

Projektbericht zum Modul Data Mining Wintersemester  
2021/2022

**Reproduktion des Papers**  
***Context-Sensitive Visualization of Deep***  
***Learning Natural Language Processing***  
***Models***[1]

Max Henze

11. März 2022

## **1 Einleitung**

Neuronale Netzwerke sind ein beliebtes Hilfsmittel im Bereich von NLP. Besonders Modelle, welche sich mit dem Einsatz von Transformern behelfen, wie BERT [2] oder GPT-2 [3], gehören schon lange zum state-of-the-art. Doch diese Modelle mit ihrer Vielzahl an Layern, Neuronen und Verbindungen, gewähren einen nicht gerade einfachen Einblick in ihre Verarbeitungsschritte. Ansätze wie Leave-One-Out oder Leave-N-Out versuchen durch Neuklassifikation von modifizierten Texten einen allgemeinen Einblick in die Verarbeitungsstrukturen von Neuronalen Netzwerken zu geben. Diese Ansätze betrachten aber nicht einen möglichen Kontext, wie er zwischen Wörtern in einem Satz existieren kann. Dunn et al. haben daher in ihrem Artikel „Context-Sensitive Visualization of Deep Learning Natural Language Processing Models“ eine Methode entwickelt um Wörter, mit ihrer unterschiedlichen Wichtigkeitsgewichtung innerhalb der Klassifizierung und unter Inbetrachtung von Kontext zu visualisieren. Als Replikationsziele wurden für diese Arbeit der gesamte Visualisierungsprozess von Dunn et al. gewählt. Zusätzlich dazu wurde ein eigenes BERT Modell auf dem gegebenen IMDB [4] trainiert um dieses für den späteren Klassifikationsprozess zu verwenden. Darüberhinaus wurden zwei zusätzliche Experimente zur Visualisierung mit Kontext durchgeführt. Durch die Replikation wird eine verständliche Codebeigabe zum Originalartikel erzeugt, welche dem Leser ein noch besseres Verständnis liefern soll. So können mit dem System alle Dokumente des Testdatensatzes klassifiziert und visualisiert werden um so eine größere Vielfalt von Beispielen und ein besseres Verständnis bereitzustellen.

## 2 Umfang der Replikation/Reproduktion

Als Ziel dieser Replikation wurde die einzige Hypothese gewählt, welche Dunn et al. in ihrem Artikel behandeln. Sie propagieren, dass sich die Wichtigkeit eines Wortes, innerhalb der Klassifikation durch ein Neuronales Netzwerk, nicht nur durch den Vergleich der sogenannten Prediction Strength (Sicherheit des NN, dass Label richtig Klassifiziert) des Originaltextes zur Prediction Strength des Textes ohne das betrachtete Wort (Leave-One-Out) ergibt, sondern dass der Einbezug von kontextuell zusammenhängenden Wörtern (Leave-N-Out) ebenfalls wichtig ist. „Unser Ansatz schaut auf die Kombination von Wörtern und Sätzen um deren Einfluss auf die Ausgabe des Modells zu erkennen, was zu einer Visualisierung führt, welche kontextsensitiver zum Originaltext ist.“ [1]

Somit ist folgende Behauptung das Ziel dieser Replikation:

- Leave-N-Out Ansatz ist geeigneter bei der Erkennung wichtiger Wörter und Strukturen als Leave-One-Out.

## 3 Methoden

Die Replikation des Originalartikels ergibt sich wie folgt. Durch die fehlende Beigabe von Code mussten alle Ideen und Modelle von Dunn et al. eigenständig implementiert werden. Dazu wurde sich an Wortangaben der Autoren wie zum Beispiel: „Der gesamte Code ist geschrieben in Python 3.8 und nutzt die Tensorflow Version der Transformersbibliothek. [...] Texttokenisierung und Abhängigkeitsbestimmung wurden mit der spaCy NLP Bibliothek durchgeführt.“ [1] gehalten.

