Kann man verstehen, wie intelligente Algorithmen entscheiden?

Ist künstliche Intelligenz gefährlich?

Seminararbeit

von

Andreas Haller

Universität Heidelberg

Fachbereich Angewandte Informatik

Heidelberg, 19. Juni 2017

Inhaltsverzeichnis

1 Vertrauen in Entscheidungen

Wie treffen Menschen und der Computer Entscheidungen und wann können wir diesen Entscheidungen trauen?

1.1 Wie treffen Menschen Entscheidungen?

Dieser Abschnitt beruht auf [Gazzaniga(2005)].
Entscheidungen werden beim Menschen instinktiv getroffen, als Summe von Sinneseindrücken und Erfahrungen. Dabei werden die einzelnen Sinneseindrücke, hören, sehen, riechen, etc. unabhängig voneinander verarbeitet. Damit sich aus diesen eine Einheit ergibt und es nicht nur eine Zusammenführung von einzelnen Eindrücken ist, verarbeitet die linke Gehirnhälfte diese zu einer Geschichte. Diese Geschichte wird vom Interpreter der linken Gehirnhälfte so gedeutet, dass diese nicht unserer Persönlichkeit oder unseren Erfahrungen widerspricht.
Zum Beispiel wurde einer Frau in der rechten Gehirnhälfte, im Zentrum für Bewegung, der Befehl „Gehen“ gegeben. Der Interpreter der linken Gehirnhälfte kennt diese Information nicht und so antwortet die Frau auf die Fragen, warum sie denn gerade gehen würde mit: „Weil ich mir eine Cola gehen hole“.

1.2 Wie treffen Computer Entscheidungen?

Neuronale Netze treffen ihre Entscheidungen auch aufgrund von Erfahrungen aus dem Training-Set und den aktuellen Sinneswahrnehmungen, also dem Input.
Der Mensch nimmt viele Dinge unterbewusst wahr, welche mit in die Entscheidung einfließen. Bei einem neuronalen Netz sind alle Inputs explizit vom Menschen vorgegeben. Sein nun ein Bild ein Input, so kann das neuronale Netz alle möglichen Informationen aus diesem Bild extrahieren, selbst wenn es für den Menschen als nicht wichtig erachtet wird.

Damit die Entscheidung eines Menschen richtig angenommen wird, vertrauen wir dieser Person und akzeptieren die Entscheidung einfach so, oder aber wir erwarten eine Begründung, welche für uns individuell plausible klingen muss.

1.3 Vertrauen und Interpretierbarkeit

Dieser Abschnitt beruht auf [Lipton(2016)].
Wie kann man nun also einer Prognose von einem Programm vertrauen? Können wir Vertrauen ansehen als

- die Performance des Modells,

- die Robustheit oder,
- dass das zugrundeliegende Modell wird verstanden?

Dabei kann man noch zwischen Vertrauen in eine einzelne Prognose und in das ganze Modell unterscheiden. Es gibt zweierlei Modelle. Die erste Art sind folgende Modelle: Der Mensch hat für gewisse Instanzen Probleme bei der richtigen Klassifikation. Genau für diese Instanzen schneidet das Modell auch schlecht ab. Die zweite Art hat Probleme bei anderen Instanzen, sodass ein Zutun des Menschen die Genauigkeit der Prognose noch verbessern würde. Daher wird der ersten Art von Modellen mehr vertraut als der zweiten. Viele Entwickler und Anwender von maschinellem Lernen nennen Interpretierbarkeit als Grundlage für Vertrauen. Aber was bedeutet nun wiederum Interpretierbarkeit?
Interpretationen sollen nützliche Informationen jeglicher Art übermitteln. Dabei haben sich zwei Arten der Interpretierbarkeit manifestiert:

1.3.1 Transparenz

Unter Interpretierbarkeit kann man die Verständlichkeit des Modells verstehen. Ist das Modell transparent oder eine Black-Box?
Unter transparent kann man wiederum folgendes verstehen:

- Konvergenz, eine einzige Lösung bzw. Oberfläche des Errors (in neuronalen Netzen nicht vorhanden)
- Repräsentation von Parametern wird verstanden
- Kann es von einem Menschen komplett nachvollzogen werden?
- Kann eine Prognose von einem Menschen in annehmbarer Zeit wiederholt werden?

Ist das Modell transparent, können wir von einem interpretierbaren Modell sprechen. Aber gilt für uns ein Algorithmus noch als intelligent, wenn er transparent ist? Man darf sich aber nicht nur auf Transparenz verlassen, denn sonst verhindert das Verlangen nach Transparenz den Fortschritt neuer, nicht transparenter Technologien.

1.3.2 Entscheidungen erklären

Eine andere Definition von Interpretierbarkeit ist nachträglich (post-hoc) die Entscheidungen erklären zu können, z. B. durch

- Erklärungen in natürlicher Sprache & Bildunterschriften
  Dies beinhaltet auch den Vergleich mit bekannten Dingen („Sieht aus wie ...“)
- Visualisierung von gelernten Repräsentationen oder Modellen (auch auf welche Bildbereiche besonderer Fokus gelegt wird)

Ich lege hier meinen weiteren Fokus auf die nachträglichen Erklärungen von Modellen. Erklärungen wie Modelle funktionieren, kann man in den Machine Learning Vorlesungen und in vielen Büchern erhalten. In vielen Fällen sind Modelle aufgrund ihrer Komplexität

einfach nicht mehr transparent, wodurch Erklärungen zum Vertrauen der Vorhersagen und der Modelle vonnöten sind.
Zum Beispiel sind lineare Modelle, wie Entscheidungsbäume, algorithmisch transparenter als neuronale Netze. Um aber die gleiche Performance in linearen Modellen zu erhalten, müssen die Features manuell angepasst werden, wodurch lineare Modelle durchaus komplexer und unverständlicher werden können.
Beispiele für Erklärungen im Nachhinein sind in Kapitel 3 zu finden.

2 Gefahren in maschinellen Entscheidungsprozessen

Schwerwiegende Probleme können entstehen, wenn man nur Prognosen und keine Begründungen als Grundlage für eine Entscheidung hat.

2.1 Grenzen Maschinellem Lernens

Dieser Abschnitt beruht auf [Ribeiro et al.(2016)] und [Lipton(2016)].

- Maschinelles Lernen ist der Versuch Lösungen zu Fragen aus der Realität durch Minimierung des Fehlers zu erhalten. Damit kann man auch nur sehr einfache Korrelationen erhalten, anstatt die komplexe Realität abzubilden.

- Der vorhandene Datensatz und die Realität weichen in der Regel voneinander ab, teils mehr, teils weniger stark. Wie gut ein Modell verallgemeinert, messen wir durch die Differenz der Genauigkeit von Training und Test-Set. Dabei werden diese beide zufällig aus derselben Distribution entnommen.

- Viele Entwickler geben ihren Algorithmen übermäßiges Vertrauen bzgl. ihrer Performance auf dem Validierungs-Set im Vergleich dazu wie der Algorithmus auf neue Daten abschneiden würde.

- Wenn man einen Klassifikator hat, welcher die höchste Genauigkeit auf Training- und Test-Set hat, weiß man noch nicht, warum dieser so gut ist.
  Erstes Beispiel: In einem Datensatz korrelierten die Patienten ID stark mit der Zielklasse aus dem Trainings- und Test-Set. Diese Korrelation aus den Rohdaten und der Prognose herauszufinden ist extrem schwer. Hier können Erklärungen deutliche Hilfestellungen bieten.
  Zweites Beispiel: Hierbei geht es darum, festzustellen, ob ein Text christlichen Inhalts oder atheistischen Inhalts ist. Der Algorithmus 2 (vgl. Abbildung 2.1) hat eine höhere Genauigkeit auf dem Trainings- und Test-Set. Durch Erklärungen erkennt man aber schnell, dass dieser Algorithmus keine relevanten Features verwendet und der Algorithmus auf einem unbekannten Datensatz erheblich schlechter abschneiden würde. Ohne Erklärungen muss man sich aber auf die Genauigkeit verlassen, welche nur ein Indikator von vielen ist.

