Digital Transformation and Ethics

Edited by
Markus Hengstschläger and the Austrian Council
for Research and Technology Development

ecowin

Contents

Ethical Deliberation for Agile Software Processes: The EDAP Method

Niina Zuber, Severin Kacianka, Alexander Pretschner und Julian Nida-Rümelin

Ethical Decision-Making

Every day humans are faced with the question of what they should do: What is a correct, appropriate, reasonable or good action? As machines and algorithms that perform actions without human supervision becoming more and more widespread, this question is especially present in the development and use of such systems. This is especially the case since these systems have a direct impact both on the life of each individual and on society as a whole. Ethics, as the science that concerns itself with questions of normativity, attempts to find answers here. Ethics makes judgements about what should be, and says nothing about what is (Henning, 2019). In contrast to jurisprudence, ethical judgements do not regulate actions ex post, but rather ex ante already serve to structure action contexts. Many everyday situations do not require normative deliberation, because the desired behavior, the appropriate action, is clearly determinable – at least at that given point in time: This subsumes for example many calculation rules, traffic regulations and rules of politeness. We know, without having to think about it, what should be done and/or which behavior is desirable.[42] These dispositional behaviors are accepted, internalized and do not require further reflection at the moment in question – which however does not exclude the possibility that the behaviors can be reconsidered and modified at a later point in time. Lifeworld complexity is thus encountered with acquired and stabilized dispositions – i.e. that which is referred to in the Aristotelian tradition as *aretai* (virtues) (MacIntyre, 1981; Nussbaum, 1999; Foot, 2003; Nida-Rümelin, ³2019; Vallor, 2016).

Different circumstances exist in the case of behavioral conflicts, i.e. in situations where it is not clear which behavior, which condition or which objective is desirable. The task of normative Ethics is thus to reflect and justify which reasons for action are to serve as points of orientation. This justification is the result of an ethical deliberation which systematizes normative arguments and ultimately leads to a normative judgement: Thus the permissibility or reasonableness of decisions is to be explored and justified. Ethical deliberation weights reasons for action and ponders different options for action. Ultimately the decision or the intention to act has to be well-founded[43] in order to be considered reasonable (Lord, 2018; Kiesewetter, 2017; Wedgewood, 2017; Nida-Rümelin, 2020). Information technologies can take these normative decisions into

[42] These rules can be relatively easily taught to machines.

[43] Machine are based on fundamental assumptions of classical decision and rationality theory, so that we can speak of »rational machine constructs«. However, we cannot speak of »reasonable machines« which justify decisions, make well-founded assumptions or which can behave contrarily to their programs (cf. Nida-Rümelin / Weidenfeld, 2018; especially the chapter »Autonomie und Determinismus in der digitalen Welt«).

account in the development of technical artifacts and implement them in practice in their designs.

Here normative judgements cannot be reduced to epistemic facts. An increase in knowledge does not necessarily lead to more clarity in decision-making or to a better judgement. In other words: Certainty cannot be found in data alone, which is why algorithms alone are insufficient for a solution; this applies among other things to the widespread machine learning algorithms that identify correlations in data or which can reduce the dimensionality of data. This limitation of algorithmization is also due to the underdetermination of normative reasons for action. This underdetermination is particularly evident in cases of fundamental norm conflicts, but is also the result of deontological constitution of lifeworld moral and juridical assessment: In contrast to consequentialist (utilitarian) systems, in deontological systems there is no general resolution of practical conflicts through optimization. Indeed cases are conceivable in which the reason for action appears to be immediately evident, for example in the obligation to help the injured. But even here situative-evaluative perspectives have to be considered in the respective individual situation, for example time pressure, danger to the person providing the help, etc. Decisions thus remain contingent. It is not possible for all facts to clearly, i.e. necessarily, determine which reason for action should be valid. Decisions thus evade predictability, which is why we must begin with the process creating technical objects. Furthermore, normatively ambiguous situations cannot be resolved through the acquisition of knowledge, since they require additional practical competencies such as robust decision-making or other characteristics.

The instructions for action and/or normativity are not limited to juridical laws[44] or social norms. It is therefore not surprising that current public discourse focuses on algorithms whose enormous scope can lets them affect and influence many people and thus develop a great normative force. Each day we find ourselves in situations requiring us to decide and in which we also *want* to decide correctly, appropriately and well. Here values and/or value systems take on an prominent role, since they can constitute conative attitudes, which means they can elicit intentions that express how we want to decide in one action situation or another. Thus for values which have been judged to be good and internalized in the course of a maturation process can individually make reasons for action appear to be particularly well-founded, so that these reasons exert a casual compulsion to perform the action. Therefore an ethical deliberation can be seen as a mode of thought, an assessment process, that is to lead to those normatively adequate actions.

