Bachelorarbeit am Institut für Informatik der Freien Universität Berlin

Human-Centered Computing (HCC)

Konzept und Implementierung einer visuellen Methode zur Verbesserung der Interpretierbarkeit der automatisierten Qualitatsbewertung mit ORES in Wikidata

Sajeera Gnanasegaram

Betreuerin und Erstgutachterin: Prof. Dr. C. Müller-Birn

Zweitgutachter: Prof. Dr. M. Margraf

Berlin, 09.09.2020

b

Eidesstattliche Erklärung

Ich versichere hiermit an Eides Statt, dass diese Arbeit von niemand anderem als meiner Person verfasst worden ist. Alle verwendeten Hilfsmittel wie Berichte, Bücher, Internetseiten oder ähnliches sind im Literaturverzeichnis angegeben, Zitate aus fremden Arbeiten sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

Berlin, den 9. September 2020

Sajeera Gnanasegaram

Zusammenfassung

Der Kontext dieser Arbeit ist Wikimedias ORES, ein ML-gestützter Dienst, der Bewertungsinformationen wie die Qualität eines Beitrags (engl. *Item*) für Wiki-Beiträge bereitstellt. Dieser Dienst wird genutzt, da in der stetig wachsenden Gemeinschaft jeder ein Item erstellt und editieren kann. Für die Redakteure ist es aber nicht möglich in kürzester Zeit die Items manuell auf ihre Qualität zu überprüfen und zu bearbeiten. Über ein Gadget wird für das ausgewählte Item die Qualität mitgeteilt. Das Gadget ist als visuelles Interface zu verstehen.

Das Problem dabei ist, dass das Gadget nicht aussagekräftig genug ist. Bisher ist es sehr schwer zu verstehen, was dessen Ausgabe bedeutet und wieso sie getroffen wurde. Einer der Gründe für dieses ungelöste Problem der Interpretierbarkeit ist, dass die Interpretierbarkeit ein sehr subjektives Konzept ist und somit schwer zu formalisieren. Je nach Kontext können verschiedene Arten von Erklärungen nützlich sein.

Eine Erklärung (engl. *Explanation*) ist nicht nur ein Produkt, sondern auch ein Prozess, der eine kognitive Dimension und eine soziale Dimension beinhaltet. Die verbesserte Version des Gadgets mit *Post hoc* Explanations soll jeden Anwender (Experten und Nichtexperten) dabei helfen, die Ausgabe und die Gründe für die Entscheidung Qualitätsbewertung zu verstehen, aber auch welche Faktoren die Entscheidung beeinflusst haben.

Mit Hilfe von *User Centred Explanations* wird ein Design für das Gadget entwickelt. Das System wird beim Start einmal die ORES-API abfragen, die Ergebnisse speichern und in Form eines Explanation Interfaces (Gadgets) präsentieren. Als Designansatz werden Explanation Interfaces aus dem Bereich *Recommender Systemes* mit *Human Friendly Explanations* für eine bessere Interpretierbarkeit kombiniert. Es ist nicht nur wichtig zu verstehen, welche Information wir über ein System bekommen, sondern eher warum diese Entscheidung getroffen wurde.

Erst erstellte ich aus dem von mir erstellen Konzept mehrere Low-Fidelity-Prototypen und entwickelte dann einen High-Fidelity-Prototyp. Mit drei Mitarbeitern der HCC-Forschungsgruppe der Freien Universität Berlin [1] führte ich einen Usability Test über Cisco Webex durch, um meinen High-Fidelity-Prototypen zu evaluieren.

Ich kam zu der Erkenntnis, dass das von mir entwickelte Explanation Interface von den Probanden verstanden wurde, sie waren in der Lage ORES nachträglich zu interpretieren und konnten mir auch sagen, warum ORES sich für die

[1] `https://www.mi.fu-berlin.de/en/inf/groups/hcc/members/researchers/index.html`

jeweilige Qualitätsklasse entschieden hat.
Das Gadget zeigt noch kleine Schwächen in der Benutzerfreundlichkeit, welche aber leicht zu beheben sind.
Insgesamt war der Designansatz aus dem Bereich der Recommender Systeme mit Human Friendly Explanations für eine bessere Interpretierbarkeit zu erreichen für mich eine gute Lösung, weil der Usability Test gezeigt hat, dass so das „Warum“ hinter der Systemausgabe vermittelt wurde. So wirkt ORES, der ML-gestützte Dienst, transparent und fair und die Nutzer*innen entwickeln ein Vertrauen gegenüber dem System.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Algorithmen bewerten Menschen. Oder bewertet der Mensch diejenigen, die den Algorithmus schreiben? Von welchen Algorithmen sprechen wir und in welcher Beziehung stehen Mensch und Algorithmen zueinander? Immer mehr gewinnen diese Fragen an Bedeutung. Mittlerweile haben wir ein Zeitalter erreicht, in welchem es nicht nur die Menschen sind, die uns bewerten oder beurteilen, sondern auch der Computer ist dazu in der Lage.

Beginnen wir mit der Frage *Was ist ein Algorithmus?*

Einfach gesagt: Ein Algorithmus ist eine endliche, diskrete Reihe von Anweisungen, die eine Eingabe empfängt und eine Ausgabe erzeugt. Heutzutage tragen Algorithmen zu Entscheidungen bei, die Millionen von Menschen betreffen, die auch negative Konsequenzen mit sich bringen [ER18].

> [..] Algorithms now contribute to decisions affecting millions of people, related to employment, housing, healthcare, education, and criminal justice, among many others negative consequences can result from decisions that depend upon algorithms, such as economic or social disadvantaging of already marginalized populations. [..]
>
> Emilee Rader et al., 2018, Seite 1 [ER18]

1.1 Motivation

Ein Algrithmus ist eine Reihe von Regeln, die eine Maschine befolgt, um ein bestimmtes Ziel zu erreichen. Algorithmen sind auch in maschinellen Lernsystemen zu finden. Machine Leaning, abgekürzt ML, ist ein Teilbereich der künstlichen Intelligenz. ML basierte Systeme sind in der Lage automatisch Daten zu erlernen und sich zu verbessern. Damit das maschinelle Lernen funktioniert und die Software Entscheidungen treffen kann, muss ein Mensch den Algorithmus trainieren. Durch das Bereitstellen von Trainings- bzw. Beispieldaten, kann der Algorithmus ein Muster erkennen und so aus Daten lernen (siehe Abbildung 1.1). Dieser Prozess wird als Modelltraining bezeichnet [1].

[1] `https://de.wikipedia.org/wiki/Maschinelles_Lernen`, besucht am 08.09.2020

[2]

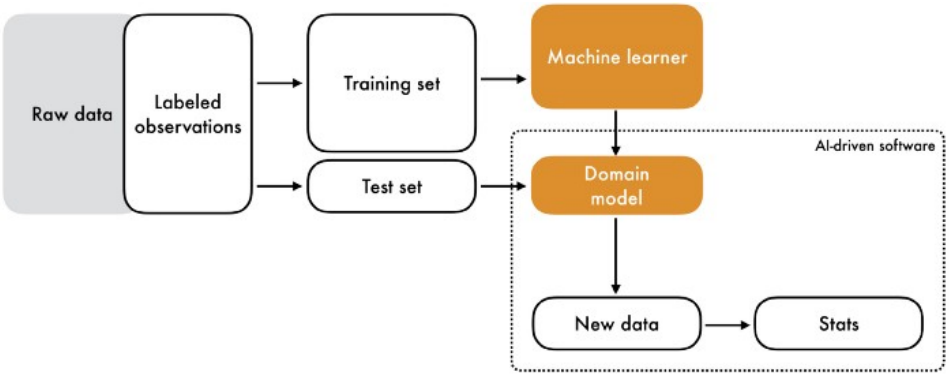


Abbildung 1.1: ML Pipeline

[2] https://medium.com/@clmb/what-is-beyond-the-human-centered-design-of-ai-driven-systems-10f90beb besucht am 08.09.2020

Sollten passende Trainingsdaten zur Verfügung stehen, kann Maschinelles Lernen z.B. leisten [3]:

- Vorhersage von Werten Basis der analysierten Daten.
- Berechnung von Wahrscheinlichkeiten für bestimmte Ereignisse.
- Erkennen von Gruppen und Clustern in einem Datensatz.
- Erkennen von Zusammenhängen in Sequenzen.
- Reduktion von Dimensionen ohne großen Informationsverlust.

Ein Modell des maschinellen Lernens ist das gelernte Programm, das Eingaben auf Vorhersagen abbildet. Dabei kann es sich um einen Satz von Gewichten für ein lineares Modell oder für ein neuronales Netz handeln. Andere Bezeichnungen für das Wort Modell sind *Prädiktor* oder - je nach Aufgabe - *Klassifikator* oder *Regressionsmodell*.

Heutzutage ist es durchaus üblich geworden, dass man moderne maschinelle Lernsysteme als „Black Boxes“ bezeichnet. Ein Black-Box-Modell ist ein System, das seine internen Mechanismen nicht offenbart. Im maschinellen Lernen beschreiben „Black Box“ Modelle, die durch die Betrachtung ihrer Parameter nicht verstanden werden können. Das bedeutet, dass die interne Logik und die inneren Abläufe dieser Modelle dem Benutzer verborgen bleiben, was ein ernsthafter Nachteil ist, da es einen Menschen, einen Experten oder einen Nichtexperten daran hindert, die Argumentation des Systems und die Art und Weise, wie bestimmte Entscheidungen getroffen werden, zu überprüfen, zu interpretieren und zu verstehen [DLH19].
Das Gegenteil einer Black Box wird als White Box bezeichnet.

Ein Datensatz ist eine Tabelle mit den Daten, aus denen die Maschine lernt. Dabei enthält der Datensatz die Eigenschaften und das Ziel der Vorhersage. Wenn der Datensatz zur Induktion eines Modells verwendet wird, wird er als Trainingsdaten bezeichnet. Eine Instanz ist eine Zeile in einem Datensatz und besteht aus Merkmalen mit ihren Ausprägungen und einem Label. Merkmale sind die für die Vorhersage oder Klassifizierung verwendeten Eingaben. Ein Merkmal ist eine Spalte in einem Datensatz.
Eine Aufgabe des maschinellen Lernens ist die Kombination eines Datensatzes mit Merkmalen und einem Ziel. Je nach Art des Ziels kann die Aufgabe z.B. eine Klassifikation oder eine Regression sein. Die Vorhersage ist das, was das Modell des maschinellen Lernens errät, was der Zielwert auf der Grundlage der gegebenen Merkmale sein sollte [ER18].

[3] `https://datasolut.com/was-ist-machine-learning/#machine-learning-arten`, besucht am 08.09.2020

1.2 Problematik und Zielstellung

Es hat uns ein Zeitalter von einer massiven Informationsflut und Social Media Konsum in Zusammenhang rasanter technologien Entwicklungen erreicht. Sich einen allumfassenden Überblick zu verschaffen, ist für viele fast unmöglich. Viele stoßen deshalb auf Widerstand. Eine Angst vor Kontrollverluste könnte entstehen, sodass es nicht immer einfach ist, diesen Modellen zu vertrauen.
Viele ML Systeme sind heutzutage Black-Box-Modelle, die weder die Funktionsweise des Prozesses transparent machen, noch zusätzliche Informationen zur Verfügung stellen. Das System funktioniert wie ein computergestütztes Orakel, die Ratschläge erteilen, aber nicht hinterfragt werden können. Einem Benutzer werden keine Indikatoren zur Verfügung gestellt, anhand derer erfragt werden kann, wann er einer Wertung trauen kann und wann ein Zweifel bestehen muss [JLHR].

Wenn ein Modell des maschinellen Lernens gut genug funktioniert und eine akzeptable Vorhersageleistung aufweist, stellt sich die Frage, warum wir nicht dem Modell vertrauen und ignorieren, warum es eine bestimmte Entscheidung getroffen hat.
Menschen möchten erfahren, warum eine Entscheidung getroffen wurde und Erklärungen erhalten. Dadurch entsteht mehr Vertrauen zum System und das Modell wirkt transparenter.

Bei einigen ML basierten Modellen kann abgelesen werden, welchen Einfluss ein Merkmal auf die Vorhersage hat. Das gilt unter anderem für Regressionsmodelle. Sollten die Anzahl der Merkmale, die in das Modell einfließen gering sein, können die Modelle visualisiert werden und dadurch Kausalitäten erkannt werden.

Es existieren aber auch Maschine-Learning-Modelle, die aber sehr komplex sind, um die Einflussfaktoren abzulesen. Die Leistung und Ergebnisse dieser Modelle sind in der Regel am besten, da die Datenmenge groß ist. Wie kann man aber einem Modell trauen, welches man nicht versteht? Es gibt die Möglichkeit das Output der Modelle nach der Trainingsphase zu beurteilen bzw. zu interpretieren [Pik19].
Interpretierbares maschinelles Lernen bezieht sich auf Methoden und Modelle, die das Verhalten und die Vorhersagen von Systemen des maschinellen Lernens für den Menschen verständlich machen.

