, an error in an algorithm powering autonomous vehicles could cause deadly crashes. An algorithm determining the extent of someone’s healthcare coverage could be based on biased inputs. Russia could hack U.S. drones, causing them to attack China and starting a war. Each of these eventualities could result from simple inaccuracies, expert hacks, or perhaps even biases. Our wired world presents all of these risks, but the full extent of them remains unknown because developers’ trade secret claims largely cloak these dangers in secrecy.

III. IN THE INTEREST OF JUSTICE

All of the secret algorithms underlying criminal justice prediction and conviction programs, Facebook’s targeted housing advertisements, and the electronic voting machines used throughout the United States make it difficult to understand the full extent of the inaccuracies, biases, and security risks embedded in these algorithms. And the questions and associated problems of secret algorithms go beyond these three areas of criminal justice, housing, and voting.


For example, in the world of healthcare, experts have found that some mobile medical apps produce erroneous results related to insulin dosage calculations or fetal heart monitoring.\textsuperscript{266} With respect to education services, research shows that Princeton Review uses discriminatory pricing, charging parents in Asian-dominated neighborhoods more for their services—perhaps because Asian “Tiger Moms” may be willing to pay more for their children’s standardized test preparations.\textsuperscript{267} And Internet-of-Things devices can probably be hacked quite easily in a number of circumstances. In fact, there are reports of wireless baby monitors being hacked for the purpose of harassing parents, threatening child abductions, and carrying out financial thefts.\textsuperscript{268} The potentially devastating effects of algorithm shortcomings call for greater transparency.

In April of 2019, Representative Yvette Clarke (D-NY-9) introduced a bill in the U.S. House of Representatives to try to address some of these concerns.\textsuperscript{269} This \textit{Algorithmic Accountability Act of 2019} would require companies such as Facebook\textsuperscript{270} to conduct “automated decision system impact assessments” to examine the risks these secret algorithms pose in “contrib[ing] to inaccurate, unfair, biased, or discriminatory decisions impacting consumers.”\textsuperscript{271} The bill lacks much specificity, leaving significant discretion to the Federal Trade Commission to regulate in this area, therefore it is unclear how rigorous

---


\textsuperscript{267} See Julia Angwin et al., \textit{The Tiger Mom Tax: Asians Are Nearly Twice as Likely to Get a Higher Price from Princeton Review}, ProPUBLICA (Sept. 1, 2015, 10:00 AM), https://www.propublica.org/article/asians-nearly-twice-as-likely-to-get-higher-price-from-princeton-review [https://perma.cc/N2V9-P3AA] (“One unexpected effect of the company’s geographic approach to pricing is that Asians are almost twice as likely to be offered a higher price than non-Asians . . . ”).


\textsuperscript{269} See \textit{Algorithmic Accountability Act of 2019}, H.R. 2231, 116th Cong. [hereinafter \textit{Algorithmic Accountability Act (House)}]. Senator Ron Wyden (D-OR) introduced an identical bill in the Senate on the same day. See \textit{Algorithmic Accountability Act of 2019}, S. 1108, 116th Cong.

\textsuperscript{270} The Act covers entities that “ha[ve] greater than [fifty million dollars] in average annual gross receipts for the 3-taxable-year period preceding the most recent fiscal year” and that “possess[] or control[] personal information on more than . . . [one million] consumers . . . or . . . [one million] consumer devices.” See \textit{Algorithmic Accountability Act (House)}, supra note 269, at § 2(2)(5).