Normative reasons do not have to be moral reasons, but all reasons – including moral reasons – are normative, i.e. they can move us towards realization with more or less justification. Economic and legal facts or social rules can thus also constitute normative reasons and offer orientation for our actions. Conflicts arise when reasons are mutually exclusive. Thus an economic reason may exclude a moral reason or two economic reasons may point in opposite directions. We must in particular speak of moral conflicts when they affect the individual freedom(s) of action and rights of others – when the performance of an action impairs freedom(s) and rights (Nida-Rümelin, 2016).

[44] Laws arise as a reaction to imbalanced situations, thus a posteriori. As soon as they become laws, they are incorporated as norms in the choice of action.

Ethically Building Machines

Underlying the terms »*robot ethics*« (Lin/Jenkins/Abney, 2017), »*moral machines*« (Wallach, 2010) and »*value sensitive design*« (Friedman/Hendry, 2019) is the question of which rules should apply in order to be able to develop ethical machines, and how these rules are to be formulated so that they can also be followed. When we speak of ethical machines, we mean machines or software systems than can meet legal, cultural and moral standards. These are systems in which the developer team has reflected on, intentionally realized and included evaluations in the system design so that the machines implement the evaluations of the developers. Thus we do not mean systems which attempt to extend the architectures and algorithms to include an ethical dimension in the form of a variable or a rule in order to model the human capability of normative deliberation (Conitzer/Sinnott-Armstrong et al., 2017; Misselhorn, 2018). The desire for a software system that independently returns an ethically secure result by attempting to imitate the human ability to deliberate is well beyond the current state of scientific research. Software systems which control a recruiting process at a university, drones which shoot at the enemy in a war or software systems that protect the social welfare systems from fraud (O'Neil, 2017; Eubanks, 2018; Noble, 2018) do not become ethical systems simply because they implement an algorithm which has been expanded to include an ethical rule or because an architectonic structure has been developed which uses certain rules to inductively and deductively cluster and subsume data items. The expectation that an ethical machine would in some way independently achieve an ethically desirable objective or even be capable of defining such an objective for itself because we have programmed it for ethics overlooks the central human contribution to ethical deliberation

and freedom to make decisions: Precisely the human ability to make well-founded decisions in an uncertain, complex lifeworld.

The fact that technical dimensions relating to societal, legal, economic, esthetic and moral contexts continuously have to be taken into consideration in the design and conception of technical objects illustrates the difficulty of reducing ethical intricacy or even delegating it to machines. Front-end and back-end design, user operation and the societal implications that arise due to the use of a system have to be considered individually and in their compositionality at different levels of urgency. Negative externalities which for example have been caused in the housing market by Airbnb (Lee, 2016) cannot be avoided or softened with the one-time implementation of an algorithm expanded to include a moral principle, which would then perform an »independent« calculation from the data available to arrive at the morally perfect solution. Technology cannot replace mental performance, reflection about the conceptualization of meaningful life forms – and in the sense of a Digital Humanism (Nida-Rümelin/Weidenfeld, 2018) the attempt is also not desirable.

Example: Machine Learning

This can be easily demonstrated using the example of what is called machine learning (ML). Machine learning refers to methods that identify patterns in training data and then classify new, unfamiliar data sets according to these patterns. The important thing is that these are algorithmically structured, statistical models that extract and order the data; here the emergence of the result is as a rule not completely understandable, even for the developer team. From a technical point of view machine learning thus does not offer a

functionality for making ethical decisions. Even seemingly straightforward issues can result in unexcepted results in ML algorithms. A simple example is freedom of speech: According to John Stuart Mill (1869), in a free society opposition to the prevailing opinion is to be tolerated, and according to Henry David Thoreau (1849) even supported. Take an abstract example, in which the majority says something is »100«, and only one single individual is of the opinion it is »–1«; here a linear ML algorithm has two possibilities: Either »–1« is treated as an outlier – the opposition is simply ignored and this opinion suppressed – or some kind of mean value is formed so that consensus would be a value which is very close to, but never reaches, »100«. Non-linear algorithms such as neuronal networks or »Decision Trees« would only be able to represent the majority opinion when no other characteristics of the individual can be identified. Therefore when training such algorithms an effort is made to avoid what is called »overfitting«, i.e. an over-adaptation of the model to the training data. An algorithm which we use to indicate that everyone except one individual describes something as »100«, but this individual describes it as »–1«, is thus not desirable for reasons of generality of the model. This means currently widespread ML algorithms are not capable of adequately representing a diversity of opinion which can be considered ethical in the sense of John Stuart Mill.