- Man muss genau aufpassen, für welchen Zweck man ein Test-Set verwendet. Denn es kann sein, dass das Ergebnis dazu verwendet wird, das Modell zu negieren. Ein Beispiel hierzu: Die Wahrscheinlichkeit, ob ein Mensch an einer Lungenentzündung stirbt, wurde mit einem Modell wahrgenommen. Dabei hatten Menschen mit Asthma eine geringere Sterbewahrscheinlichkeit. Dies lag aber daran, dass sie aufgrund

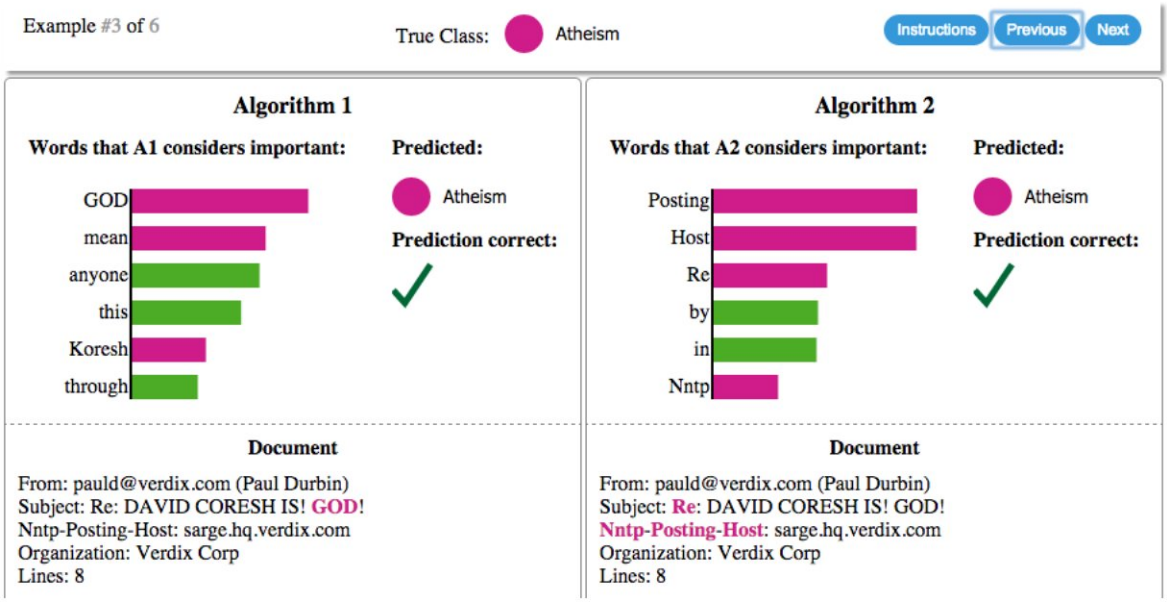


Abbildung 2.1: Verschiedene Klassifikatoren benutzen unterschiedliche Wörter für ihre Prognosen. Grün ist für die Kategorisierung eines Textes als christlich, rot für atheistisch. Die Balken geben die Wichtigkeit für die Klassifikation an. Entnommen aus [Ribeiro et al.(2016)]

ihres Asthmas eine aggressivere Behandlung erhielten. Würde nun das Ergebnis dieser Sterbewahrscheinlichkeiten dazu verwendet, um die Intensität der Behandlung von Lungenentzündungen zu steuern, würden an Asthma Erkrankte eine schwächere Medikation erhalten. Dies würde zu einer übermäßigen Sterberate von Asthmatikern führen, womit sich das eingesetzte Modell selbst widerlegen würde. Daher muss man sehr genau aufpassen, welche Korrelationen erkannt werden und wie diese miteinander verknüpft werden. Erklärungen und Hintergrundwissen können hier helfen, um ein Modell richtig zu stellen und Fehler aufzudecken bzw. dass Ergebnisse bestimmter Modelle nur mit diesen Informationen weiterverwendet werden dürfen.

2.2 Daraus entstehende Diskriminierung

Dieser Abschnitt beruht auf [Goodman and Flaxman(2016)] und [Lipton(2016)].

- Problem der Datensätze:
  In den meisten Fällen beruhen die Datensätze auf sozialen Daten. Soziale Daten beinhalten automatisch Ungleichheiten, Ausschluss oder andere Arten von Diskriminierung. Diese Muster der Diskriminierung werden dann auch in neuen Datensätzen reproduziert, da ein guter Klassifikator diese extrahiert und zur Kategorisierung nutzt.

- Lösung: Bearbeiten von Datensätzen:
  - Weglöschen von sensiblen Daten, die direkte Rückschlüsse auf z. B. die Rasse und Einkommensverhältnisse schließen lassen. Es bleiben Variablen übrig, wel-

che direkt mit sensiblen Daten korrelieren. Durch dieses Verfahren verringert man keine Diskriminierung.

  - Alle Variablen löschen, welche auch mit sensiblen Daten korrelieren. Dabei muss man aber auch z. B. die PLZ weglöschen, da bestimmte Gebiete Ghettos oder Armenviertel sind. Am Schluss erhält man einen Datensatz, auf dem kein Klassifikator sinnvoll arbeiten kann. Beispiele für weitgreifende Korrelationen:
    * Bei genügend großen Datensätzen korreliert die IP-Adresse mit der eigenen Rasse.
    * Es ist möglich durch Drittanbierterinformationen, wie der Kaufhistorie, einen ähnlich guten Gesundheitszustand festzustellen, wie wenn man sich beim Arzt untersuchen lässt. Dies sind interessante Informationen z. B. für Versicherungsunternehmen.

- Lösung: Repräsentation von Randgruppen erhöhen:
  Sind Gruppen in einem Datensatz nur sehr gering vertreten, steigt die Unsicherheit der Vorhersage für diese Gruppen. Dies kann bei der Vergabe von Krediten essentiell sein, so dass diese mit einer geringeren Wahrscheinlichkeit an Randgruppen vergeben werden, obwohl die Angehörigen solch einer Randgruppe den Kredit gleich wahrscheinlich zurück zahlen können, wie die Hauptgruppe (vgl. Abbildung 2.2).

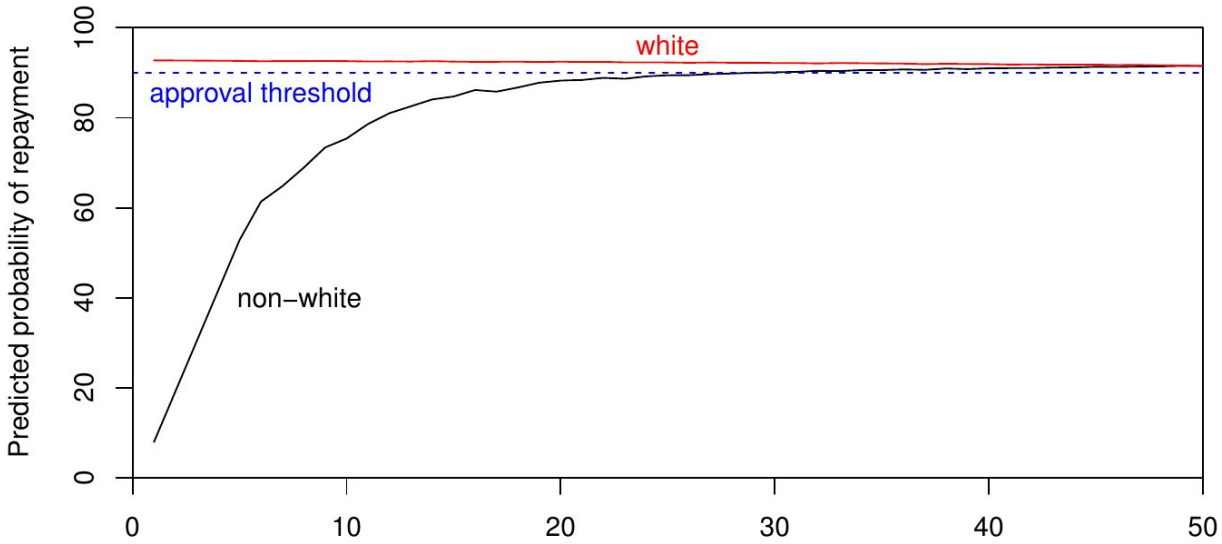


Abbildung 2.2: Prozentualer Anteil der nicht-weißen Bevölkerung; Weiße und nicht-weiße Menschen zahlen jeweils gleich wahrscheinlich einen Kredit zurück. Ein Risiko abgeneigter Algorithmus benötigt 95% Überzeugung, dass ein Kredit zu mindestens 90% zurückgezahlt wird, um einen Kredit zu verteilen. Ist ein Bevölkerungsteil unter 30% vertreten, so würde dieser Algorithmus aufgrund von zu hohen Unsicherheiten diesen Gruppen keinen Kredit geben. Entnommen aus [Goodman and Flaxman(2016)]

Deshalb ist die Wahl des Datensatzes essentiell, sowie die Begründungen, warum eine Entscheidung getroffen wurde, um Diskriminierung zu minimieren.
Folgende Beispiele motivieren zur Vorsicht:

- Maschinelles Lernen wird bereits genutzt, um die Kriminalitätsraten durch gezielte Polizeipräsenz zu verringern. Diese führt auch zu einer geringeren Kriminalität. Dabei wird aber Diskriminierung in den Trainingsdaten nicht beachtet, so dass es zu einer Einkerkerung mancher Gebiete durch ein Überaufkommen von Polizei kommt. Die gefühlte Freiheit geht verloren.
- Man darf den nachträglichen Erklärungen aber auch nicht blind vertrauen, denn diese können so modelliert sein, dass sie plausible Erklärungen geben aber irreführend sind, indem sie von Diskriminierung weg täuschen.

3 Erklärungen zu Entscheidungen

Dieser Absatz beruht auf [Gazzaniga(2005)].
Wir möchten nun komplizierte Sachverhalte von komplexen Algorithmen lösen lassen. Ein Algorithmus mit einfacher Grundstruktur wird durch die komplexen Anforderungen nicht mehr nachvollziehbar und verliert seine Transparenz. Man kann den Entscheidungsprozess nicht nachvollziehen und hat deshalb kein Vertrauen in die Prognosen und benutzen den Algorithmus nicht. Dabei kann Angst über die Auswirkungen von falschen Entscheidungen aufgrund fehlender Erklärungen in den Vordergrund treten. Auch deshalb und um die zuvor genannten Gefahren möglichst gering zu halten, müssen gute Erklärungen für die Prognosen vorhanden sein. Wie diese erstellt werden, sollte nachvollziehbar sein.