Miller, 2017 definiert Interpretierbarkeit wie folgt:

> [..] Interpretability is the degree to which a human can understand the cause of a decision [..] [Mil17]

Interpretierbarkeit, die manchmal als Synonym für Erklärbarkeit verwendet wird, wird von Doshi und Kim wie folgt definiert

> [..] Interpret means to explain or present in understandable terms. In the context of ML systems, we define interpretability as the

> ability to explain or to present in understandable terms to a human. [..] [Kri19]

Je höher die Interpretierbarkeit eines maschinellen Lernmodells ist, desto leichter ist es für jemanden zu verstehen, warum bestimmte Entscheidungen oder Vorhersagen getroffen wurden. Ein Modell ist besser interpretierbar als ein anderes Modell, wenn seine Entscheidungen für eine Person leichter zu verstehen sind als die Entscheidungen des anderen Modells.

Dennoch sollte man bedenken, dass, obwohl die Relevanz und Bedeutung der Interpretierbarkeit klar erklärt wurde, nicht alle ML-Systeme Interpretierbarkeit erfordern, da es Situationen gibt, in denen es ausreicht, eine hohe Vorhersageleistung zu erbringen, ohne dass Entscheidungen erklärt werden müssen. Aus diesem Grund gibt es laut Doshi-Velez und Kim zwei Arten von Situationen, in denen die Interpretierbarkeit und damit die Erklärungen nicht notwendig sind [Kim17]:

1. Wenn es keine signifikanten Auswirkungen oder schwerwiegende Folgen für falsche Ergebnisse gibt.

2. Wenn das Problem so gut untersucht und in realen Anwendungen validiert ist, dass wir den Entscheidungen des Systems vertrauen, auch wenn das System nicht perfekt ist.

Die erstgenannte Situation bezieht sich auf Systeme mit geringem Risiko, wie z.B. Produktempfehlungen und Werbesysteme, bei denen ein Fehler keine schwerwiegenden oder sogar tödlichen Folgen hat.

Letzteres bezieht sich auf gut erforschte Systeme, die schon seit einiger Zeit eingesetzt werden, wie Postleitzahlensortierung und Flugzeugkollisionssysteme, die ihren Output ohne menschliches Eingreifen berechnen.

Wie zu Beginn des Abschnitts festgestellt wurde, ergibt sich die Notwendigkeit der Interpretierbarkeit aus einer Unvollständigkeit der Problemformalisierung, was bedeutet, dass es für bestimmte Probleme oder Vorhersageaufgaben nicht ausreicht, die Vorhersage (das „Was“) zu erhalten. Das Modell muss auch erklären, wie es zu der Vorhersage gekommen ist (das „Warum“), denn eine korrekte Vorhersage löst das ursprüngliche Problem nur teilweise. Darüber hinaus kann diese Unvollständigkeit in der Problemspezifikation in verschiedenen Szenarien gezeigt werden, von denen einige die folgenden umfassen[Kim17]:

- Sicherheit - denn das System ist nie vollständig testbar, da man keine vollständige Liste von Szenarien erstellen kann, in denen das System versagen könnte.

- Ethik - weil die menschliche Vorstellung von z.B. Fairness zu abstrakt sein kann, um sie vollständig in das System einzufügen.

- Nicht übereinstimmende Ziele - weil der Algorithmus möglicherweise ein unvollständiges Ziel optimiert, d.h. eine stellvertretende Definition des tatsächlichen Endziels.

Ziel ist es, die Vorhersagen eines maschinellen Lernmodells zu erklären. Um dies zu erreichen, stützen wir uns auf einige Explanationsmethoden, die ein Algorithmus sind, die Erklärungen (engl. *Explanations*) generiert. Explanations werden dafür genutzt die Merkmalswerte einer Instanz mit ihrer Modellvorhersage einem Menschen verständlich zu erklären.

Die vorliegende Arbeit ist folgendermaßen aufgebaut:

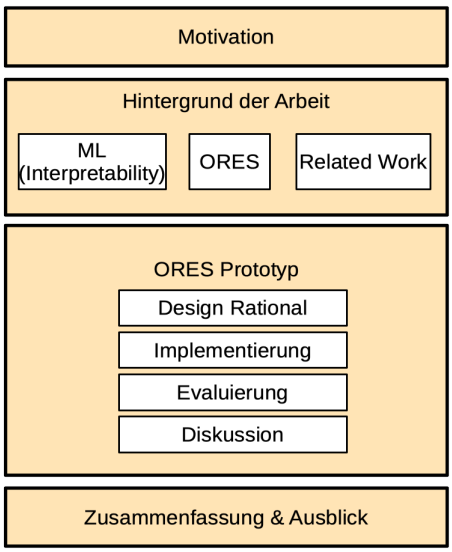


Abbildung 1.2: Aufbau der Arbeit

2 Hintergrund der Arbeit - Theoretische Grundlagen

2.1 Wikidata und automatisierte Qualitätsbewertungen

Wikidata ist eine Datenbank auf die jeder zugreifen kann. Zu beachten ist, dass die Daten von jedem aus der Wikidata-Community geschrieben werden können. Es ist ein offenes System, das viele Vorteile bietet, aber auch den Nachteil, dass die Richtigkeit von Daten nicht sofort überprüft wird. Es ist nicht möglich, dass Menschen alle neu hinzugefügten Einträge nach ihrer Qualität bewerten können. Dafür entstehen in kürzester Zeit zu viele neue Einträge. Beispielsweise entstehen auf der englischen Wikipedia Seite am Tag 160 000 neue Einträge, die sofort veröffentlich werden. Jede schädliche oder anstößige Bearbeitung gefährdet die Glaubwürdigkeit der Community und ihres Produkts, sodass alle Bearbeitungen so schnell wie möglich überprüft werden müssen.

Von den Wikidata Redakteuren wurde ein Bewertungsschema zusammengestellt, das die Qualität der Items anhand von unterschiedlichen Merkmalen klassifiziert wird[1]. Insgesamt existieren fünf unterschiedliche Klassen (A, B, C, D, E). Folgende Eigenschaften werden hierbei auf Ihre Qualität überprüft:

- Vollständigkeit von relevanten Aussagen (Statements)
- Quellen (References)
- Übersetzungen (Translation)
- Aliases
- Sitelinks
- Medien (Images)

A entspricht hierbei der Klasse mit der höchsten Qualität und E steht für die schwächste Qualitätsklasse.

[1] `https://www.wikidata.org/wiki/Wikidata:Item_quality`, besucht am 08.09.2020

Class	Criteria	Readers experience
A	Items containing all relevant statements, with solid references, and complete translations, aliases, sitelinks, and a high quality image.	All available information is recorded with reliable references.
B	Items containing all of the most important statements, with good references, translations, aliases, sitelinks, and an image.	All of basic information and some extended information with references.
C	Items containing most critical statements, with some references, translations, aliases, and sitelinks.	Most of the basic information that you'd expect is available. May not be well referenced or complete.
D	Items with some basic statements, but lacking in references, translations, and aliases.	The statements need to provide enough information to easily identify the item.
E	All items that do not match grade "D" criteria.	

Tabelle 2.1: ORES Bewertungsschema

https://www.wikidata.org/wiki/Wikidata:Item_quality, besucht am 08.09.2020

2.2 ORES

ORES *Objective Revision Evaluation Service* ist ein Webservice, der maschinelles Lernen als Dienstleistung für Wikimedia-Projekte, wie Wikipedia und Wikidata, zur Verfügung stellt [2]. Das System wurde entwickelt, um menschliche Redakteure bei der Durchführung kritischer Wiki-Arbeiten zu unterstützen und ihre Produktivität zu erhöhen, indem Aufgaben wie die Erkennung von Vandalismus und die Entfernung von böswilligen Änderungen automatisiert werden. ORES wird von der Wikimedia Scoring Plattform[3] entwickelt, einem Team, das sich auf den Aufbau transparenter, prüfbarer, offener und ethischer künstlicher Intelligenz (KI) zur Unterstützung menschlicher Entscheidungen spezialisiert hat.

ORES entkoppelt drei Aktivitäten, die über einen längeren Zeitraum hinweg durchgeführt werden: Auswahl oder Aufbereitung von Trainingsdaten, Erstellung von Modellen zur Unterstützung von Vorhersagen und die Entwicklung von Schnittstellen oder automatisierten Agenten, die auf diese Vorhersagen reagieren [Lip17].

[2] `https://www.mediawiki.org/wiki/ORES`, besucht am 08.09.2020
[3] `mediawiki.org/wiki/Wikimedia_Scoring_Platform_team`, besucht am 08.09.2020

Modellbauer entwerfen einen Designprozess, um Scoring Modelle mit Trainingsdaten zu trainieren (siehe Abbildung 2.1). ORES hostet Scoringmodelle und stellt sie Forschern und Entwicklern zur Verfügung.

[4]

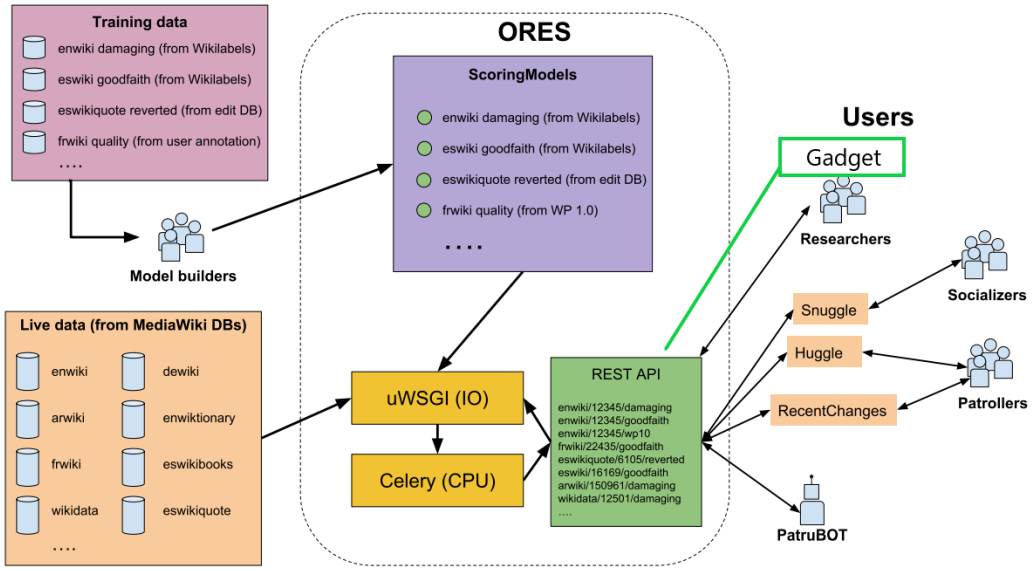


Abbildung 2.1: ORES - konzeptioneller Überblick

2.2.1 Quality Prediction

Im Kern besteht ORES aus einer Sammlung von Klassifikation von Maschinenbelegungsmodellen und einer API. Diese Modelle wurden von einer Vielzahl von Modellbauern entworfen und konstruiert unter Verwendung verschiedener Quellen von Trainingsdaten.
Um diese Modelle den Benutzer*innen zur Verfügung zu stellen, implementiert ORES einen einfachen Container Service, wobei der „Container“, der als Scoring Modell bezeichnet wird, ein vollständig trainiertes und getestetes Vorhersagemodell ist.
Alle Scoring Modelle enthalten Metadaten darüber, wann das Modell trainiert und getestet wurde und welche Merkmale für eine Vorhersage notwendig sind. Die Vorhersagen haben die Form eines JSON-Dokuments. Der ORES Dienst bietet Zugang zum Scoring Model über eine RESTful HTTP Schnittstelle.

[4] `https://pdfs.semanticscholar.org/3cef/57c5ae8a4d569c9f5231dc77255f97280fe0.pdf`, besucht am 08.09.2020

JSON am Beispiel des Wikidata Items Sri Lanka [5]:


```
{
  "wikidatawiki": {
    "models": {
      "itemquality": {
        "version": "0.4.0"
      }
    },
    "scores": {
      "1116542220": {
        "itemquality": {
          "score": {
            "prediction": "A",
            "probability": {
              "A": 0.9812577177162157,
              "B": 0.010730259309006904,
              "C": 0.006663974701529073,
              "D": 0.0007423983195287161,
              "E": 0.0006056499537198
            }
          }
        }
      }
    }
  }
}
```


ORES verwendet für die Berechnung den Gradient Boost Algorithmus [Hal17]. Dabei handelt es sich um einen überwachten Lernalgorithmus, der auf Grundlagen von Hypothesen möglichst zielsichere Vorhersagen trifft. Die Methode richtet sich also nach einer festgelegten zu lernenden Aufgabe, deren Ergebnisse bekannt sind. Die Ergebnisse des Lernprozesses können mit den bekannten und richtige Ergebnissen verglichen und somit „überwacht“ werden.

Das Ziel von Wikidata Redakteuren ist es, aus schädlichen und nicht schädlichen Einträgen (engl. *Edits*) die schädlichen Einträgen zurückzusetzen und aus den Revisionen (engl. *Revisions*) von nicht schädlichen Einträgen eine Wiki, wie Wikidata, mit hoher Item-Qualität zu entwickeln.

Ein automatisiertes Scoring-System, wie ORES kann diesen Arbeitsaufwand drastisch reduzieren. Zum Beispiel gibt ORES eine Punktzahl für eine Revision, welche die Wahrscheinlichkeit angibt, dass es sich um Vandalismus handelt. Über diese einfache Webschnittstelle macht es ORES einfach, eine leistungsstarke künstliche Intelligenz zu nutzen, die darauf trainiert ist, schädliche Änderungen zu erkennen [SP12].