Machine learning (Mohri/Rostamizadeh/Talwalkar, 2018) is essentially a collection of information science methods which make predictions about future data items based on existing or past data items. This process is common to all methods, which is why the core task is to draw the most elegant multidimensional curve through the usually multidimensional data with the expectation that this particular curve will also put new data items into the »right« class. Machine learning is especially successful in tasks

which consist in the classification of data (e.g. image or language recognition), in identifying regressions (e.g. the price development of products) and sequences (e.g. the most relevant web page for a search query) as well as in the clustering of data (e.g. dividing people into certain groups).

Machine learning procedures are thus to be considered purely statistical and are not in any form (artificially) intelligent. Just the opposite: Pearl shows quite clearly that current processes are not capable of reaching beyond simple correlations (Pearl/Mackenzie, 2018, 27 sqq.). They may be excellent means of detecting patterns and associations in data and 'learning' these patterns, but it is not possible to use them to answer counterfactual questions. To take Pearl's example, it is thus very easy to use an ML algorithm to identify which products are purchased together in a supermarket. It is however impossible to use such an algorithm to answer the question of the probability that the buyer of product A would have also purchased this product if it had been twice as expensive. Precisely such questions of alternative options for action and for other motivational structures are essential to ethical deliberation. The reason here is that data alone cannot make this issue explainable; that requires a theory or a model of the world and its causal relationships – in this case a solid economic theory and a model of the theory. These forms of causal conclusions remain as yet reserved for the human.[45]

[45] Even if there is also research on the subject, for example Dasgupta et al. (2019).

Software systems, regardless of how smart the algorithm is, do not perform any assessment of motivations; they cannot dependently switch between different modes of thinking and assessment and justify their methods in order to ultimately become meaningfully effective. Software systems are not autonomous, moral agents: They always remain executive. Making reasonable, i.e. ethically desirable, decisions is thus reserved for the human, which is why currently existing development processes must integrate normative consideration processes at the level of the developer as well.

Codes of Conduct and Ethics Canvas: Ethical Deliberation for Agile Processes (EDAP Method)

There are already processes and/or extensions for existing system development methods with regard to *safety* and *security*[46] requirements which are to ensure that a system is developed to be both *safe* and *secure*. We actually make an automatic door *safe* by using technical means to make sure that it cannot close on our limbs. This does not require the door to have an intuition in the sense of »making my environment safer«, but rather simply has to be built in compliance with certain safety criteria. Analogously, an ML algorithm cannot and should not make ethical decisions, but rather simply realizes the valuations of the developer team. The current state of research does not allow for the development of algorithms that independently »reflect« and »formulate theories«. It

[46] In fact »safety« refers to »functional safety«, i.e. that a system causes no harm, while »security« refers to »information security«, i.e. that a system ensures the confidentiality, integrity and availability of dataand functions.

is thus absolutely crucial that developers, administrators and users[47] pose the question of desirable objectives in the development and use of software. Consequently the compatibility of algorithms, data sources and input, such as control commands, with that which is considered ethically desirable must also be kept in mind. This also means that developers have to ask themselves the self-critical question of whether or not the contribution they are making is in keeping with their professional ethics[48], i.e. among other things, whether the test cases are actually good and adequate, but also which responsibility they can and must take on in the development process. In particular because of the wide variety of central workflows, handling data-based technologies calls for a variety of skills with different requirements – ranging from program development and data storage to the maintenance of technical artifacts all the way to the profitable market launch. This is why consideration of the individual activities and their significance for a sustainable product is necessary, since this division of labor can quickly result in a diffusion of responsibility (Battaglia/Mukerji/Nida-Rümelin, 2014). Responsibility does not diffuse exclusively among entrepreneurs and developers, since handling the technical product responsibly is also a moral obligation on the part of the user. The end-users can misuse the product, they can simply operate it incorrectly or misunderstand the proper application of the product. In the present paper we would however like to concentrate on the ethical re-

[47] Developers work on a system until it »goes into production«. Then administrators take over the maintenance and upkeep of the system. The term »DevOps« (*Development* and *Operations*) refers to a management approach in which this strict separation is softened. Users of the system use the functionalities which are made available.