3.1 LIME – Local Interpretable Model-agnostic Explanations

Dieser Abschnitt beruht auf [Ribeiro et al.(2016)].
Ziel dieses Abschnittes ist es eine lokale Approximation des Models zu finden, welche interpretierbar ist und gut approximiert. Eine Instanz wird in ihrer Umgebung betrachtet und das Modell wird an diesen Stellen ausgewertet. Nun wird ein einfacheres, z. B. lineares Modell gesucht, welches die Instanz und ihre Umgebung bezüglich des Ursprungsmodells gut approximiert. Dabei muss das neue Modell möglichst einfach bleiben.

Formeln:

Black-Box-Modell (siehe Bild 3.1)	$f : \mathbb{R}^d \to \mathbb{R}$
Approximationsmodell	$g : [0,1]^{d'} \to \mathbb{R}$
(1=interpretierbare Komponente vorhanden, 0=nicht vorhanden)	
g ist aus interpretierbaren Modellen G	
Komplexität von g	Ω
Instanz	x
Abstandsfunktion für Umgebung von x	Π_x
lokaler Loss	$\mathcal{L}(f, g, \Pi_x)$
(wie unglaubwürdig g das Modell f mit der Metrix Π_x in der Umgebung von x approximiert)	

Um eine lokal gute und interpretierbare Approximation zu erhalten muss der *Loss* und die Komplexität, wie in Gleichung 3.1 minimiert werden:

$$\xi(x) = \arg \min_{g \in G} \mathcal{L}(f, g, \Pi_x) + \Omega(g) \tag{3.1}$$

Vorgehensweise zur Berechnung des *Loss*es:
Man sucht sich eine Instanz x heraus, zu dessen Prognose man eine Begründung haben

möchte. Dann transformiert man sie vom Definitionsbereich von f, D(f) zu D(g): $x \rightarrow x'$. Dann wird um x' gleichförmig gesamplet. Diese z' werden in den Definitionsbereich von f zurück transformiert. Der Loss kann dann z. B. der lokal gewichtete quadratische Loss sein:

$$\mathcal{L}(f, g, \Pi_x) = \sum_{z,z' \in Z} \Pi_x(z)(f(z) - g(z'))^2 \tag{3.2}$$

Sollen die Erklärungen in natürlicher Sprache sein, z. B. Eigenschaften die vorhanden sind oder nicht, so kann man die maximale Anzahl an Wörtern auf einen fixen Schwellenwert K setzen und erhält damit folgende Notation für die Komplexität von g

$$\Omega(g) = \begin{cases} \infty & \#\text{words} > K \\ 0 & \#\text{words} \leq K \end{cases} \tag{3.3}$$

Um solche K Wörter bestimmen zu lassen, kann man diese mit LASSO lernen lassen (setzt Koeffizienten von Features auf Null und nicht nur sehr kleine Werte). Für Bilder können anstatt von Wörtern Superpixel betrachtet werden.

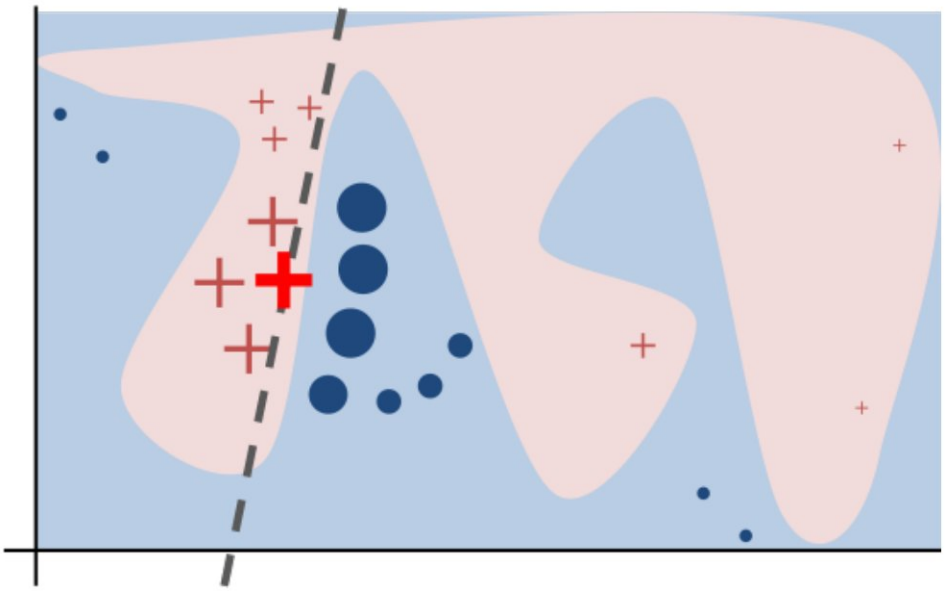


Abbildung 3.1: Die Black-Box Funktion f ist in hellblau/rosa dargestellt. Diese ist unbekannt und hochgradig nicht linear. Instanz x ist mit einem fetten roten Kreuz markiert. Die anderen Kreuze und Punkte sind aus der Umgebung von x' gesamplet und sind größenteschnisch nach ihrem Abstand zur Instanz x dimensioniert. LIME lernt durch die lokale Approximation von f die gestrichelte Linie, welche in diesem Fall lokal und nicht global gültig ist. Entnommen aus [Ribeiro et al.(2016)]

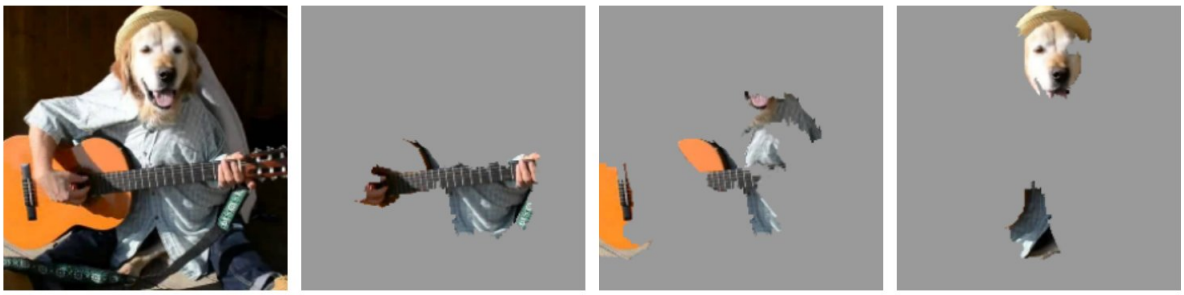
Nachteil:
Die Art der Modellfamilie G kann zu einfach sein um gute Ergebnisse zu erzielen:

- Als Beispiel können Superpixel keine Aussage darüber machen, warum ein Sepia-Bild als Retro von Modell f klassifiziert wird

- Die erstellte Erklärung kann komplett falsch sein, wenn f in der Umgebung von x nicht linear approximiert werden kann, und somit das gewählte Modell nicht geeignet ist

Vorteil:

- Man bekommt eine Erklärung, warum die Prognose auf dem Trainings-Set gut funktioniert hat und auf dem Test-Set nicht z.B. Fokus auf falsche Wörter in Textvergleich (Christen/Atheisten, siehe Abbildung 2.1)
- Welche Bildbereiche bei einer Prognose ausschlaggebend waren (siehe Abbildung 3.2).



(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Abbildung 3.2: Die Top 3 der Prognosen sind Elektische Gitarre (p=0,32), Akustische Gitarre (p=0,24) und Labrador (p=0.21). Das Griffbrett erklärt die falsche Prognose für Elektrische Gitarre. Entnommen aus [Ribeiro et al.(2016)]

Wenn nun nicht nur die Glaubwürdigkeit für eine Prognose mit einer Erklärung aufgezeigt werden soll, sondern das ganze Modell als glaubwürdig eingestuft werden soll, dann muss folgendes getan werden: Damit der User nur wenige, aber repräsentative Beispiele für dieses Modell und diesen Datensatz anschauen muss, sucht ein Algorithmus aus den Erklärungen aller Instanzen die wichtigsten Features heraus. Diese haben am meisten Einfluss auf die Prognose. Dann sollen mit möglichst wenigen Instanzen sehr viele und vor allem die wichtigsten Features abgedeckt sein. Die Anzahl der Beispiele soll dabei möglichst gering sein, damit der User nicht gelangweilt wird. Ein Beispiel dafür kann man in Abbildung 3.3 wiederfinden.

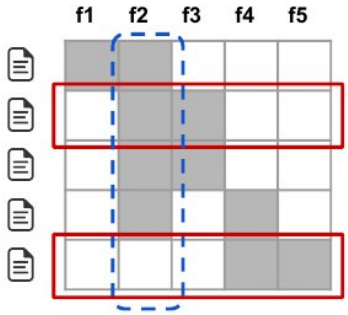


Abbildung 3.3: Beispiel um möglichst wenige, aber repräsentative Instanzen zu finden. f2 ist das wichtigste Feature und mit Instanz 2 und 5 (rot eingerahmt) bekommt man möglichst viele Features und vor allem auch das wichtige f2 abgedeckt. Entnommen aus [Ribeiro et al.(2016)]

3.2 VQA - Visual Question Answering

Hier geht es darum, eine gegebene Frage in natürlicher Sprache mit einem Bild in natürlicher Sprache zu beantworten (siehe Abbildung 3.4). Um diese Aufgabe zu lösen, wird meiner Meinung nach ein intelligenter Algorithmus benötigt.