Zur Klassifizierung von Dokumenten wurde ein Modell unter der Verwendung von BERT trainiert. Da keine weitere Angaben zu finden waren und eine große Auswahl an unterschiedlichen BERT Modellen zu finden ist wurde das *BERT uncased L-12 H-768 A-12* Modell gewählt, welches auf Tensorflow Hub<sup>1</sup> zu den am häufigsten verwendet BERT Modellen zählt. (über 214.000 Downloads)

Entwickelt wurde innerhalb eines Jupyter Notebooks mit Python. Die folgenden essentiellen Packages fanden dabei Anwendung:

---

<sup>1</sup><https://tfhub.dev>

Package Name	Package Funktion
tensorflow_hub	Einbindung des BERT Modells
tensorflow	Modellerzeugung und Training
official.nlp	Trainingsoptimisierung
spacy	Abhängigkeitsbestimmung (Dependency Parsing)
pandas	Arbeiten mit Dataframes
matplotlib	Visualisierung der Texte

Abbildung 1: Verwendete Packages

Das BERT Modell auf einer Nvidia Geforce RTX 3070 mit 8 GB Arbeitsspeicher trainiert.

### 3.1 Modellbeschreibung

Innerhalb des Originalartikels sind keine Angaben bezüglich der Zielfunktion und Parameter zu finden. Angaben zum Modell beruhen auf der Benennung eines BERT Modells und einer Modellbeschreibung, welche auf das Anhängen eines Dropout-Layers und Dense-Layers verweist.

Die beschriebene Methodik ist wie folgt:

Ein Text wird durch das Modell klassifiziert und die damit korrespondierende Ausgabestärke, der Score, wird notiert. Nun werden mit Hilfe einer Abhängigkeitsbestimmung alle Beziehungen zwischen Wörtern aufgedeckt. Anschließend werden neue Texte erzeugt, in denen jeweils ein Wortpaar, welches eine Verbindung zueinander aufweist, entfernt wurde. Die nun erhaltene Sammlung an neuen Texten wird wieder durch das Modell klassifiziert und die neuen Ausgabestärken werden mit der des Ausgangstextes verglichen. Texte mit größeren oder gleichen Ausgabestärken als der des Originals tragen scheinbar nicht zur Klassifikation bei und werden entfernt. Dies geht mit unserer Intuition einher wie das folgende Beispiel erklärt.

Nehmen wir an der Satz *I love this film so much.* wurde durch das Modell mit einem Score von 0.9 bewertet. Das Neuronale Netzwerk ist sich somit sehr sicher, dass dieser Satz das Label 1, also positiv bekommen sollte. Nehmen wir nun an, dass wir den Text so modifizieren: *I love film so much.* und dass das Modell nun einen Score von 0.95 vergibt. Durch das weglassen von *this* ist die Sicherheit, dass es sich hier um ein positives Label handelt, gestiegen. Somit lässt sich annehmen, dass *this* keinen Beitrag zur Klassifikation des Labels leistet. Somit können wir es von unseren Betrachtungen entfernen.

Je größer nun eine Differenz ist, umso wichtiger war das Wortpaar für die Klassifizierung. Mit Hilfe einer Linearisierung der Differenzen und einer Colormap können Wörter somit bezüglich ihrer Wichtigkeit farblich kenntlich gemacht werden. Je wichtiger umso grüner, je unwichtiger, desto blauer.

## 3.2 Datenbeschreibung

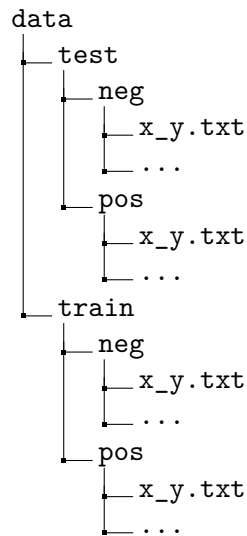


Abbildung 2: Ordnerstruktur des Datensatzes. x ist die Dokumentenid und y ist eine Sternbewertung von Null bis Zehn.

