**Projektbericht zum Modul**

**Information Retrieval und Visualisierung**

Sommersemester 2022

**Bearbeitetes Thema:**

**Netflix TV Shows and Movies**

Viet-Anh Do

Matrikelnummer: 220228228

GitHub Repository: <https://github.com/do46/IR_Netflix_TVShows_Movies_VAD>

Inhalt

[1. Einleitung 3](#_Toc121651445)

[1.1. Anwendungshintergrund 3](#_Toc121651446)

[1.2. Zielgruppe 3](#_Toc121651447)

[1.3. Überblick und Beiträge 4](#_Toc121651448)

[2. Daten 5](#_Toc121651449)

[2.1. Technische Bereitstellung der Daten 6](#_Toc121651450)

[2.2. Datenaufbereitung 6](#_Toc121651451)

[3. Visualisierung 7](#_Toc121651452)

[3.1. Analyse der Anwendungsaufgaben und Anforderungen an Visualisierung 7](#_Toc121651453)

[3.2. Präsentation der Visualisierung 8](#_Toc121651454)

[3.2.1. Visualisierung Eins 8](#_Toc121651455)

[3.2.2. Visualisierung Zwei 9](#_Toc121651456)

[3.2.3. Visualisierung Drei 11](#_Toc121651457)

[3.3. Interaktion 12](#_Toc121651458)

[4. Implementierung 13](#_Toc121651459)

[5. Anwendungsfälle 17](#_Toc121651460)

[5.1. Anwendung Visualisierung Eins 17](#_Toc121651461)

[5.2. Anwendung Visualisierung Zwei 17](#_Toc121651462)

[5.3. Anwendung Visualisierung Drei 17](#_Toc121651463)

[6. Verwandte Arbeiten 17](#_Toc121651464)

[7. Zusammenfassung und Ausblick 17](#_Toc121651465)

[Literatur 17](#_Toc121651466)

# Einleitung

Streaming-Dienste wie Netflix, Amazon Prime oder Disney+ dominieren heutzutage den TV-Markt. Hinter der Erfolgsgeschichte dieser Video-on-Demand Plattformen stehen nicht nur ihre innovative Programme, sondern auch die Ausnutzung von Technologien, um ihre Kunden zu verstehen und ihnen maßgeschnittene TV Shows bzw. Filme zu produzieren bzw. vorzuschlagen. Mit weltweit über 200 Millionen Abonnements bis zum 2. Quartal 2022 ist Netflix neben Amazon Prime die Marktführer im Streaming-Dienst-Bereich. JustWatch, eine Firma mit Sitz in Berlin, verfügt inzwischen über die Datenbank mit über 3.600 Filme und 1.800 Serien [2]. In Rahmen dieser Arbeit wird dieser genannte Datensatz von JustWatch genauer betrachtet, analysiert und schließlich aufbereitet für die Visualsierung, wobei unterschiedliche Visualisierungstechniken angewendet werden.

Das Hauptziel dieser Arbeit ist es, die individuellen Fragen zum Thema Netflix Filme und Serien von der Zielgruppen dieser Arbeit mithilfe einem Visualisierungsprogramm zu beantworten. Um dieses Ziel zu erfüllen, werden unterschiedliche Eigenschaften des Datensatzes durch zwei- oder mehrdimensionale Darstellungen gegenübergestellt. Zuerst einmal werden lediglich zwei Eigenschaften der Filme und/oder Serien in einem einfachen Scatterplot visualisiert. Weiterhin sollte die Anzahl der Dimensionen nicht nur auf zwei beschränkt werden. Um ein tieferes Verständnis des Datensatzes zu erhalten, werden die Daten mittels parallelen Koordinaten dargestellt werden, wobei die Achsen ebenfalls beliebig modifizieren lassen. Schließlich um eine Überblick über die gesamte Datenstruktur des Datensatzes zu gewinnen, werden die Daten durch einen Iconplot präsentieren.

## Anwendungshintergrund

Die in dieser Projektarbeit verwendete Visualisierungstechniken sind die des Scatterplots, der Parallelen Koordinaten und der Iconplot. Damit ein besseres Verständlnis in diesen Techniken geschafft wird, werden sie im Folgenden kurz vorgestellt.

Bei einem Scatterplot geht es um eine Grafikdarstellung von beobachteten Wertepaaren zweier statistischer Merkmale mithilfe von Punkten auf Kartesischem Koordinatensystem. Aus diesen Punkten bilden sich eine Punktwolke, welche auch die Streuung der Daten auf dem

## Zielgruppe

Im Folgenden werden die Zielgruppen dieser Visualisierungsarbeit ausführlich beschrieben. Diese sind namentlich Verbrauchs- und Medienanalytiker und junge Konsumentinnen und Konsumenten.

Die Verbrauchs- und Medienanalytiker sind die Gruppe, die hauptsächlich mit Medienbewertung im Allgemeinen beschäftigen. Diese Visualisierungen können diese Zielgruppe eine Einblick in den Filmtrend über Jahre hinweg, in den Zusammenhänge zwischen ihren Eigenschaften sowie eine gesamte Überblick über allen Eigenschaften der gesamten Filmdatenbank von Netflix. Erkenntnisse aus dieser Visualisierung könnten sowohl für Filmproduzenten als auch für Streamingplattform vorteilhaft sein, denn sie können aus diesen Entdeckungen oder Trenderkennungen dem Zuschauern maßgeschneiderte Filme oder Serien produzieren bzw. in den Portfolio aufnehmen.