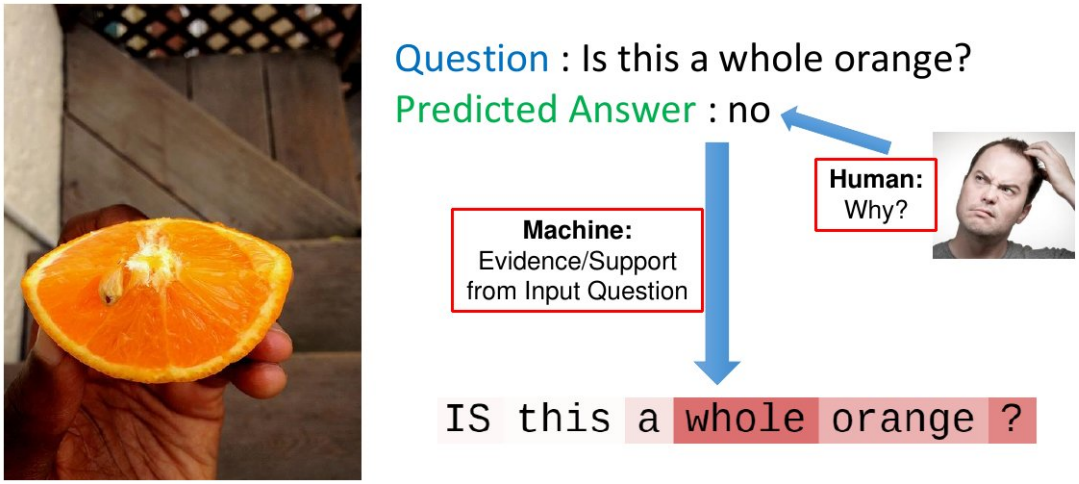


Abbildung 3.4: Exemplarische Funktionsweise von VQA. Aus einem gegebenen Bild und einer Frage soll eine Antwort generiert werden. Warum der Algorithmus zu diesem Ergebnis kommt, ist hierbei nicht immer klar. Entnommen aus [Goyal et al.(2016)]

3.2.1 Abdecken von Inputs

Dieser Abschnitt beruht auf [Goyal et al.(2016)].
Eine Herangehensweise um Erklärungen zur Prognose zu erhalten, ist das Abdecken von Inputs. Man möchte dadurch herausfinden, welche Teile der Frage bzw. des Bildes am wichtigsten für die Antwort sind.
Umsetzungen:

- Man deckt ein Wort in der Frage ab und betrachtet die Veränderung in der Wahrscheinlichkeit für diese Prognose. Dabei bleibt das Bild unverändert (beispielhaft in Abb. 3.5 dargestellt).
- Das Gleiche kann man auch für das Bild machen. Dabei bleibt die Frage unverändert und im Bild werden Superpixel mit grauen Pixeln überdeckt. Dabei setzt sich das Grau als Durchschnittsfarbwert aller Bilder im genutzten Datensatz zusammen.

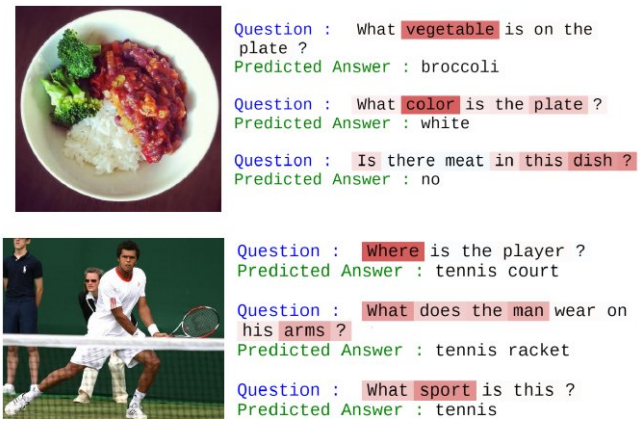


Abbildung 3.5: Die Abbildung hebt diejenigen Wörter in der Frage hervor, welche für die richtige Prognose am wichtigsten sind. Entnommen aus [Goyal et al.(2016)]

Man konnte feststellen, dass bei der Fragenstellung Wh-Wörter, Adjektive und Substantive am wichtigsten waren (siehe Abbildung 3.6).
Dies lag daran, dass die Fragen häufig nach Charakteristika des Objekts oder nach dem

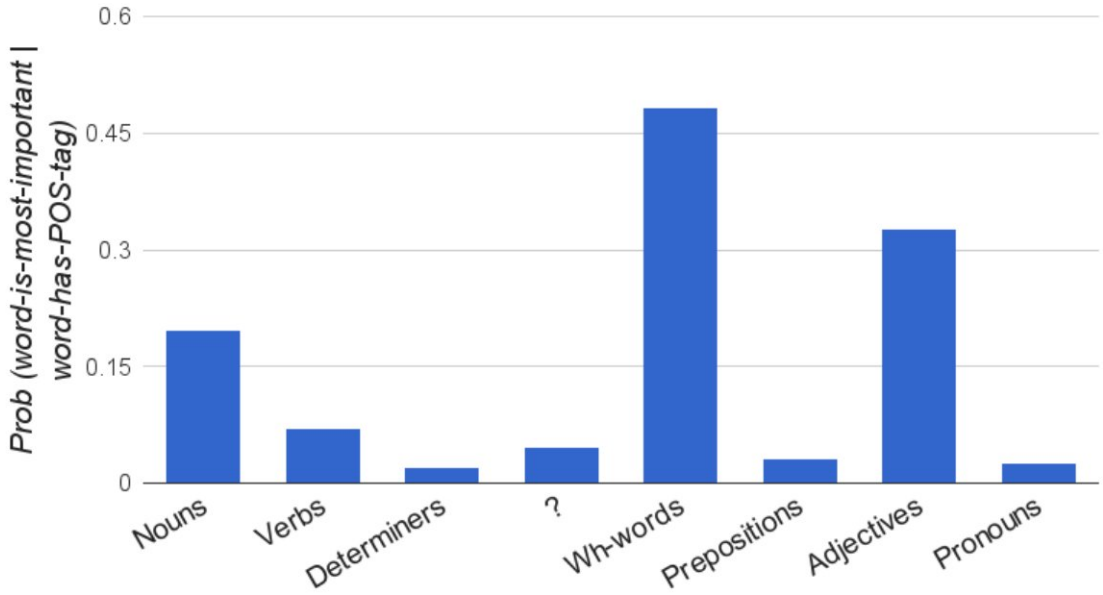


Abbildung 3.6: Wahrscheinlichkeit dafür, dass ein Wort, welches einer bestimmten Wortgruppe angehört wichtig zur Klassifizierung ist. Entnommen aus [Goyal et al.(2016)]

Objekt selbst gefragt haben.

3.2.2 Erklärungen in Sprache und mit Aufmerksamkeitsabbildungen

Dieser Abschnitt beruht auf [Park et al.(2016)].
Es wird zur Antwort noch eine Erklärung in natürlicher Sprache gegeben. Für die Prognose und die Erklärung wird jeweils eine Aufmerksamkeitsabbildung geliefert, welche die Bildbereiche hervorhebt, welche zur Prognose bzw. für die Erklärung am wichtigsten sind. Dabei bleibt zu erwähnen, dass der Fokus auf einem Bild für Menschen ein anderer ist als für ein neuronales Netz. Jedoch stimmen teilweise wichtige Bildbereiche bei neuronalen Netzen und der menschlichen Fokussierung bis zu 25 % überein.
Für manche Tätigkeiten wie z. B. Yoga, kann man an der Pose erkennen, um welche Sportart es sich handelt. Für andere Sportarten, wie Mountain Biking, braucht man ein Fahrrad und eine Umgebung um diese richtig zu klassifizieren.

Wie dieser Algorithmus von [Park et al.(2016)] umgesetzt wird, kann in Abbildung 3.7 gefunden werden.