[48] Cf. for example the guidelines of the *Gesellschaft für Informatik*: https://gi.de/ueber-uns/organisation/unsere-ethischen-leitlinien (last accessed on 4.2.2020)

sponsibility of the software developer. To do this, the various assignment areas have to be addressed in ethical guidelines, so that normative concerns can be identified and then taken into consideration at the appropriate point. This is a prerequisite to living out and internalizing a professional ethics. The »Swiss Alliance for Data-Based Services« follows exactly this approach in the formulation of an ethical code which is oriented to the lifecycle of the technical product in order to specifically link work roles and responsibilities with key ethical questions (Loi/Heitz et al., 2019). The majority of codes of conduct in numerous private and public alliances[49] as well as of the five major tech giants[50] are however based on respective individually differentiated canons of values. These exclusive weighting lists differ in terms of their concrete definitions and creation of hierarchies for ethically desirable conceptualization of technical objects. The divergences are probably due to the lack of distinction between workflows or technologies used (e.g. between machine learning and rule-based systems), or among product categories, although the normatively desirable orientation of autonomous weapons systems requires a different emphasis in its ethical direction from the emphasis required for the development of controlling software for business economics.

The values and ethical principles which emerge from the virtually uncountable number of these codes of conduct and guidelines

[49] For example private alliances and organizations such as the IEEE (Institute of Electrical and Electronics Engineers), the Partnership on AI (founded in 2016 as an alliance of industry and non-profit organizations with academic institutions such as IBM, Google's DeepMind, Microsoft; Apple joined in 2017) or Open AI (founded in 2015 as a non-profit organization for researching AI; largest funder is Elon Musk). Also global and public alliances such as the Future of Life Institute with the »Asilomar AI Principles« (23 principles for ethical treatment of AI, adopted at the Asilomar Conference in 2017 and signed by 1273 AI/Robotics researchers and 2541 others).

[50] Apple, Amazon, Facebook, Google and Microsoft.

have as yet to be investigated and statistically evaluated in a literary survey.[51] Although the majority of the codes contain fundamental values, for example welfare and autonomy, as well as ethical principles such as consequentialist or deontological rules of assessment, they do so without listing specific possible applications or offering support in learning transfer. An attempt is made to address this situation using ethical outlines, like those developed for example by the Open Data Institute[52], the Center for Humane Technology[53] or by The Ethics Canvas[54]. These outlines in turn function as action elements which call for reflection and call attention to the multidimensionality of technical objects: They are intended to help reveal cross links and mutual interactions of complex relationships between the technical object and its ecosystem by for example asking for psychological conditions, stakeholder requirements, legal science basics and policies, since it is only with the embedding of a technical artifact that unintentional distortions or justified goal conflicts arise.[55] However, these question catalogs are of no help in prioritizing values or in conflict resolution or in the formulation of appropriateness or even in the embedding of ethical deliberation cycles in business-economic workflows. They remain at a level of primarily descriptive value formulation, *without systematically addressing normative judgements*. It is therefore important to distin-

[51] The authors of the present paper will conduct in 2020 a descriptive, Artificial Intelligence-based value analysis.

[52] Open Data Institute https://theodi.org/article/data-ethics-canvas/ (last accessed on 4.2.2020).

[53] Center for Humane Technology https://humanetech.com/ (last accessed on 4.2.2020).

[54] The Ethics Canvas https://www.ethicscanvas.org/download/handbook.pdf (last accessed on 4.2.2020).

[55] In 2020 the authors of the present paper will scientifically evaluate the various ethical schemata as a part of the project »*Ethik in der agilen Softwareentwicklung*« of the Bavarian Research Institute for Digital Transformation (bidt).

guish between a phase of descriptive value formulation and a phase of the critical inspection of values as well as between the transfer of values and principles to technical objects. *Values in Design* is exactly the field of research which addresses the integration of ethical values in technical objects.[56] This places the focus on the design and development process, and the ethical evaluation is not reduced to stakeholder and value analyses, even though these elements have to play a central role. Using descriptive system and value analyses it is possible to localize initial value conflicts and then to subject value trade-offs, value tensions or value conflicts to a scientific, normative examination.[57] Thus the connection of a descriptive value analysis with a subsequent normative deliberation in the technical formulation of objectives is essential to the transfer of learning and implementation of ethically desirable aspects in technical artifacts so that well-founded codes of conduct can be formulated. These are the steps which have to be methodically analyzed, structured and systematized.