Die Gruppe der junge Konsumentinnen und Konsumenten können mithilfe der Visualisierung die Filmwelt selbst erkunden. Für viele junge Leute sind neben brandneue auch alte Filme und Serien von Interesse. Hierfür könnten die Bewertungen von beiden Datenbank eine gute Referenz für ihre Filmwahl.

## Überblick und Beiträge

Diese Arbeit ist ein Versuch, die gelernte Visualisierungstechniken auf ein echtes Anwendungsbeispiel anzuwenden. Der verwendete Datensatz befindet sich in der Webseite *kaggle.com*, eine zur Google gehörende Online-Community, die sich an Datenwissenschaftler richtet. Diese Datensatz liefert dabei Auskünfte über Netflix Filmen- und Serienportfolio, etwa Name, Produktionsjahr, verschiedene Bewertungsmetriken von zwei Filmdatenbanken, also IMDb und TMDb.

Zudem können die entsprechende Zielgruppen die Kategorien auswählen und dementsprechend die Filme auflisten für sie sich interessieren. Die Datensätze liefern neben der Kategorisierung der Filmen einen Überblick über die Bewertung der Filme. Insgesamt haben zwei Unternehmen die Filme bewertet. Mann kann auf dem ersten blick erkennen, dass die Bewertung der Filme unterschiedlich ausfallen. Die dargestellten visualisierungstechniken ermöglichen den Zielgruppen eine Möglichkeit die Produkte zu unterscheiden in filme oder Serien. Interessiert man nur für Serien kann man die Daten so filtern, dass man nur die Serien bekommt und je nach dem, was man gerne anschauen möchte, bekommt man die entsprechenden dargestellten Daten. Zudem sind sowohl für die produzierende Filme als auch die Serien das Jahr des Veröffentlichung angegeben. Weiterhin kann man anhand der Daten den Besucheranzahl, die die Filme gestreamt haben, herauslesen.

Die Anwendungshintergrund der ausgewählten Filme und Serien wird in diesem Kapitel angerissen und kurz erklärt und in Kapitel 3 wird den Lesern ein genauer Überblick gegeben. Der Auswahl des Scatterplots für diesen Datensätzen kann so argumentiert werden, dass anhand dessen viele Informationen über diese Sreamingsdienst gewonnen können. Dabei ist zu beachten, dass je nach dem welche Zielgruppe man gehört, kann man dementsprechend die Achsen auswählen, um besondere Erkenntnisse hinsichtlich der Streaming-Angebote aus einem Blick zu erkennen. Der Parallelen Koordinaten ermöglicht du die Zielgruppen mehr als zwei Eigenschaften miteinander zu vergleichen, um ein besseres Verständnis hinsichtlich der Datensätze zu bekommen. Diesbezüglich können auch Erkenntnisse bezüglich Trends und Besonderheiten sowohl der Filme als auch den Serien den Interessierten ermöglichen. Mithilfe der dritten Visualisierungstechnik können die Zielgruppen Informationen bei den verschiedenen Merkmalen der Filmen hinsichtlich der zwei angegebenen Bewertungsunternehmen in x- und y-Achse abgebildet werden.

# Daten

Wie bereits erwähnt stammt der verwendete Datensatz einem Nutzer von der Plattform *kaggle.com*, wobei dessen Ursprung nach seiner Angabe aus der Film- und Seriendatenbank JustWatch zurückzuführen ist. Dieser Datensatz enthält 15 Spalten mit 5.978 Film- und Serieneinträge.

Der Datensatz fängt an mit der Spalte „id“, wobei sich die IDs von Filmen und Serien durch die Präfix „tm“ für Filmen und „ts“ für Serien unterscheiden. Neben „title“ für Titel, „type“ für Typen und „release\_year“ für Erscheinungsjahr beinhaltet der Datensatz noch die folgenden Spalten: „description“, „age\_certification“, „runtime“, „genres“, „production\_countries“, „seasons. Eine kurze Beschreibung erhält man durch die Spalte „description“, während die Spalte „age\_certification“ die Auskunft über Altersbeschränkung liefert. „runtime“ gibt die Dauer des einzelnen Film oder die Standarddauer einer Serienfolge an und „genres“ umfasst alle „tags“ von dem Film und Serien, welche sich selbstverständlich auf deren Genres bezieht. „production\_countries“ gibt die Information über Produktionsländer sowohl der Filmen als auch der Serien an, wohingegen in der Spalte „seasons“ nur die Angaben über die Anzahl an Staffeln der Serien eingetragen werden.

Anschließend befinden sich die Informationen bezüglich der zwei Datenbanken, IMDb und TMDb. „imdb\_id“ gibt an, mit welcher ID der Film oder die Serie in der IMDb bezeichnet wird. Die IMDb-Bewertung befindet sich in der Spalte „imdb\_score“. Registrierte IMDb-Nutzer können für jeden veröffentlichten Titel in dieser Datenbank eine Stimme (von 1 bis 10) abgeben. Einzelne Stimmen werden dann aggregiert und zu einer einzigen IMDb-Bewertung zusammengefasst. Nutzer können ihre Stimmen unbegrenzt aktualisieren. Die neue Bewertung wird die vorherige aber dann überschreibt, sodass eine gültige Stimme lediglich pro Titel und pro Nutzer ist [2]. Die dazugehörige Anzahl der abgegebenen Stimme findet man in der Spalte „imdb\_votes“. Von seitens TMDb enthält der Datensatz zwei Spalten, namentlich „tmdb\_popularity“ und „tmdb\_score“. Diese beiden Metriken werden anhands zahlreichen Daten kontinuerlich fortgeschrieben und täglich um 7 Uhr UTC aktualisiert. Zu diesen Daten gehören beispielweise die Anzahl der Stimme pro Tag, Anzahl der Abrufe pro Tag, Bewertung des vorherigen Tags, Anzahl aller abgegebenen Stimmen, usw. [3]