[5] `https://ores.wikimedia.org/scores/wikidatawiki/?models=itemquality&revids=1272528577`, besucht am 08.09.2020

Zudem trägt ORES dazu bei, Informationen über den aktuellen Stand der Itemqualität zu ermitteln und vorhersagen über das Item zu treffen. ORES bietet eine einfache Benutzerschnittstelle für den Erhalt von Bewertungen [6], [7]. Die Vorhersage (engl. *Prediction*), teilt uns die Klasse mit der höchsten Wahrscheinlichkeitsverteilung mit.

Die gewichtete Summe gibt den Schwellenwert, d.h. eine Tendenz zur vorherigen bzw. nachfolgenden Klasse, an. Dafür wird angenommen, dass die von den Wikidata-Redakteuren entwickelte Qualitätsskala für Artikel ungefähr kardinal und gleichmäßig verteilt ist.

Folgende Formel wird für die Messung verwendet [Hal17]:

$$\text{weightedsum}=\sum_{c \in C} I(c) * P(c)$$

Um zur Messung der gewichteten Summe zu gelangen, wird die Vorhersagewahrscheinlichkeit für jede Klasse mit einer Aufzählung geordneter Klassen, die bei (1) für E beginnt und bei (5) für A endet, multipliziert und dann addiert.

Das obenstehende JSON-Dokument zeigt die gewichtete Summe 4,97 an.

Das Ergebnis, das über das Gadget mitgeteilt wird, bedeutet somit zu welcher Klasse das Item tendiert. Wenn das Item mit folgender Revision ID „649884“ betrachtet wird und für die Klasse D die gewichteten Summe 2.01 berechnet wurde, bedeutet es, dass der Wert eher zur Klasse C tendiert als zu E.

Erklären kann sich das mit folgender Überlegung: Wenn jede Klasse einzeln betrachted wird und nur eine Klasse die Wahrscheinlichkeit 1 erhält, während sie für die restlichen 0 beträgt, entsteht der maximale Wert für diese Klasse.

- Klasse E (1*1) = 1
- Klasse D (2*1) = 2
- Klasse C (3*1) = 3
- Klasse B (4*1) = 4
- Klasse A (5*1) = 5

[6] `https://ores-staging.wmflabs.org/ui/`, besucht am 08.09.2020
[7] `https://ores-staging.wmflabs.org/`, besucht am 08.09.2020

2.3 Gadget

Die Vorhersage über die Itemqualität von ORES wird in Form eines Gadgets (siehe Abbildung 2.2) mitgeteilt.

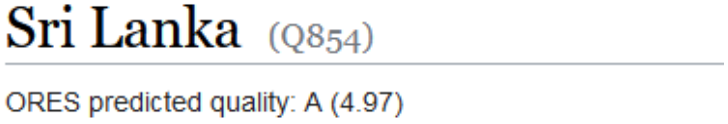


Abbildung 2.2: Gadget
https://www.wikidata.org/wiki/Q854, besucht am 08.09.2020

Die Klasse mit der höchsten Wahrscheinlichkeitsverteilung wird als verifizierte Klasse zurückgegeben, sowie die gewichtete Summe rechts daneben.

In der Abbildung 2.2 wurde das Item der Klasse A zugewiesen. Das Item beinhaltet somit die wichtigsten Aussagen mit guten Referenzen, Übersetzungen, Aliasnamen, Sitelinks und einem Bild beinhalten.
Wofür steht aber die weighted-sum, auf deutsch „gewichtete Summe“, rechts neben der Klasse?
Das Gadget deutet auch nicht darauf hin, basierend auf welcher Grundlage die Vorhersage für die Qualitätsklasse getroffen wurde und was die gewichtete Summe neben der Klasse zu bedeuten hat.

2.3.1 Motivation für das Gadget

Wenn die Funktionsweise von ML-Systemen den Menschen verborgen bleiben, dann kann der Mensch seine Entscheidungen nicht selbst treffen und kann zudem vom System negativ beeinflusst werden. Größere Tranzparenz ermöglicht den Nutzer*innen, ein System zu hinterfragen, aber auch zu kritisieren. Ein angemessenes Vertrauen wird entwickelt, statt einem System blind zu vertrauen. In Form eines Explanation Interfaces soll den Nutzer*innen eine Erklärung dafür gegeben werden, wie die Qualitätsbewertung des eingetragenen Items zustande kam. Die Nutzer*innen sollen mit Hilfe der Visualisierung ebenso erkennen, welche Merkmale Einfluss auf die Entscheidung genommen haben und was der Wert neben der Klassifizierung bedeutet und im besten Fall auch wie er zustande kam. Die Nutzer*innen sollten somit in der Lage sein, nachträglich das System zu interpretieren und zu verstehen.
Wie sollte aber so ein Gadget gestaltet sein? Bevor ein visuelles Explanation Interface erstellt werden kann, muss die Ursache der Problematik (Black-Box-Modell) in einzelne Teilmengen dieser Ursache unterteilt und überlegt werden,

wie diese Probleme gelöst werden können. Das Hauptziel ist, dass der Erklärende genügend Informationen vom Erklärer enthält.

2.4 Interpretability

Wieso kann einem Modell des maschinellen Lernens nicht blind vertraut werden, sobald das Modell eine bestimmte Entscheidung getroffen hat? Wir erwarten nicht, dass unsere Mitmenschen ihren Denkprozess immer erklären. Wir vertrauen ihnen intuitiv und diese Art von Vertrauen haben wir bei Maschinen nicht.

Ein Machine-Learning-Prozess besteht aus mehreren Schritten [8]:

1. Definition und Ziele (Welche Technologie soll optimiert werden)
2. Daten vorbereiten (Daten sammeln, transformieren und nach Merkmalen extrahieren)
3. Lernphase (Muster werden erkannt und Wahrscheinlichkeiten oder Werte vorhergesagt)
4. Interpretation (Ergebnisse werden ausgewartet, um zu verstehen, was im Algorithmus passiert
5. Nutzung in der Praxis (Modell wird in Betrieb genommen und auf unbekannte Daten übernommen)

Der Fokus dieser Arbeit liegt im 4. Schritt - der Interpretation (engl. *Interpretability*). Sie ist ein wichtiger Schritt, um auch Vertrauen und Akzeptanz für maschinelles Lernen zu schaffen. Der Mensch möchte und sollte verstehen, was in dem Algorithmus passiert.

In vielen Fällen kann den Nutzer*innen die Kenntnis des „Warums“ dabei helfen, mehr über das Problem, die Daten und den Grund für das Scheitern eines Modells zu erfahren. Einige Modelle sind möglicherweise nicht erklärungsbedürftig, weil sie in einer Umgebung mit geringem Risiko eingesetzt werden, wodurch ein Fehler keine schwerwiegende Folgen hat.

2.4.1 Taxonomy von Interpretability Methoden

Eine Taxonomie ist ein einheitliches Verfahren oder Modell, das dazu dient, Objekte nach bestimmten Kriterien zu klassifizieren [9].

Methoden zur Interpretierbarkeit des maschinellen Lernens können auch nach verschiedenen Kriterien klassifiziert werden[10].

[8] `https://datasolut.com/was-ist-machine-learning/#machine-learning-arten`, besucht am 08.09.2020

[9] `https://de.wikipedia.org/wiki/Taxonomie`, besucht am 08.09.2020

[10] `https://christophm.github.io/interpretable-ml-book/taxonomy-of-interpretability-methods.html`, besucht am 08.09.2020

Intrinsisch oder Post-hoc

Dieses Kriterium unterscheidet, ob die Interpretierbarkeit durch eine Beschränkung der Komplexität des Modells des maschinellen Lernens (intrinsisch) oder durch die Anwendung von Methoden erreicht wird, die das Modell nach dem Training analysieren (Post-hoc).

Intrinsische Interpretierbarkeit kann durch die Gestaltung selbsterklärender Modelle erreicht werden, welche die Interpretierbarkeit direkt in die Modellstruktueren einbeziehen. Diese konstruieren interpretierbare Modelle und sind entweder global interpretierbar oder könnten Explanation liefern, wenn sie individuelle Vorhersagen machen. Post-hoc Interpretierbarkeit hingegen bezieht sich auf die Anwendung von Interpretationsmethoden nach dem Modelltraining.

Modellspezifisch oder modell-agnostisch

Modellspezifische Interpretationswerkzeuge sind auf bestimmte Modellklassen beschränkt. Dabei ist die Interpretation von Regressionsgewichten in einem linearen Modell eine modellspezifische Interpretation. Die lineare Regression ist eines der vielseitigsten, statischen Verfahren und ist ein nützliches Verfahren für Prognosen. Werkzeuge bzw. Tools, die eine spezielle Struktur eines Entscheidungsbaumes ausnutzen und dadurch nur für diese verwendet werden könnnen, sind ebenfalls modellspezifisch.

Modell-agnostische Tools können auf jedes maschinellen Lernverfahren nach dessen Traning (Post-hoc) angewendet werden.

Lokal oder global

Hier stellt sich die Frage, ob die Interpretationsmethode eine einzelne Vorhersage (lokal) oder das gesamte Modellverhalten (global) erklärt wird. In der Regel stellt solch eine Erklärung eine Beziehung zwischen den Input-Featuren und dem Modell-Output her. Die Erklärung präsentiert diese in einer für Menschen verständlichen Art und Weise.

2.4.2 Post-hoc Interpretability und Explanation

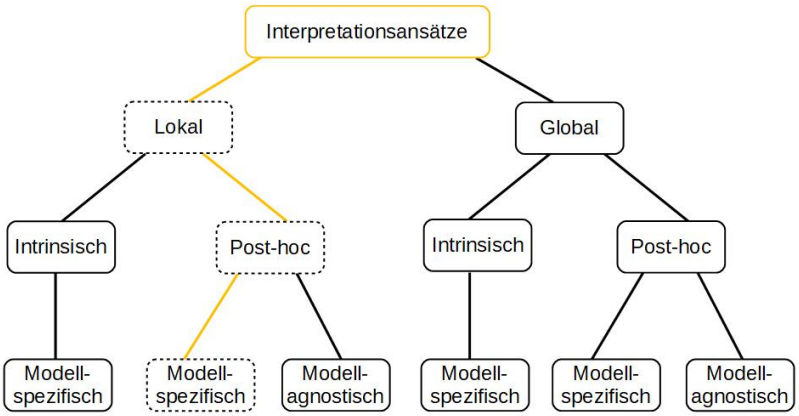


Abbildung 2.3: Taxonomy von Interpretability Methoden

Um die Frage, wie ein undurchsichtiges Modell wie das ML System von ORES nachträglich interpretiert werden kann, ohne dass die Vorhersageleistung beeinträchtigt wird, zu beantworten, nutze ich den Ansatz von *Post-hoc interpretability* und gehe den gelben Pfad aus Abbildung 2.3 entlang. Post-hoc Interpretability bietet einen guten Ansatz, um Informationen aus erlernten Modellen zu extrahieren. Während Post-hoc Interpretationen oft nicht genau erklären, wie ein Modell funktioniert, können dennoch nützliche Informationen für Endnutzer*innen von ML-basierten Systemen vermitteln. Bei allem, was wir wissen, können sich die Prozesse, durch die wir Menschen Entscheidungen treffen, und die, durch die wir sie erklären, unterscheiden.

2.5 User-Centred Explanations

Um ein möglichst verständliches Explanation Interface für die Endnutzer*innen zu entwickeln, stehen Erklärungen (Explanation) und die Interpretierbarkeit dieser Explanations im Vordergrund.

Das Ziel ist es, dem Benutzer die Vorhersagen von ML-Modellen verständlich zu machen, was durch Erklärungen erreicht wird. Dafür dient eine Erklärungsmethode, die nichts anderes ist als ein Algorithmus, der Erklärungen generiert. Die Rolle der Erklärungsmethoden und der generierten Erklärungen innerhalb der ML-Pipeline ist in Abbildung 2.4 dargestellt[DVCC19].

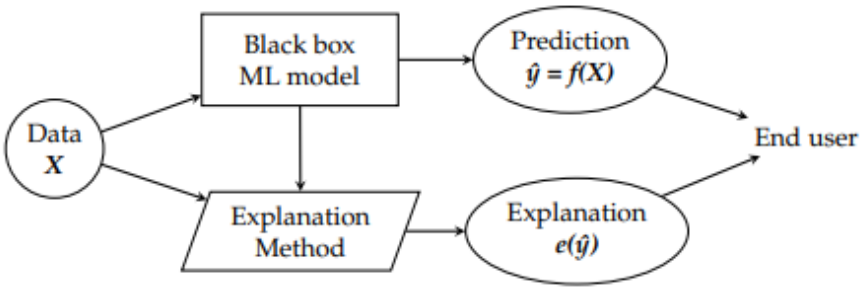


Abbildung 2.4: Explanation Pipeline
[DVCC19]

Mireia Ribera et al., 2012 [MR12] definieren fünf Hauptaspekte, die im Mittelpunkt der jüngsten Erhebungen und theoretischen Rahmen zur Erklärbarkeit stehen:

1. Was sind Explanations?
2. Was sind die Zwecke und Ziele von Explanations?
3. Welche Informationen müssen in Explanations enhalten sein?
4. Welche Art von Explanations kann ein System geben?
5. Wie können wir die Qualität von Explanations bewerten?