\textsuperscript{271} \textit{Id.} § 2(2)(C). It would also require such companies to “assess[] . . . the risks posed by the automated decision system to the privacy or security of personal information of consumers.” \textit{Id.}
The inaccuracy, bias, and security risks that secret algorithms pose cry out for intervention, but each different scenario has varying nuances that must be taken into account. For example, where criminal justice algorithms are concerned, the problem of secrecy pits the profit motives of algorithm developers against the justice concerns associated with incorrect and discriminatory recidivism predictions and wrongful convictions. An unfair criminal justice algorithm also negatively impacts the public because it may invalidate the bases for justified punishment, let the guilty go free, and erode the public’s confidence in the criminal justice system. Similarly, where housing algorithms are at work, the issue of secrecy impacts the algorithm developer, the person who is directly affected by the algorithm, and even the public more broadly by perpetuating the pernicious effects of discrimination. In both of these contexts, it seems that the primary justification for maintaining secrecy is supporting the creative work by the algorithm developers—the same reason for providing IP protection under the Constitution. Otherwise, transparency could mitigate many of these con-


274 See James R. Acker, The Flipside Injustice of Wrongful Convictions: When the Guilty Go Free, 76 ALB. L. REV. 1629, 1631 (2013); see also Meghan J. Ryan, Proximate Retribution, 48 Hous. L. Rev. 1049, 1066–67 (2012) (noting the “public conception of crime … which is reflected in the practices of allowing only government prosecutors, not victims, to prosecute defendants and of captioning criminal cases as the government entity against the offender”). Professor James Acker has explained:

Wrongful convictions entail profound social costs in addition to the hardships borne by the unfortunate individuals who are erroneously adjudged guilty. When innocents are convicted, the guilty go free. Offenders thus remain capable of committing new crimes and exposing untold numbers of additional citizens to continuing risk of victimization. Public confidence in the administration of the criminal law suffers when justice miscarries. At some point, as cases mount and the attendant glare of publicity intensifies, the perceived legitimacy of the justice system and the involved actors is jeopardized. Associated monetary costs, paid from public coffers, represent yet another tangible social consequence of wrongful convictions.

Acker, supra, at 1631.

cerns. With respect to electronic voting machines, however, even total transparency would not solve the problem. Putting aside electronic voting machine manufacturers’ desires to preserve the secrecy of their algorithms in order to maintain profits, publicizing these algorithms could actually do more public harm than good. Certainly, the risk of inaccuracies that secrecy hides and protects is problematic, but if would-be hackers were to have access to the algorithms, this could make it even easier for them to hack the machines. In this circumstance, they essentially could have a blueprint for successfully hacking an election.276

Addressing the full spectrum of threats that secret algorithms pose in various public-interest contexts requires balancing the risks of both secrecy and transparency. Algorithm secrecy threatens defendants’ lives and liberty in the criminal justice context, equal treatment of persons in the housing arena, and the very heart of our democracy where voting is concerned. But algorithm transparency also poses threats. Beyond exposure of algorithm-developers’ IP, making this secret algorithm information publicly available may pose risks such as the threat of election-hackers using voting machine algorithms to fix an election. Such risks should also be taken into account in trying to find the proper balance between algorithm secrecy and transparency.

Balancing the risks of algorithm secrecy and transparency is not an easy task. Courts might address each situation individually to create a patchwork of remedies using traditional approaches like allowing documents to be filed under seal or even applying more innovative solutions in some circumstances. This ad hoc approach could certainly lead to unpredictable results as to when algorithms ought to be disclosed, however. Courts might instead take a more systemic approach and target the recent changes in IP law that are contributing to the swelling secrecy concern. The availability of patents for software is also currently unpredictable, but loosening subject matter eligibility restrictions on software patents could perhaps improve predictability here, as well as further transparency goals.

A. Patchwork of Remedies

Examining the risks of secrecy and transparency in each case would allow courts to tailor disclosure requirements to the unique facts of individual cases, creating a broad patchwork of remedies. Such an ad hoc approach is not foreign to judges in the trade secret context. It involves a balancing of interests, with judges considering factors such as the relevance of the trade secret, its im-

276 Moreover, the concerns of security probably outweigh the inaccuracy issues, because a very high unintentional error rate would likely have to exist to change an election result, which is the most important metric in the area of voting.
portance to litigation, and the harms that disclosure will cause a litigant or some third party.

