The Education of the Human Factors Engineer in the Age of Data Science

Daniel Hannon[1], Esa Rantanen[2], Ben Sawyer[3], Ashley Hughes[4], Katherine Darveau[1,5], Robert O'Donnell[1], James Intriligator[1], and John D. Lee[6]

[1]Tufts University, [2]Rochester Institute of Technology, [3]University of Central Florida, [4]University of Illinois, [5]GE Aviation, [6]University of Wisconsin-Madison

The continued advances in artificial intelligence and automation through machine learning applications, under the heading of data science, gives reason for pause within the educator community as we consider how to position future human factors engineers to contribute meaningfully in these projects. Do the lessons we learned and now teach regarding automation based on previous generations of technology still apply? What level of DS and ML expertise is needed for a human factors engineer to have a relevant role in the design of future automation? How do we integrate these topics into a field that often has not emphasized quantitative skills? This panel discussion brings together human factors engineers and educators at different stages of their careers to consider how curricula are being adapted to include data science and machine learning, and what the future of human factors education may look like in the coming years.

 10.1177/1071181320641109

INTRODUCTION

The growing influence of data science (DS) and machine learning (ML) on automation continues, generating new opportunities, and leading, potentially, to new automation failures. As before, human factors engineers (HFEs) are able to guide the design and implementation of human-machine-systems to avoid catastrophe and to assure effective solutions.

However, like other aspects of the field, HFEs must learn how to work with ML practitioners, understand the tools, and understand how ML and DS techniques frame approaches to problem solving. This knowledge allows us to steer toward satisfying, safe, and useful solutions.

Unfortunately, many graduate programs in HFE, as well as those in related fields, have not emphasized quantitative skills as admission criteria nor have they developed DS content within them. Presently, many HFEs are unprepared to participate in research and development (R&D), or consulting efforts that utilize ML for lack of knowledge, training, or a sense of how to contribute.

This paper builds on Hannon et al. (2019), as a call to examine graduate curricula with respect to DS and ML and to develop a framework for positioning HFEs to participate in the future of automation development. A panel of HFEs from academia, government and industry, provides insights on challenges in the redesign of graduate HFE training regarding DS and ML in the area of automation and artificial intelligence (AI). Topics included are what attracts students to HFE, a reexamination of admission criteria, the relationship between ML, DS and current HF program content, and the role for HFEs in future human-machine systems, automation, and AI.

The Challenge of ML to HFE

The use of the term ML here refers to the application of quantitative techniques applied to input data for the purpose of modeling and predicting outcomes. ML techniques are designed to find patterns and relationships, often in multi-dimensional data spaces, that are not obvious when using more conventional analytics, such as those employed in behavioral science. A common feature of ML approaches is that they attempt to minimize a cost function over many iterations through the data, adjusting internal parameters until a criterion is met. Unlike conventional behavioral statistics that rely on "strong assumptions" (e.g., assuming linear relationships), ML approaches can accommodate nonlinear relationships.

Different types of ML have been developed, with one of the distinctions being between supervised learning, in which the outcomes learned are identified in advance, and unsupervised learning in which the algorithm finds patterns without prior knowledge of the data. Both approaches pose unique and similar challenges regarding the assumptions made about the relationships in the data and the utility of the results.

Hannon et al. (2019) used examples of classifier systems (i.e., supervised learning) and neural network models (in this case, unsupervised learning) to highlight specific challenges in ML approaches with respect to the design of automated systems. With respect to supervised learning, open questions include understanding the impact of how data are labeled prior to model training, defining representative training sets, understanding how optimization influences model performance, and what unanticipated consequences may result from using the models. Regarding unsupervised learning, there are similar concerns with preparing data for use in the model, deciding on criterion performance levels of the model, and utilization of results. Regardless of the approach, there are decisions that can have a large influence on the ML process and outcome.