In der Literatur wird der Begriff Explanations oft in Zusammenhang mit Transparenz, Interpretierbarkeit, Vertrauen und Fairness und Rechenschaftspflicht in Verbindung gebracht. In jede Explanation sind zwei Subjekte beteilgt, derjenige, der sie liefert (das System) oder der Erklärer und derjenige, der sie erhält (der Mensch).

Der Bedarf an Explanations ist in vier Punkten zu begründen:

1. Verifiaktion des Systems, d.h. die Explanations helfen sicherzustellen, dass die Algorithmen die erwartete Leistung erbringen.
2. Verbesserung des Systems, d.h. das Model zu verstehen und des Datensatzes, um verschiedene Modelle zu vergleichen und Fehler zu vermeiden.
3. Lernen von Systemen.
4. Einhaltung der Rechtsvorschriften.

Wachter et al., 2018 [SWR18] beschreiben wichtige Ziele für Explanations. Explanations sollen dabei helfen, den Erklärungsempfänger zu informieren, warum eine Entscheidung getroffen wurde. Außerdem sollte es Gründe dafür geben, negative Entscheidungen anzufechten und zu verstehen, was geändert

werden könnte, um ein gewünschtes Ergebnis in der Zukunft basierend auf dem gegenwärtigen Entscheidungsmodell zu erhalten.

Eine mangelnde Systemverständlichkeit führt dazu, dass die Benutzer*innen dem System misstrauen, es missbrauchen oder gar ganz aufgeben. Offensichtlich ist es wichtig zu analysieren, was eine Explanation menschenfreundlich (engl. *Human-Friendly Explanation*) macht, da Explanations, so korrekt sie auch sein mögen, nicht notwendigerweise in einer leicht verständlichen Weise präsentiert werden.

Miller, 2017 [Mil17] führte eine Umfrage unter Publikationen über Explanations durch. Aus seiner Umfrage ergeben sich die folgenden Human-Friendly Merkmale von Explanation [DVCC19]:
Constrastiveness, Selectivity, Social, Focus on the abnormal und Trust

Constrastiveness

Die Menschen fragen nicht, wieso eine Vorhersage gemacht wurde, sondern eher warum diese Vorhersage kam und keine andere. Sie sind an den Faktoren interessiert, die sich im Input ändern müssen, damit auch die Vorhersage sich ändert.

Also eine Explanation, die einen gewissen Kontrast zwischen der zu erklärenden Instanz und einem Referenzpunkt darstellen, sind vorzuziehen.

Selectivity

Die Menschen erwarten keine Explanation, welche die tatsächliche und vollständige Liste der Ursachen eines Ereignisses abdecken. Sie ziehen es vor, ein oder zwei Hauptursachen aus einer Vielzahl von möglichen Ursachen als Explanation auszuwählen.

Das Explanation Modell sollte in der Lage sein, ausgewählte Explanations zu liefern oder zumindest deutlich zu machen, welche die Hauptursache für eine Vorhersage sind.

Social

Explanation sind ein Teil einer sozialen Interaktion zwischen Erklärenden und dem Erklärten. Das bedeutet, dass der soziale Kontext den Inhalt, die Kommunikation und die Art der Explanation bestimmt.

Im Bezug auf die Interpretierbarkeit von ML basierten Systemen bedeutet das, dass man bei der Beurteilung die am besten geeignete Explanation das soziale Umfeld und die Zielgruppe berücksichtigen sollte.

Die beste Explanation kann also je nach Anwendungsgebiet und Anwendungsfall variieren.

Focus on the abnormal

Menschen konzentrieren sich mehr auf anormale Sachen/Ursachen, um Ereignisse zu erklären. Dies sind die Ursachen, die eine geringe Wahrscheinlichkeit haben, aber trotzdem eingetreten sind. Im Hinblick auf die ML-Interpretierbarkeit sollte, wenn eine der Eingabemerkmale für eine Vorhersage in irgendeiner Weise abnormal war (z.B. eine seltene Kategorie) und das Merkmal das Vorhersageergebnis beeinflusst hat, dieser Wert in die Erklärung einbezogen werden, auch wenn andere, häufigere Merkmalswerte denselben Einfluss haben, wie der abnormale.

Trust

Gute Explanations sind in der realen Welt erwiesenermaßen wahr. Dies bedeutet nicht, dass die ganze Wahrheit in der Erklärung enthalten sein muss, da sie die Auswahl der Explanations beeinträchtigen würden. Selectivity ist daher ein wichtigeres Merkmal als Trust.

Für die Interpretierbarkeit von ML bedeutet dies, dass ein Explanation sinnvoll und geeignet sein muss, um Vorhersagen für andere Fälle zu treffen.

Inhaltlich sollte eine Explanation nicht nur das „Was“ beantworten können, sondern auch das „Warum“ und „Warum nicht“.

„Warum“:

- Warum/Wie hat die Instanz diese Prediction gegeben?
- Welche Feature von dieser Instanz führten zur Vorhersage?
- Warum haben Instanz A und B die gleiche Prediction

„Warum nicht“:

- Warum/Wie hat die Instanz keine Predicion?
- Warum ist die Vorhersage von P anstelle von Q?
- Warum haben die Instanzen A und B unterschiedliche Prediction.

Lim et al., 2012 [MR12] haben Fragen aufgelistet, welche eine Explanation beantworten können sollte.

1. Was hat das System P getan?
2. Warum hat das System X getan?
3. Warum hat das Systen nicht X getan?
4. Was würde das System tun, wenn das Ereignis Y eintritt?

5. Was kann ich machen, um das System dazu zu bringen Z zu tun in der aktuellen Situation?

Nachdem ein Explanation Modell entwickelt wurde, geht es um die Modellinterpretation.

Modellinterpretation bedeutet, Vernunft und die Logik dahinter zu liefern, um die Rechenschaftspflicht und Transparenz des Modells zu ermöglichen. LIME (*Local Interpretable Model-Agnostic Explanations*) ist ein Algorithmus, der die Vorhersagen jeder Klassifikatoren erklärt, indem er lokal mit einem interpretierbaren Modell sich Instanzen annähert (siehe Abbildung 2.5). Die LIME-Autoren Ribeiro et al., 2016 [MTR16] erwähnten, dass LIME bei einigen Szenarien nicht in der Lage ist, das Modell korrekt zu erklären. Daher schlugen sie eine neue Art der Modellinterpretation vor, nämlich *Anchors*.

Anchors erklärt einzelne Vorhersagen eines beliebigen Black-Box Klassifikationsmodells, indem eine Entscheidungsregel gefunden wird, die die Vorhersage ausreichend "verankert"(siehe Abbildung 2.5). Eine Regel verankert eine Vorhersage, wenn Änderungen in anderen Merkmalswerten die Vorhersage nicht beeinflussen. Anchors verwendet Techniken des Reinforcement Learning in Kombination mit einem Algorithmus für die Graphensuche, um die Anzahl der Modellaufrufe (und damit die erforderliche Laufzeit) auf ein Minimum zu reduzieren und gleichzeitig in der Lage zu sein, sich von lokalen Optima zu erreichen [11].

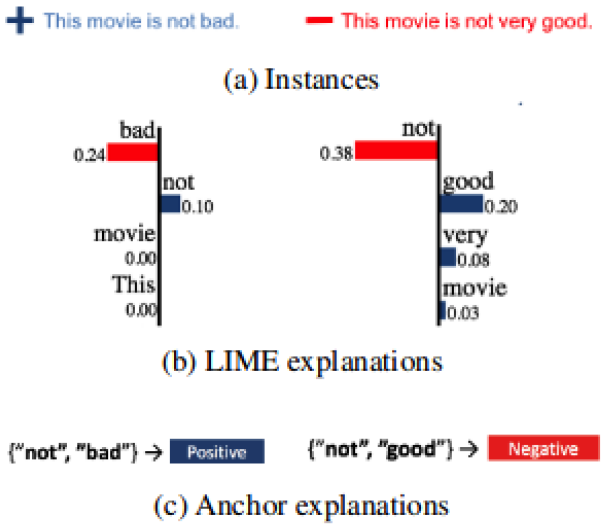


Abbildung 2.5: LIME und Anchor Explanation
https://docs.seldon.io/projects/alibi/en/stable/methods/Anchors.html, besucht am 08.09.2020

[11] `https://christophm.github.io/interpretable-ml-book/anchors.html`

Obwohl sie einfache Funktionen verwenden können, um Komplexität lokal zu interpretieren, können sie nur die jeweilige Instanz erklären. Das bedeutet, dass sie möglicherweise nicht für eine unsichtbare Instanz geeignet sind. Von der obigen LIME-Erklärung zur Stimmungsvorhersage bietet „not“ einen positiven Einfluss auf der linken Seite, während es in der rechten Seite einen starken negativen Einfluss hat.

Anders als LIME verwendet Anchors die „lokale Region“, um zu lernen, wie das Modell zu erklären ist. Die „lokale Region“ bezieht sich auf eine bessere Konstruktion des erzeugten Datensatzes zur Erklärung.

Für die Evaluierung stellen Dosh-Velenz und Kim folgende Ansätze auf [MR12]:

1. Anwendungsbezogene Evaluierung mit realen Menschen und realen Aufgaben

2. Menschlich begründete Evaluierung mit realen Menschen, aber vereinfachten Aufgaben

3. Funktional begründete Evualierung ohne Menschen und Stellvertreteraufgaben: alle immer inspiriert von realen Aufgaben und Beobachtungen realer Menschen.

Bessere Explanations können geschaffen werden, wenn man neue Richtungen einschlägt. Oftmals hilft es mehr als Explanation zu geben, die sich jeweils an eine Benutzergruppe richtet oder Explanations zu geben, die kooperativen Prinzipien der menschlichen Konversation folgen.

Dafür können die Explanations in drei Hauptgruppen kategorisiert werden, die auf ihren Zielen, ihrem Hintergrund und ihrer Beziehung zum Produkt basieren.

a) Entwickler*innen und Forscher*innen

   Forscher, Softwareentwickler oder Datenanalytiker, die das System erstellen

b) Fachgebietsexperten

   Spezialisten auf dem Gebiet der Expertise, zu dem die Entscheidungen des Systems gehören, wie zum Beispiel Physiker oder Juristen

c) Laienbenutzer*innen

   Die Endempfänger der Entscheidungen

Das System zielt auf Explanations für verschiedene Arten von Benutzern ab, wobei ihre unterschiedlichen Ziele berücksichtigt werden und ihnen relevante und maßgeschneiderte Informationen zur Verfügung gestellt werden (siehe Abbildung 2.6).

[12]

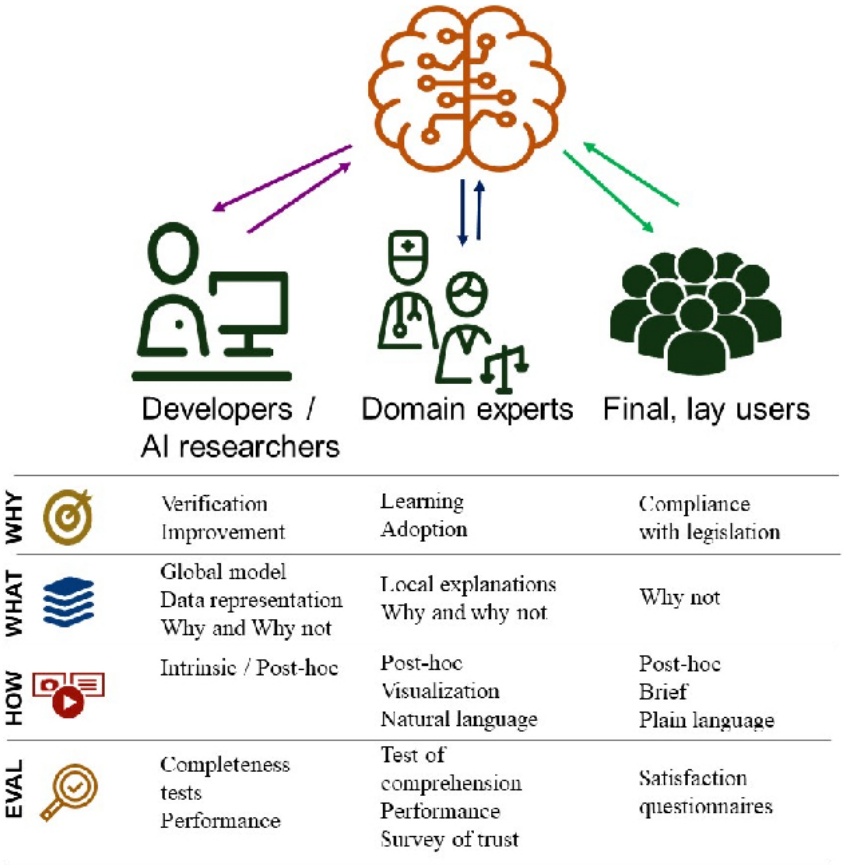


Abbildung 2.6: Zielgruppe im Überblick

Zusammengefasst sollten Explanations folgende Eigenschaften besitzen:

- Die Informationen müssen von hoher Qualität sein. Es darf nichts gesagt werden, von der geglaubt wird, dass diese Aussage falsch ist bzw. keine ausreichenden Beweise haben.