Pursuant to this approach, there are several possibilities for addressing the problems of algorithm secrecy, depending on the particular nuances of the case. Perhaps the most obvious solution is turning to well-established remedies that judges typically employ when issuing protective orders to help safeguard litigants’ trade secrets. Tools that judges might employ here include making the algorithm available for *in camera* examination, or making it available under seal. Indeed, these are both typical approaches to dealing with trade secrets in various types of litigation. When documents are produced for *in camera* examination, only the judge has access to the information; the opposing party does not. Keeping trade secret information closely held in this fashion should mitigate algorithm developers’ concerns about the release of trade secret information. However, in only very limited circumstances would allowing the judge—and no one else—to have access to the information ease the algorithm secrecy problem. And in most such cases, the judge would need to make use of an independent expert who would also need access to the algorithm at issue. Filing documents under seal might be a more practical means by which to place a check on secret algorithms. This mechanism of filing under seal means that the public will generally not have access to the information even though the parties, their attorneys, and the judge will. However, there is always the possibility that these individuals privy to the trade secret information will violate

277 See Elizabeth A. Rowe, *Striking A Balance: When Should Trade-Secret Law Shield Disclosures to the Government?*, 96 *IOWA L. REV.* 791, 821 (2011) (“The courts consider a wide range of factors that are neither definitive nor exhaustive in deciding whether good cause exists to grant a protective order.”).

278 See, e.g., Fed. R. Civ. P. 26(c)(1)(G) (stating that “[t]he court may, for good cause, issue an order to protect a party or person . . . [by] requiring that a trade secret or other confidential research, development, or commercial information not be revealed or be revealed only in a specified way”); Fed. Open Mkt. Comm. of Fed. Rsvr. Sys. v. Merrill, 443 U.S. 340, 356 (1979) (“The federal courts have long recognized a qualified evidentiary privilege for trade secrets and other confidential commercial information.”).

279 See 2 *E-COMMERCE AND INTERNET LAW* § 10.16, Westlaw (database updated Apr. 2020) (“A protective order allowing for *in camera* proceedings and permitting parties to file records under seal may be entered in trade secret cases, so long as the measures taken to maintain the integrity of the trade secrets are narrowly tailored to not needlessly encroach upon the strong public policy objective of maintaining open courts.”); 1 *JOHN GLADSTONE MILLS, III ET AL., PATENT LAW FUNDAMENTALS* § 4:20, Westlaw (2d ed. database updated Aug. 2020) (“Where litigation involves trade secrets the court proceedings may be conducted in camera and the court files impounded, so that the litigation does not compromise the trade secrets.”); Malla Pollack, *Litigating Misappropriation of Trade Secret*, in 127 *AM. JUR. TRIALS* § 56, Westlaw (database updated Oct. 2020).


281 See *Seal, BLACK’S LAW DICTIONARY* (11th ed. Bryan A. Garner, ed., 2019) (defining “*seal*” as “*to close up tightly or keep secret*” and “[*t*]o prevent access to (a document, record, etc.)”).
their obligations and reveal the secret. And once a trade secret is out, it is “lost forever.” But for this reason, algorithm developers may not be entirely comfortable with this potential solution to the algorithm secrecy problem. While software developers may trust their trade secrets with employees who have signed confidentiality and non-competition agreements, they likely will not feel as comfortable trusting opposing parties with this information, even if a potentially pilfering party would be subject to contempt orders, sanctions, breach of contract claims, or even misappropriation of trade secret claims if the secret got out. Success on each of these remedies is often difficult. For example, prevailing on a misappropriation claim would generally require the algorithm developer to establish that the opposing party actually caused the release of the trade secret information, and this is often difficult to establish. Because of these concerns, courts may be less likely to find filing documents under seal or ordering in camera examination to tip the scale in the direction of requiring transparency.