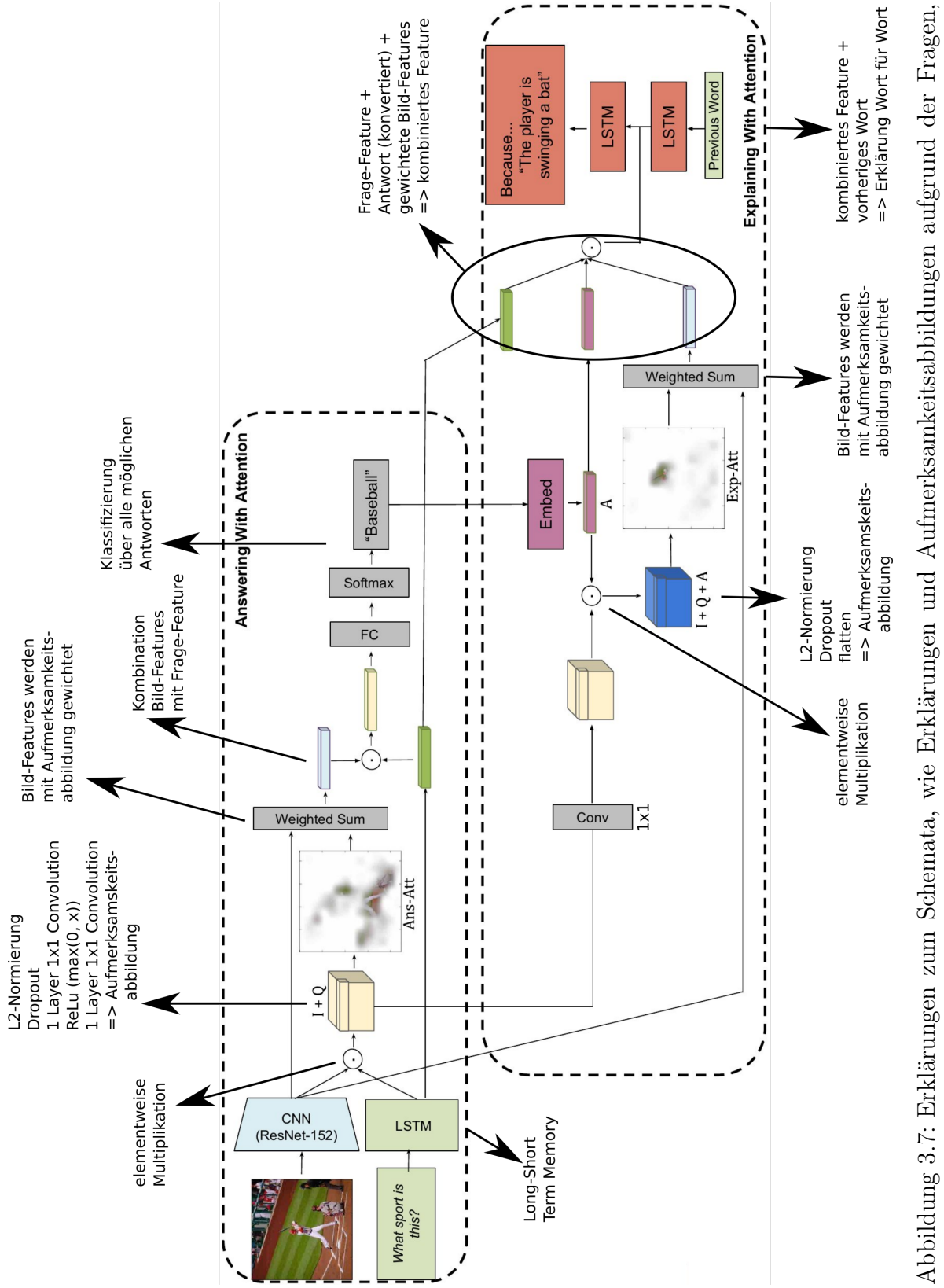
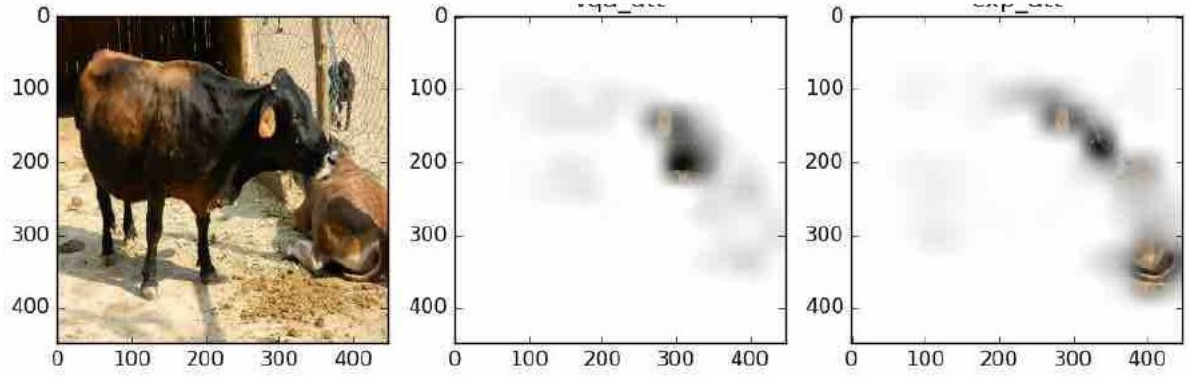


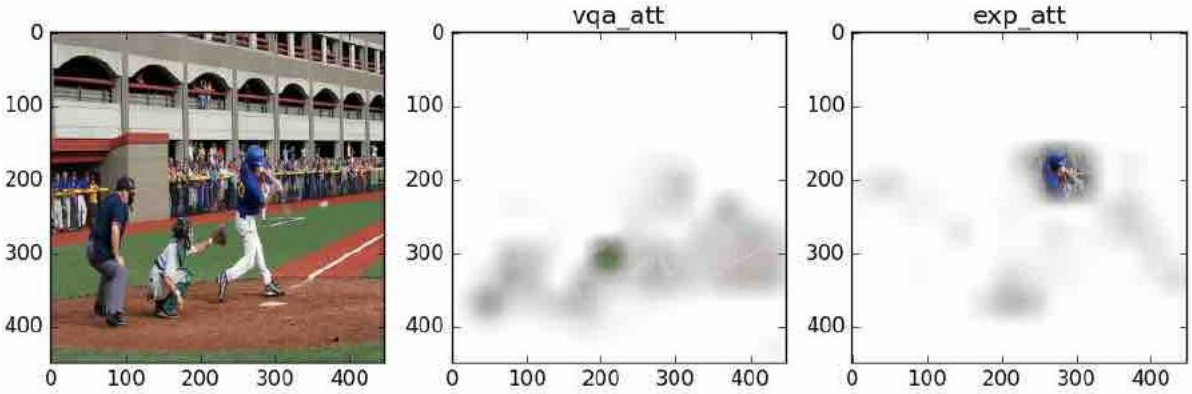
Abbildung 3.7: Erklärungen zum Schemata, wie Erklärungen und Aufmerksamkeitsabbildungen aufgrund der Fragen, Bilder und Antworten entstehen. Entnommen aus [Park et al.(2016)]

Als Ergebnis erhält man lediglich Erklärungen, welche die Prognose begründen. Ein tieferes Verständnis die internen Prozesse eines Modellen erlangt man dadurch nicht.
Der hier verwendete Datensatz legt seinen Fokus auf Bildern mit Aktivitäten. Dieser

Datensatz besteht aus 25.000 Bilder, welche aus YouTube Videos entnommen wurden (MPI Human Pose (MHP)).
In Abbildung 3.8 werden richtig beantwortete Fragen und ihre Begründungen mit den Aufmerksamkeitsabbilungen dargestellt. 3.9 zeigt falsch beantwortete Fragen. Abbildung 3.10 zeigt zwei unterschiedliche Bilder mit der gleichen Antwort zur selben Frage und unterschiedlichen Begründungen. In 3.11 werden zum selben Bild zwei unterschiedliche Fragen gestellt.

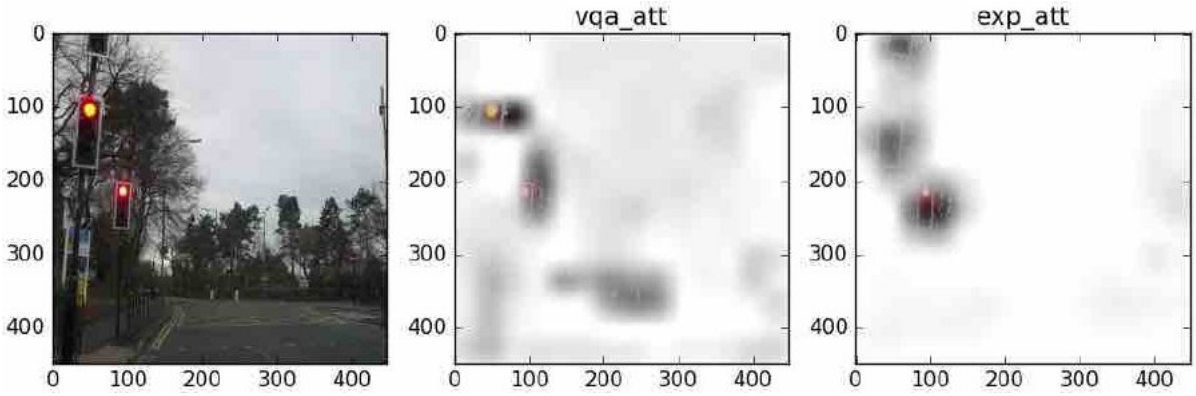


(a) Q: What kind of animal is lying on the ground?
A: Cow. (correct)
E: Because it has four legs and looks like a cow.