- Die richtige Menge an Informationen sollte zur Verfügung gestellt werden. Der Beitrag sollte so informativ sein, wie erforderlich ist. Die Art und Weise bezieht sich darauf, wie man Informationen liefert und nicht darauf, was geliefert wird.

[12] `http://ceur-ws.org/Vol-2327/IUI19WS-ExSS2019-12.pdf`, besucht am 08.09.2020

- Diese vier kooperativen Prinzipien sollten auch mit anderen erwünschten Eigenschaften von Explanation wie Treue und Verständlichkeit in Verbindung gebracht werden.

    a) Unklare Ausdrucksweisen vermeiden

    b) Mehrdeutigkeit vermeiden

    c) Kurz fassen

    d) Ordnung beibehalten

3 Gadget Prototyp

3.1 Design Rationals

Basierend auf die zuvor gewonnenen Erkenntnissen kann eine Strategie für den gezielten Entwurf entwickelt werden. Diese Strategie wird als Design Rationales bezeichnet.

Mit dem bisherigen Gadget wird als Ausgabe die Klasse, die dem Item entspricht, ausgegeben, sowie einen nummerischen Wert.

[1]

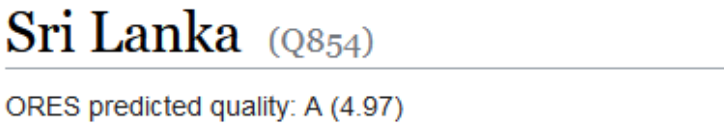


Abbildung 3.1: Gadget original

Ich möchte 4 der 5 Hauptaspekte von Mireia Ribera et al., 2012 [MR12] aufgreifen und auf meinen Anwendungsfall anpassen, um die Design Rationales zu definieren.

1. Was sind die Zwecke und Ziele von Explanation?
2. Welche Informationen müssen in Explanation enhalten sein?
3. Welche Art von Explanation kann ein System geben?
4. Wie können wir die Qualität von Explanation bewerten?

Das Gadget dient dazu den Endnutzer*innen darüber zu informieren und zu helfen zu verstehen, warum die Qualitsklasse für das Item ausgewählt wurde. Die Wikidata Redakteure haben hierfür ein Bewertungsschema angelegt, worin erklärt wird, welche Merkmale für die jeweiligen Klassen stehen. Diese Merkmale werden in dem bisherigem Gadget nicht erwähnt. Was auch hier für eine gute Explanation fehlt, ist ein Verständnis, was man an den Merkmalen ändern könnte, um ein gewünschtes Ergebnis in der Zukunft zu erhalten.

In diesem Zusammenhang möchte ich zwei Human Friendly Explanations von Miller, 2017 [Mil17] erwähnen, die ich mit einbinden möchte:

[1] `https://www.wikidata.org/wiki/Q854`, besucht am 08.09.2020

Contrastiveness und Selectivity.
Das Interface sollte folgende Fragen beantworten können:

Constrastiveness

> Ist es möglich, Items zu zeigen, die genau eine gegenteilige Eigenschaft haben? Was muss sich im Input verändern damit ein gegenteiliges Ergebnis ensteht?

Selectivity

> Ist es möglich, herauszubekommen, welches Feature besonders stark zu dieser Bewertung beigetragen hat.

Wie in Abschnitt 2.5 erwähnt, spielen besonders die
„Was-" und „Warum-" Fragen eine wichtige Rolle, um Systemausgaben besser zu verstehen.

1. Was hat ORES gemacht?
2. Warum hat ORES diese Klasse angezeigt?
3. Warum hat ORES keine andere Klasse gewählt?
4. Was würde ORES tun, wenn die Merkmale sich ändern?
5. Wie kann ich ORES dazu bringen eine andere Klasse in der aktuellen Situation anzuzeigen?

Explanations können Post-hoc erstellt werden, wenn die Entscheidung bereits getroffen wurde. Durch unterschiedliche Explanation Interfaces können Erläuterungen dargestellt werden, die sie interpretieren.

Als Ausgangssituation für eine geeignete Visualisierung dient die Idee des Recommender Papers von Julie Daher et al., 2017 [JBDB17] .

Traditionelle Explanation Interfaces sind in der Regel:

- Text-Explananation
- Tag basiertierte Explanations
- Histrogramme
- Radargrafiken
- Tortendiagramme

- Baumdiagramme

Es existieren 6 definierte Kriterien für die Bewertung von Explanation in Recommender Systemen:

- Transparenz
- Überprüfbarkeit
- Effizienz
- Überzeugungskraft
- Zufriedenheit
- Vertrauen

Diese Maße bewerten die Qualität der Explanation und werden als Vorteile betrachtet, die Explanation bieten können, da sie die Fragen nach dem „Warum“ in Explanations beantworten können.

In dem Paper werden unterschiedliche Beispiele für Explanation Interfaces im Bereich der Recommender Systeme im Bezug auf Ratings gegeben, die für unseren Anwendungsfall genutzt werden können. Nicht alle aber einige werde ich hier erwähnen, die für die Visualisierung der Wahrscheinlichkeitsverteilgung geeignet sind.

Von den vielen Visualisierungsmöglichkeiten empfand ich das Confidence Display [2] (siehe Abbildung 3.2) als besonders geeignet, weil es Vertrauen in die Empfehlung schafft, indem sie Prozentsätze der Fälle, in denen sich die Merkmale der Systeme als richtig erwiesen haben, gezeigt wird.

[3]

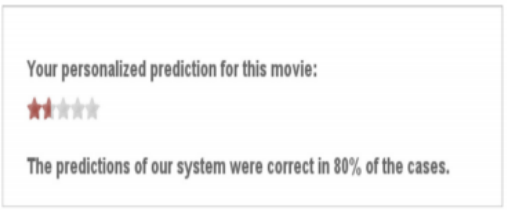


Abbildung 3.2: Confidence display

Die gewichtete Summe (Schwellwert), die bislang als nummerischen Wert neben der identifizierte Klasse angezeigt wird, kann für das Explanation Interface

[2] `https://hal.archives-ouvertes.fr/hal-01836639/document`, besucht am 08.09.2020
[3] `https://hal.archives-ouvertes.fr/hal-01836639/document`, besucht am 08.09.2020

genutzt werden. In einer Skala die mit den Klassen von E bis A gekennzeichnet wurde, kann der Schwellwert als Punkt für die Füllmenge genutzt werden.

Eine der traditionellen Explanation Interfaces ist die Erklärung in Form einer textlichen Beschreibung (siehe Abbildung 3.3), die in natürlicher Sprache die Gründe für die Abgabe dieser Empfehlungen angibt. Der Vorteil dieses Explanation Interfaces ist, dass sie in alle Bereiche passt und allen Fällen verwendet werden kann, da sie die am wenigsten riskante Schnittstelle ist.
[4]

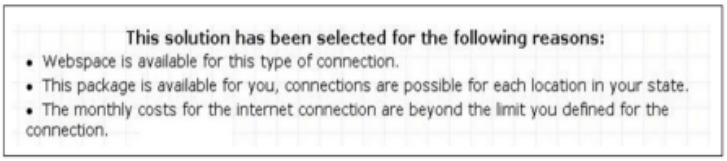


Abbildung 3.3: Textual explanation

In dem Gadget werden zwei große Bereiche miteinander verbunden. Zum einen soll ein visuelles Interface entstehen, in dem ich auf Explanation Interfaces aus dem Bereich der Recommender Systeme zurückgreife und zum anderen die Merkmale der Interpretierbarkeit anwenden, die wichtig sind, um das System zu verstehen.

In der Tabelle 3.1 habe ich die Anforderungen aus beiden Bereichen, die oben in Textform verfasst wurden, übersichtlich zusammengefasst und gekennzeichnet, wie groß der Aufwand hierfür wäre und ob eine Umsetzung möglich ist.

[4] `https://hal.archives-ouvertes.fr/hal-01836639/document`, besucht am 08.09.2020

Anforderungen	**Wichtigkeit** der Anforderungen (niedrig/mittel/hoch)	**Aufwand** der Umsetzung (niedrig/mittel/hoch)	**Designansätze**	**Umsetzung** (ja/nein)
Constrastiveness	hoch	niedrig	Textual Explanation, Range Slider	ja
Selectivity	hoch	Hoch, ORES API gibt keine Auskunft über die Gewichtigung der einzelnen Merkmalen, die zu dem Ergebnis geführt haben, bietet	Range Slider	nein
Social	niedrig			nein
Focus on the abnormal	niedrig			nein
Trust	hoch	mittel	Confidence Display, Range Slider	ja
Warum hat diese Instanz diese Predicition gegeben?	hoch	mittel	Text Explanation	ja
Welche Features dieser Instanz haben zu dieser System Prediction geführt?	hoch	mittel	Text Explanation, (Anchor Explanation)	ja
Warum geben die Instanz A und B die gleiche Prediction?	niedrig	niedrig		nein
Warum hat die Instanz nicht ... vorhergesagt?	niedrig	niedirg	Range Slider	ja
Warum wurde für die Instanz P vorhergesagt und nicht Q?	hoch	hoch	Confidence Display	ja

Tabelle 3.1: Design Rationals

Auf Grundlage der erarbeiteten Design Rationals habe ich einen Low-Fidelity-Prototyp entworfen. Low-Fidelity-Prototyp sind einfach und schnell zu entwickeln und ermöglichen das Konzept interaktiv zu gestalten. Wie bereits erwähnt, habe ich mich an den Explanation Interfaces aus dem Recommender Systemen orientiert und als Grundlage für das neue Interface im Anwendungsfall ORES angewendet. Dabei habe ich meine Priorität auf die Wichtigkeit der Anforderungen gesetzt und anhand meiner Tabelle 3.1. die Anforderungen gezogen, die zutreffen.

Durch die Tabelle wird ersichtlich, dass der Fokus der Arbeit auf Constrativeness und Trust gesetzt wird. Selectivity wäre für eine bessere Explanation von Vorteil, ist aber leider nach dem aktuellen Stand von ORES nicht möglich, da wir nicht in der Lage sein werden, das Merkmal des Modells direkt mit den Kriterien für jede Itemkategorie gleichzusetzen.

So habe ich den Low-Fidelity-Prototyp (siehe Abbildung 3.4) entworfen:

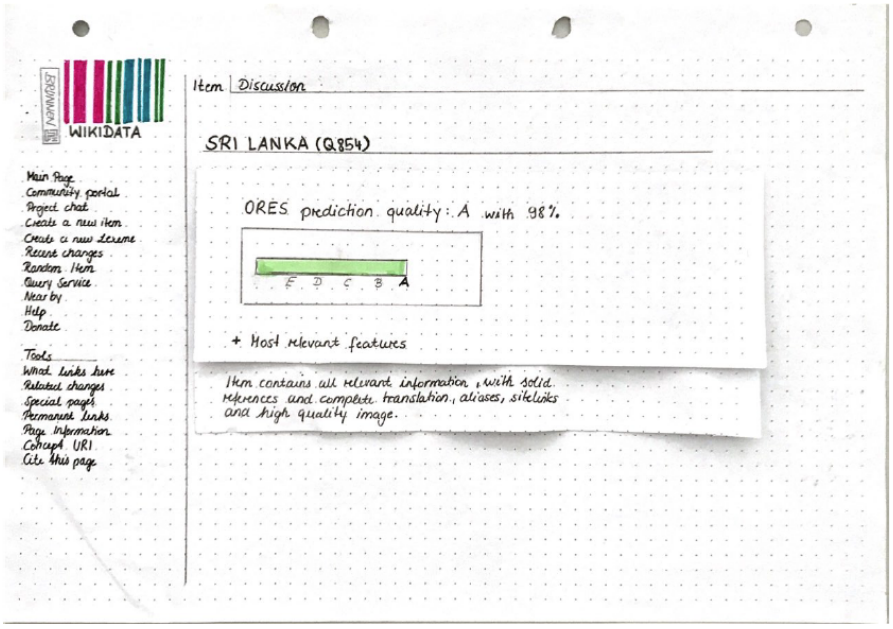


Abbildung 3.4: Low-Fidelity-Prototyp v1

Eine mögliche Idee für Selectivity war es einen Regler für die einzelnen Features einzubauen, der die aktuellen Gewichtungen der einzelnen Merkmale, die die Qualitätsklasse bestimmt, anzeigt. Dem Endnutzer*innen ist es möglich

den Regler, den ich als *Range Slider* bezeichnen werde zu verschieben. Das kann dem Endnutzer*innen dabei helfen zu verstehen, wie viel noch für eine bessere Qualitätsklasse fehlt. Damit könnte auch die Frage beantwortet werden, warum diese Instanz diese Predicition gegeben hat und warum für die Instanz P vorhergesagt hat und nicht Q.

Zusätzlich sollten, sofern möglich, folgende Fragen beantwortet werden:

1. Warum hat diese Instanz diese Predicition gegeben?
2. Welche Features dieser Instanz haben zu dieser System Prediction geführt?
3. Warum wurde für die Instanz P vorhergesagt und nicht Q?