In some circumstances, such as in the context of HUD examining possible Fair Housing Act violations by Facebook and its algorithms, a government official could be charged as a confidant who has access to the trade secret to examine it for legal, or other, relevance, but who would also be subject to the remedies of trade secret law if he inappropriately disclosed the secret information. Again, from the algorithm developer’s perspective, divulging trade secret information beyond what is necessary to carry out its business is likely undesirable, but this may be an acceptable compromise where there is some basic showing of, for example, a Fair Housing Act violation. The severity of the grounded allegation may justify this required additional limited disclosure by the algorithm developer.

Considering another dimension could very well be relevant in determining the appropriate balance to strike between secrecy and transparency and in considering the variety of remedies that might be available in each context. And that is the identity of the primary algorithm users. For example, in the area of voting, individual governments are likely the primary users relying on the algorithm developers’ software to accomplish their tasks of conducting elections,

282 FMC Corp. v. Taiwan Tainan Giant Indus. Co., 730 F.2d 61, 63 (2d Cir. 1984) (“A trade secret once lost is, of course, lost forever.”).
284 See Robert A. Kearney, Why the Burden of Proving Causation Should Shift to the Defendant Under the New Federal Trade Secrets Act, 13 HASTINGS BUS. L.J. 1, 4 (2016) (“[C]ausation is difficult to prove and, as it stands, it is too difficult for many misappropriation plaintiffs.”) (footnotes omitted).
285 See Rowe, supra note 277, at 801 (explaining that “none of these options [for redressing a misappropriation claim against the government or its actors] may provide a satisfactory remedy, and legal recourse against the government may also be tenuous.”). For a good discussion of balancing interests in determining whether a company ought to be required to disclose a trade secret to the government, see generally id.
and this could very well be an important consideration. In the criminal justice context, too, it could be relevant if government actors are the primary users of tools like breathalyzers and probabilistic genotyping systems. In contrast, the primary users of Facebook’s software are generally Facebook itself and its private advertisers. (In each of these cases, individuals are also users—secondary users—when they cast their votes, benefit from a criminal conviction or fight against a wrongful one, or answer a housing advertisement on Facebook.) But when the government is the primary user, and where total transparency can address all but the algorithm developers’ profit concerns, perhaps the government, on behalf of the public, could pay a premium to gain access to these trade secrets. This would probably make the most sense in the criminal justice arena, where it is often the government itself making use of these programs to do its business of convicting the guilty and freeing the innocent. In this context, if the government is the primary purchaser or licensor of the product anyway, perhaps the developer would not lose much if the government were locked into a long-term, exclusive license arrangement with these developers.

Beyond balancing these and other interests in determining whether and to what extent to compel trade secret disclosure, courts have occasionally found that certain important public interests—like freedom of expression—can override trade secret interests and justify full disclosure. The Third Restatement of Unfair Competition suggests the existence of such a privilege “in connection with the disclosure of information that is relevant to public health or safety, or to the commission of a crime or tort, or to other matters of substantial public concern.” Professor Pamela Samuelson offers the example of “a firm [that]

---


287 See Ryan, supra note 12, at 341.

288 Such an arrangement might also work well for voting algorithms, except that the security concerns associated with algorithm transparency in this context pose additional difficulties.


290 RESTATEMENT (THIRD) OF UNFAIR COMPETITION § 40 cmt. c (1995); see also Samuelson, supra note 289, at 787–88 (stating that this is a “well-recognized privilege”). The Restatement provides:

[Disclosure of another’s trade secret for purposes other than commercial exploitation may implicate the interest in freedom of expression or advance another significant public interest. . . . The existence of a privilege to disclose another’s trade secret depends upon the circum-


considers certain chemicals used in its manufacturing process to be trade secrets, but those chemicals are toxins whose use violates environmental protection laws.” Courts seem to have relied on this exception only rarely, though, and it is even rarer that a court would find the privilege outside of the First Amendment or whistleblower contexts. When courts do find the privilege applicable, however, they engage in interest balancing to determine whether these important interests outweigh the trade secret interests.