The essential point is that without consideration of what is referred to here as the human factors elements of ML models, they will continually fall short of expectations. It is not uncommon for a ML practitioner to receive a labeled data set, such as a medical record database, and apply modeling techniques without being able to fully explor the validity or reliability of the data. The results then may be generalized beyond a prudent scope. Model performance that correctly classifies results at 85% or even 95%, for example, often are considered to be good results, without any consideration of the downstream consequences of when the models make mistakes. A 95% error rate would be unacceptable in many practical situations from the user's perspective. Indeed, failures by ML models to match the needs of human teammates is often couched as 'edge cases' of the model, and the familiar 'human error', an all-to-familiar explanatory pattern ignoring the need to match operator and system capabilities. It is, however, unsurprising that these foundational concerns of our community are neglected, given HFEs' lack of training which could provide them a voice.

Historical HF contributions to aviation did not require each contributor to be an aeronautical engineer, but did rely on HFEs understanding the functional relationship between aircraft and human. Likewise, it may not be critical for every HFE to be able to write or run the ML algorithms, but it is essential that HFEs understand how the human engages with their generation and use. This includes knowing what the data are, what they mean, how labels are assigned, etc. It also includes knowing the outputs, in terms of system performance and the implications for the system users. Herein lies the challenge for the HFE community; how do we ensure new HFEs understand when many existing members do not?

What HFEs are Taught about Automation

HFE educators continue to seek input from industry and the wider research community in an effort to keep the curriculum both current and relevant (e.g., see Rantanen 2016; Rantanen, et al. 2016; and Rantanen et al. 2018). Hannon et al. (2019) pointed out that HFE has continued to focus on "first generation" automated systems. Noting that the pace of change in technology "has been running well ahead of HFE", we now ask whether we are focusing on the right topics.

Noting Sheridan (2002), among others, the topics of automation and HF are interwoven. Indeed, the concept of an autonomous system with no human input is only one of many automation levels, with most having some user awareness of the system. Human-in-the-loop supervisory control is central in automation, and is taught as a fundamental principle.

Much of automation education is based on lessons from past automation failures that provide a cautionary tale. However, as noted by Hannon et al. (2019), the goal of automation with limited human input continues in industry and it is time for HFE educators to revise our thinking.

The Structure of HFE Curricula

A recent discussion by Rantanen et al. (2016) highlighted the need for cross training for HFEs. They noted that beyond awareness of the work of designers (notably, Industrial and Graphic Design), HFEs need to be exposed to the skill sets of designers. This awareness fosters an appreciation for the value brought by designers and an understanding of how to blend HFE work with the work of these colleagues. A similar approach is needed with ML and DS. The question is how to revise the current HFE curriculum to meet this challenge?

Many of the methods utilized in the design process are shared with those from a design background, creating a natural affinity between HFEs and Designers. The same may not be true for HFEs and engineers who may champion AI-based automation excluding the human. How then to prepare the HFEs to interact effectively with engineering colleagues?

A good starting place is to consider what attracts students to graduate study in HFE. As noted, an affinity between HFE and Design is one touch point in which students with technical, psychological and artistic inclinations find a comfortable balance. Therefore, what are the affinities between HFE and AI? Although early AI was an outgrowth of cognitive and computer sciences, it presently appears to be very non-human and almost antithetical to the HFE mission. How do we (re)humanize the study of AI?

One might presume that an understanding of ML and DS may require some prior quantitative skills and an affinity for quantitative problems. Presently, the admission criteria for graduate education vary considerably, depending on the specific school, but also stemming from the type of department or school in which the program is found. Is there a common background in DS that future HFEs will need upon admission to graduate school? If so, how should we encourage this?

Within our programs, what should we teach about ML and DS? Is it sufficient to point our students to programs in computer science or informatics and expect that they will transfer and integrate learnings within HFE on their own? Or do we need specific ML/DS course content relevant to HFE?

Lastly, how do we train HFEs to interact with AI engineers in order to add value? With Designers, shared work processes help to equalize conceptual understanding. Is there an equivalent set of procedures that are shared between the HFE and AI engineers? Are there unique ways in which the HFE can offer insights not presently captured in AI development, such as applications of behavioral statistics or a perspective of the entire DS pipeline that will add value? The panelists here share their insights into their respective programs as they continue to wrestle with the challenges outlined above.

