Validierung von ML-Libraries

Mit Professor Hasse, Prof. Dr. Christian Johner

Transkript

00:00:05 Sprecher 1

Medical Device Insights.

00:00:08 Sprecher 1

Ein Podcast des Johner Instituts für Medizinprodukteh ersteller, Behörden und benannte Stellen.

00:00:18 Sprecher 1

Unser Podcast hier ist sicherlich dafür bekannt, dass wir immer dünne Bretter bohren und wir planen zwar für die Zukunft auch ein paar

00:00:27 Sprecher 1

einführende Themen für Beginner, für Anfänger, aber dieses Mal wird es nicht der Fall sein.

00:00:33 Sprecher 1

Dieses Mal bohren wir extrem dicke Bretter und all diejenigen, die denken, Machine Learning und Regularien, das interessiert mich nicht besonders, die dürfen ausnahmsweise mal diese Podcastfolge überspringen.

00:00:47 Sprecher 1

Alle anderen dürfen sich freuen auf echten dicken Content und ich hab heute bei mir den Oliver Haase,

00:00:55 Sprecher 1

mit dem ich schon eine Weile zusammenarbeite im Kontext maschinellen Lernens und auch der Validierung von Machine Learning Libraries.

00:01:02 Sprecher 1

Oliver, willst du dich ganz kurz vorstellen, damit unsere Hörer wissen, wen ich hier zu Gast haben darf?

00:01:07 Sprecher 2

Ja, sehr gern, lieber Christian.

00:01:08 Sprecher 2

Vielen Dank, dass ich heute hier sein darf.

00:01:10 Sprecher 2

Ich habe einen Hintergrund in Informatik.

00:01:12 Sprecher 2

Ich bin Informatiker, habe eine Professur für Software Engineering und beschäftige mich seit vielen Jahren mit Softwareverifikation und Softwarevalidierung.

00:01:22 Sprecher 2

Und seit einiger Zeit auch angewandt auf Machine Learning, das heißt eben Machine Learning, vor allem auch im Kontext von regulierten Märkten wie Medizingeräten.

00:01:33 Sprecher 2

Und dabei treibt mich die Leitfrage, vor allem die Leitfrage, um, wie man die Korrektheit einer Machine Learning Anwendung überprüfen und nachweisen kann.

00:01:44 Sprecher 2

Und dieser Nachweis geht weit über die Modellevaluation hinaus, wie wir sie

00:01:51 Sprecher 2

vom Machine Learning her kennen.

00:01:53 Sprecher 1

Ich vermute mal, dass du jetzt vor allem auf die Libraries anspielst, die es auch zu validieren gibt.

00:01:59 Sprecher 1

Für was braucht man denn die, was, was sind das für Libraries, die die meisten da mit einsetzen und die eben wahrscheinlich auch einer Validierung bedürfen?

00:02:07 Sprecher 1

Zumindest habe ich dich jetzt gerade gehört, als du gesagt hast, es geht jetzt nicht nur um das Modell selber.

00:02:12 Sprecher 2

Ja genau, Christian, ich spiele natürlich auch auf diese Libraries an, die man auch Frameworks nennt, also Machine Learning Libraries oder Machine Learning Frameworks.

00:02:20 Sprecher 2

Und das sind solche Dinge wie beispielsweise PyTorch, TensorFlow, Carras für neuronale Netze oder X.G.

00:02:29 Sprecher 2

Boost für Gradient Boosting.

00:02:31 Sprecher 2

Und diese Libraries sind Sammlungen von Machine Learning Algorithmen, mit deren Hilfe man Modelle, Machine Learning Modelle erzeugen kann, wenn man Trainingsdaten, wenn man gute Trainingsdaten

hat.

00:02:44 Sprecher 2

Und diese Machine Learning Modelle dann wiederum können dann am Ende Vorhersagen machen, indem sie

00:02:50 Sprecher 2

aus den Trainingsdaten Muster extrahieren, Muster erkennen und extrahieren.

00:02:54 Sprecher 2

Und diese Vorhersagen sind dann beispielsweise gut, um auf C.T.

00:02:59 Sprecher 2

Bildern Krebsgeschwüre zu erkennen.