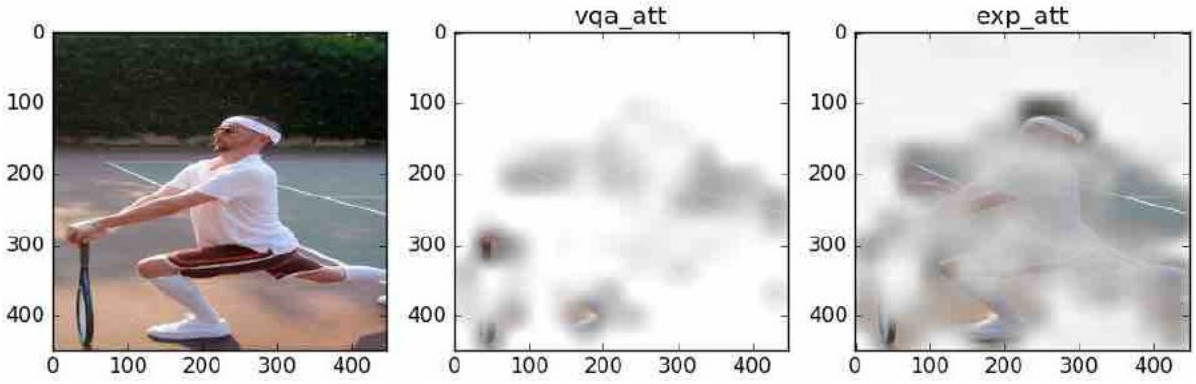


(b) Q: What game is this?
A: Baseball. (correct)
E: Because the player is holding a bat.

Abbildung 3.8: Abbildungen für korrekte Klassifikationen und deren Begründungen. Entnommen aus [Park et al.(2016)]



(a) Q: Should we stop?
A: No. (wrong: Yes)
E: Because the light is green.



(b) Q: What is the person doing?
A: Playing tennis. (wrong: Streching)
E: Because he is holding a tennis racket.

Abbildung 3.9: Abbildungen für inkorrekte Klassifikationen und deren Begründungen. Diese Begründungen können Aufschluss über Fehler im Algorithmus geben. Entnommen aus [Park et al.(2016)]

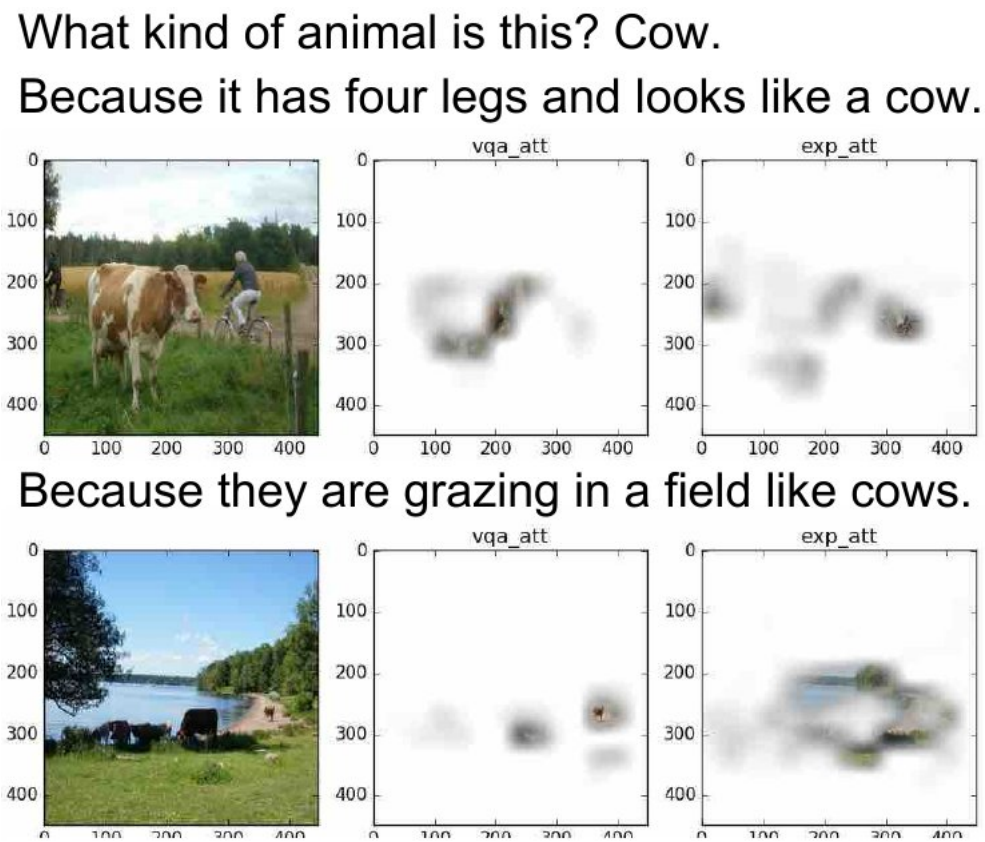


Abbildung 3.10: Zwei verschiedene Bilder mit der gleichen Antwort auf dieselbe Frage erhalten durch andere Spezifikationen auf dem Bild unterschiedliche Begründungen. Entnommen aus [Park et al.(2016)]

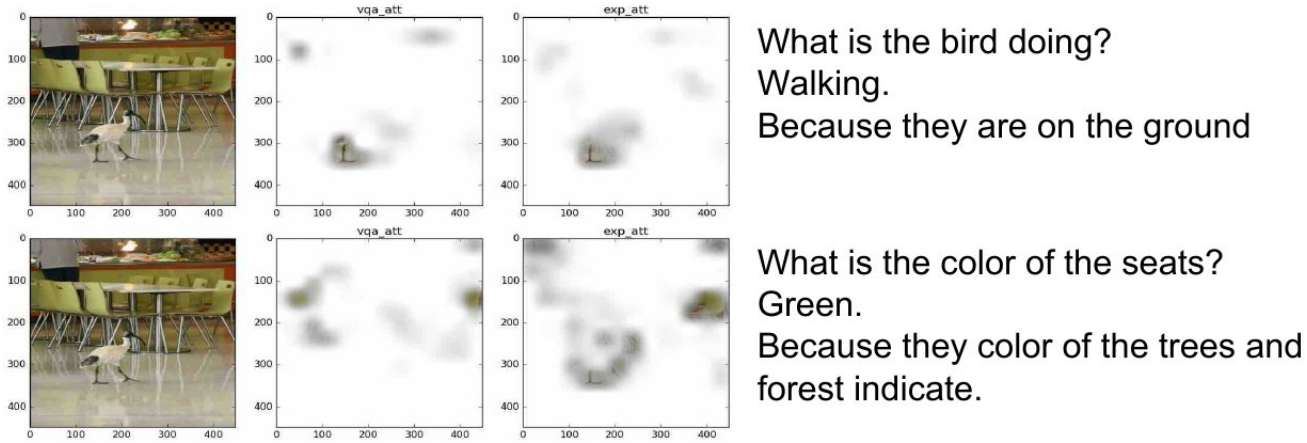


Abbildung 3.11: Zu einem Bild werden zwei unterschiedliche Fragen gestellt, um unterschiedliche Fokusse festzustellen. Entnommen aus [Park et al.(2016)]

3.2.3 Verbesserungen von neuronalen Netzen aufgrund von Erklärungen

Dadurch dass man eine Erklärung bekommt, kann man auf falsche Rückschlüsse im neuronalen Netz schließen. Wie kann man neuronale Netze nun anhand von falschen Klassifizierungen und deren Erklärungen verbessern:

1. Man kann die Datensätze erweitern bzw. verbessern. Man kann z. B. mehrere Instanzen von ähnlichen, aber falsch klassifizierten Instanzen hinzufügen. Damit reduziert man auch die größere Unsicherheit bei Randgruppen.

2. Man stellt fest, dass der verwendet Klassifikator für diesen Datensatz nicht geeignet ist. Man wechselt demnach den Klassifikator oder das gesamte Modell.

3. Man stellt fest, dass der Datensatz nicht geeignet ist, um für seinen Einsatzzweck gute Ergebnisse zu liefern und entscheidet sich, diesen zu ersetzen.

4. Man nutzt die Begründung nur als Unterstützung im Entscheidungsprozess, welcher weiterhin voll und ganz dem Menschen überlassen wird. Denn oft ist der Fokus des neuronalen Netzes gut, nur dessen Schlussfolgerung ist falsch.

4 Täuschung neuronaler Netze - universelle Störungen

Dieser Abschnitt beruht auf [Moosavi-Dezfooli et al.(2016)].

Es besteht aber noch das Problem der absichtlichen Täuschung von solchen neuronalen Netzen bzw. von hoch-dimensionalen Algorithmen.
Wollte man ein Bild durch ein neuronales Netz absichtlich falsch klassifiziert haben, so konnte man zu einem bestimmten Bild ein spezielles Rauschen erstellen, welches durch Lösen eines Optimierungsproblems gefunden wurde. Addiert man diese Störung zum Bild, so wurden hohe Falschklassifizierungsraten erzielt. Der Prozess ist jedoch individuell für unterschiedliche Bilder und daher sehr aufwändig.

4.1 Implementierung und Ergebnisse

Mit dem folgenden Algorithmus kann man eine universelle Störung generieren, welche bei allen Bildern für eine Netzwerkarchitektur zu hohen Klassifizierungsfehlern führt.