Die Datenqualität dieser Daten eignet sich für beide Zielgruppen der Arbeit gut. Dieser Datensatz verfügt sich über eine klare Struktur, die einheitliche Formatierung des Zellenwertes innerhalb einer Spalte und die Vollständigkeit der Einträge, was die Qualität der Daten auch nach der Datenaufbereitung garantiert. Weiterhin mit über 5.000 Einträge bietet der Datensatz eine angemessene Tiefe sowohl für die Forschungs- als auch für die Selbsterkundungszwecke geeignet. Somit hat der Datensatz die Anforderungen der beiden Zielgruppen erfüllt.

## Technische Bereitstellung der Daten

Die technische Bereitstellung der Daten erfolgt mithilfe des GitHubs Repository, in dem das Visualisierungsprojekt umgesetzt worden ist. Alle Daten, sowohl im Original („titles.csv“) als auch die Aufbereiteten, befinden sich im Ordner „Data“, wobei die letzteren im Unterordner „AufbereiteteDaten“ gespeichert wird. Der Prozess der Datenaufbereitung wird im anschließenden Unterkapitel 2.2 detailiert erklärt.

Das Dateiformat ist „comma separated values“ mit der Dateinamenserweiterung *.csv*, wobei die Werte der nebeneinanderliegenden Spalten mit einem Kommazeichen separiert werden. Da der Delimiter in diesem Fall „Komma“ ist, soll das Dezimaltrennzeichen ein Punkt sein, um Mehrdeutigkeit zu vermeiden. Leerfelder sind allerdings nicht zu vermeiden und Einträge mit Leerfelder werden auch bei der Datenaufbereitung entfernt.

## Datenaufbereitung

Die originalen Daten werden im Python mit dem Package „pandas“ aufbereitet. Visual Studio Code und Jupyter Notebook finden auch Einsatz, um das Code besser zu visualisieren. Die Nutzung von Excel wird hier vermieden, da diese möglicherweise die Datenstruktur ändern könnte. Hinzukommt, dass Excel eigentlich für die Interaktion zwischen Maschine und Menschen gedacht wurde. In diesem Fall fungiert das Aufbereitungstool allerdings um eine Schnittstelle zwischen .csv Datei und Elm-Programmierung, darum ist die Nutzung von Excel nicht geeignet.

Aus dem großen Datensatz, die nach der Verarbeitung „titleslesslessdf.csv“ umbenannt wird, entstehen dazu noch zwei kleinere Datensätze, nämlich „moviedf.csv“ und „showdf.csv“. Der erste enthält Daten zu den Filmen, während in dem zweiten die Daten zu den Serien zu finden sind. Somit gibt es insgesamt drei Datensätze, ein gesamter Datensatz, der eine für Filmen und der andere für Serien. Die Unterteilung der Daten wird durch die Spalte „type“ ermöglicht, da diese genau angibt, ob es sich bei dem Eintrag um einen Film („MOVIE“) oder um eine Serie („SHOW“) handelt.

Wie bereits erwähnt sollen alle Einträge mit Leerfelder entfernt werden, damit alle Einträge gleichbehandelt werden können. Nach der Bereinigung reduziert sich die Anzahl der Einträge bei dem großen Datensatz von 5806 auf 5041 Einträge. Auch zu beachten ist die Daten bezüglich der Anzahl des Staffels, welche beim Filmen immer mit Leerfelder versehen und im Python mit *na* (not assigned) gekennzeichnet werden. Die Filmeneinträge sollen daher nicht aufgrunddessen gelöscht werden. Die Funktion *fillna()* sorgt dafür, dass die Spalte *seasons* von Filmen durch einen bestimmten Wert ausgefüllt werden (hier den Wert „0“).

Weiterhin wird eine weitere Spalte namens *numberTags* generiert und am Ende der Tabelle zugefügt. Diese Spalte ist der Spalte *genres* zurückzuführen, wobei der Wert der generierten Spalte entspricht die Listenlänge der originalen Spalte. Dieser Wert gibt an, an wieviele Filmkatgorien wird ein Film oder eine Serie zugeordnet. Filme oder Serien mit vielen Tags könnten beispielweise angeschaut werden. In Betracht kommen nach der Datenaufbereitung nur noch die folgenden Spalten: *id, title, type, release\_year, runtime, genres, seasons, imdb\_score, imdb\_votes, tmdb\_popularity, tmdb\_score* und *numberTags*. (còn thời gian thì giải thích)

# Visualisierung

In diesem Kapitel wird den Lesern einen Überblick verschaffen wie die Visualisierungsthe-chniken bei den oben genannten Zielgruppen Anwendung finden und ob die ausgewählten Visualisierungen die Zielgruppen aussagekräftige Informationen liefern können.

## Analyse der Anwendungsaufgaben und Anforderungen an Visualisierung

Um die individuellen Fragen der Zielgruppen zu beantworten, bestehen die Hauptaufgaben der Arbeit darin, die Zusammenhänge zwischen unterschiedlicher Merkmalen sowie die gesamte Struktur von dem Datensatz zu erkennen.