Ultimately, through a balancing of interests, a variety of traditional, as well as more innovative, approaches could be used to deal with the algorithm secrecy problem on a case-by-case basis. Indeed, managing trade secrets in litigation is nothing new. But judges have generally been hesitant to order any disclosure of the algorithms underlying instruments like breathalyzers and other tools that can have real impacts on the public interest where the trade secret holder is not party to the litigation. Moreover, an ad hoc balancing approach ordinarily leads to unpredictable results. Accordingly, a more comprehensive solution might be desirable here.

B. The Systemic Intellectual Property Solution

More effectively addressing the problem of algorithm secrecy requires moving beyond the patchwork of remedies growing out of an ad hoc balancing approach. Instead, it would be useful to examine the system that has played a role in the algorithm secrecy problem: IP law. The recent surge in algorithm secrecy has corresponded with shifts in IP jurisprudence over the past several years. If the Supreme Court’s decisions in Alice and related cases make it significantly more difficult to obtain patent protection for software, then it is not surprising that software developers now seek to protect their work through trade secret law instead. Perhaps using the levers of IP law, then, would be a useful way to address the legal and policy concerns that the secrecy of algorithms has raised.

As the Supreme Court has acknowledged, excluding software—or even basic algorithms—from the realm of patentable subject matter is not expressly

stances of the particular case, including the nature of the information, the purpose of the disclosure, and the means by which the actor acquired the information.


291 Samuelson, supra note 289, at 788.

292 See Rowe, supra note 277, at 819–20 (“Even in the private-party discovery disputes, the courts are more protective toward permitting discovery of third parties who are not a party to the litigation and are far less likely to compel disclosure of their trade secrets.”); Ryan, supra note 12, at 307 (“Despite this need for the program information that controls breathalyzers, courts have generally refused to grant defendants access to these algorithms and source codes.”).

293 See supra Part I.

294 See supra Section I.C.
required by either the Constitution or the Patent Act.\footnote{295} Instead, the Constitution just vests power in Congress “[t]o promote the Progress of Science and useful Arts” by offering patent rights to inventors for their “[d]iscoveries,”\footnote{296} and the Patent Act purports to provide protection for “\textit{any} new and useful process, machine, manufacture, or composition of matter.”\footnote{297} Moreover, even the House and Senate reports accompanying the passage of § 101 articulated the law’s substantial breadth, stating that “anything under the sun that is made by man” should be patentable so long as it meets the other requirements of the Act.\footnote{298} Accordingly, the stringent test the Court used in \textit{Alice} to generally exclude software from patentability not only was a sharp change in law, but it was also contrary to the explicit terms of the governing statute and potentially at odds with the spirit of the Constitution.\footnote{299}

The \textit{Alice} test largely sweeping software outside the bounds of patentability was based on policy rather than constitutional or statutory grounds. The broad exceptions to § 101 prohibiting the patentability of principles can be traced back to the 1852 case of Le Roy v. Tatham,\footnote{300} where the Court stated (without references): “A principle, in the abstract, is a fundamental truth; an original cause; a motive; these cannot be patented, as no one can claim in either of them an exclusive right.”\footnote{301} The Court explained that, where such principles are involved, it is “the processes used to extract, modify, and concentrate natural agencies” that “constitute the invention”; “the invention is not in discover-


\footnote{296} U.S. CONST. art. I, § 8, cl. 8.

\footnote{297} 35 U.S.C. § 101 (emphasis added).

\footnote{298} S. REP. NO. 82-1979, at 5 (1952); H.R. REP. NO. 82-1923, at 6 (1952); see also Diamond v. Chakrabarty, 447 U.S. 303, 309 (1980) (referencing the committee reports). Although Congress overhauled the Patent Act with the Leahy-Smith America Invents Act in 2011, it mostly left the subject matter eligibility question untouched. See Pub. L. No. 112-29, 125 Stat. 284 (2011) (amending various parts of the Patent Act, but, with respect to subject matter eligibility, providing only that, “no patent may issue on a claim directed to or encompassing a human organism”).