00:03:02 Sprecher 2

Und dieses Trainieren von den Modellen, das ist eine, das ist eine sehr komplexe Aufgabe, eine algorithmisch sehr komplexe Aufgabe.

00:03:10 Sprecher 2

Und deswegen sind diese Machine Learning Libraries auch der eigentliche Grund, warum heute vergleichsweise einfach jeder

00:03:18 Sprecher 2

Modelle, Machine Learning Modelle erzeugen kann, wenn er gute Trainingsdaten zur Verfügung hat, weil eben diese Algorithmen schon da sind.

00:03:25 Sprecher 2

Man kann das Gleiche theoretisch auch ohne Bibliotheken machen, also man braucht die nicht unbedingt, aber das würde Personen Jahre an Entwicklungsaufwand bedeuten, wenn man das alles per Hand noch mal nach implementieren wollte.

00:03:39 Sprecher 1

Ähnlich wie bei den anderen Bibliotheken, stellt sich natürlich auch hier die Frage, ja, wie ist das jetzt regulatorisch zu betrachten und

00:03:48 Sprecher 1

wir kennen natürlich jetzt alle die die Subanforderung, also die Anforderung an die Software of Unon Provenance.

00:03:55 Sprecher 1

Da gibt uns ja die 62 304 schon mal einiges vor, was wir da machen müssen, was wir an Anforderung spezifizieren müssen, wie wir die ja validieren oder verifizieren müssen.

00:04:06 Sprecher 1

Wenn ich dich richtig verstehe, haben wir hier aber ein komplexeres Szenario.

00:04:11 Sprecher 1

Also hier geht es nicht nur um die 62 304 und die Sub

00:04:16 Sprecher 1

Validierung.

00:04:17 Sprecher 1

Kannst du unseren Hörern dann noch mal ‚ne Einführung dazu geben, wie das regulatorisch zu betrachten ist und was man dann auch ganz konkret machen muss, um die regulatorischen Voraussetzungen zu schaffen?

00:04:27 Sprecher 2

Ja, sehr gerne.

00:04:28 Sprecher 2

Ja, das ist genau richtig.

00:04:30 Sprecher 2

Also, das ist anders einzuschätzen als bei der traditioneller Software.

00:04:34 Sprecher 2

Insofern,

00:04:35 Sprecher 2

als dass diese Machine Learning Bibliotheken 2 verschiedene Rollen spielen.

00:04:40 Sprecher 2

Sie spielen nicht nur die Rollen, die Rolle einer SOOP, wenn sie dann im fertigen Medizinprodukt eingesetzt werden, sondern sie spielen außerdem auch noch die Rolle eines Software-Tools während des Trainingsprozesses.

00:04:53 Sprecher 2

Also, um oder um das konkreter zu sagen, wenn ein Modell, solange ein Modell trainiert wird,

00:05:00 Sprecher 2

Mithilfe einer Machine Learning Library spielt diese Library im Trainingsprozess die Rolle eines Software-Tools und ist deswegen nach ISO 13485 reguliert, also nach der harmonisierten Norm für Qualitätsmanagement.

00:05:15 Sprecher 2

Und das fertig trainierte Modell, das im Medizingerät verwendet wird, das spielt die Rolle einer SUB, wie du schon erwähnt hast, einer Software-Fun-on-Provenance.

00:05:27 Sprecher 2

und muss dann demnach nach 62304 validiert werden.

00:05:33 Sprecher 2

Und die insgesamt ist die gute Nachricht, dass wenn man, wenn man die Machine Learning Library nach diesen Regularien validiert, dass man sie dann durchaus auch verwenden darf im fertigen Produkt und auch für den Entwicklungsprozess, obwohl es eben Drittanbietersoftware ist.

00:05:50 Sprecher 2

Und das ist natürlich eine sehr gute Nachricht, weil wie du schon gesagt hast, das ist eine

00:05:55 Sprecher 2

ein ein großartiger Quell an Wiederverwendung.

00:05:59 Sprecher 1

Damit haben wir ja eigentlich eine gedankliche Teilung, die wir machen müssen bei diesen Bibliotheken.