In der Abbildung 3.5 ist der Low-Fidelity-Prototyp zu sehen, der diesen Anwendungsfall abdeckt:

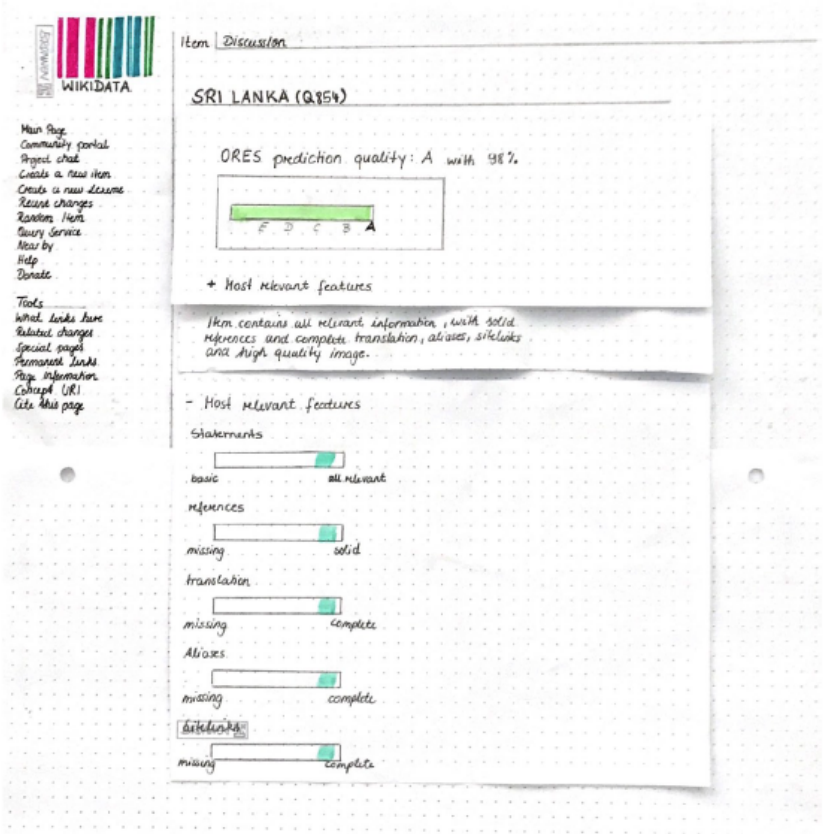


Abbildung 3.5: Low-Fidelity-Prototyp v2

So ist ein Explanation Interface entstanden, das folgende Komponenten beinhaltet:

- Confidence Display
- Textual Explanation
- Range Slider

3.2 Implementierung

Nachdem unterschiedliche Low-Fidelity-Prototypen entstanden sind, habe ich mich an die Implementierung des High-Fidelity-Prototypen gesetzt. Beim High-Fidelity-Prototyp können die Funktionen aus dem Low-Fidelity-Prototyp simuliert werden.

Inwiefern eine Umsetzung möglich ist, habe ich in der letzten Spalte in der Tabelle 3.1. gekennzeichnet.

Kurz zusammengefasst erhalte ich von ORES folgende Informationen:

- Qualitätsklasse A-E
- Schwellenwert
- Wahrscheinlichkeitsverteilungen für die Qualitätsklassen A-E

Mit den Informationen, die ORES gibt, habe ich mich dafür entschieden, meinen Schwerpunkt auf die Anforderung „Trust" zu legen und ein Confidence Display zu erstellen. Gestützt wird das Ganze mit Text Explanation [5], um zusätzliche Informationen darüber zu geben, wieso diese Qualitätsklasse ausgewählt wurde.

Die Implemetierung erfolgt über die Webanwendung von Wikidata. Aaron Halfaker[6] hat einen öffentlichen Link [7] zur Bearbeitung des Codes zur Verfügung gestellt. Diesen habe ich genutzt, um meine Ideen anzuwenden und so mein Explanation Interface zu realisieren.

Bisher gibt ORES die Qualitätsklasse an. Die Information ist wichtig, aber ich empfand es als noch wichtiger zu erwähnen, mit welcher Wahrscheinlichkeit diese Klasse gewählt wurde, weil anhand dieser Prozentzahl Nutzer*innen eher verstehen können, welchen Score ORES für das Item berechnet hat. In Textform erhalten wir die für das Item identifizierte Klasse, sowie den Score mit welcher Wahrscheinlichkeit das Item in dieser Klasse liegt.

```
1 formatScoreHeader: function(score){
2             var prediction = this.extractPrediction(score);
3             var weightedSum = this.computeWeightedSum(score);
4             var probaArray = this.computeProba(score);
5             return this.assessment_system + ": " +
```

[5] `https://hal.archives-ouvertes.fr/hal-01836639/document`, besucht am 08.09.2020
[6] `https://en.wikipedia.org/wiki/Aaron_Halfaker`, am 08.09.2020 gesehen
[7] `https://github.com/halfak/gadgets-ArticleQuality`, am 08.09.2020 gesehen

```
6                   this.names[prediction] + " with " + Math.round(Math.max.apply(
                        Math, probaArray)*100) + " % ";
7           },
```

Neben der Qualitätsklasse erhalten wir den Schwellenwert als nummerische Zahl (gewichtete Summe). Da der Wert ohne weitere Kennzeichnung angegeben wurde, ist es für den Nutzer*innen wohlmöglich schwer zu verstehen, welche wichtige Information sie liefert. Wie bereits erwähnt, teilt uns der Schwellenwert mit zwischen welchen die Qualitätsklassen das Item tendiert.

Ich habe mich dafür entschiedlichen das Confidence Display zu nutzen, um diesen Wert visuell darzustellen. Einen Balken[8], welcher mit den Klassen E - A (die Klasse entsprechen dem nummerischen Wert 1-5) gekennzeichnet sind, wird gefüllt mit dem Schwellenwert (weighted sum).

```
1   renderBarHeader: function(score){
2               var rawText = this.formatBarHeader(score);
3               var qualityBlock = $('<div>').addClass("article_quality_bar").attr('id', '
                    myProgress');
4               var progressBar = $('<div>').addClass("innerDiv").attr('id', 'myBar');
5               var progressClass = $('<div>').addClass("progressClass").attr('id', '
                    myClass');
6
7               var progressClassA = $('<div>E</div>').addClass("progressClassName"
                    ).attr('id', 'myClassA');
8               var progressClassB = $('<div>D</div>').addClass("progressClassName"
                    ).attr('id', 'myClassB');
9               var progressClassC = $('<div>C</div>').addClass("progressClassName"
                    ).attr('id', 'myClassC');
10              var progressClassD = $('<div>B</div>').addClass("progressClassName"
                    ).attr('id', 'myClassD');
11              var progressClassE = $('<div>A</div>').addClass("progressClassName"
                    ).attr('id', 'myClassE');
12
13              var weightedSum = this.computeWeightedSum(score);
14              var res = weightedSum / 5 * 100;
15              $('#bodyContent').prepend(qualityBlock);
16              this.parseText(rawText)
17                  .done(function(html){
18                      qualityBlock.html(html);
19                      $('#myProgress').append(progressBar);
20                      $('#myProgress').append(progressClass);
21                      $('.progressClass').append(progressClassA);
22                      $('.progressClass').append(progressClassB);
23                      $('.progressClass').append(progressClassC);
24                      $('.progressClass').append(progressClassD);
25                      $('.progressClass').append(progressClassE);
```

[8] `https://www.w3schools.com/howto/howto_css_skill_bar.asp`, besucht am 08.09.2020

```
                $('#myBar').css('width', res + '%');


            })
            .fail(function(error){console.error(error)});
    },
```

Damit die Nutzer*innen wissen, welche Instanzen die Prediction beeinflusst hat, habe ich mit Hilfe eines Textual Explanation das Gadget mit den Kriterien aus dem Bewertungsschema ergänzt.

```
    renderDescriptionsHeader: function(score){
        var rawText = this.formatDescriptionsHeader(score);
        var qualityBlock = $('<br><div><br>').addClass("
            article_quality_descriptions");
        $('#bodyContent').prepend(qualityBlock);
        this.parseText(rawText)
            .done(function(html){qualityBlock.html(html)})
            .fail(function(error){console.error(error)});
    }

descriptions: {
            E: "Items with less basic statements, but lacking in references,
                translations, and aliases.",
            D: "Items with some basic statements, but lacking in references,
                translations, and aliases.",
            C: "Items containing most critical statements, with some references,
                translations, aliases, and sitelinks.",
            B: "Items containing all of the most important statements, with good
                references, translations, aliases, sitelinks, and an image.",
            A: "Items containing all relevant statements, with solid references,
                and complete translations, aliases, sitelinks, and a high quality
                image."
        }
```

In der Abbildung 3.6 ist das Gadget für die Interpretierung der automatisierten Qualitätsbewertung mit ORES in Wikidata zu sehen:
[9]

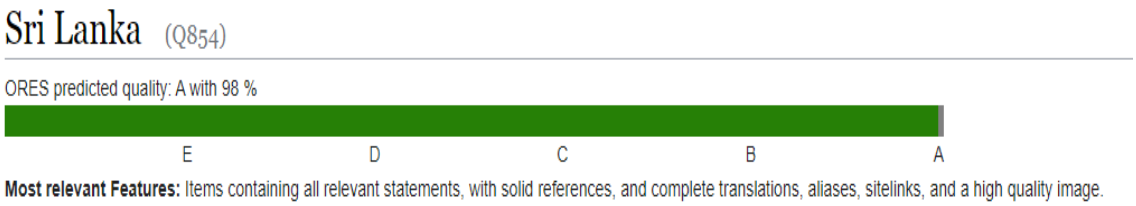


Abbildung 3.6: Gadget v1

[9] `https://www.wikidata.org/wiki/Q854`, besucht am 08.09.2020

Allerdings schien mir das Gadget nicht aussagekräftig genug, da allein die Angabe der Qualitäsklasse den Nutzer*innen nicht ausreichen könnte zu verstehen, welche Features zur Entscheidung beigetragen haben. Deshalb habe ich mich dafür entschieden, das Gadget mit weiteren Explanations zu ergänzen, mit Merkmalen, die ORES nicht liefert. An dieser Stelle ist zu erwähnen, dass es sich bei den Werten für die Features um fiktive Daten handeln.

Ich setzte meinen Fokus darin, herauszufinden, welche Features besonders stark zu dieser Bewertung beigetragen haben, mit dem Ziel folgende Fragen zu beantworten:

- Warum hat ORES keine andere Klasse gewählt?
- Was würde ORES tun, wenn die Merkmale sich ändern?
- Wie kann ich ORES dazu bringen eine andere Klasse in der aktuellen Situation anzuzeigen?

Ich entschied mich dafür einen Range Slider[10] für die Visualierung zu nutzen, weil ich damit die Möglichkeiten habe, mit den Werten der einzelnen Features zu ändern. Es ist eine Möglichkeit, herauszufinden, wie die Features die Qualitätsklasse beinflussen. Ich bin davon ausgegangen, dass die Gewichtung der einzelnen Features gleichverteilt sind, sodass der Durchschnitt aus dem Features den Schwellenwert des Items bestimmen und so können die Nutzer*innen mit dem Regler testen, wie sich die Qualität ändert, sobald sich die Gewichtung der einzelnen Features ändert. Zudem habe ich für mein Beispiel Item Sri Lanka feste Werte für die einzelnen Features gesetzt.

Aus dem Bewertungsschema entnahm ich folgende Features:

- Statements
- Reference
- Translation
- Aliases
- Sitelinks
- Image

[10] `https://www.w3schools.com/howto/howto_js_rangeslider.asp`, besucht am 08.09.2020

```
        getAndRenderFeatureBars: function(){
            var revId = mw.config.get('wgCurRevisionId');
            this.oresScore(revId)
                .done(this.renderFeatureBars.bind(this))
                .fail(function(error){console.error(error)});
        },
        renderFeatureBars: function(score){
            this.renderFeatureBarsHelper("Statements", 98);
            this.renderFeatureBarsHelper("References", 96);
            this.renderFeatureBarsHelper("Translation", 100);
            this.renderFeatureBarsHelper("Aliases", 96);
            this.renderFeatureBarsHelper("Sitelinks", 99);
            this.renderFeatureBarsHelper("Image", 99);


        },
        renderFeatureBarsHelper: function(feature, value) {
            var rawText = "";
            var container = feature + "Slider";
            var containerId = feature + "SliderId";

            var sliderRange = "range" + feature;

            var sliderContainer = $('<div>').addClass("slidecontainer").attr('id',
                containerId);

            var start = "missing";
            var end = "complete";
            switch (feature) {
                case ("Statements"):
                    start = "basic";
                    end = "all relevant";
                    break;
                case ("References"):
                    end = "solid";
                    break;
                default:
                    start = "missing";
                    end = "complete";
                    break;
            }

            var htmlText = '<p>' + feature + '</p><input type="range" min="1"
                max="100" value=' + value + ' class="slider" id=' + sliderRange +
                '><div><p style="float:left; margin-top: 0;">' + start + '</p><p
                style="float:right; margin-top: 0;">' + end + '</p></div><div
                style="clear:both;"/>'

            $('#mw-content-text').before(sliderContainer);
```

So wurde das Gadget mit einer weiteren Funktion ergänzt (siehe Abbildung 3.7).