System analysis should integrate the ethical deliberation process in the development of software systems in particular and should structure both the systematization and implementation of normative judgements and their verifiability. In the best case this will lead to well-founded preference relations and defined option spaces, which is in the interests first of an innovative change in perspective and second of the humane conceptualization of our living spaces. Our proposal is to make targeted use of the benefits of iterative agile management processes to arrive at the desired results: As an example we would like to take a brief look at one of these processes, *Scrum*. Like other management methods of this type (other familiar methods include Kanban, Extreme Programming and Feature Driven Development), the objective of Scrum is to reduce hierarchically structured bureaucracies in work organizations to enable reaction to changes dynamically and without delay (i.e. to be »agile«). At its core Scrum parses product development into smaller features. Each of these features is to be completed in short discrete iterations (typically two weeks long; no longer than four weeks), referred to as Sprints. All features are recorded in the »Backlog« and then prioritized by a domain expert or customer, the Product Owner. The Scrum Master then takes over the relevant communication between the Product Owner and the development team and supports the latter in acting as efficiently as possible in project realization. Scrum involves several different types of meetings. In the Daily Scrum the entire team meets to align on the current status, in the Sprint Review the product is considered and the current results are presented to the Product Owner and other stakeholders, and in the Sprint Retrospective the Sprint itself is analyzed and possible process improvements are investigated. It is conceivable that normative deliberation units could be included in these agile work processes and the continuous review phases, so that existing corporate or societal expectations can be synchronized and new development approaches, possible applications and handling methods may arise.[58] The entire development process, from the conceptual phase all the way to final use, is inte-

[56] Cf. Simon (2016a), Friedman / Hendry (2019) and Friedman (1997).

[57] Examples of how an ethical change in values can take place can be found among others places in Friedman/Kahn/Borning/Huldtgren (2013), Simon (2016b) and Simon (2012).

[58] The authors of the present paper conduct research at the bidt as part of the project »*Ethik in der agilen Softwareentwicklung*« on the question of how normative deliberations can be optimally integrated in agile process management structures.

grated in systematization, ensuring a normatively desirable conceptualization of the technical product. Software developers are sensitized to normatively appropriate design. Ethical deliberation thus becomes an integral part of product development. The method is open, i.e. there is no advance definition of which ethically desirable evaluation criteria or principles are to finally be applied; much more such criteria or principles are to be identified based on the available facts and arguments. The method makes it possible to integrate different attitudes of motivation and is to counter associative-intuitive brainstorming. The humanistic foundation of normative ethics does not describe norms, rules or laws according to which humans are to orient their behavior de facto, but calls on us to reflect about which act would be desirable. For the algorithm designer this would be for example to find out what output is possible or desirable. It is this deliberation, this practice, that is ethics.

The EDAP Method[59]

This *reasoning* process – i.e. the normative deliberation or reflection about which technical orientation is desirable or which conceptualization would be desirable and how it can be technically implemented – is to take place under the application of the EDAP method[60] in a goal-oriented and structured manner. It is precisely this deliberation which constitutes the human ability to be the author of own's own life (Nida-Rümelin, 2005). And precisely this practice is at the same time the humanistic moment which must be seen as a reference point and which is to be technically supported and not undermined. The EDAP method is intended to support this deliberation process. The EDAP method achieves this by structuring the associative deliberation process and performing a prioritization of the possible actions. The process is intended to result in a simple, manageable and effective recommendation for action which is to be taken into consideration in technical realization. This makes it possible for example to protect privacy by eliminating data or to strengthen the autonomy of the user through transparent design of algorithms. In addition, the careful selection of content can counteract possible extremist tendencies. The development process is not restricted, much more the assumption of normative perspectives can give rise to innovative conceptual possibilities.

The EDAP method is based on rationality theory, ethics, economics and developmental psychology theories. It is divided into eight phases which are oriented towards the various development steps: They begin with a survey of the overall system and the description of the requirement profile. The second phase is concerned with an inventory of the values which are to be taken into consideration: What values and recommendations are relevant? Here we will also use our descriptive analysis of codes of conduct to enable a sound entry into the topic of goal conflicts and value conflicts. Then we will look at the specific concrete case in order to localize those ethical values which are to be taken into consideration in this specific technical object. This then leads directly to Phase III, in which the desirable orientations, which are undermined by a variety of biases, are to be captured and accounted for. Dealing critically with values and possible goal conflicts is then the topic of Phase IV: Which values are in conflict with one another? What does this

[59] Am bidt forschen die Autor*innen dieses Beitrags im Rahmen des Projektes »Ethik in der agilen Softwareentwicklung« an der Frage, wie normative Deliberationen in agile Prozessmanagementstrukturen optimal eingebunden werden können.