00:06:04 Sprecher 1

Ich glaube, da wird man nachher noch mal kurz drauf zu sprechen kommen, dass wir einen Teil letztlich betrachten müssen als ja durch Kapitel 416 reguliertes Computerized System, das validiert werden muss, also das oder als Prozesswerkzeug und auf der anderen Seite einen anderen Teil, den wir dann aus 62304 Sicht als Sub

00:06:25 Sprecher 1

validieren oder verifizieren müssen und das ist sicher ,ne ,ne Besonderheit.

00:06:29 Sprecher 1

Siehst du noch andere Unterschiede zu klassischer Software als im Sinne Drittbibliotheken, was weiß ich, zum Beispiel ein Statistikpaket, die jetzt da drüber hinausgehen?

00:06:41 Sprecher 2

Ja, absolut.

00:06:42 Sprecher 2

Also auf diese Unterscheidung gehen wir nachher tatsächlich gern noch mal genauer ein, was das dann für die Validierung bedeutet.

00:06:49 Sprecher 2

Der weitere Unterschied oder ein wesentlicher weiterer Unterschied

00:06:53 Sprecher 2

ist einfach die deutlich größere Rolle, die so eine Library einnimmt im Gesamtprodukt.

00:07:01 Sprecher 2

Bei traditioneller Software ist es so, dass der Hersteller einen Großteil, üblicherweise einen Großteil der Software per Hand entwickelt und dann bestimmte Teilfunktionalitäten durch Drittanbieterbibliotheken erledigt.

00:07:15 Sprecher 2

Und das ist bei Machine Learning komplett anders.

00:07:17 Sprecher 2

Da ist nämlich das nahezu oder nicht nur nahezu, da ist das komplette

00:07:22 Sprecher 2

Ergebnis ist eigentlich Soup.

00:07:24 Sprecher 2

Der im Machine Learning Entwicklungsprozess entwickelt der Hersteller nicht wirklich Software selbst, sondern er verwendet seine Trainingsdaten.

00:07:36 Sprecher 2

Er konfiguriert die Machine Learning Library und am Ende ist der Code, der dabei entsteht, ist komplett Drittanbieter-Code.

00:07:46 Sprecher 2

Das heißt, das Modell

00:07:48 Sprecher 2

ist eigentlich in seiner Gänze Sub.

00:07:50 Sprecher 2

Das ist eine völlig andere zentrale Rolle, die diese Bibliothek einnimmt im Vergleich zu traditioneller Softwareentwicklung und damit kommt natürlich auch dieser Subvalidierung ein ganz anderer Stellenwert zu.

00:08:05 Sprecher 1

Das ist, glaub ich, ein ganz gutes Stichwort, nämlich jetzt sind wir schon bei der Validierung und was, glaub ich, den meisten klar ist, dass es regulatorisch gefordert ist, dass das Modell nachher validiert wird, hat man damit nicht

00:08:18 Sprecher 1

dann auch die Bibliothek indirekt mit validiert.

00:08:24 Sprecher 2

Nein, leider nein, und zwar aus aus 2 aus 2 verschiedenen Perspektiven nicht, einmal regulatorisch nicht und einmal auch nicht inhaltlich.

00:08:33 Sprecher 2

Also regulatorisch ist es völlig eindeutig vorgegeben von der 62 304, da gibt es klare Anforderungen an SOOPs,

00:08:43 Sprecher 2

unter denen man sie einsetzen darf.

00:08:45 Sprecher 2

Und dazu gehört vor allem, dass man die erwartete Funktionalität spezifizieren und dann auch validie-

ren muss.

00:08:52 Sprecher 2

Und das ist etwas, woran sich Auditoren halten werden bei bei einem Audit.

00:08:59 Sprecher 2

Und das ist auch durchaus berechtigt und das ist absolut berechtigt.

00:09:03 Sprecher 2

Und da sind wir bei der inhaltlichen Begründung, warum das nicht reicht.

00:09:07 Sprecher 2

Und das ist das Argument, ist dasselbe wie bei traditioneller Softwareentwicklung.

00:09:12 Sprecher 2

Da wird ja auch nicht nur die komplette Software getestet, sondern es wird Testen durchgeführt auf verschiedenen Ebenen der Softwareentwicklung.

00:09:21 Sprecher 2

Es werden die einzelnen Units getestet, dann wird den Integrationstest bis hoch zum kompletten System werden immer größere Komponenten getestet und das macht man deshalb, weil Testen.