Der im Originalartikel und dieser Replikation verwendete Datensatz ist das Large Movie Review Dataset [4] der Universität Stanford. Dieser umfasst 50.000 Dokumente, darunter 25.000 Trainingsdokumente und 25.000 Testdokumente. Er ist unter <https://ai.stanford.edu/~amaas/data/sentiment/> verfügbar.

Der Datensatz hat eine vorgegebene Ordnerstruktur, siehe Abbildung 2. So befinden sich die Trainingsdokumente und Testdokumente in eigenen Ordnern, wobei positive und negative Dokumente nochmals in eigene Ordner unterteilt sind. Die Dokumente unterscheiden sich stark in der Länge, so gibt es Dokumente mit knapp über 50 Zeichen aber auch solche mit über 13.000 Zeichen. Die Dokumente an sich sind nicht aufbereitet, enthalten englische Alltagssprache und Sonderzeichen.

Fair drama/love story movie that focuses on the lives of blue collar people finding new life thru new love.The acting here is good but the film fails in cinematography,screenplay,directing and editing.The story/script is only average at best.This film will be enjoyed by Fonda and De Niro fans and by people who love middle age love stories where in the coartship is on a more wiser and cautious level.  
It would also be interesting for people who are interested on the subject matter regarding illiteracy.....

Abbildung 3: Beispieltext eines positiven Trainingsdokuments

Bei der Verwendung der Daten zum Training des Modells, wurde der Trainingsdatensatz zusätzlich in einen Validierungsdatensatz aufgeteilt. Dieser umfasst 20 Prozent der Trainingsdaten und somit 5.000 Dokumente.

### 3.3 Hyperparameter

Folgende Hyperparameter wurden gesetzt:

Parameter	Wert
Batch Size	16
Epochs	1
Learningrate	3e-5
Dropoutrate	10%

Die Batch Size definiert die Menge an Trainingsdaten, welche von Netzwerk aufeinmal verarbeitet werden bevor es sich aktualisiert. Diese wurde auf 16 festgesetzt, da das Modell auf der zuvor schon erwähnten Nvidia Grafikkarte trainiert werden sollte. Größere Batch Sizes haben sich als Problem entpuppt, da diese nicht mehr in den Speicher passten. Bei der Epochenanzahl wurde nur eine festgelegt. Beginnend lag dieser Wert bei fünf. Doch verschiedene Durchläufe und eine Einbindung von Early Stopping ergaben, dass das Modell nach der ersten Epoche die beste Leistung aufwies. Mit zunehmenden Training stieg auf dem Trainingsdatensatz die Accuracy und im selben Moment sank der Loss. Doch auf dem Validierungsdatensatz stieg der Loss bei gleichbleibender Accuracy in jeder Epoche. Dies war ein eindeutiges Zeichen für Overfitting.

### 3.4 Implementierung

Der Code zur Implementierung ist abrufbar unter: <https://github.com/maxhenze/Klausurleistung.git> Die wichtigsten verwendeten Packages finden sich in Abbildung 1.

Durch die Verwendung eines Jupyter Notebooks ist der Code interaktiv gehalten. Parameter können angepasst werden und dadurch erzeugte Modelle können sofort neu trainiert werden, falls die Hardware dies zulässt. Falls nicht sind im Projekt zwei fertige Modelle vorhanden, welche eigenständig trainiert wurden.

Diese können im Notebook geladen und verwendet werden. Der entsprechende Flag muss gesetzt werden ob ein Modell trainiert oder geladen werden soll. Je nachdem werden ungebrauchte Codeteile übersprungen.

Der Datensatz wird automatisch heruntergeladen und entpackt, je nachdem ob Daten schon vorhanden sind.