PANELIST INPUTS

John D. Lee, University of Wisconsin—Madison: The highly publicized failures and stunning achievements of ML systems point to important opportunities for HFEs to contribute and benefit from this technology. To engage in this field, HFEs must develop familiarity with basic concepts of ML and DS. Gaining this requires investing time to learn new methods, but this is repaid by ML techniques improving HFE research and methods and by HFE techniques and considerations improving designs that rely on ML.

Core techniques of ML remain unfamiliar to HFEs even though they apply directly to our research. Machine learning has a focus on predicting outcomes and not estimating p-values. This *prediction mindset* could help HFEs think about data analysis in a way that focusses attention on whether the variables of interest contribute to predictive accuracy in a practically significant manner and under what conditions does the model fail. This prediction mindset also highlights the need for cross validation in a way that can weed out spurious associations. ML has also developed techniques to support the analytic process that could promote more replicable research, such as assertive data checking and opinionated data analysis.

Most HFEs have a deep appreciation for human behavior and experimental design. This includes an appreciation for concepts of representative sampling, generalization, and the challenge of extracting causal explanations from observational data. HFEs can bring this *causal mindset* into code reviews to help ML experts identify shortcomings of their models. Similarly, deep knowledge of how people trust (or not) advice and control from machines can guide algorithmic tradeoffs. More specifically, HFEs can suggest how the type of error, the understandability of the algorithm, and subtle definitions of fairness can be included in the cost function that trains algorithms. With a modest investment in learning ML concepts HFE practitioners can engage in important design conversations.

In developing a new course entitled *Data Science for People*, I have found that students with a modest background in statistics can learn the basic concepts that underly ML without too much pain. The pervasive application of ML makes it easy to identify case studies of successful and less successful ML applications, including: Stitch Fix, Facebook, Uber, and COMPAS. These case studies highlight how HFE involvement might improve systems that rely on ML.

Esa M. Rantanen, Rochester Institute of Technology: The problem with lack of HFE impact on new technologies, such as AI- and ML-based automated agents, is not new. AI, ML, and DS merely represent new domains where HFE is not a part of R&D as it is in more traditional domains (e.g., aviation). This problem may be traced to the education system, where professionals first learn the knowledge, skills, and abilities (KSAs) relevant to their work domains.

It is of course very necessary for universities to produce generalists, that is, graduates with KSAs that are applicable and marketable in as many and diverse domains as possible. This best serves the students' interests and gives them greatest return for their tuition investment. Yet, we cannot avoid the notion of tradeoffs between generalists and specialists. This problem may be illustrated by a couple of schematic plots.

Figure 1 illustrates the hypothetical tradeoff between HFE and domain-specific KSAs. The ideal professional is marked with a star in the upper right-hand corner, but we may assume that such ideal is humanly unattainable, save for some rare polymaths. Most likely the solid line represents the present situation: We may hope to train HFE specialists with limited domain expertise, or domain experts with limited HFE knowledge. The goal is to "push the envelope" towards the top right-hand corner in university curricula, as depicted by the dashed line.

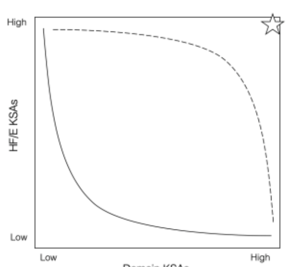

Figure 1. Hypothetical tradeoffs between HF/E and domain-specific KSAs. The solid line depicts presumed current situation, and the dashed a goal for educational institutions, closer to the ideal (star).

Alternatively, we may view the problem in a Venn diagram with two sets, one for HF and the other for any other specific domain. The goal is to increase the size of their intersection in university curricula as much as possible (Figure 2)

With this attempt to define the problem and the goals, we may now discuss ways towards the goals. Figures 1 and 2 suggest two different ways. The first is to include domain-specific content to HFE program curricula. This is what Hannon et al. (2019) proposed and what is also suggested in this paper and panel discussion. One particular initiative in my university is redesign of a graduate Engineering Psychology course for online format. As part of the redesign I have included several lab experiments in the syllabus, programmed in Python, and requiring students to make changes to the code to manipulate independent variables according to their experimental designs. Furthermore, the students are required to learn R to analyze the results from their lab experiments. This way students will gain some familiarity with coding while learning about human performance models in the course.