00:09:35 Sprecher 2

immer nur die Anwesenheit und nie die Abwesenheit von Fehlern zeigen kann.

00:09:40 Sprecher 2

Das heißt, es wird nie ein vollständiger Beweis sein für die Abwesenheit von Fehlern und deswegen macht man es auf vielen Ebenen, einfach um die Wahrscheinlichkeit zu erhöhen, dass man Fehler findet, weil man eben immer nur Teile der tatsächlichen Eingaben wird testen können.

00:09:59 Sprecher 2

Und das gilt für Machine Learning Modelle in ganz besonderem Maße und das wissen wir spätestens

00:10:06 Sprecher 2

seit die Anfälligkeit von Machine Learning Modellen für sogenannte Adversarial Attacks bekannt ist, bei denen es relativ leicht ist, durch durch kleine Unterschiede in den Eingabedaten zu falschen Vorhersagen zu kommen.

00:10:24 Sprecher 2

Und allein diese Adversarial Attacks zeigen, dass dass wir beim Testen immer nur einen kleinen Ausschnitt aus der Wirklichkeit abbilden können.

00:10:33 Sprecher 2

Und deswegen ist es wichtig,

00:10:35 Sprecher 2

dass wir das auf den verschiedensten Ebenen tun.

00:10:39 Sprecher 1

Jetzt könnte aber ein Hersteller vielleicht sagen, ja, die werden doch irgendwie schon tausendfach oder millionenfach eingesetzt, warum muss ich jetzt da noch mal ‚ne Validierung machen, kann ich da nicht irgendwie risk-based argumentieren und sagen, die Wahrscheinlichkeit, dass ich da noch was finde, ist so beliebig gering?

00:10:57 Sprecher 2

Ja, auch hier leider nein,

00:10:59 Sprecher 2

dass diese Bibliotheken häufig eingesetzt werden, ist durchaus ein wichtiger Baustein in der Gesamtbetrachtung, reicht aber alleine nicht.

00:11:08 Sprecher 2

Und weil du Risk-Based angesprochen hast, Christian, da schlägt dann der Unterschied zwischen 13 485 der Toolvalidierung und der 62 304, nämlich der Subvalidierung, zu.

00:11:22 Sprecher 2

Bei der Toolvalidierung darf man das durchaus tun, darf man riskbasiert, also risikobasiert argumentieren.

00:11:28 Sprecher 2

Bei der Sub-Validierung darf man das einfach nicht.

00:11:31 Sprecher 2

Da kommt das nicht im Konzept vor, dass man eine Risikoabwägung macht.

00:11:36 Sprecher 1

Außer, falls man natürlich über die Software-Sicherheitsklasse argumentiert, aber sonst steht im Kontext von Sub in der Tat nichts zum Thema Risk-Based.

00:11:44 Sprecher 2

Ja genau, genau.

00:11:45 Sprecher 2

Wenn es um die eigentliche Validierung geht, dann ist die einfach, dann ist die zu, dann ist die Sub zu spezifizieren und zu validieren.

00:11:51 Sprecher 2

Und das hat im Übrigen auch wieder nicht nur einen regulatorischen Hintergrund, sondern auch tatsächlich inhaltlichen Hintergrund.

00:11:56 Sprecher 2

Also, diese Regulatorien sind auch hier wieder berechtigt.

00:11:59 Sprecher 2

Der ein oder andere erinnert sich vielleicht noch an den sogenannten Heartbeat Bug aus dem Jahr 2012.

00:12:05 Sprecher 2

das war ein Fehler in einer Open in der Open SSL Implementierung und Open SSL ist eine Bibliothek, die von Webservern eingesetzt wird, die sehr, sehr weit verbreitet war, noch heute weit verbreitet ist und mindestens eine Bekanntheit und Verbreitungsgrad hat, wie die genannten Machine Learning Bibliotheken.

00:12:25 Sprecher 2

Und dieser Fehler, der hatte zum Ergebnis oder der hatte zur Folge, dass über eine halbe 1000000 Webserver angreifbar wurden und der wurde erst nach langer Zeit entdeckt,

00:12:35 Sprecher 2

Der ging zurück auf den Fehler eines Doktoranden, der ein neues Feature programmiert hat, und genau eines Mitglieds der Open SSL Community.