Die Einteilung der Trainingsdaten in Training- und Validierungsmenge wird durch einen Seed bestimmt. Durch unterschiedliches setzen werden unterschiedliche Daten zum Training bzw. zur Validierung benutzt. Hauptsächlich ist dieser Wert aber dazu da, damit kein Dokument in beiden Datensätzen auftaucht.

Ein anderes BERT Modell kann ebenfalls geladen werden, dazu muss nur der passende Link von Tensorflow Hub für die Variable `tfhub_handle_encoder` ersetzt werden.

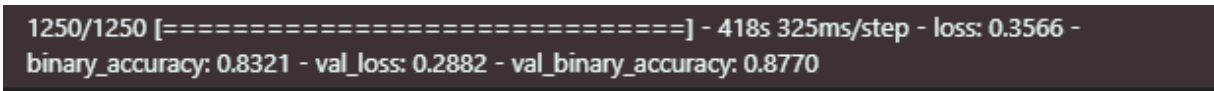
Bei dem zuvor schon erwähnte Optimierer handelt es sich um den AdamW [5] Optimierer, welcher die Parameter des Modells dynamisch während des Lernprozesses anpasst.

### 3.5 Aufbau der Experimente

Zur Durchführung der Experimente des Originalartikels wurden die Dokumente aus der Testdatensatzmenge verwendet. Eine Zelle des Notebooks hat dabei die Aufgabe ein zufälliges Dokument zu wählen. Durch das Nacheinanderausführen der dahinter liegenden Zellen werden die Klassifikations und weiteren Rechenschritte zur Visualisierung automatisch abgearbeitet und man erhält am Ende ein fertiges Bild des eingefärbten Textes. Dabei wird auf der einen Seite der Leave-One-Out und auf der anderen der Leave-N-Out Ansatz durchlaufen, so dass am Ende zwei Texte herauskommen, welche miteinander verglichen werden können.

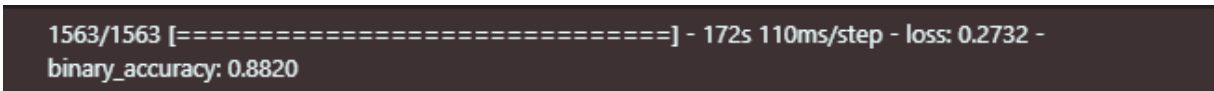
## 4 Ergebnisse

### 4.1 Ergebnis 1



```
1250/1250 [=====] - 418s 325ms/step - loss: 0.3566 -  
binary_accuracy: 0.8321 - val_loss: 0.2882 - val_binary_accuracy: 0.8770
```

Abbildung 4: Trainingsergebnisse des Klassifikationsmodells mit BERT und angehängtem Dropout und Dense Layer.



```
1563/1563 [=====] - 172s 110ms/step - loss: 0.2732 -  
binary_accuracy: 0.8820
```

Abbildung 5: Testergebnisse des Klassifikationsmodells mit BERT und angehängtem Dropout und Dense Layer.

Die Replikation des Klassifikationsmodells ergab nach einer Trainingsepoche ein Modell mit einer Accuracy von 0.877 und einem Loss von 0.2882. Es wird also ein Großteil der Daten richtig klassifiziert. Diese Ergebnisse konnten innerhalb der Testdaten bestätigt werden, siehe Abbildung 4 und 5.

Training mit größerer Epochenanzahl ergab keine Verbesserung der Accuracy, aber einer Steigerung des Losses. Dies deutet auf Overfitting hin.

## 4.2 Ergebnis 2

Die Ergebnisse des Originalartikels konnten durch die Replikation bestätigt werden. Leichte Abweichungen ergeben sich in der genauen Einfärbung der Wörter. Dies lässt sich auf ein unterschiedliches Klassifikationsmodell, welches leicht abweichende Werte produziert oder ein abweichendes Farbschema. Ebenso gibt es Beispiele des Originalartikels, welche falsch vom Modell klassifiziert werden.