Um Fragen zwischen zwei Merkmalen zu beantworten, wurde Technik des Scatterplots in der Visualisierung Eins angewendet, um zwei unterschiedliche Eigenschaften miteinander zu vergleichen. So könnten verschiedene Zusammenhänge zwischen den Filmen- und Serienmerkmalen aufgedeckt und somit ein besseres Verständnis für diese geschaffen werden. Zudem muss man beachten, dass die visualisierte Variablen auch sinnvoll ausgewählt wurden. Aus diesem Grund wurde die zum visualisierenden Variablen-Paare bei dieser Visualisierung vordefiniert. Eine Gegenüberstellung von dem Paar *TMDb score* und *IMDb votes* scheint beispielweise nicht sinnvoll zu sein, da die Anzahl der Stimmen (vote) bei IMDb offensichtlich kaum Einfluss auf die Bewertung von TMDb-Nutzern hat. Imfolgedessen wird dieses Paar ausgeschlossen. Des Weiteren sollten die Datenpunkte so dargestellt werden, dass eine Trendlinie zwischen Merkmale, falls vorhanden, gut erkennbar ist. Zum Schluss sollten auch die tatsächlichen Daten unkompliziert angezeigt werden, etwa wie der Film oder die Serie heißt, zu welchem Genre gehören diese, usw.

Wenn sich die Fragen um mehr als zwei Merkmale beziehen, benötigt man mehr als nur ein Koordinatensystem. Weswegen findet die Visualisierungstechnik Paralel Coordinate Anwendung. Mithilfe dieser Technik lassen sich mehrere Merkmale auf einem Plot gegenüberstellen. Genau wie bei dem Scatterplot sollten die visualisierten Daten auch leicht abrufen lassen, damit die Erkenntnisse nochmal unkompliziert geprüft werden können. Optimalerweise sollten auch Ausreißer ausgeschlossen oder ausgeblendet werden, damit der Trend besser erkennbar wird.

Zum Schluss sollte auch die gesamte Struktur der Daten visuell dargestellt werden. Neue Erkenntnisse könnten möglicherweise mithilfe eines Überblicks über alle Merkmale entdeckt werden. Ein *icon plot* wird diese Anforderung erfüllen, wobei wie bei den anderen Darstellungen die Datenpunkte möglich ersichtlich sowie die Werte bestmöglich abrufbar gemacht werden sollten.

## Präsentation der Visualisierung

### Visualisierung Eins

Bei der ersten Visualisierung des Projekts handelt es sich um ein Scatterplot (oder auch als Streudiagramm genannt). Mithilfe eines Scatterplots können stets zwei verschiedene Eigenschaften eines Filmes oder einer Serien gegenübergestellt werden, um deren Zusammenhänge bzw. Korrelationen erkennen zu können. Dieses Wertpaar der Eigenschaften wird in ein kartesisches Koordinatensystem eingetragen und bildet sich daraus einen Punkt, welchen in dieser Visualisierung durch einen Kreis repräsentiert wird. Wenn man mit der Maus auf diese Punkte schwebt, werden diese farblich hervorgehoben und deren Informationen werden zeitgleich über dem Scatterplot angezeigt. Die anzuzeigenden Eigenschaften können darüber hinaus auch angepasst werden, indem man ein anderes Eigenschaftspaar aus der Dropdown-Liste auswählt. Diese Liste befindet sich über dem Diagramm, wobei deren Einträge mit den Namen von Eigenschaftspaare beschriftet werden. Durch die Buttons über der Dropdown-Liste kann man auswählt, ob man nur Filmen, nur Serien oder beiden Typen visualisiert. Bei Filmen hat man beispielweise nur eine einzige Filmeinheit, wohingegen haben Serien oft viele Folgen. Als ein weiteres Beispiel haben Serien oft mehr als bloß ein Saison, während der Begriff „Saison“ gar nicht bei Filmen existiert. Aus diesen Gründe sollte man auch die Möglichkeit bekommen, die Filmen und Serien unabhängig voneinander zu beobachten. Das Scatterplot kann hierfolgend in Abbildung 1 zu erkennen.