\footnote{299} Some scholars argue that the Constitution’s language limits Congress to enact only patent-related laws that further “progress” in the sense that it advances civilization. See Jake Linford, \textit{Datamining the Meaning(s) of Progress}, 2017 BYU L. REV. 1531 (2017). For a good summary as to various plausible interpretations of “progress,” see generally \textit{id}.

\footnote{300} Le Roy v. Tatham, 55 U.S. 156 (1852).

\footnote{301} \textit{Id.} at 175.
ing [the principles], but in applying them to useful objects.” Although the Court’s discussion about the patentability of principles dates back almost two centuries, its discussions about the more specific exception related to algorithms is newer. In the 1939 case of Mackay Radio & Telegraph Co., Inc. v. Radio Corp. of America, the Court alluded to algorithms when explaining that, “[w]hile a scientific truth, or the mathematical expression of it, is not patentable invention, a novel and useful structure created with the aid of knowledge of scientific truth may be.” And it finally specifically addressed algorithms in the 1972 Benson case, where it suggested that it would be improper to patent the computer program at issue. The concern there was that allowing the patent would essentially grant a monopoly on an algorithm, preventing other inventors from building on this basic principle, thereby arresting innovation. Despite concluding that the program at issue was not patentable, however, the Court indicated that perhaps software should be patentable but that this is “a policy matter to which [the Court is] not competent to speak.” This nuance seems to have been lost when the Court decided Parker six years later. When the Court subsequently decided Diehr in 1981, it clarified that, although the policy decision that algorithms are unpatentable stood, that did not mean that inventions based on algorithms could not be patented. The Court

302 Id.
304 Id. at 94.
    It is conceded that one may not patent an idea. But in practical effect that would be the result if
    the formula for converting BCD numerals to pure binary numerals were patented in this case.
    The mathematical formula involved here has no substantial practical application except in con-
    nection with a digital computer, which means that if the judgment below is affirmed, the patent
    would wholly pre-empt the mathematical formula and in practical effect would be a patent on
    the algorithm itself.
    Id.
306 See id. at 67–68, 71 (reciting precedent providing that one should not be granted a legal
    monopoly on an idea and stating that “in practical effect [the patenting of an idea] would be
    the result if the formula for converting BCD numerals to pure binary numerals were patented
    in this case.”)
307 Id. at 72 (“It may be that the patent laws should be extended to cover these programs, a
    policy matter to which we are not competent to speak. The President’s Commission on the
    Patent System rejected the proposal that these programs be patentable . . . .” (footnote omi-
    tted)).
308 See Parker v. Flook, 437 U.S. 584, 585 (1978) (“In Gottschalk v. Benson, we held that
    the discovery of a novel and useful mathematical formula may not be patented.” (internal
    citation omitted)). But cf. Diamond v. Diehr, 450 U.S. 175, 185 (1981) (“This Court has un-
    doubtedly recognized limits to § 101 and every discovery is not embraced within the statu-
    tory terms. Excluded from such patent protection are laws of nature, natural phenomena, and
    abstract ideas.”).
309 See Diehr, 450 U.S. at 185 (reciting the notion that ideas and principles are not patent-
    able, explaining that the Court’s precedents “stand for no more than these long-established
    principles,” and stating that the invention at issue is patentable even though, “in several steps
    of the process[,] a mathematical equation and a programmed digital computer are used”).
thus delimited the policy exception, finding that, so long as an algorithm were paired with a physical step—even one already known in the art—the invention was patent-eligible. The Alice Court broadened the policy exception, however, making it much more difficult to patent algorithm-based inventions.