00:12:44 Sprecher 2

Das war ein Entwickler, der genau dieses Feature dann eben akzeptiert hat und in den Open SSL Code aufgenommen hat.

00:12:50 Sprecher 2

hat.

00:12:50 Sprecher 2

Manche Webserver sind im Übrigen bis heute noch für diesen Bug anfällig, was eben am Rande bemerkt auch ein gutes Argument ist für die Wichtigkeit von Post-Market-Surveyance.

00:13:02 Sprecher 2

Das heißt also, selbst in solchen extrem weit verbreiteten Bibliotheken kommt es vor, dass es Fehler gibt, die erst nach langer Zeit erkannt werden und die dann verheerende Folgen haben.

00:13:16 Sprecher 2

Und das bringt mich auf den nächsten Teilaspekt, nämlich, wenn Sie sich beispielsweise TensorFlow angucken, dann gibt es da derzeit ungefähr 4000 sogenannte Open Issues, die innerhalb der letzten 5 Jahre entstanden sind.

00:13:31 Sprecher 2

Also, die Bibliothek gibt es seit circa 5 Jahren und in der Zeit sind ungefähr 4000 Open Issues verzeichnet worden.

00:13:39 Sprecher 2

Das sind alles Probleme, die bis heute nicht gelöst sind.

00:13:44 Sprecher 2

Und die meisten dieser Probleme sind sind keine ernsthafte Bedrohung.

00:13:50 Sprecher 2

Viele haben was damit zu tun, dass TensorFlow unter bestimmten Konfigurationsvoraussetzungen in besonders seltenen Fällen merkwürdiges Verhalten zeigt.

00:14:01 Sprecher 2

Aber das zeigt eben, dass trotz dieses vielfältigen Einsatzes auch eine Bibliothek mit TensorFlow bis heute weit davon entfernt ist, komplett fehlerfrei zu sein.

00:14:13 Sprecher 1

Damit haben wir also weder die Ausrede, dass die Bibliotheken indirekt mit dem Modell mit validiert würden, noch haben wir die Ausrede, dass sie sagen können, ja, die sind ja millionenfach im Einsatz und alles ist gut, wir müssen offensichtlich dann doch die Ärmel hoch krempeln und wir müssen validieren.

00:14:31 Sprecher 1

Ja, jetzt kommen wir, glaub ich, zur interessanten und wesentlichen Frage und wie macht man denn das jetzt?

00:14:36 Sprecher 1

Also, wie validiert man eine Machine Learning Library?

00:14:41 Sprecher 2

Genau, das ist tatsächlich die zentrale Frage und da komme ich gerne nochmal auf diese Unterscheidung zurück oder ich muss auch auf die Unterscheidung zurückkommen zwischen der Tool-Validierung nach ISO 13485 und der Subvalidierung nach IEC 62304.

00:14:58 Sprecher 2

Ich fange mal mit der Subvalidierung an.

00:15:00 Sprecher 2

Für die Subvalidierung muss gezeigt werden, dass die Inferenzfunktionalität der Bibliothek korrekt arbeitet.

00:15:09 Sprecher 2

Das heißt, man muss zeigen,

00:15:11 Sprecher 2

dass die Vorhersagen, die das Modell macht, seinen Gewichten entspricht.

00:15:16 Sprecher 2

Das hat nichts zu tun mit der Vorhersagequalität des Modells und das ist auch einer der Gründe, warum ich vorhin gesagt habe, dass diese Korrektheitsüberprüfung über die Modellevaluation hinausgeht.

00:15:28 Sprecher 2

Hier geht es darum zu zeigen, dass das Modell die Vorhersagen macht, die es gemäß seiner Gewichte tun müsste.

00:15:38 Sprecher 2

Und das können gute und es können schlechte Vorhersagen sein.

00:15:41 Sprecher 2

Das hängt eben davon ab, ob dieses Modell gut trainiert ist oder nicht gut trainiert ist.

00:15:45 Sprecher 2

Aber das kommt, das ist nicht die Betrachtung in diesem Kontext.

00:15:50 Sprecher 2

Und für diese Verifikation braucht man 3 Dinge: Man braucht erstmal eine Spezifikation des erwarteten Verhaltens.

00:15:57 Sprecher 2

Was müsste denn eigentlich vorhergesagt werden?

00:16:00 Sprecher 2

Das muss man spezifizieren.