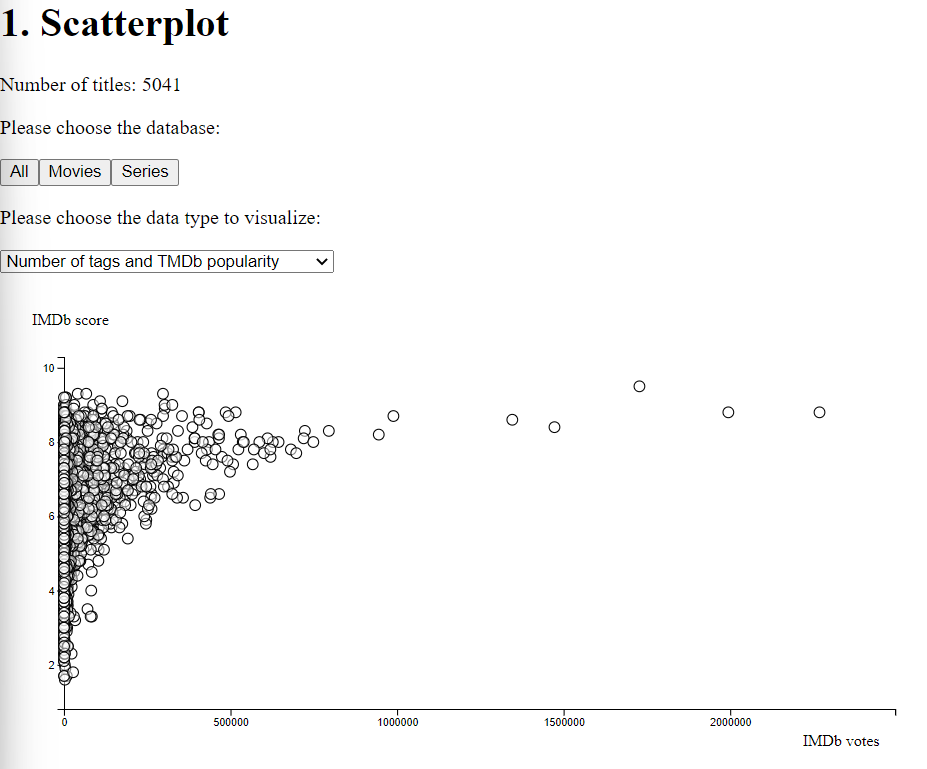


Abbildung : Das Scatterplot (Quelle: Eigene Darstellung)

Die Anforderung an das Scatterplot ist somit erfüllt. Man kann durch Auswahl der Optionen die Merkmalen der Filmen oder Serien gegenüberstellen, um mögliche Zusammenhänge zwischen ihnen zu entdecken. Die Seite wird nach der Auswahl nicht neugeladen und die Verzögerung ist nur in milisekunden zu rechnen. Die Anzahl der Punkten wird ebenfalls angezeigt und die entsprechenden Trends oder Besonderheiten der Daten lassen sich auch erkennen.

### Visualisierung Zwei

Die Visualisierung Zwei wird mittels eines Parallels Koordinaten Diagramm dargestellt. Bei solcher Darstellungsform erhält man einen Überblick über mehreren ausgewählten Eigenschaften des Filmes und der Serien. Die auf den Achsen eingetragenen Datenpunkt werden mitenander durch Linien verbunden, sodass man eine Darstellung wie in Abbildung 2 erhält, wobei jeder Film oder jede Serie durch eine Linie repräsentiert. Darüber hinaus hat man die Möglichkeit, individuell den Eigenschaften zuzuweisen. Die zu beobachtenden Eigenschaften von den Achsen können ebenfalls angepasst werden. Wird ein Button geklickt, so wird auch die Eigenschaft von der entsprechenden Achse auch geändert. Damit die Übersichtlichkeit der anzuzeigenden Daten gewährleistet wird, werden die Linien farblich markiert und die entsprechenden Daten über dem Diagramm angezeigt, wenn man die Maus auf die Linien schwebt.

Die Achsen können dabei individuell den Eigenschaften zugewiesen werden. Dies erfolgt mithilfe von verschiedenen Buttons, welche über der Darstellung zu finden sind. Wenn dabei ein Button gedrückt wird, nimmt die entsprechende Achse das gewünschte Merkmal an. Zusätzlich werden, wenn mit der Maus über eine Linie gefahren wird, die Eigenschaften angezeigt und die Linie farblich hervorgehoben. Die Parallelen Koordinaten sind in Abbildung 2 zu erkennen.

Ein Bild, das Text, Himmel enthält.

Automatisch generierte Beschreibung

Abbildung : Parallele Koordinaten (Quelle: eigene Darstellung)

Weiterhin bietet Parallele Koordinaten noch die Möglichkeit, den Diagramm im X-Ray-Modus wie in Abbildung 3 zu repräsentieren. Die Linien werden verblasst und im weiß anstelle von schwarz dargestellt, während der Hintergrund schwarz gefärbt wird. Mithilfe dieses Modus lassen sich die Linien von Ausreißer ausblenden. Der Trend wird wohingegen deutlich angezeigt. Sobald der Trend erkannt geworden ist, kann man die Attributen genauer betrachten mit der 1. Visualisierung.

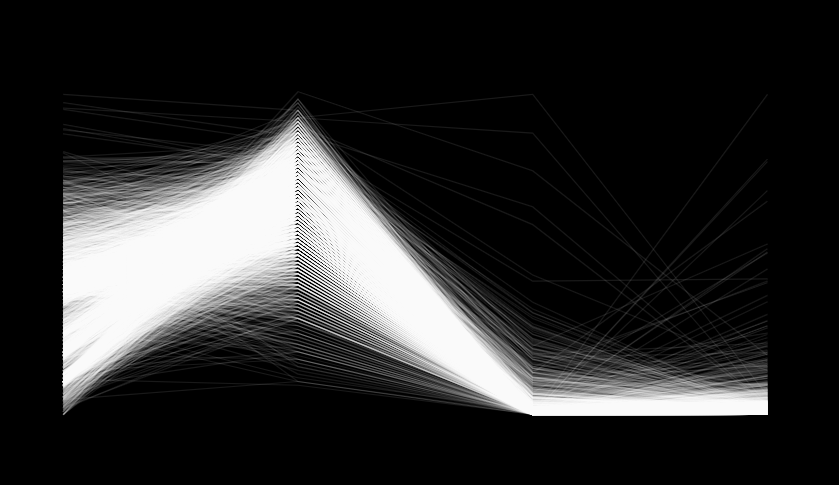


Abbildung : Parallele Koordinaten in X-Ray-Modus (Quelle: eigene Darstellung)

Die Anforderungen an diese Darstellung konnten erfüllt werden. So ist es mithilfe dieser Darstellungsform möglich, die verschiedenen Eigenschaften gegenüberzustellen und individuell anzupassen. Weiterhin werden Werte angezeigt, wenn innerhalb dieses Diagramms über eine Linie drübergefahren wird. Somit ist die Verfolgbarkeit und Nachvollziehbarkeit der Daten gegeben. Durch das Erfüllen der Übersichtlichkeit und der Nachverfolgung der Werte sind die Anforderungen an dieses Diagramm erfüllt.