[60] This method is being developed in the course of the »*Ethische Deliberation in agilen Softwareprozessen*« project of the Bavarian Research Institute for Digital Transformation. The method in question is the EDAP method 2019, which the authors of the present paper are currently investigating in practice.

mean in terms of implementation? And are the values identified morally binding? How are moral motivations in conflict with other motivations? In most cases it is possible to move on directly to Phases VI and VII, where the phases of technical conceptualization and verification have to be integrated in considerations as early as the initial phase. The phase of normative-theoretical system testing (Phase VI) is primarily helpful in the context of serious conflicts and in the context of continuing education and ethical sensitization. The phases are not to be regarded as linear, i.e. during the development process it is necessary to switch between the deliberation phases in order to be able to highlight the mutual interactions.

Our EDAP method is suitable for brief but pointed reflection of everyday and typical decisions made in developer teams and to render explicitly the scope of one's own actions. Here we will use as a thought experiment[61] a software system which records the phone conversations of call center employees for training purposes. We assume that the recordings and usual analyses, for example length of the conversation and customer feedback have already been established and work properly. At this point the developer team is asked to expand the system to include a *sentiment analysis* of the conversations[62]. The objective is to further improve employee training and for example to make a preselection of audio recording after aggressive conversations. In the deliberation template we now present examples relating to our thought experiment, in addition to the general key questions.

61 A detailed theoretical description will be published in late 2020.

62 Here the feelings behind spoken language are measured, frequently using machine learning algorithms. For an audio and video sample see: Zadeh, A. / Chen, M. / Poria, S. / Cambria, E. / Morency, L.P. (2017): Tensor fusion network for multimodal sentiment analysis. *arXiv preprintarXiv:1707.07250.*

	Phase I: Descriptive System Analysis
Universe	Description of the overall situation *see above*
Stakeholders	Who influences the technical system or is influenced by it? *E.g.: Customers, call center staff, training supervisor...*
Technical Strategies	What are the technical possibilities for reacting to the formulated objective? *E.g.: Sentiment Analysis*
	Phase II: Descriptive Value Analysis[63]
Universe	What values have to be considered? Human rights, etc. Which values are to be declared desirable? Value analysis and clustering of codes of conduct / guidelines. *E.g.: ACM Code of Ethics, IEEE, et al.*
Stakeholder Analysis	Which values are held by the various stakeholders? Corporate Social Responsibility / Digital Responsibility / Investor Relationships / Guiding Principles, etc. *E.g.: Employees via employee representatives:* *Management tier: Head of training:*

63 Evaluation of codes of conduct and ethical guidelines according to ethical values and theories using substantive analyses in work.

Phase III: Technical Analysis

Preexisting bias	Founded in institutions, practices, attitudes. *E.g.: In the past within the company all available data – to the extent legally possible – was also used to measure employee efficiency and performance. There was also one case of illegal use of data.*
Technical bias	Arises due to technical limitations and considerations. *E.g.: The sentiment analysis is not precise and frequently classifies statements incorrectly. Human accuracy is at 85.7 %, ML algorithms reach up to 77.1 %*[64].
Emergent bias	Arises in connection with users and through incorporation in real life situations. *E.g.: This would not appear until during use.*

Phase IV:
Value Conflicts and/or Goal Conflicts

Key Questions	Evaluative attitude: Which stakeholder values are in a trade-off, in a tension situation or in conflict with one another? *E.g.: The employee representative defends the employees' right to privacy and the training officer represents the profit interests of the company.* *Critical value reflection: Value-immanent conflicts*

> Volitional attitude: Realization of the values
> Pre-theoretical deliberation:

Assessment and ordering based on empirical content; Explication of options for action and justification (preference relations).
E.g.: As the developer team the well-being of the company that has hired us is more important than a possible but improbable abuse or negative consequences for the call center employees.
Or: We don't want our name to be associated with a product that is used to violate human dignity.

Phase V: Ethical System Testing
(This step is necessary when no reasonable recommendation for action is generated in Phase IV.)

Key questions	Which values and/or which reasons should we follow?

	Deontological	**Consequentialist**	**Virtue ethics/ Professional ethics**
Arguments for	*E.g.: Privacy is under absolute protection*	*E.g.: Smaller compromises are possible in the interest of the success of the company.*	*E.g.: We should fulfill the requirements of the assignment as well as possible.*
Arguments against	*E.g.: The customer is always king.*	*E.g.: A data protection scandal can do permanent damage to the company and also to our individual reputations.*	*E.g.: Do not write software that can be abused for surveillance purposes.*[65]

Theoretical Deliberation:	Is it desirable that such a technology should become a part of our lifeworld?