Algorithmus 4.1.1 : Computation of universal perturbations from [Moosavi-Dezfooli et al.(2016)]

1 **Data :** Data points X,
classifier $\hat{k}$,
desired l_p norm of the perturbation ξ,
desired accuracy on perturbed samples δ
2 **Result :** Universal perturbation vector v
3 Initialize $v \leftarrow 0$.
4 **while** $Err(X_v) \leq 1 - \delta$ **do**
5 **for** x_i *in* X **do**
6 **if** $\hat{k}(x_i + v) = \hat{k}(x_i)$ **then**
7 $\Delta v_i \leftarrow \arg\min_r ||r||_2$ s.t. $\hat{k}(x_i + v + r) \neq \hat{k}(x_i)$
8 $v \leftarrow \arg\min_{v'} ||v + \Delta v_i - v'||_2$ subject to $||v'||_p \leq \xi$
 end
 end
end

Zeile 4: Verbessere solange die universelle Störung, bis genug Bilder falsch klassifiziert werden.
Zeile 6: Wenn die vorhandene Störung für diese Instanz nicht für eine Fehlklassifikation ausreicht, dann verändere die Störung.

Zeile 7: Verändere die vorhandene Störung v um das minimalste r, sodass $x + v + r$ falsch klassifiziert wird, also dass $x + v + r$ über die nächste Entscheidungsgrenze gebracht wird.
Zeile 8: Die neue Störung wird auf den durch die l_p Norm aufgespannten Ball mit Radius ξ projiziert und auf Null zentriert. Dies bezweckt mitunter, dass die Störung nicht zu groß wird (Werte für ξ aus [Moosavi-Dezfooli et al.(2016)]: $\xi = 2000$ für l_2 und $\xi = 10$ für l_∞).

Wird die Reihenfolge der Bilder zur Erstellung einer Störung verändert, so entsteht eine neue, universelle Störung. Diese besitzen zwar ähnliche Muster, aber das normalisierte innere Produkt zweier unterschiedlicher, universeller Störungen blieb nach [Moosavi-Dezfooli et al.(2016)] dabei immer unter 0,1 (vgl. Abbildung 4.1).

Abbildung 4.1: Illustration verschiedener, universeller Störungen für ein und dasselbe neuronale Netz. Entnommen aus [Moosavi-Dezfooli et al.(2016)]

Genutzt wurde hier der Datensatz ILSVRC 2012 mit 50.000 Bildern. Zur Erstellung wurden 10.000 Bilder genommen, mit ca. 10 Bildern pro Klasse und einem Validationsset von den restlichen 40.000 Bildern. Tabelle 4.2 beinhaltet die Missklassifikationsraten unterschiedlicher neuronaler Netze. Aber auch mit nur 500 Bilder (1000 verschiedene Klassen),

		CaffeNet [8]	VGG-F [2]	VGG-16 [17]	VGG-19 [17]	GoogLeNet [18]	ResNet-152 [6]
ℓ_2	X	85.4%	85.9%	90.7%	86.9%	82.9%	89.7%
	Val.	85.6	87.0%	90.3%	84.5%	82.0%	88.5%
ℓ_∞	X	93.1%	93.8%	78.5%	77.8%	80.8%	85.4%
	Val.	93.3%	93.7%	78.3%	77.8%	78.9%	84.0%

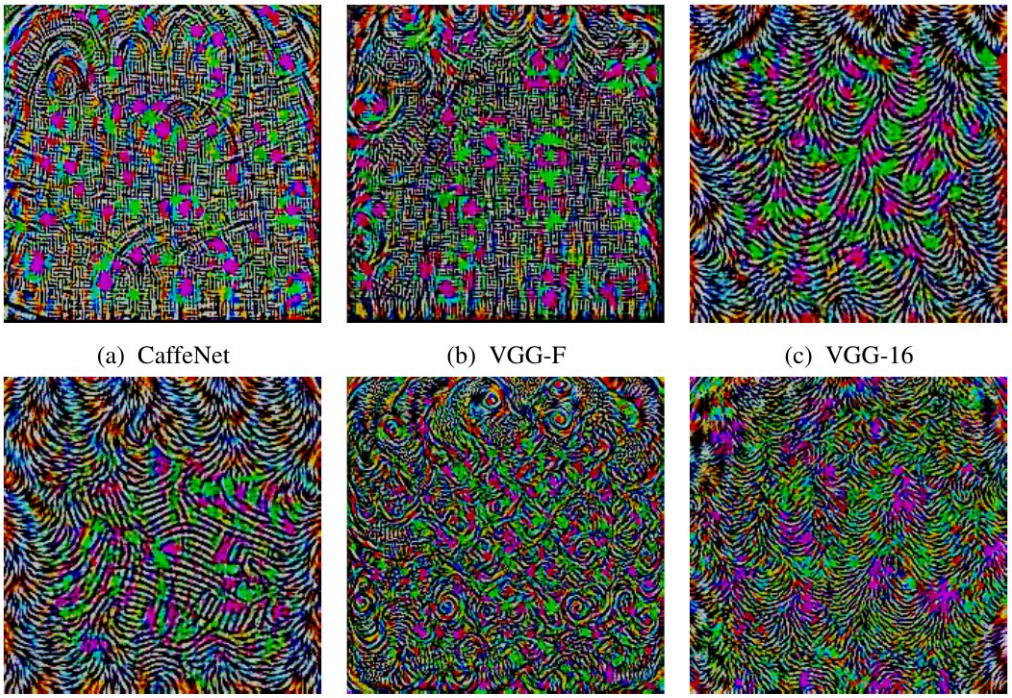
Abbildung 4.2: Die Missklassifikationsraten aus unterschiedlichen neuronalen Netzen und unter verschiedenen Normen liegen zwischen 77,8% und 93,7%. Diese sind sehr hoch. Entnommen aus [Moosavi-Dezfooli et al.(2016)]

wurden 30 % der restlichen 49.500 Bilder falsch klassifiziert. Man kann hier eine hohe Generalisierung feststellen, da die universelle Störung auch für komplett ungesehene Bilder ansehnlich funktioniert.
Universale Störungen können bei anderen Netzwerken auch noch sehr gut funktionieren (vgl. Tabelle 4.3).

	VGG-F	CaffeNet	GoogLeNet	VGG-16	VGG-19	ResNet-152
VGG-F	**93.7%**	71.8%	48.4%	42.1%	42.1%	47.4 %
CaffeNet	74.0%	**93.3%**	47.7%	39.9%	39.9%	48.0%
GoogLeNet	46.2%	43.8%	**78.9%**	39.2%	39.8%	45.5%
VGG-16	63.4%	55.8%	56.5%	**78.3%**	73.1%	63.4%
VGG-19	64.0%	57.2%	53.6%	73.5%	**77.8%**	58.0%
ResNet-152	46.3%	46.3%	50.5%	47.0%	45.5%	**84.0%**

Abbildung 4.3: Universelle Störungen eines neuronalen Netzes funktionieren teilweise auch sehr effektiv auf anderen neuronalen Netzen, obwohl die Architektur nicht bekannt ist. Entnommen aus [Moosavi-Dezfooli et al.(2016)]

In Abbildung 4.4 sind universelle Störungen verschiedener neuronaler Netze abgebildet.



(a) CaffeNet (b) VGG-F (c) VGG-16

(d) VGG-19 (e) GoogLeNet (f) ResNet-152

Abbildung 4.4: Universelle Störungen zu verschiedenen neuronalen Netzen. Entnommen aus [Moosavi-Dezfooli et al.(2016)]

Die universellen Störungen sind nicht nur universell bezüglich der Bilder eines Datensatzes, sondern auch stückweise universell gegenüber der eingesetzten Netzwerkarchitektur.

4.2 Funktionsweise

Einfach gesprochen, werden viele Instanzen zu einigen, wenigen Labels falsch klassifiziert. Man kann daraus schlussfolgern, dass diese dominanten Labels wahrscheinlich eine größere Fläche im Bildraum einnehmen.
Mathematisch betrachtet, sieht das folgendermaßen aus:
Ein binärer Klassifikator hat nur einen Normalvektor zur Entscheidungsgrenze, da alle

anderen Vektoren kollinear (parallel/antiparallel) sind.
Sei nun N die Matrix aus Normalvektoren r zur Entscheidungsgrenze in der Umgebung von n Datenpunkten $(x_1, \ldots, x_n)$ im Validierungsset für einen Klassifikator mit n Klassen.

$$N = \left[\frac{r(x_1)}{||r(x_1)||_2} \cdots \frac{r(x_n)}{||r(x_n)||_2} \right] \tag{4.1}$$

Eine Singulärwertzerlegung (vergleichbar mit einer Eigenwertzerlegung einer quadratischen Matrix) kennzeichnet Längenänderung und Richtung im Bildraum durch die lineare Transformation, welche durch die Matrix N bezüglich eines Orthonormalsystems im Urbildraum induziert wird. Die Singulärwerte nehmen sehr schnell ab im Vergleich zu Singulärwerten einer Matrix, welche aus zufällig und gleichförmig ausgewählten Vektoren aus dem Einheitskreis stammen (vgl. Abbildung 4.5).