Today, where software touches basically every facet of life, IP policy considerations should be viewed more broadly. Courts should look beyond the basic trade-offs of the innovation incentive created by a limited monopoly versus the costs to innovation when preventing another from building on existing principles and consider the secrecy consequences of generally sweeping software outside the boundaries of patentability. Essentially closing off this avenue to inventors creates almost an entire industry of secrecy that affects and potentially negatively impacts the public interest. Secrecy can undermine trust in our criminal justice system. It can allow unfair biases to flourish in our society. And it can erode the trust in our government that is essential to a functioning democracy. When engaging in a decision about the scope of subject matter eligibility, these broader policy considerations should also be deemed relevant.

For example, when determining the patentability of software powering a probabilistic genotyping system used to convict criminals, a decisionmaker—whether that be the PTO or a court—might take into account considerations beyond just the extent to which issuing a software patent might stifle innovation by effectively preempting use of the underlying algorithm. This narrow focus on stifling innovation was of course the concern behind Alice’s less-than-clear test concentrating on whether the combination of claims includes an “inventive concept.” This test has left a wake of uncertainty about where exactly the line between patent eligibility and non-eligibility lies. Within this area of uncertainty, decisionmakers might also consider the consequences of the inventor turning to trade secret law to protect his invention. In this example, where the software is used to convict criminals, a lack of transparency and the shield of trade secret law could have devastating effects, such as wrongfully convicting an innocent person. And the disadvantages of finding subject matter eligibility for the software could very well pale in comparison. This balancing might mirror some of the balancing that takes place when courts determine whether and under which conditions a party must disclose a trade secret. It might also reflect the balancing that

---

310 See id. at 184; see also supra text accompanying note 99.
311 See supra text accompanying notes 305–306.
313 Alice Corp. v. CLS Bank Int’l, 573 U.S. 208, 217, 221 (2014); supra text accompanying note 140.
314 See supra Section ILA.
315 See supra text accompanying notes 277–79.
courts occasionally engage in when determining whether the secrecy of trade secret law must yield to First Amendment or whistleblower concerns. But when the balancing takes place under the umbrella of patent eligibility, at least the property owner potentially receives something in return for disclosure: a twenty-year patent.

Certainly, a more lenient approach to subject matter eligibility for patents would not be a cure-all for harms created by software secrecy. Depending on the nature of the invention, the would-be patentee might continue to prefer trade secret protection over patent protection. But, as in other areas of law, patent law can affect individuals’ decisions. It can be used to nudge inventors toward transparency when that decision will better serve the public interest. Ultimately, assessing the potentially negative effects from narrowly construing patent eligibility could push courts back in the more patent-permissive direction of Diehr and the Federal Circuit cases suggesting that the software industry is not largely excluded from successfully pursuing patents on their inventions. And this would likely induce at least some software inventors to pursue patent protection rather than trade secret protection, thus increasing transparency.

The Constitution focuses on promoting “[p]rogress” through science and the arts, and transparency is key to such progress. Not only is transparency important so other would-be innovators can build upon one’s inventions, but it is also key to ensuring that an invention is truly useful—in the case of breathalyzers, for example, that it provides accurate results. While the usefulness, or utility, requirement of § 101 is ordinarily not stringently applied

316 See Sandeen & Mylly, supra note 31, at 42; see also Samuelson, supra note 289, at 839–40 (2007) (suggesting that certain factors set out in case law “will obviously play out differently in varying factual contexts, but they provide a sound mechanism for balancing the First Amendment interests of journalists and publishers and those of trade secret claimants in weighing whether confidential source information should be disclosed to trade secret claimants”).

317 See supra text accompanying notes 36–44.

318 See, e.g., Julia Patterson Forrester, Mortgaging the American Dream: A Critical Evaluation of the Federal Government’s Promotion of Home Equity Financing, 69 Tul. L. Rev. 373, 397 (1994) (“Federal income tax law promotes home ownership in a number of ways, the most notable being the allowance of a deduction for home mortgage interest.”). Indeed, before the Court found it to be a First Amendment violation, Congress required decisionmakers to consider the effects of negative words on the general public by prohibiting disparaging trademarks. See 15 U.S.C. § 1052(a); see also Matal v. Tam, 137 S. Ct. 1744, 1765 (2017) (“[W]e hold that the disparagement clause violates the Free Speech Clause of the First Amendment.”).