Die folgenden Beispiele sind die selben wie im Originalartikel, welche jeweils mit dem Leave-One-Out und Leave-N-Out Ansatz visualisiert wurden.

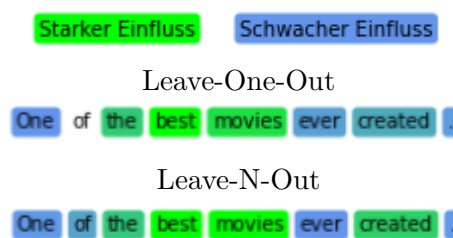


Abbildung 6: Wie im Originalartikel wurde hier mit dem leave-n-out der Zusammenhang von *best* und *movies* besser gekennzeichnet. Im Gegensatz zum Originalartikel hingegen, wurde hier bei beiden das Wort *created* als beeinflussend gekennzeichnet.

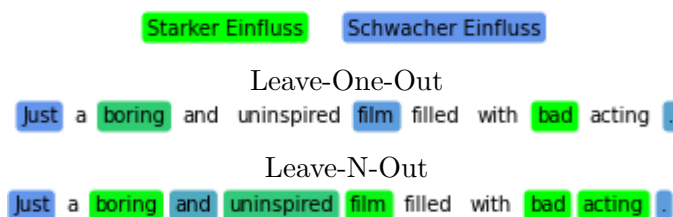


Abbildung 7: In diesem Beispiel konnten die kontextuellen Zusammenhänge von *boring* und *film*, sowie *bad* und *acting* durch den leave-n-out Ansatz besser verdeutlicht werden.

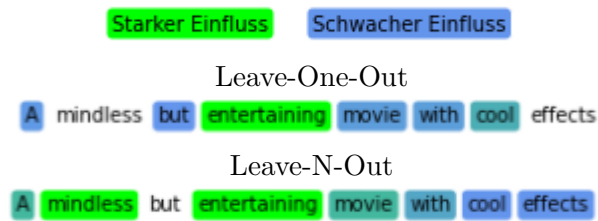


Abbildung 8: Auch in diesem Beispiel findet sich eine Übereinstimmung zum Originalartikel. *mindless* wird hier mit *entertaining* stärker in Beziehung gesetzt.

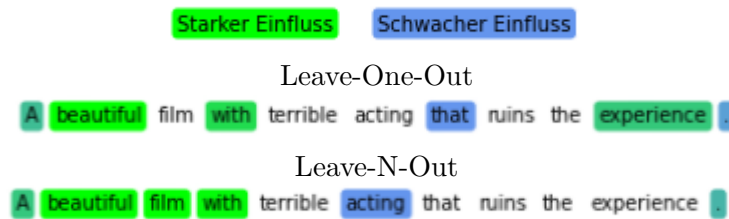


Abbildung 9: Dieses Beispiel bietet gar keine Übereinstimmung mit dem Originalartikel. *terrible* und *ruins* werden gar nicht in Betracht gezogen. Der Fehler scheint hier beim Modell zu liegen. Es klassifiziert bei diesem Beispiel das falsche Label. Vermutlich aufgrund der Einleitung mit *a beautiful film*.

Wie in den Abbildungen 6 bis 8 zu sehen ist, sind die Ergebnisse nahezu äquivalent zu denen des Originalartikels. Bei denen wo das Klassifikationsmodell richtig liegt lässt sich auch die spätere Visualisierung des Originalartikels, in leicht abweichenden Farbnuancen, replizieren. Das Experiment wo das Modell schon im Vorhinein scheitert, siehe Abbildung 9, ergibt auch eine stark abweichende Visualisierung.