00:16:02 Sprecher 2

Man braucht zweitens ein Testorakel, das heißt, man muss

00:16:07 Sprecher 2

zeigen können, wie dieses erwartete Verhalten eigentlich aussehen müsste.

00:16:12 Sprecher 2

Also, man braucht einen Vergleichswert zu dem, was man beobachtet und man braucht drittens geeignete Testdaten.

00:16:20 Sprecher 2

Das sind diese 3 Komponenten, die benötigt werden für die Subvalidierung.

00:16:25 Sprecher 2

Bei der Tool-Validierung ist es ein bisschen anders, wie wir vorhin schon mal kurz angesprochen haben.

00:16:31 Sprecher 2

Bei der Tool-Validierung darf und soll man auch risikobasiert vorgehen.

00:16:36 Sprecher 2

Das heißt, es ist nicht notwendig, die die vollständige Trainingsfunktionalität der Bibliothek zu validieren.

00:16:45 Sprecher 2

Das wäre auch gar nicht plausibel, das wäre auch gar nicht praktikabel, weil das Training den Großteil dieser Bibliotheken ausmacht.

00:16:53 Sprecher 2

Zur Toolvalidierung kann man nun tatsächlich geeignete Techniken der Modellevaluierung heranziehen, um zu zeigen,

00:17:03 Sprecher 2

dass das Ergebnis des Trainings den Trainingsdaten entspricht.

00:17:07 Sprecher 1

Also, wir haben 2 Validierungen, die wir jetzt eigentlich hier zu tun haben, nämlich wir müssen einmal schauen, dass die Bibliothek wirklich auch tatsächlich trainiert und das nachher, dass wenn das Modell trainiert ist, dass dann auch die Predict Funktion, vielleicht wenn man das jetzt mal auf die konzentriert, dann auch das Predicted entsprechend, wie das Modell eben trainiert worden ist.

00:17:27 Sprecher 1

Und was es so schwierig macht, ist eben, dass der eine Teil

00:17:31 Sprecher 1

letztlich von der 13485 reguliert wird und der andere Teil von der 62304.

00:17:37 Sprecher 1

Ich würd mal sagen, das wird einigen Auditoren die Schweißperlen auf die Stirn treiben und so sicher spannenden Diskussionen beim Audit führen, damit man sich immer klar ist, welche Regularie greift jetzt hier, weil ja diese Demarkationslehne mitten durch diese Bibliothek durchläuft.

00:17:55 Sprecher 1

Ich glaub, das ist wirklich ‚ne Besonderheit, die wir sonst noch in keinem Fall irgendwie so hatten.

00:18:01 Sprecher 1

Es hört sich aber schon ‚n bisschen nach einiger Arbeit an, die hier auf Hersteller zukommt und ich denk mal, dass ‚n Teil zumindest dieser Validierung jetzt ja relativ unabhängig von dem konkreten Medizinprodukt ist, wo das Modell zum Einsatz kommt.

00:18:17 Sprecher 1

Da stellt sich die Frage, muss es jeder noch mal neu machen, kann man sich irgendwie diese Aufwände teilen, kann man das vielleicht sogar schon fertig einkaufen, was wär da deine Empfehlung?

00:18:28 Sprecher 2

Ja, genau.

00:18:28 Sprecher 2

Ja, danke, danke für die Frage, Christiana.

00:18:30 Sprecher 2

Das ist tatsächlich so, dass diese Validierung auf diese 2 verschiedenen Arten Arbeit bedeutet.

00:18:39 Sprecher 2

Das Gute ist, dass das nicht jeder komplett neu machen muss oder das Rad neu erfinden muss, zumindest nicht.

00:18:45 Sprecher 2

Dann erst noch mal vielleicht eine schlechte Nachricht.

00:18:48 Sprecher 2

Es ist nicht so, dass man eine Machine Learning Bibliothek, wie beispielsweise TensorFlow, einmal fertig validieren würde.

00:18:56 Sprecher 2

und dann ist es erledigt für den Einsatz als Sub oder als Tool.

00:19:00 Sprecher 2

Das wäre, das wäre überhaupt nicht praktikabel, dafür sind diese Bibliotheken viel zu groß, viel zu umfangreich, viel zu, viel zu komplex.