64 Cf. Zadeh/Chen/Poria/Cambria/Morency (2017).
65 https://www.gnu.org/proprietary/proprietary-surveillance.en.html

Phase VI: Judgment Phase (Coherence)

Should we (moral reasons) / Do we want to (economic reasons, etc.) deliberate about a technical implementation of the feature at all? Do the costs and benefits for the stakeholder fit together? How does the technical realization fit into the central corporate image (Keyword: Corporate Digital Responsibility) / How will we handle the judgments from Phase IV and when appropriate Phase V?
If the judgement is negative → Begin in Phase II
If das judgement is positive → Phase VII

Phase VII: Technical Feasibility

(1) Technical problem: Outline whether and how you would like to proceed technically. Transfer and integration of the analysis to the design: Classification of users, front-end/interface design, back-end design, environment
(2) Can you technically realize your normative judgements, i.e. can you develop a feature that e.g. tracks work time without violating the rights of the employee? How much can be expected of the employees?
(3) Is a technical realization feasible which reflects the normative judgement? What normative aspects cannot be integrated, and why? What does this mean for you: Do you want to develop the feature?

Phase VIII: Verification

— The system is tested in technical and empirical terms to ensure that the standards can be met: What could a test look like? How technically complicated is such a test?
— Technical: »What are good test cases for the system? «

Now the decision on whether or not and how the system should be built is up to the developer team. The important thing is that the reasons have been explicated in transparent and understandable form. We assume that joint ethical deliberation alone will lead to responsible handling of software systems on the part of companies, developers as well as the users.[66]

Conclusion

At present it is technically impossible to program machines and/or algorithms in such a way that they can independently generate ethically desirable results. Nevertheless machines are increasingly becoming a part of our lifeworld and are integrated more and more deeply in our society and our everyday lives. Because of their enormous scope and their influence on human decisions, these systems have a strong normative power. Software systems intentionally or unintentionally steer decision-making, which is why in both the development process and in the application of technology care should be taken to avoid unwanted as well as intentional implications. At present these design decisions are often arbitrary, i.e. usually made on an intuitive and associative basis, through interaction of developers and technology. Very often the usually implicit ethical understanding of the supervisors or rhetorically strong minorities prevails. It is therefore crucial that this often unstructured process be counteracted with a semi-structured process based on scientific findings in order to localize normative requirements and

[66] A corresponding study is already in planning.

– whenever and wherever necessary – to discuss them and take such requirements into account. The EDAP method presented here can be seamlessly integrated in existing development processes, supports ethical deliberation not only on the part of the developer team, but also on the part of the entire company – and this without significant extra effort. It is as such a contribution to the debate on ethical systems and offers a pragmatic and practically viable approach to developing systems under the governance of ethical considerations.

Literature

Battaglia, F. / Mukerji, N. / Nida-Rümelin, J. (Hrsg.) (2014): Technology and Responsibility. Pisa: Pisa University Press.

Betzler, M. (2011): Erziehung zur Autonomie der Elternpflicht. In: Deutsche Zeitschrift für Philosophie, 59, Heft 6, S. 937–953.

Conitzer, V. / Sinnott-Armstrong, W. / et al. (2017): Moral Decision Making Frameworks for Artificial Intelligence. In: Thirty-first AAAI-Conference on Artificial Intelligence.

Dasgupta, I. / et al. (2019): Causal reasoning from meta-reinforcement learning. arXiv preprint arXiv:1901.08162.

Donaldson, T. / Preston, L. E. (1995): The Stakeholder Theory of the Corporation: Concepts, Evidence, and Implications. In: Academy of Management Review, 20(1)1, S. 65–91.

Eubanks, V. (2018): Automating Inequality – How High Tech Tools Profile, Police, and Punish the Poor. New York: St. Martin's Press.

Foot, P. (2003): Virtues and Vices. And Other Essay in Moral Philosophy. Oxford: Oxford University Press.

Freeman, R. E. (2010): Strategic Management: A Stakeholder Approach. Cambridge: Cambridge University Press.

Friedman, B. (1997): Human Values and the Design of Computer Technology. Stanford: Center for the Study of Language and Information Stanford University.

Friedman B. / Kahn P. H. / Borning A. / Huldtgren A. (2013): Value Sensitive Design and Information Systems. In: Doorn N. / Schuurbiers D. / Van de Poel I. / Gorman M. (Hg.): Early engagement and new technologies: Opening up the laboratory. Philosophy of Engineering and Technology, Vol. 16. Dordrecht: Springer.

Friedman, B. / Hendry, D. G. (2019): Value Sensitive Design – Shaping Technology with Moral Imagination. Cambridge, MA: MIT University Press.