319 U.S. CONST. art. I, § 8, cl. 8 (“The Congress shall have Power . . . To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries”).

320 Indeed, the majority of scholars believe that “progress” means further advancing knowledge. See Linford, supra note 299, at 1545 (“[M]ost scholars have embraced a series of meanings that coalesce around the notion that progress means advancement in knowledge, using phrases like “the encouragement of learning’ to refine the concept.” (footnotes omitted) (quoting Edward C. Walterscheid, The Nature of the Intellectual Property Clause: A Study in Historical Perspective 150–51 (2002))).
to limit patent eligibility.\textsuperscript{321} § 101’s focus on promoting “new and useful” inventions highlights how the transparency of patents can further usefulness, and thus “[p]rogress,” through science and the arts.\textsuperscript{322} Rather than sharply limiting patents on software powering instruments ranging from breathalyzers to electronic voting machines, returning to a more generous and workable standard for software that acknowledges secrecy’s impact on the public interest would further progress in this field and benefit the public interest by fostering greater transparency where important considerations like equality, life, and liberty are at stake.

CONCLUSION

The Supreme Court’s recent decision in \textit{Alice} significantly curtailed subject matter eligibility for patents. This involved exempting from patent protection an overwhelming swath of software that was previously patentable. Making it more difficult to patent software has pushed at least some developers to protect their intellectual property through trade secrets instead. This has resulted in considerable secrecy now cloaking the workings and uses of numerous software offerings and their underlying algorithms. Many of these algorithms affect the public interest, but the secrecy shrouding them has made it close to impossible to assess whether the algorithms suffer from risks related to inaccuracies, biases, or security flaws. And the devastating effects of inaccurate, biased, or insecure algorithms and software are heightened in the public interest realm, leading to severe effects such as wrongful convictions, ubiquitous biases in housing, and undermining the sacred act of voting in a democracy. One might address these concerns in a patchwork manner by balancing the secrecy and transparency risks in each unique situation. But attacking one source of the secrecy problem—recent changes in IP law—could effectively address the matter in a more global fashion. The \textit{Alice} Court’s shift in doctrine was not driven by the Constitution or even the terms of the governing statute. Instead, policy concerns drove the Court’s decision. Yet the Court neglected to examine the significant impact that propelling software and algorithm developers toward secrecy would have on the public interest overall. These effects should have

\textsuperscript{321} See Michael Risch, \textit{A Surprisingly Useful Requirement}, 19 Geo. Mason L. Rev. 57, 58 (2011) (“In the broad scheme of things, . . . the requirement that an invention be useful has been nearly nonexistent—essentially ignored. The level of ‘utility’ an applicant must currently demonstrate to obtain a patent is extremely low: the invention need only operate as described and potentially provide some de minimis public benefit.” (footnote omitted)). \textit{But cf.} Sean B. Seymore, \textit{Making Patents Useful}, 98 Minn. L. Rev. 1046, 1049–50 (2014) (arguing that characterizing the usefulness requirement as de minimis is inaccurate and stating that “[i]t is more correct to say that the utility threshold is decidedly biased—a de minimis threshold for some inventions but a considerably more stringent one for others” (emphasis omitted)). Note that some experts suggest that utility is only part of the usefulness requirement. \textit{See} Risch, supra, at 58–59 (arguing that § 101’s “utility requirement (as currently interpreted) is only a part of an invention’s usefulness”).

\textsuperscript{322} See supra note 320.
moved the Court in *Alice*, and these effects should influence the Court in re-shaping subject matter eligibility for patents going forward.