00:19:08 Sprecher 2

Das würde letzten Endes bedeuten, die Testaktivitäten, die die Tensorflow Community zum Beispiel, die die durchführen, noch mal alles selbst zu machen und besser zu machen und das ist nicht realistisch.

00:19:21 Sprecher 2

Die gute Nachricht ist aber, dass man das auch nicht muss,

00:19:25 Sprecher 2

dass man für ein konkretes Produkt nicht die Bibliothek einmal generisch validieren muss, sondern man muss für ein konkretes Produkt nur zeigen, dass die Bibliothek für das konkrete Modell korrekt arbeitet.

00:19:40 Sprecher 2

Das heißt, man muss zeigen, dass dieses konkrete Modell richtig trainiert wurde und man muss zeigen, dass dieses konkrete Modell richtig vorhersagt.

00:19:49 Sprecher 1

Weil ich da ganz kurz einhaken darf, also das heißt zum Beispiel, es besteht jetzt keine Notwendigkeit, wenn man jetzt sich für ,n ganz speziellen Modelltyp, ne, für ,ne Architektur entscheidet.

00:20:00 Sprecher 1

Ich sag jetzt mal vielleicht irgendwie ein Convolutional Neural Network mit einer bestimmten Architektur, da muss man sich nicht um die X.

00:20:07 Sprecher 1

G.

00:20:07 Sprecher 1

Boost Variante kümmern, da muss man sich nicht um die normalen, möglicherweise nicht um alle Varianten neuronaler Netzwerke kümmern, sondern eben genau um die Architektur, die man hier gewählt hat.

00:20:19 Sprecher 2

ganz genau und sogar noch weiter.

00:20:20 Sprecher 2

Also man muss sich sogar noch nicht mal komplett um alle Modelle dieser Architektur kümmern, sondern nur um dieses eine konkrete Modell.

00:20:29 Sprecher 2

Das heißt, man macht den die Tests, also die hat beispielsweise Vergleichs Vergleiche mit den Oracle, die ich vorhin angesprochen habe, die macht man konkret für dieses eine Modell.

00:20:40 Sprecher 2

Das klingt jetzt auch wieder ein bisschen wie eine schlechte Nachricht, weil das ja sehr projekt oder produktspezifisch aussieht.

00:20:47 Sprecher 2

Und es ist in der Tat so, dass man das für jedes Modell und für jedes Projekt neu machen muss.

00:20:53 Sprecher 2

Aber die Art, wie man das tut, ist immer gleich.

00:20:57 Sprecher 2

Das heißt, man muss immer die die Spezifikation anfertigen des gewünschten Verhaltens.

00:21:04 Sprecher 2

Man braucht immer ein Testorakel und man braucht immer die Testdaten.

00:21:07 Sprecher 2

Und die Art, wie man diese 3 Komponenten erzeugt, folgt immer demselben Vorgehen.

00:21:15 Sprecher 2

und dafür haben wir eine Blaupause und Baustein entwickelt, die man wiederverwenden kann und die man an seine konkrete, an seine konkreten Modelle und damit an seine konkreten Produkte anpassen kann.

00:21:27 Sprecher 1

Wenn ich dich richtig verstehe, hast du jetzt ein Prozess hier oder Arbeitsanweisung entwickelt, wie man hier ganz konkret vorgeht.

00:21:36 Sprecher 2

Ja, es geht allerdings auch über den Prozess hinaus, sondern wir haben außerdem auch tatsächlich schon Code-Bausteine.

00:21:45 Sprecher 2

Also wir haben Bausteine, mit denen man Testorakel zusammenbauen kann, für beispielsweise für neuronale Netze.

00:21:55 Sprecher 2

Wir haben

00:21:57 Sprecher 2

sind Generatoren, mit denen man passende Testdaten erzeugen kann.

00:22:01 Sprecher 2

Wir haben beispielhafte Testvalidierungspläne, also textliche Dokumente, indem man beschreibt, wie diese Validierung aussieht, also die dann quasi diese Komponenten noch mal zusammenfasst und beschreibt.

00:22:14 Sprecher 2

Genau, also wir haben ein Konzept und darüber hinausgehend auch schon konkrete Bausteine für die Lösung.