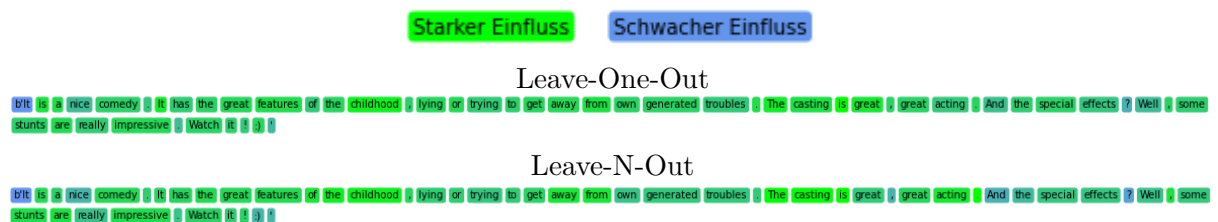


Abbildung 10: Vergleich längerer Texte. Veränderte Einflüsse fallen hier nur in Nuancen auf.

Bei längeren Texten fallen die veränderten Zusammenhänge nicht sofort auf. Ebenso fällt bei längeren Texten auf, dass oft nur einzelne Wörter vom Modell als wichtig interpretiert werden. Wodurch sich Visualisierungen wie in Abbildung 11 ergeben.



Starker Einfluss      Schwacher Einfluss

### Leave-N-Out

b'Someone definitely has it in for The Mind and I can not believe that what I saw on the screen has much to do with it. This is a better and more solid movie than most of the independents I watch all year long. The cinematography displays a genuine love and mastery of the craft and the casting was just fine. I would love to see more of these folks, especially Zeke Rippy. As far as the story and script, I'm not so sure that the negative comments preceding this post were written with the intent of informing anybody else about the movie. The long drawn out nit-picky bashing posts that must have taken hours to compose and are the only comments ever left for any movie on this site by the reviewer, are obvious slander directed at the producers of the film. I don't know the inside story, but it would probably make a good movie. What I do know from being around this biz is, productions that try to make everybody happy usually end up being awful and when the filmmaker has the guts, drive and common sense to kill the babies, someone always ends up with hurt feelings. That's part of the biz too, and one of the finest learning tools available to those truly dedicated to making it in the movie business. Of course, the failures have nothing better to do than to sit at home and write false reviews on the internet as a form of vendetta against those that snubbed them. I read were honest and truthful with them. And that is my best guess to explain the nasty, nasty reviews. In as much as there is a grain of truth behind everything, there is a point to be made, but these exaggerations of the grains are so over the top that they become obviously fictitious. It's not a dasher blood bath if that's what you're looking for, it's more the psychological suspense thriller, which typically is not appreciated by the lowest common denominators out there. The best way to see this movie is to try to expunge any pre-conceived notions, pop it in and let it unfold seriously, characters are defined by their actions and words and when you see what these people do and what they say, I don't think you can come away with the conclusion that these characters were poorly developed, poorly understood perhaps. Overall, I do agree that this is a nicely done, compelling movie. Perhaps I would not have given it a 9 under normal circumstances, but the severe negative comments actually attracted me to the picture. I have a secret love for really bad cinema. To me, the ratings below 5 should be reserved for the shlocky, inept, poorly acted and stupidly written movies. None of that applies here, it is quality movie making with some real talent in there. I gave it the nine to tip the scales back in the correct direction. Watch the movie and tell me I am wrong.

Abbildung 11: Visualisierung langer Texte. Einzelne Wörter fallen hier stark ins Gewicht, wohingegen alle anderen Wörter als nicht stark beeinflussend gekennzeichnet werden.

## 4.3 Zusätzliche Ergebnisse, die nicht im Originalartikel enthalten waren

Zusätzlich zu den im Experimenten des Originalartikels wurde zwei weitere Methodiken des Leave-N-Out Ansatzes implementiert.