11

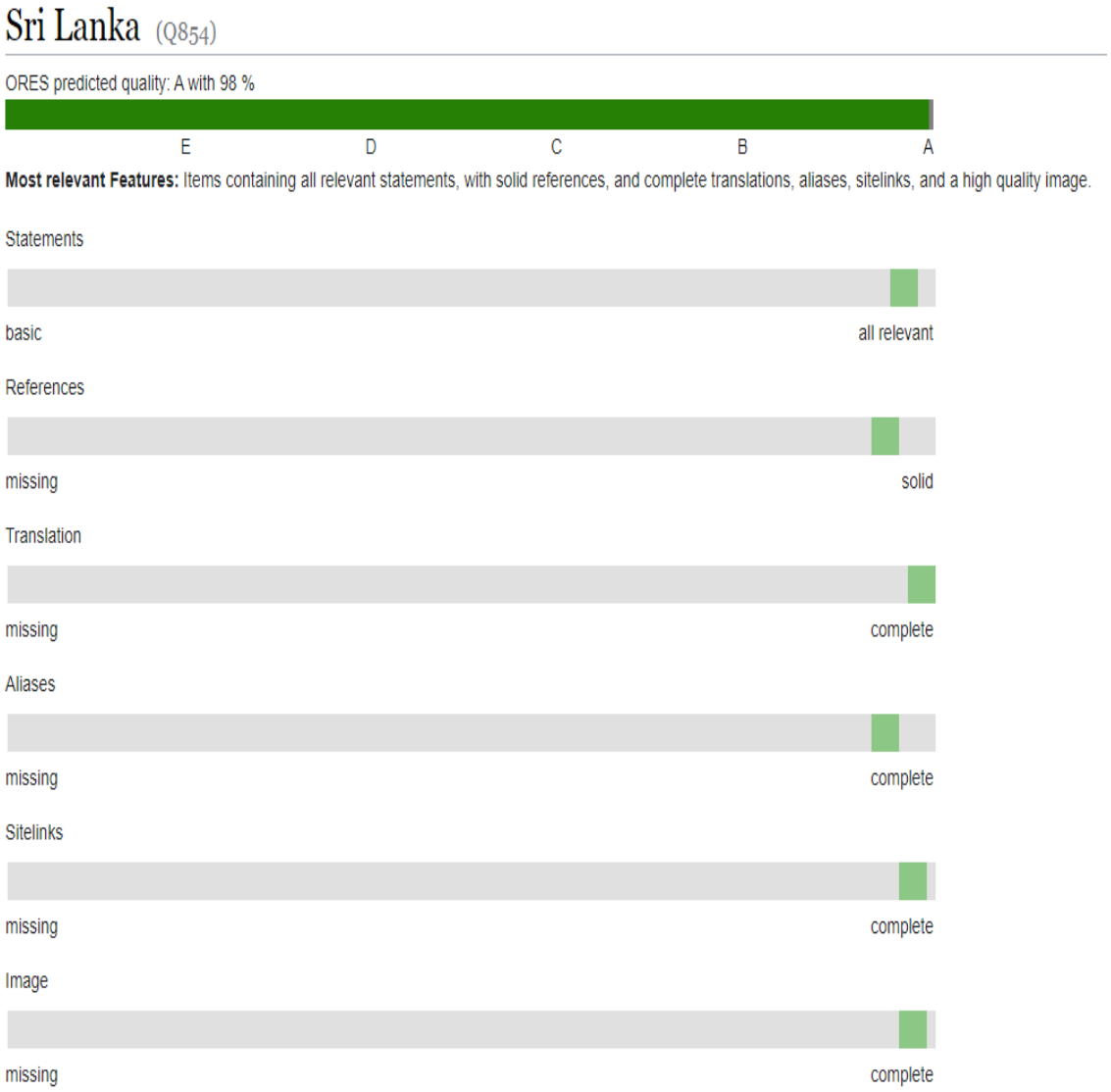


Abbildung 3.7: Gadget v2

Die Nutzer*innen können per Mausklick die Werte der Features ändern und erkannen am ersten oberen Balken, inwiefern das Feature die Qualitätsklasse verändert und zu welcher Qualitätsklasse das Item tendiert.

[11] `https://www.wikidata.org/wiki/Q854`, besucht am 08.09.2020

3.3 Evaluierung

Bei der Evaluation liegt der Fokus auf der Benutzerfreundlichkeit meines High-Fidelity-Prototypens, d.h. wie leicht ist die Verwendung, aber auch welches Wissen die Nutzer*innen aus dem System ziehen.

Mireia Ribera et al., 2012 [MR12] erwähnen, dass Explanations oft in Zusammenhang mit Transparenz, Interpretierbarkeit, Vertrauen und Fairness und Rechenschaftspflicht in Verbindung gebracht werden. Mit der Evaluation möchte ich folgende Fragen beantworten und überprüfen, ob die genannten Kriterien mit dem Explanation Interface erreicht werden können.

1. Entspricht die Funktionalität des Systems die Anforderungen unserer Nutzer*innen?

   - Transparenz

2. Sind Nutzer*innen auch in der Lage diese Funktionen zu finden?

   -Transparenz

3. Wie empfinden Nutzer*innen diese Art dieser Interaktionen? Wie wird es bewertet?

   - Interpretierbarkeit

4. Wie benutzerfreundlich ist es?

   - Vertrauen und Fairness

5. Spezifische Probleme: Werden Benutzer*innen überlastet? Muss ich irgendwas anpassen ?

   - Vertrauen und Fairness

3.3.1 Usability Test

Auf Grund der aktuellen coronabedingten Situation habe ich mich dafür entschieden, den Usability Test remote stattfinden zu lassen. Als Probanden stellten sich einige Mitarbeiter der Forschungsgruppe des HCC's der Freien Universität Berlin zur Verfügung. Ich habe mich für sie entschieden, da meines Erachtens nach, die Mitarbeiter am Besten in meine Zielgruppe passen. Wie zuvor erwähnt, beschränkt sich meine Zielgruppe auf die Domain Experten, die Redakteure der Wikidata Plattform. Zwei von drei Probanden sind aus vergangenen Projekten mit der Domain Wikidata bereits vertraut [12].

[12] `https://www.mi.fu-berlin.de/en/inf/groups/hcc/members/researchers/index.html`, besucht am 08.09.2020

Zudem habe ich mich für einen synchronen Test entschieden, da mir die Zusammenarbeit mit den Probanden wichtig ist. Ich wollte aktiv dabei sein, um die Gedankengänge besser nachzuvollziehen und die Think Aloud Methode zu nutzen. Dabei handelt es sich um eine Forschungsmethode, in der die Probanden gebeten werden seine Gedanken laut auszusprechen [13].

Der Usability Test fand mit folgendem Setting statt:
Die Probanden nutzen zwei Browser. Ein Browser mit dem High-Fidelity-Prototypen und im zweiten Browser war ich über das Tool *Cisco Webex*[14] anwesend. Cisco Webex habe ich als Tool gewählt, da ich damit mit den Probanden problemlos über die Live-Webcamschaltung ohne großen Aufwand kommunizieren konnte. Außerdem war es mir möglich, den Usability Test aufzuzeichen, was sehr hilfreich für die spätere Evaluation war. Dafür habe ich mir zuvor über eine Einverständniserklärung die Erlaubnis eingeholt. Die Teilnehmer habe ich vor dem aktiven Testen darum gebeten, ihr zweites Browserfenster über Cisco Webex freizugeben, damit ich die Bewegung mit der Maus mitverfolgen kann. So konnte ich meine gesamte Aufmerksamkeit auf die Probanden richten und die Anweisungen für den Test geben. Die Zugangsdaten zum Prototyp haben die Probanden von mir bekommen. Diese waren die Anmeldedaten für die Wikidata Plattform, worin das Gadget bereits integriert war. Durchschnittlich dauerte ein Usability Test ca. 20 Minuten.

Um ein möglichst qualitatives Feedback für das Explanation Interface zu bekommen, habe ich die „Was-“ und „Warum-“ Fragen aus meinem Konzept (Tabelle 3.1), auf die ich meinen Schwerpunkt gelegt habe, gestellt.

Zu Beginn habe ich eher allgemeine Fragen gestellt, um zu erfahren, ob die Probanden erkennen, zu welchem Zweck ORES in diesem Gadget eingesetzt wird. Meine Absicht dahinter war es, herauszufinden, ob die automatisierte Qualitätsbewertung mit ORES in Wikidata nachträglich interpretiert werden kann.

- Was hat ORES gemacht?
- Warum hat ORES diese Klasse angezeigt?
- Warum hat ORES keine andere Klasse gewählt?
- Was würde ORES tun, wenn die Merkmale sich ändern?

[13] `https://www.usability.de/usability-user-experience/glossar/concurrent-think-aloud.html#:~:text=Die%20Think%2DAloud%2DMethode%20ist,Teilnehmer%20mit%20dem%20Testgegenstand%20interagiert.`, am 08.09.2020 besucht

[14] `https://www.webex.com/de/index.html`, besucht am 08.09.2020

- Wie kann ich ORES dazu bringen eine andere Klasse in der aktuellen Situation anzuzeigen?

Anschließend stellte ich gezielte Fragen zu einem Item und nutze auch hier den Anwedungsfall „Sri Lanka“. Dabei ging ich folgende Fragen ein:

- Warum wurde für das Item „Sri Lanka“ die Qualitätsklasse A ausgewählt?
- Welche Merkmale von dem Item haben zu dieser Systemvorhersage geführt?
- Warum hat das Item nichte eine andere Klasse vorhergesagt?
- Welche Feature von dieser Instanz führten zur Vorhersage?

Ich habe die Antworten zu den „Was-“ und „Warum-“ Fragen in der Tabelle 3.2 zusammengefasst und werde im Abschnitt 3.4 darauf eingehen. Insgesamt haben die Probanden ähnliche Antworten gegeben, weshalb ich die Antworten nicht nochmal unterteilt haben. Wenn zusätzliche Informationen kamen bzw. erkenntlich war, welche Komponente des Gadgets die Probanden für die Beantwortung der Frage genutzt haben, habe ich Sie innerhalb der Spalte „Bemerkungen“ hinzugefügt.

Frage	**Antwort**	**Bemerkung**
Was hat ORES gemacht?	„ORES hat an Hand der Features bestimmt, wie die Qualität dieses Eintrags ist.“, „ORES gibt es um eine Prozentzahl an und sagt uns wie gut das Item bewertet wurde.“	Range Slider, Textual Explanation
Warum hat ORES diese Klasse angezeigt?	„Weil die Verteiltung der Features so ist.“	Hatte eine Nachfrage, was unter dem Begriff „Klasse“ zu verstehen ist, Range Slider
Warum hat ORES keine andere Klasse gewählt?	„Dafür müsste die Verteilung der einzelnen Features anders liegen, die Features bestimmen die Qualitätsklasse.“	Range Slider
Was würde ORES tun, wenn die Merkmale sich ändern?	„Je nachdem wie die Verteilung der einzelnen Features sind, ändern sich die Qualität von dem Item, die Gesamtbewertung wird verändert. “	Range Slider
Wie kann ich ORES dazu bringen eine andere Klasse in der aktuellen Situation anzuzeigen?	„Die relevant Features ändern.“	Zu Beginn war nicht klar, welche Motivation dahinter stecken sollte, hat aber dann verstanden, dass die Interesse da wäre, zu überprüfen, welches Feature für eine bessere Qualitätsbewertung benötigt wird, Range Slider, Confidence Display
Warum wurde für das Item „Sri Lanka“ die Qualitätsklasse A ausgewählt?	„Da alle einzelne Features hochbewertet waren, wurde diese Qualitätsklasse ausgewählt.“	Range Slider
Welche Merkmale von dem Item haben zu dieser Systemvorhersage geführt?	„Die Merkmale der einzelnen Features.“	Range Slider
Warum hat das Item nicht eine andere Klasse vorhergesagt?	„Die Verteilung der Features waren zu gut für eine schlechte Bewertung.“	Range Slider
Welche Feature von dieser Instanz führten zur Vorhersage?	„Statements, References, Translation, Aliases, Sitelinks und Images.“	Textual Explanation, Range Slider

Tabelle 3.2: Usability Test - Ergebnis

3.4 Diskussion

Das Explanation Interface wurde visuell aus drei Komponenten zusammengestellt:

- Textexplanation
- Confidence Display (Bewertungsskala)
- Range Slider

Die Evaluation hat gezeigt, dass die Auswahl der einzelnen Komponenten seine Vor- und Nachteile haben, auf die ich gerne eingehen möchte.

Insgesamt haben die Probanden die automatisierte Qualitsätsbewertung richtig interpretieren können. Die Porbanden konnten mir direkt die Qualitätsklasse das Items nennen und mit welcher Score die Klasse bestimmt wurde. Allerdings gab es vereinzelnd unterschiedliche Anliegen, die sie sich mit dem neuen Gadget erhofft haben.

Die Probanden haben verstanden, wofür die Bewertungsskala dient. Allerdings haben Sie die Skala so interpretiert, dass der Score die Skala ausfüllt und nicht, dass es sich hierbei um eine gewichtete Summe handelt, die uns mitteilt, zwischen welchen Qualitätsklassen das Item tendiert. Zudem kam die Anmerkung, dass es doch sehr irritierend ist, dass man die Skala von E nach A liest, anstelle von A nach E. Das hat zu Beginn für ein wenig Verwirrung gesorgt. Von den Probanden kam als Feedback der Hinweis, dass eine Art Legende dabei helfen könnte, das Prinzip der Qualitätsverteilung, so wie sich die Wikidata Redakteure mit ihrem Bewertungsschema vorgestellt haben, zu verstehen.