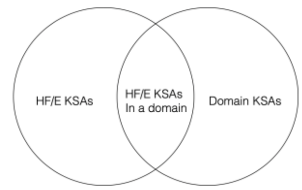

Figure 2. A Venn diagram depicting an intersection between HF/E and other specific domains. The goal is to make the intersection as large as possible within university curricula.

A second approach is to introduce HFE content to the curricula in other domains. Although in my university programs in Human-Computer Interaction (HCI) and Human-Centered Computing (HCC) have dedicated courses in HFE with "standard" HFE content, programs in AI typically do not. I am presently working on a new course proposal for "Cognitive Systems Engineering in Cybersecurity", to be offered as an elective in computing and information science curricula.

A common constraint to both of these approaches is the very short time our students spend in academic programs: only 4-5 years as undergraduates and a mere 2 years as master's students, each semester packed with myriad courses, ranging from general education to highly specific technical courses. Doctoral programs involve lengthy and very narrow dissertation research. Engineering disciplines in particular offer few opportunities for students to add elective courses to the accredited program curricula. However, the current trends towards increasingly interdisciplinary university programs may offer innovative opportunities to introduce HFE content in different academic programs. Reform of college education in this manner is in and of itself a human factors problem, where we may contribute from our expertise in the HFE discipline.

Katherine Darveau, GE Aviation & Tufts University: As HFE and Design have been shown to blend the interests of prospective HFE graduate students, the emergence of AI and associated DS and ML techniques for problem solving are creating a new relationship with HFE that may be very appealing to the modern student. An HFE education is attractive for many reasons, one of which is the human-centric approach to the design of products and systems, a perspective that more traditional engineering curricula do not always offer. The level of technical depth and quantitative skill development offered by an HFE program may be an important element to today's prospective students. The option to integrate DS and ML with HFE may provide a more appealing, and highly relevant, quantitative alternative to existing HFE curricula.

In an era where automation is part of everyday life and computer code enables so many actions and interactions in the day, students may be inclined to consider graduate programs in the data science or computer engineering. Students constantly see this technology in practice and are aware of the opportunities. While some may be more content oriented in developing the code, others are equally, if not more, interested in the human aspects of AI: data collection and the impact of data quality, dissection and selection of inputs, human interaction with the AI, and implications of model output and performance. For prospective HFE students, DS and AI curricula present a clear intersection that will offer the development of a valuable, complimentary, multidisciplinary skillset.

It will be important to develop curricula that utilizes industry examples to build a skillset for effective interaction with AI engineers. Discussions with industry professionals in AI and Applied Statistics show that bringing an "HFE mindset" to such teams is of great value. HFE professionals can provide a framework for utilizing knowledge of human cognition and behavior in the design of AI systems. Such value will undoubtedly be proven quantitatively through improved modeling performance and more effective outputs. While the AI engineer may be inclined to fully automate solutions, an HFE professional should be equipped to offer human-machine teaming approaches that combine elements of AI with human feedback or oversight. HFEs may also be privier to ML modeling techniques that utilize human domain-specific knowledge to improve feature extraction (e.g., creating domain-specific dictionaries or weighting key terms).

There is significant opportunity to develop future HFEs into informed contributors whose unique perspectives may identify unseen shortcomings and opportunities for improved modeling. Developing an appropriate level of technical DS/ML/AI awareness might include a look into the mindset of an AI engineer. What are their focus areas? How do they approach modeling? What questions might they ask, and what considerations might they miss? This knowledge would help HFEs to be effective collaborators on AI problems.

Robert O'Donnell, U.S. Coast Guard & Tufts University: The application of machine learning can be a game changer in the way aviation safety systems are introduced and integrated into the cockpit. By utilizing ML, military and civilian aviation can take a more proactive stance on how to recognize critical, human degraders in order to ensure operational success.