### Visualisierung Drei

Für die Dritte Visualisierung werden die sogenannte Stickfigure-Technik angewendet. Im Unterschied zu der ersten Visualsierung werden Datenpunkte mithilfe eines Stickfigure in einer mehrdimensionalen Darstellung visualisiert. Bei dieser Technik werden zwei Attribute verwendet, um das Stickfigure auf das Koordinatensystem platziert. Das Stickfigure selbst wird durch die restlichen Attribute gezeugt, wobei einzelne Attribute durch einen Winkel des Stickfigure abgebildet wird. Wie in Abbildung 3 zu sehen, werden *runtime*, *TMDb popularity*, *release year, IMDb votes* und *number of tags* durch den Körper, die linke und rechte Hände bzw. die linke und rechte Füße des Stickfigures dargestellt. Diese Stickfigures werden anschließend auf dem durch *IMDb Score* und *TMDb Score* abgebildeten Koordinatensystem platziert und somit erhält man den Iconplot. Um Daten anzuzeigen, kann man wiederum die Maus auf Stickfigure schwebt. Das angewählte Stickfigure wird gleichzeitig mit vergrößter Länge und dunkler Farbe hervorgehoben. Darüber hinaus ist es möglich, die Default-Größe von Stickfigure anzupassen. Wie bei die anderen Plots kann man mit Buttons die Filmen- und Seriendaten unabhängig voneinander betrachten.

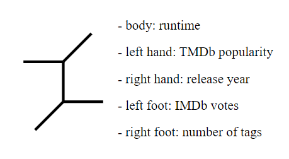
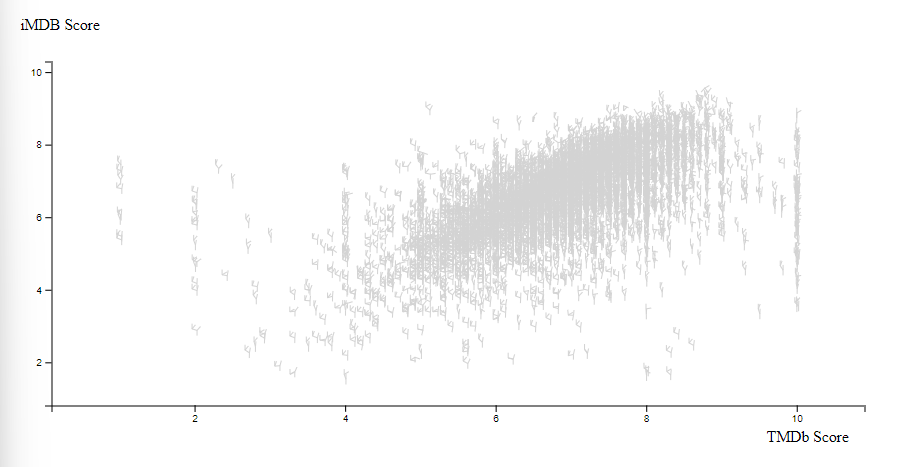


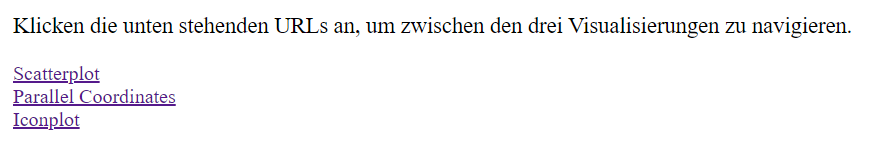
Abbildung : Iconplot (Quelle: eigene Darstellung)

Iconplot bietet eine Möglichkeit, den Gesamtdatensatz vollständig durch eine Übersicht zu betrachten. Man kann zwar nicht mit Iconplot den Zusammenhang anschauen, erkennt ihn aber trotzdem durch Bewegungsrichtung oder -tendenz der Körperteilen. Sobald sie erkannt geworden ist, kann man die Attributen mit den ersten und zweiten Visualisierung nochmal genauer betrachten. Schließlich ist die Anforderung an Visualisierung Drei wird wieder erfüllt und Visualisierung Drei wird mit Iconplot repräsentiert.

## Interaktion

Interaktion zwischen Visualisierungen ist notwendig. Der Trend wird zuerst bei der Visualisierung Drei erkannt. Für eine detailierte Ansicht kann man den Trend mithilfe der Visualisierung Eins und Zwei verdeutlichen. Ein schneller und problemloser Wechsel zwischen Visualisierungen ist erforderlich. Bei dem homepage erhält man die Möglichkeit, Visualisierungstechniken auszuwählen. Innerhalb der Visualisierung kann sich frei zu den nächsten Visualisierungen oder zu der Hauptseite, ohne die Seite neuzuladen durch Anklicken auf die entsprechenden Hypertexte wechseln.