Die Probanden haben ihre Antworten größtenteils über den Range Slider beantwortet, den ich zusätzlich implementiert habe. Die Probanden haben aktiv mit dem Range Slider gearbeitet, um das „Warum„ dieser Prediction zu beantworten. Sie waren an den Faktoren interessiert, die sich im Input ändern müssen, damit sich auf die Prediction ändert und das sind in dem Fall die Features, die für die Entscheidung der Qualitätsbewertung beitragen (*Constrastiveness*).

Einer der Probanden hat erkannt, dass die Range Slider bewegt werden können. Die restlichen Probanden sind von einer Display Anzeige ausgegangen. Als Schlussfolgerung ziehe ich daraus, dass es für die Nutzer*innen erkennbar sein muss, dass man diesen Range Slider bewegen kann, um noch mehr Informationen über das Item zu bekommen. So kann man beispielsweise ermitteln, welches Feature für eine bessere Qualität beitragen müsste. Einer der Probanden hat die Hover Funktion als visuelle Umsetzung der einzelnen Skalen vorgeschlagen. Mit der Maus Hover Funktion kann dann abgelesen werden, in

welchem Bereich sich der Range Slider in der Gasamtwertung befindet, mit einem Hinweis, dass man diesen auch bewegen kann.

Die Probanden hatten kein Verlangen danach zu verstehen, wie der Algorithmus hinter dem System funktioniert, d.h. wie z.B. der Score für die Bestimmung der Qualitätsklasse berechnet wird. Aus einer Vielzahl an Möglichkeiten für eine Ursache, hat ihnen als Grund für die Entscheidung der Einfluss der einzelnen Features ausgereicht. Den Probanden wurde deutlich gemacht, dass die Features die Hauptursache für die Predicition sind (*Selectivity*).

Insgesamt entspricht die Funktionalität des System den Anforderungen unserer Probanden. Die Probanden kennen das Ergebnis der Systemausgabe und die Gründe dafür. Da sie auch verstanden haben, was ORES berechnet hat und wieso es zu dieser Ausgabe kam, gehe ich davon aus, dass das System insgesamt transparenter erscheint als zuvor, da sie auch in der Lage waren, die Funktionalitäten zu finden.

Die Probanden fanden diese Art der Interaktion sehr interessant. Gerade wenn es darum geht, als Redakteur zu erfahren, welche Features noch für eine bessere Qualität benötigt werden. Die Benutzerfreundlichkeit könnte noch verbessert werden. Da hat noch die ein oder andere Hilfestellung gefehlt das Interface aktiv zu nutzen.

Ich hatte nicht den Eindruck, dass die Probanden überlastet werden. Eher im Gegenteil. Sie konnten relativ schnell Informationen ablesen und diese auch nennen, sodass für mich keine spezifischen Probleme herausgestellt haben.

Alles in Einem konnten die Probanden mit der neuen Version des Gadgets meine Fragen richtig beantworten, sodass eine Grundlange für eine bessere Interpretierbarkeit der automatisierten Qualitätsbewertung mit ORES in Wikidata gelungen ist. Im Vergleich zum ursprünglichen Gadget ist das neue Explanation Interface für Nutzer*innen transparenter und vertrauenswürdiger. Es wurde auch deutlich eine bessere Interpretierbarkeit erreicht.

4 Zusammenfassung

Ziel dieser Arbeit war es ein Konzept für eine visuelle Methode zur Verbesserung der Interpretierbarkeit der automatisierten Qualitätsbewertung mit ORES in Wikidata und aus dem Konzept ein *Explanation Interface* zu erstellen. Mit Hilfe von *User Centred Explanation* habe ich ein Redesign für das bereits vorhandene Gadget entwickelt. Hierfür habe ich Explanation Interfaces aus dem Bereich der *Recommender Systeme* mit *Human Friendly Explanation* für eine bessere Interpretierbarkeit kombiniert, da ich meinen Schwerpunkt darin gelegt habe, nicht nur zu zeigen, was das System macht, sondern auch warum.

Es entstand ein neues Gadget mit zwei Versionen. Die erste Version, ist die, die man tatsächlich mit den gegebenen Daten umsetzen kann. In der zweiten Version wurde eine Ergänzung hinzugefügt, die ich für sehr sinnvoll und nützlich empfinde, aber zum jetzigen Zeitpunkt nicht realisiert werden kann, da ORES keine Auskunft über die Verteilung der einzelne *Features* gibt. In beiden Versionen werden die Nutzer*innen darüber informiert, mit welchem Score das Item in die jeweilige Qualitätsklasse zugeordnet wurde. Zusätzlich wird die gewichtete Summe über einen ausgefüllten Balken demonstriert, der darauf hinweist, zu welcher Klasse das Item tendiert.

Der Mensch sollte für eine bessere Interpretierbarkeit immer Wissen warum das System diese Entscheidung getroffen hat, statt ihm blind zu vertrauen und mit meinem Konzept, bestehend aus Human Friendly Explanation und Explanation aus dem Bereich der Recommender System und die dazu gehörige Implementierung eines Explanation Interface wurde das Vertrauen zu den Nutzer*innen aufgebaut und die Gründe für die Systemausgabe geliefert.

4.1 Ausblick

Die Evaluation hat gezeigt, dass der Prototyp von den Probanden akzeptiert wurde. Die Systemausgabe wurde verstanden und das Gadget wurde aktiv genutzt. Damit die Version mit dem Range Slider auch umgesetzt werden kann, wird noch die Information über die Gewichtung der einzelne Features, die die Qualitätsklasse bestimmen, von ORES benötigt. Während der Evaluation ist aber aufgefallen, dass gerade die einzelnen Features wichtig sind, um nicht nur zu verstehen „Was“ ORES berechnet hat, sondern auch „Warum“ es zu dieser Ausgabe gekommen ist.

Die einzelnen Komponenten, wie das Gadget aufgebaut ist, hat die Probanden zufrieden gestellt, allerdings könnte an kleineren Details gearbeitet werden, wie z.B. einer Legende, die das Bewertungsschema der Wikidata Redakteure erklärt. Außerdem sollte deutlich zu sehen sein, dass man den Range Slider bewegen kann. Ein Mouse Hover Effekt könnte dabei helfen.

Literaturverzeichnis

[DLH19] Mengnan Du, Ninghao Liu, and Xia Hu. *Techniques for Interpretable Machine Learning*, volume 63. Association for Computing Machinery, New York, NY, USA, December 2019.

[DVCC19] Eduardo M. Pereira Diago V. Carvalho and Jaime S. Cardoso. *Machine Learning Interpretability: A Survey on Methods und Metrics.* Electronics2019,8,832, 2019.

[ER18] Janghee Cho Emilee Rader, Kelley Cotter. *Explanations as Mechanisms for Supporting Algorithmic Transparency.* Montréal, QC, Canada, April 21–26, 2018.

[Hal17] Aaron Halfaker. *Interpolating Quality Dynamics in Wikipedia and Demonstrating the Keilana Effect.* OpenSym, Galway, Ireland, 2017.

[JBDB17] Armelle Brun Julie Bu Daher and Anne Boyer. *A Review on Explanations in Recommender Systems.* Université de Lorraine, Loria lab., Nancy, France, July 31, 2017.

[JLHR] Joseph A. Konstan Jonathan L. Herlocker and John Riedl. *Explaining Collaborative Filtering Recommendations.* Dept. of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55112 USA.

[Kim17] Finale Doshi-Velezand Been Kim. *Towards A Rigorous Science of Interpretable Machine Learning.* arXiv:1702.08608v2 [stat.ML] 2 Mar 2017, 2017.

[Kri19] Maya Krishnan. *Against Interpretability: a Critical Examination of the Interpretability Problem in Machine Learning.* D1 All Souls College, High Street, Oxford, Oxfordshire OX1 4AL, UK, 2019.

[Lip17] Zachary C. Lipton. *The Mythos of Model Interpretability.* https://dl.acm.org/citation.cfm?id=3241340, 2017.

[Mil17] Tim Miller. *Explanation in artificial intelligence: Insights from the social sciences.* arXiv Preprint arXiv:1706.07269, 2017.

[MR12] Agata Lapedriza Mireia Ribera. *Can we do better explanations? A proposal of User-Centered Explainable AI.* ACM, 2012.

[MTR16] Carlos Guestrin Marco Tulio Ribeiro, Sameer Singh. *"Why Should I Trust You?" Explaining the Predictions of Any Classifier*. DKDD 2016 San Francisco, CA, USA, 2016.

[Pik19] Machine-learning: So verstehen sie die black-box, 01.03.2019.

[SP12] Wei Zhou Sjouke Mauw Selwyn Piramuthu, Gaurav Kapoor. *Input online review data and related bias in recommender systems*. Decision Support Systems 53 (2012) 418–424, 2012.

[SWR18] Brent Mittelstadt Sandra Wachter and Chris Russell. *Counterfactual Explanations Without Opening the Black Box: Automated*. Decisions and the GDPR. Harvard Journal of Law Technology, 2018.

Anhang

Einverständniserklärung

Einverständniserklärung zur Durchführung von Audio-Video-Aufzeichnungen und der Verarbeitung personenbezogener Daten im Rahmen der Durchführung von Usability-Test.

Hiermit erteile ich meine Einwilligung, dass im Rahmen eines Usability-Tests die von mir aufgezeichnete Audio-Video Aufzeichnungen für die Evaluation der Bachelorarbeit von Sajeera Gnanasegaram verarbeitet werden dürfen.

Im Zuge dessen werden folgende personenbezogene Daten verarbeitet:

- Biographische Daten (z.B. Ausbildung, Beruf)
- Inhaltsdaten aus den audio-visuellen Aufzeichnungen (z.B. die aufgerufene Webseite, das gesprochene Wort)
- Daten, die zum Nachweis Ihrer Einwilligung in die Datenverarbeitung (z.B. diese Einwilligungserklärung)

Die vorbenannten personenbezogenen Daten werden ausschließlich zu Forschungszwecken verwendet.

Diese Einwilligung ist freiwillig und ich kann sie ohne Angabe von Gründen verweigern, ohne dass ich deswegen Nachteile zu befürchten hätte. Der einfachste Weg zum Widerruf ist eine formlose E-Mail an sajeera@zedat.fu-berlin.de.

Berlin, den 31.08.2020	Jiane Linke
Ort, Datum	Unterschrift

Einverständniserklärung

Einverständniserklärung zur Durchführung von Audio-Video-Aufzeichnungen und der Verarbeitung personenbezogener Daten im Rahmen der Durchführung von Usability-Test.

Hiermit erteile ich meine Einwilligung, dass im Rahmen eines Usability-Tests die von mir aufgezeichnete Audio-Video Aufzeichnungen für die Evaluation der Bachelorarbeit von Sajeera Gnanasegaram verarbeitet werden dürfen.

Im Zuge dessen werden folgende personenbezogene Daten verarbeitet:

- Biographische Daten (z.B. Ausbildung, Beruf)
- Inhaltsdaten aus den audio-visuellen Aufzeichnungen (z.B. die aufgerufene Webseite, das gesprochene Wort)
- Daten, die zum Nachweis Ihrer Einwilligung in die Datenverarbeitung (z.B. diese Einwilligungserklärung)

Die vorbenannten personenbezogenen Daten werden ausschließlich zu Forschungszwecken verwendet.

Diese Einwilligung ist freiwillig und ich kann sie ohne Angabe von Gründen verweigern, ohne dass ich deswegen Nachteile zu befürchten hätte. Der einfachste Weg zum Widerruf ist eine formlose E-Mail an sajeera@zedat.fu-berlin.de.

Berlin, den 31.08.2020 _____

Ort, Datum Unterschrift

Einverständniserklärung

Einverständniserklärung zur Durchführung von Audio-Video-Aufzeichnungen und der Verarbeitung personenbezogener Daten im Rahmen der Durchführung von Usability-Test.

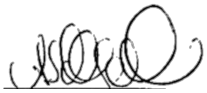
Hiermit erteile ich meine Einwilligung, dass im Rahmen eines Usability-Tests die von mir aufgezeichnete Audio-Video Aufzeichnungen für die Evaluation der Bachelorarbeit von Sajeera Gnanasegaram verarbeitet werden dürfen.

Im Zuge dessen werden folgende personenbezogene Daten verarbeitet:

- Biographische Daten (z.B. Ausbildung, Beruf)
- Inhaltsdaten aus den audio-visuellen Aufzeichnungen (z.B. die aufgerufene Webseite, das gesprochene Wort)
- Daten, die zum Nachweis Ihrer Einwilligung in die Datenverarbeitung (z.B. diese Einwilligungserklärung)

Die vorbenannten personenbezogenen Daten werden ausschließlich zu Forschungszwecken verwendet.

Diese Einwilligung ist freiwillig und ich kann sie ohne Angabe von Gründen verweigern, ohne dass ich deswegen Nachteile zu befürchten hätte. Der einfachste Weg zum Widerruf ist eine formlose E-Mail an sajeera@zedat.fu-berlin.de.



_____	_____
Ort, Datum	Unterschrift