By utilizing the predictive abilities of ML, the aircraft system can be used as a safety tool, providing information about the aircraft, and alerting pilots and aircrew about their own physiology. Imagine a pilot speaking into the aircrafts communication system and the aircraft alerts the pilot that they are fatigued – or of some other degraded state. This will allow the pilot to make an appropriate decision, while avoiding unnecessary risk. In order to develop and implement this type of process, it is imperative that future HFEs are exposed to the breadth of possibilities that ML and DS offer.

Ben D. Sawyer, University of Central Florida: In teaching 'Probability and Statistics for Engineers' and 'Human-Computing Interaction', I consider the integration of ML and AI concepts into both. One of my primary goals is to move ML from seeming magical and mysterious, to approachable and amenable to scrutiny. This can be challenging. In statistics, for example, one recent test assignment explored ideas about eigenvalues and computer vision, aspects of an actual face recognition project (see Sawyer, Teo, & Mouloua, 2012). While I intended for students to 'discover' the idea that ML is the extension of the statistics we have learned, none did. Instead most wrote about the miraculous performance of a system built from simple concepts, and nearly all touching upon the idea that as 'black boxes', these systems defy attempts at understanding. From this foundation of challenge, I am now redesigning a larger portion of my course. A work in progress, these changes focus on four goals: 1) helping students to understand that they can understand machine learning and apply it, using simple examples; 2) an attempt at systematic processes and a tool which is capable of being learned within the tight time confines of my class; 3) I provide the datasets and time for individuals to develop confidence; 4) I use this simple example as a platform from which to examine more complex types of ML and AI. Note that this is competing for time in a complex and demanding class, in which we move from the basics of distributions through running ANOVA, and time and student attention constraints are limited. Nonetheless, I believe that ML and AI concepts are worth the journey, for me and my fledgling undergraduates.

My graduate-level "Human-Computing Interaction" class is more challenging, with diverse levels of expertise. This class attracts students from Industrial-organizational Psychology through Electrical Engineering, with many levels of ML skill and understanding present in the class. A team-based experience, partners propose world-changing ideas hackathon-style, and then poach one another until only a few teams remain. I expected the real challenge was less about education, and more on fostering communication and collaboration between classmates with divergent backgrounds. I was partially right, but far above any other challenge has been that of the use of 'magical' ML to solve problems. All projects have a mandate to remain 'technologically feasible', and conversations abound regarding how AI will allow feats including instantly curing major diseases, enabling nonexistent sensor technologies, and eliminating the restrictions of space and time. These conversations have me designing a stronger up-front component focusing on what ML and AI can reasonably accomplish, common pitfalls in human-agent teaming (see Sawyer & Hancock, 2018), how data inputs inform outputs, and other recommendations in this very publication.

There is a fine line here, which I urge readers to recognize: what a professor says is impossible may not in fact be so. Nonetheless, I am often surprised to find even competent builders of ML, and practitioners of DS, attribute seemingly fantastical properties to their systems. This, to be clear, is true both in the classroom, and in my work with industry and startups. I would suggest that higher levels of education in building ML/AI technology do not eliminate blind spots in terms of capability of such technologies. Indeed, in my class I have discovered a microcosm for some challenges of my professional life. I invite the reader to join me in acquiring the data and developing the techniques to train more robust partners.

Ashley M. Hughes, University of Illinois: With high demand for healthcare human factors engineering in research and quality care initiatives, we turn to the overlapping yet similar field of health informatics (HI) to instruct HFE professionals and students on health data science. Health data bears specific considerations for security, privacy, access, and in many cases, fragmented, incomplete, or text-based data systems. A key example is understanding the need, role, purpose, and limitations of ML, AI, and natural language processing (NLP). For instance, data captured in the Electronic Health Record (EHR) is both structured (e.g., blood pressure) and unstructured (e.g., clinical notes) data. HFE professionals and students who wish to pursue an HFE career in healthcare require additional training (e.g., health data security or health insurance privacy and portability act [HIPPA] training). In particular, the primary purpose for which data is captured influences its quality and utility for which the HFE needs to be prepared. For example, a clinical decision support (CDS) system- including use of alerts- hinge upon the timeliness and quality of data entered into the EHR. However, oftentimes, key data are entered late in the day by overworked, burned-out clinicians. Further, much of this information is captured via free text notes, expanding the need for data mining techniques to handle unstructured data (e.g., NLP; ML).