Abbildung : Interaktionen zwischen Visualisierungen (Quelle: Eigene Darstellung)



# Implementierung

## Gesamtaufbau

Ein Bild, das Text enthält.

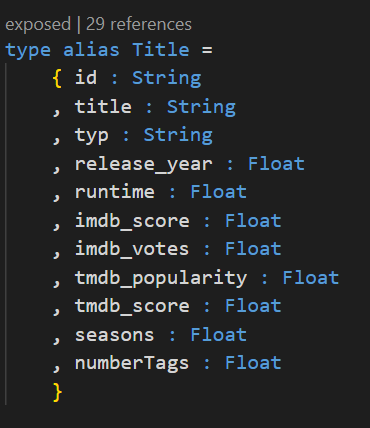
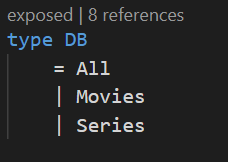
Automatisch generierte BeschreibungDas Projekt wird in fünf unterschiedlichen Elm-Modul aufgeteilt. Diese sind nämlich Main.elm, Data.elm, Scatterplot.elm, ParallelCoordinate.elm und Iconplot.elm. Die Daten werden mit Python-Jupyter vorverarbetet, wobei das Packages Pandas verwendet wird. Die Vorverarbeitungsschritte sind in dem Modul datapreprocessing.ipynb zu finden. Die verarbeitete Daten werden anschließend in dem Ordner AufbereiteteDaten gespeichert. Eine Übersicht über des Aufbaus von dem Projekt befindet sich in der Abbildung 6.

Daten wurde zuerst bereinigt durch dem Modul datapreprocessing.ipynb. Werte, die sich nicht visualisieren lassen wie z.B. NaN-Werte (not a number) oder Na-Werte (not assigned) werden entweder gelöscht oder mit 0 ersetzt. Eine neue Spalte (numberTags) wird ebenfalls auch mithilfe dieses Moduls generiert. Eine Funktion wird verwendet, um die Anzahl der „genres“ zu zählen. Diese gelten als die Anzahl des Tags pro Film oder pro Serie. Anschließend wird die Datei wieder als .csv-Datei zurückgegeben.

Abbildung : Gliederung des Projekts

Im Modul Data.elm werden verschiedene Parameter festgelegt, sodass diese global definiert und somit auch bei den anderen Modul wiederverwendet werden. Diese sind unter anderem der Standardabstand, der Radius von Kreise usw. Das gleiche gilt auch für die Funktionen, die ebenfalls in den anderen Module immer Anwendung findet. Diese sind beispielweise Funktionen, die Datentypen manipulieren, oder die Funktion, die die Länge der Achse automatisch anpassen. Auch bei dem Modul Data.elm werden die Daten aus .csv-Datei eingelesen und dekodiert, sodass diese später auch als Elm-Typen verstanden wurden und somit in der Elm-Umgebung verwendet werden können. Zu guter Letzt werden auch bei diesem Modul Elm-Typen und Elm-Alias-Typen definiert, sodass diese nicht nochmals bei den Visualisierungen definiert werden müssen.

Abbildung : Elm-Typen Definition



## Implementierung Visualisierung Eins

Der Großteil der ersten Visualisierung basiert auf das Quellcode von den Übungsstunden. Da die Anforderung auch darin besteht, dass Filmen und Serien einzeln betrachten lassen, wird ChangeDB-Buttons in allen drei Visualisierungen implementiert. Diese Buttons verleiht den Nutzern die Möglichkeit, zwischen Filmen, Serien bzw. beiden zu wechseln. Die Implementierung dieses Buttons erfolgt durch Namenersetzen der zu visualisierenden Dateien in der Funktion getData, wobei die Namen der csv-Dateien im Data.elm als eine Liste definiert werden.

Weiterhin wird eine Dropdown-Liste in diesem Modul hinzugefügt. Eine Vorteil der Liste gegenüber Button ist, dass die UI-Fläche mit der Liste eingespart werden kann. Die Liste lässt sich zeigen erst beim Betätigen des Dropdown-Pfeils.

Darüber hinaus werden die beim Hovern angezeigten Texte über den Diagramm umgestellt. Diese Änderung sorgt dafür, eine bessere Lesbarkeit des Texts zu gewähren, vor allem wenn sich der betrachtete Punkt in einem dichtbesetzten Bereich befindet. Die Sichtbarkeit des angezeigten Textes wird in diesem Fall stark durch die benachbarten Punkte beeinträchtigt.

* 1. **Implementierung Visualisierung Zwei**

Basierend auf die erste Visualisierung sowie auf der bearbeiteten Übung wird die Visualisierung Zwei aufgebaut. Um die aktuell gezeigte Achsen zu ändern, werden unterschiedliche Buttongruppe implementiert. Es gibt insgesamt vier Gruppe. Diese gilt jeweils für eine Achse, sodass sich das bei einer Achse angezeigte Attribut ändern kann, wenn ein Button von derselben Gruppe betätigt wird.