00:22:22 Sprecher 1

Was würdest du jetzt herstellen

00:22:25 Sprecher 1

auch genauso konkret empfehlen, was die jetzt tun sollten, wenn sie Libraries bereits einsetzen oder planen, die einzusetzen.

00:22:33 Sprecher 2

Ja, also für die Subvalidierung eben, wie gesagt, sind eben diese 3 Komponenten zu erstellen.

00:22:40 Sprecher 2

Man braucht ‚ne Spezifikation des gewünschten Verhaltens, also ganz konkret der der Predict Funktion in den allermeisten Fällen.

00:22:49 Sprecher 2

Man braucht ein Testorrakel, das ist zu erstellen und man braucht passende Testdaten dazu.

00:22:55 Sprecher 2

Und diese Testdaten müssen dann sowohl das Modell als auch das Orakel auf eine möglichst breite Art quasi durchlaufen und ansteuern und und triggern.

00:23:07 Sprecher 2

Das ist das, was man für die Subvalidierung braucht, die Toolvalidierung, weil die risikobasiert ist, hängt es dann auch

00:23:16 Sprecher 2

stärker vom vom konkreten Produkt ab.

00:23:19 Sprecher 2

Bedeutet aber letztendlich immer, dass man mit Hilfe von Modell-Evaluations und Modell-Transparenzmaßnahmen zeigt, dass das Modell entsprechender Trainingsdaten richtig trainiert wurde.

00:23:36 Sprecher 1

Weil das ja sich schon noch ein bisschen nach Arbeit anhört, wäre es für dich O.

00:23:40 Sprecher 1

K., wenn wir deine Kontaktdaten da veröffentlichen, dass ich

00:23:43 Sprecher 1

die Hersteller auch an dich wenden können, um da vielleicht zu lernen oder vielleicht sogar einige Dinge wiederverwenden.

00:23:50 Sprecher 2

Ja, klar, sehr gerne.

00:23:52 Sprecher 2

Du, du hast natürlich recht, Christian, das ist Arbeit, das ist tatsächlich nicht zu unterschätzende Arbeit.

00:23:57 Sprecher 2

Aber ich möcht noch mal betonen, dass das einfach auch ein wichtiger Bestandteil ist des Gesamtprodukts.

00:24:04 Sprecher 2

Also diese, diese Bibliotheken sind der wesentliche Kern,

00:24:09 Sprecher 2

Des Modells und müssen deswegen auch sorgfältig validiert werden.

00:24:14 Sprecher 2

Und wie gesagt, ein großer Teil der Arbeit ist ja auch schon gemacht.

00:24:19 Sprecher 2

Also, wir haben eben diese Beispiele, die gut wiederverwendbar sind, das heißt.

00:24:24 Sprecher 2

diese Bibliotheken, die Machine Learning Bibliotheken, sind nicht nur ein großartiger Quell von Wiederverwendung auf der auf der funktionalen Ebene, sondern auch auf der Validierungsebene, weil eben diese Validierung immer wieder sehr ähnlich aussieht.

00:24:40 Sprecher 2

Und natürlich helfen wir gerne, unsere Lösung anzupassen und anzuwenden für andere Produkte.

00:24:48 Sprecher 1

Ja, dann wär mein Empfehlung, dass Sie vielleicht mal auch auf die Webseite gehen.

00:24:52 Sprecher 1

Wir haben ja schon ,ne

00:24:53 Sprecher 1

Beitrag geschrieben, wo da ganz vieles beschrieben ist.

00:24:57 Sprecher 1

Den verlinken wir auch noch mal unten in der Beschreibung.

00:25:00 Sprecher 1

Und wenn Sie da ein Shortcut suchen und die Arbeit nicht alle machen wollen, dann wäre meine Empfehlung, dass Sie einfach meinem Professor Hase Kontakt aufnehmen.

00:25:09 Sprecher 1

Oliver, ich danke da von ganzem Herzen für die wirklich sehr wertvollen Insights.

00:25:14 Sprecher 1

Danke auch für deinen deine Einschätzung meines sympathischen badischen Einschlags und in diesem Sinne bis am nächsten Züßdick.

00:25:22 Sprecher 1

Danke dir.

00:25:23 Sprecher 2

Ja, vielen Dank, lieber Christian.

00:25:25 Sprecher 2

War mir ein Vergnügen.