I teach capstone students, upper level graduate courses, and work closely with Veterans Health Administration on research involving health data infrastructures. The burgeoning need for ML, AI, and NLP in understanding the complexity of health information is pivotal to the progression of the HFE. As data quality and structure inform available analytic techniques and potential interpretations, I will speak to the selection of techniques as appropriate to data infrastructures, including tradeoffs in methods employed. These key skills are necessary for the progression of HFE as a field and particularly in its application within the healthcare sector.

James Intriligator, Tufts University: Like many, I envision a future where ML helps to make it possible for people and machines to interact naturally and effortlessly. In that future, ML will help systems function efficiently and organically. However, we need a pathway there.

First, future HFEs must see machine learning as a tool in the toolbox or building block in the system. Our students know they can make a system “better” by changing the form, shape, display type, or input mechanism. They should think of ML as another way to change a system. ML is as another element that can make a system better matched to its user.

By way of example, consider cognitive task analysis (CTA) in which an HFE might seek to identify points of decision. At any of these decision points, there may be opportunities to include an ML building block. In the case of online shopping, current systems include a machine learning block that makes a recommendation. The system mines past purchases, actions, or queries and uses that information to suggest a specific selection or action, and based on recent purchases, it offers recommendations for other items. When we look at a CTA, we should look for any points where the operator needs to make a choice or perform an action, and we should ask whether machine learning can facilitate that choice or action.

A second place where ML building blocks are valuable is in the design of information flow. At system branch points, ML building-blocks can help decide how to channel the information. Imagine a system that analyses the voice of a caller and classifies them into a category and routes the caller to the best agent. Or, the system prompts the agent with what to say next. The reader might notice that this is, in some sense, actually the same kind of decision-making facilitation as in the CTA method. But, here it is not the operator’s decision making that is being facilitated - instead, it is the machine’s decision making, that is why I refer to this as “machine-decision”.

We need to be sure the next generation of HFEs understand where ML building-blocks can come into play in their designs. The HFE professional may never need to write ML code. But, they will need to identify opportunities for including specific ML functions that solve specific problems.

How can we get our students thinking like this? The first step is ensuring that they are trained to understand two important things: first, <u>the basic logic of ML</u>. For example, they should become familiar with classic neural-net models and back-propagation. This will give them a deeper understanding of the crafting and training ML systems. Secondly, they should be trained in <u>the strengths and abilities of ML systems</u>. These are: the ability to recommend and the ability to classify. They can gain this by exposure to a range of ML systems.

Our students should think of ML as building-blocks in their arsenal of tools. The ability to use these *well* requires an understanding of the logic of ML and an understanding of the strengths and abilities of ML systems. We must build that understanding coupled with training in the methods and tools for identifying opportunities for incorporating ML into systems. If we can give our students these tools, then they can build the unimaginable operator-machine systems of the future.

REFERENCES

Hannon, D., Rantanen, E., Sawyer, B., Ptucha, R., Hughes, A., Darveau, K., & Lee, J. D. (2019, November). A Human Factors Engineering Education Perspective on Data Science, Machine Learning and Automation. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 63(1), 488-492.

Rantanen, E., Rantanen, A., & Lau, M. (2018, September). Cross-training in Design and Human Factors. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 62(1), 364-368.

Rantanen, E., Boehm-Davis, D., Boyle, L. N., Hannon, D., & Lee, J. D. (2016, September). Education of Future Human Factors Professionals. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 60(1), 418-421.

Rantanen, E. M. (2016, September). Trends in Knowledge and Skill Expectations for New Human Factors Professionals. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 60(1), pp. 370-374.

Sawyer, B., Teo, G. and Mouloua, M., (2012). DriveID: Safety innovation through individuation. Work, 41(2012), 4273–4278.

Sawyer, B. D., & Hancock, P. A. (2018). Hacking the Human: The Prevalence Paradox in Cybersecurity. Human Factors, 60(5), 597-609.

Sheridan, T. B. (2002). Humans and automation: System design and research issues. Human Factors and Ergonomics Society.