Außerdem wird ein X-Ray-Modus wie bereits erwähnt noch implementiert. Der Diagramm an sich wird inhaltlich nicht geändert, sondern nur deren Eigenschaft. Im diesem Modus werden die Linien in Weiß anstatt in Schwarz dargestellt und die Linienopazität wird zeitgleich auf 0,1 gesetzt. Diese Eigenschaften (stroke, stroke-width, opacity) kann man durch css-style festlegen. Der Hintergrund sollte dabei auch in Schwarz anstatt in Weiß angezeigt. Beide Änderungen werden mit den Buttons verknüpft, sodass sie gleichzeitig geändert wird, wenn man die Buttons betätigt. Schwierigkeit besteht darin, dass sich die Eigenschaften in einer gleichen Stelle befinden, sonder verteilt im Skript. Sie miteinander zu verbinden mithilfe eines Msg (ChangeMode) bei *update* hat bei mir ein bisschen gedauert. Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Abbildung : Änderung der Linieneigenschaften durch css-style

* 1. **Implementierung Visualisierung Drei**

Das Quellcode von der ersten Visualisierung wird weiterverwendet für die Implementierung von Visualisierung Drei, da anstelle von Kreisen sollten die Stickfiguren die Datenpunkten repräsentieren. Eine Schwierigkeit besteht darin, dass Stickfigures sollten mithilfe von svg gezeichnet werden und ich mir noch gar keine Erfahrung mit svg gemacht habe. Die Winkel von Körper und Körperteile sollten dabei noch dynamisch sein, d.h. sie ändert sich, wenn die Daten sich ändern. Nach dem mathematischen Hinweis von Herrn Hinneburg wird das Code unter Einsatz von *polyline* zum Erstellen des Stickfigures geschrieben wie in Abbildung 9 zu sehen.

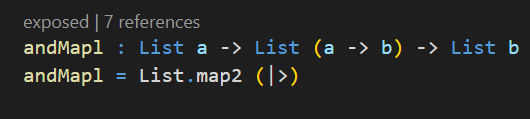
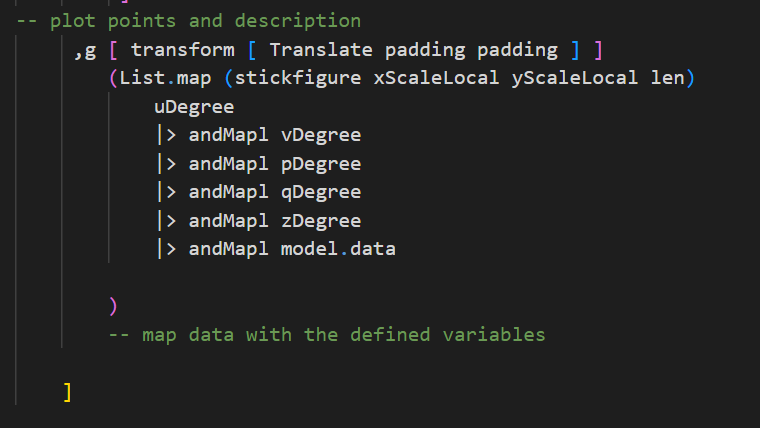
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Abbildung : svg-Code für Stickfigures

Eine andere Schwierigkeiten besteht darin, die Datenwerte mit der Zeichnen-Funktion *stickfigure* zu mappen. Insgesamt gibt es sechs Variablen, die mit dem Funktion *stickfigure* wiein Abbildung 10gemappt werden sollen. Eine schnelle Lösung wie List.map6 ist nicht gegeben, sondern nur höchsten bis List.map5. Aufgrunddessen habe ich eine Hilffunktion andMapl verwendet. Das Mapping findet dann sequenziell anstatt gleichzeitig statt durch Nutzen von der Funktion *pipe* (|>)*.* Analog wird auch ein andMap verwendet anstelle von Maybe.map6, was gar nicht existiert.

Abbildung : andMapl-Funktion und Implementierung



Weiterhin wird noch Buttons in der Visualisierung hinzugefügt, um die Länge der Linien durch Nutzer anpassen zu lassen. Diese erfolgt durch Hinzufügen eines weiteren Parameter *len* in dem Zeichnen-Funktion. Diese soll die Länge der Linien definieren und wird geändert beim Betätigen des Buttons.

* 1. **Implementierung der Interaktion**

Das Modul Main.elm sorgt für die Interaktionen zwischen den anderen Modulen unter Einsatz von Html. Hypertext wird vor allem verwendet, damit man von einem Modul zu dem anderen Modul ohne Neuladen wechseln kann. Bei den anderen Modul wurde ein Hypertext „Back to Homepage“ hinzugefügt, sodass man problemlos zu dem Hauptseite jederzeit zurückkehren kann. In der Hauptseite werden außerdem noch zwei andere Hypertexte dargestellt. Das eine führt zu dem Kaggle-Seite, wo die Daten zu finden ist. Das andere ist eine URL zu diesem Projekt im Github.

# Anwendungsfälle

In diesem Kapitel werden die Visualisierungen durch ein Anwendungsbeispiel demonstriert. Zur Erinnerung wurde das Attribut „number of tags“ generiert, indem die Anzahl der Genre

# Verwandte Arbeiten

# Zusammenfassung und Ausblick

Literatur

1. IMDb.com Help Center. Ratings FAQ. https://help.imdb.com/article/imdb/track-movies-tv/ratings-faq/G67Y87TFYYP6TWAV#

2. Rabe L (2022) Anzahl der zahlenden Streaming-Abonnenten von Netflix weltweit vom 3. Quartal 2011 bis zum 2. Quartal 2022. https://de.statista.com/statistik/daten/studie/196642/umfrage/abonnenten-von-netflix-quartalszahlen/. Zugegriffen: 28. Juli 2022

3. themoviedb.org Getting Started. https://developers.themoviedb.org/3/getting-started/popularity