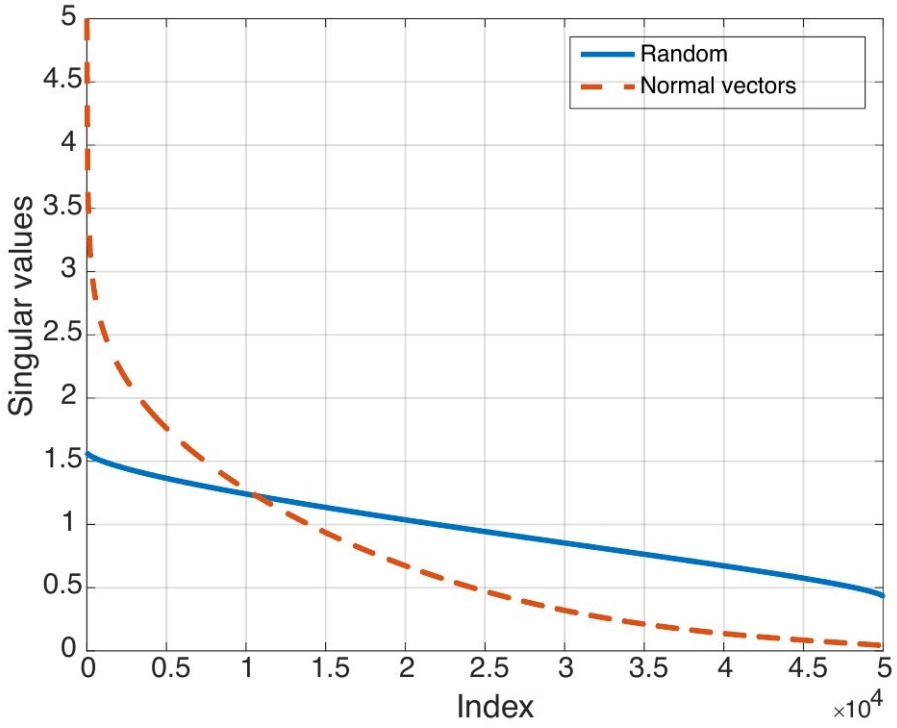


Abbildung 4.5: Singulärwerte der Matrix N. Entnommen aus [Moosavi-Dezfooli et al.(2016)]

Dies bestätigt die Existenz von großen Korrelationen und Redundanzen im neuronalen Netz. Man kann auf eine Existenz eines Unterraums folgern mit Dimension $d' << d$, welches die meisten Normalvektoren zur Entscheidungsgrenze in der Umgebung natürlicher Bilder enthält. Wird nun ein Vektor aus diesem Unterraum zufällig gewählt, so kann dieser eine Fehlklassifizierungsrate von ca. 38% erzielen (auf Bildern, die nicht genutzt wurden, um den Unterraum zu berechnen). Ein normaler, zufälliger Vektor kommt auf eine Rate von 10%. Der vorgestellte Algorithmus hingegen nimmt nicht einen zufälligen Vektor aus dem Unterraum, sondern versucht aktiv eine maximale Falschklassifizierungsrate zu erreichen.

4.3 Steigerung der Robustheit gegen universelle Störungen

Um neuronale Netze robuster gegen solche universelle Störungen zu machen, wurde auf 50% der Trainingsbilder zufällig eine aus 10 universellen Störungen addiert. Das neuronale Netz trainiert nun fünfmal auf diesen neuen Daten und wird mit einer 11ten universalen Störung getestet.
Man konnte damit die Rate der falsch klassifizierten Instanzen von 93,7 % auf 76,8% (auf dem VGG-F Netz) senken. Weitere Runden mit anderen universellen Störungen lagen bei ca. 80%, aber nicht besser. Die Robustheit konnte etwas verbessert werden. Aufgrund der Redundanzen im neuronalen Netz ist es nicht verwunderlich, dass eine solche universelle Störung nur geringfügig abtrainiert werden kann.

5 Zusammenfassung

5.1 Wo werden intelligente Algorithmen (in Zukunft) genutzt

- Zur Bekämpfung von Verbrechen bevor sie ausgeübt werden können
- Personenerkennung anhand des Gesichts, Bewegungsmuster, Surfverhalten
- Stephen Hawkins und Elon Musk warnen vor einer Verselbständigung von militärischer KI, die ihre Ziele selbständig ausmacht und angreift
- Übernahme von schwierigen bzw. gefährlichen Aufgaben und Jobs
- Qualitätssicherung
- Gewinnmaximierung
- etc.

5.2 Welche Gefahren entstehen durch diese Algorithmen

- Diskriminierung
- Vortäuschung falscher Tatsachen
- falsche Beratung bis hin zum Tod (Arzt)
- Überwachung und Verfolgung auf Schritt und Tritt
- Verurteilung vor Gericht aufgrund von schlechten Datensätzen
- etc.

5.3 Chancen, die sich aus den neuen Algorithmen mit Erklärungen ergeben

- Vertrauen in Algorithmen wird gesteigert
- Algorithmen und Datensätze können aufgrund der Erklärungen angepasst oder ersetzt werden

- Beratung mit Begründung für Ärzte
- Automatisierung von Prozessen (da nun Vertrauen vorhanden ist)
- dem Fortschritt wird kein Stein in den Weg gelegt, daher können alte und neue Fragen gelöst werden
- Verurteilung vor Gericht beruht auf einer extrem großen Datenbank zu ähnlichen Fällen und fällt gerechter aus
- etc.

5.4 Warum die breite Masse der Bevölkerung die Algorithmen nicht versteht?

Dieser Abschnitt beruht auf [Goodman and Flaxman(2016)].

1. Algorithmen und Datensätze werden nicht offengelegt.
2. Code ist zwar öffentlich zugänglich, aber es ist für Otto Normalverbraucher nicht verständlich, was hier gemacht wird.
3. Multi-dimensionale mathematische Begründung passt nicht zum menschlichen Denken, wie wir Dinge begründen. Daher bleiben diese Überlegungen für uns teilweise unergründlich.

5.5 Was wird zur Verbesserung der Erklärung seitens der Politik gemacht?

1. Verbraucherzentrale des Bundesverbandes fordert TÜV für Algorithmen, immer dann, wenn etwas über den eigenen Kopf hinweg entschieden wird.
2. Ab April 2018 tritt die General Data Protection Regulation (GDPR) EU-weit in Kraft. Dabei besteht die Pflicht für Algorithmen, dass diese nicht-diskriminierend sind. Außerdem bekommt jeder das Recht auf Erklärung, welche Daten über einen gesammelt werden und dass dies überhaupt passiert. Weiterhin hat man auch ein Recht auf Erklärung, wie eine Entscheidung zustande kommt, damit, wenn nötig, diese Prognosen korrigiert werden können. Wenn diese Regulationen nicht eingehalten werden, können Strafen von bis zu 4% des Umsatzes erhoben werden. Der gerade vorgestellte Punkt beruht auf [Goodman and Flaxman(2016)].

5.6 Schlussplädoyer

Meine persönliche Antwort auf die Frage: “Kann man verstehen, wie intelligente Algorithmen entscheiden?” lautet “Nein”. Denn die Transparenz, die für die Nachvollziehbarkeit eines Entscheidungsprozesses vonnöten ist, geht durch die Komplexität der intelligenten

Algorithmen verloren. Jedoch können wir Erklärungen für die Entscheidungen finden, welche uns zwar keine Aufschlüsse auf die internen Prozesse liefern, aber unser Vertrauen in die Algorithmen bestärkt. Die Chancen, welche uns künstliche Intelligenz mit solchen Algorithmen ermöglichen, werden in der Zukunft enorm anwachsen und ein großes Potenzial beherbergen. Aber da man diese Algorithmen nicht komplett beherrscht, besteht die Möglichkeit, dass man schlussendlich die Kontrolle über diese künstliche Intelligenz verlieren wird. Wenn man also künstlicher Intelligenz zu viel Macht gibt, ist sie für mich gefährlich, da man ihren Entscheidungsfindungsprozess nicht mehr nachvollziehen kann, so wie beim Menschen eben auch.

Abbildungsverzeichnis

Literaturverzeichnis

[Gazzaniga(2005)] Michael S Gazzaniga. *The ethical brain.* Dana press, 2005.

[Goodman and Flaxman(2016)] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a"right to explanation". *arXiv preprint arXiv:1606.08813*, 2016.

[Goyal et al.(2016)] Yash Goyal, Akrit Mohapatra, Devi Parikh, and Dhruv Batra. Towards transparent ai systems: Interpreting visual question answering models. *arXiv preprint arXiv:1608.08974*, 2016.

[Lipton(2016)] Zachary C Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.

[Moosavi-Dezfooli et al.(2016)] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. *arXiv preprint arXiv:1610.08401*, 2016.

[Park et al.(2016)] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. Attentive explanations: Justifying decisions and pointing to the evidence. *arXiv preprint arXiv:1612.04757*, 2016.

[Ribeiro et al.(2016)] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.

Erklärung

Ich versichere hiermit, dass ich die anliegende Arbeit mit dem Thema "Kann man verstehen, wie intelligente Algorithmen entscheiden?" selbstständig verfasst und keine anderen Hilfsmittel als die angegebenen benutzt habe. Die Stellen, die anderen Werken dem Wortlaut oder dem Sinne nach entnommen sind, habe ich in jedem einzelnen Fall durch Angaben der Quelle als Entlehnung kenntlich gemacht.

Andreas Haller Heidelberg, 19. Juni 2017