Henning, T. (2019): Allgemeine Ethik. Paderborn: Wilhelm Fink Verlag.

Hildebrandt, F. / Musholt, K. (2020): Teaching Rationality – Sustained Shared Thinking as a Means for Learning to Navigate the Space of Reason. In: Journal of Philosophy of Education, Vol. 00, No. 0.

Kiesewetter, B. (2017): The Normativity of Rationality. Oxford: Oxford University Press.

Kohlberg, L. (1996): Die Psychologie der Moralentwicklung. Frankfurt am Main: Suhrkamp.

Lee, D. (2016): How Airbnb short-term rentals exacerbate Los Angeles's affordable housing crisis: Analysis and policy recommendations. In: Harvard Law & Policy Review. 10, S. 229.

Lin, P. / Ryan, J. / Abney, K. (Hrsg.) (2017): robot ethics 2.0. Oxford: Oxford University Press.

Loi, M. / Heitz, C. / et al. (2019): Towards an Ethical Code for Data-Based Industry. In: 2019 6th Swiss Conference on Data Science. IEEE, S. 612.

Lord, E. (2018): The Importance of Being Rational. Oxford: Oxford University Press.

MacIntyre, A. (1981): After Virtue. Notre Dame: University Press of Notre Dame.

Mill, J. S. (Erstveröffentlichung 1869): Über die Freiheit. Stuttgart: Reclam Verlag.

Misselhorn, C. (2018): Grundfragen der Maschinenethik. Stuttgart: Reclam Verlag.

Mohri, M. / Rostamizadeh, A. / Talwalkar, A. (2018): Foundations of Machine Learning. Cambridge: MIT University Press.

Nida-Rümelin, J. (2005): Über menschliche Freiheit. Stuttgart: Reclam.

Nida-Rümelin, J. (2015): Theoretische und angewandte Ethik: Paradigmen, Begründungen, Bereiche. In: Angewandte Ethik – Die Bereichsethiken und ihre theoretische Fundierung, S. 2–88.

Nida-Rümelin, J. (2016): Gründe und Lebenswelt. In: Information Philosophie, Heft 2, S. 8–19.

Nida-Rümelin, J. / Weidenfeld, N. (2018): Digitaler Humanismus – Eine Ethik für das Zeitalter der Künstlichen Intelligenz. München: Piper.

Nida-Rümelin, J. (^{3}2019): Die Optimierungsfalle – Philosophie einer humanen Ökonomie. München: btb Verlag.

Nida-Rümelin, J. (i. E. 2020): Theorie einer praktischen Vernunft. Berlin: De Gruyter.

Nissenbaum, H. (2005): Values in Technical Design In: Encyclopedia of Science, Technology and Ethics, hg. von C. Mitcham. New York: Macmillan, S. ixvi-ixx.

Noble, S. U. (2018): Algorithms of Oppression: How Search Engines Reinforce Racism. North Yorkshire: Combined Academic Publ.

Nussbaum, M. C. (1999): Virtue Ethics: A Misleading Category? In: The Journal of Ethics Vol.3, No.3, S. 163–201.

O` Neil, C. (2017): Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy. London: Penguin.

Pearl, J. / Mackenzie, D. (2018): The Book of Why: the New Science of Cause and Effect. New York: Basic Books.

Simon, J. (2012): E-Democracy and Values in Information Systems Design. In: Proceedings of the XXV World Congress of IVR, Special Workshop on »Legitimacy 2.0: E-democracy and Public Opinion in the Digital Age«, S. 40–64.

Simon, J. (2016a): Values in Design. In: Heesen J. (Hrsg.): Handbuch Medien- und Informationsethik. Stuttgart: J. B. Metzler.

Simon, J. (2016b): Value-Sensitive Design and Responsible Research and Innovation. In: Hansson, S.-O.: The Ethics of Technology- Methods and Approaches. London: Rowman & Littlefield International, S. 219–236.

Thoreau, H. D. (Erstveröffentlichung 1849): Civil Disobedience. CreateSpace Independent Publishing Platform.

Vallor, S. (2016): Technology and the Virtues. A Philosophical Guide to a Future Worth Wanting. New York: Oxford University Press.

Wallach, W. (2010): Moral Machines. Teaching Robots Right from Wrong. New York: Oxford University Press.

Wedgewood, R. (2017): The Value of Rationality. Oxford: Oxford University Press.

Zadeh, A. / Chen, M. / Poria, S. / Cambria, E. / Morency, L. P. (2017): Tensor fusion network for multimodal sentiment analysis. arXiv preprint arXiv:1707.07250.