Ersteres versucht das Weglassen eines Elternwortes mit allen von ihm abhängigen Kindern. Dabei wird deutlich, dass hierbei zu viele Wörter entfernt werden, wodurch Abhängigkeiten schlechter dargestellt werden. Diese Methodik ähnelt in der Visualisierung stark der Leave-One-Out Methodik.

Letzteres hingegen versucht zusätzlich zum Elternwort, alle Nachfahren dieses Wortes weggelassen. Somit werden auch implizite Verbindungen von Wörtern in Betracht gezogen, welche nicht durch eine direkte Abhängigkeit miteinander verbunden sind. Diese Methodik erzeugt ähnliche Darstellungen wie die Leave-N-Out Methodik des Originalartikels, wobei direkte Abhängigkeiten etwas schwächer visualisiert werden. Indirekte werden hingegen hervorgehoben, wodurch größere Teilabschnitte bzgl. der Farbgebung zusammengefasst werden. Dies ist sichtbar in Abbildung 12

## Starker Einfluss      Schwacher Einfluss

### Leave-N-Out

b'I thought this movie was perfect for little girls . It was about a magical place where Genevieve and all her sisters could do what they wanted to do the most anytime they 'd like . Most little girls would like this story . even though there is the thought of death in it . Although no one dies . the king almost does . but little girls would not understand it . so it adds up to make a perfect story . All the events add up . creating a great plot that can have a meaning if you dig deep enough . This story is perfect for little girls . and since it is a barbie movie . the kids can have more fun with it . especially if they have barbies of their own . Anyone can have fun with it . though . because it is so cute and understandable . Overall . I think this movie is a good movie for everyone . especially little girls . and will give anyone a smile at least once during it . "

### Leave-Ancestors-Out

b'I thought this movie was perfect for little girls . It was about a magical place where Genevieve and all her sisters could do what they wanted to do the most anytime they 'd like . Most little girls would like this story . even though there is the thought of death in it . Although no one dies . the king almost does . but little girls would not understand it . so it adds up to make a perfect story . All the events add up . creating a great plot that can have a meaning if you dig deep enough . This story is perfect for little girls . and since it is a barbie movie . the kids can have more fun with it . especially if they have barbies of their own . Anyone can have fun with it . though . because it is so cute and understandable . Overall . I think this movie is a good movie for everyone . especially little girls . and will give anyone a smile at least once during it . "

### Leave-One-Out

b'I thought this movie was perfect for little girls . It was about a magical place where Genevieve and all her sisters could do what they wanted to do the most anytime they 'd like . Most little girls would like this story . even though there is the thought of death in it . Although no one dies . the king almost does . but little girls would not understand it . so it adds up to make a perfect story . All the events add up . creating a great plot that can have a meaning if you dig deep enough . This story is perfect for little girls . and since it is a barbie movie . the kids can have more fun with it . especially if they have barbies of their own . Anyone can have fun with it . though . because it is so cute and understandable . Overall . I think this movie is a good movie for everyone . especially little girls . and will give anyone a smile at least once during it . "

### Leave-Children-Out

b'I thought this movie was perfect for little girls . It was about a magical place where Genevieve and all her sisters could do what they wanted to do the most anytime they 'd like . Most little girls would like this story . even though there is the thought of death in it . Although no one dies . the king almost does . but little girls would not understand it . so it adds up to make a perfect story . All the events add up . creating a great plot that can have a meaning if you dig deep enough . This story is perfect for little girls . and since it is a barbie movie . the kids can have more fun with it . especially if they have barbies of their own . Anyone can have fun with it . though . because it is so cute and understandable . Overall . I think this movie is a good movie for everyone . especially little girls . and will give anyone a smile at least once during it . "

Abbildung 12: Visualisierung eines Textes mit zusätzlich erdachten Methodiken. In dieser Abbildung werden die Methodiken der Leave-N-Out und Leave-One-Out des Originalartikels mit den selbsterdachten Methodiken Leave-Ancestors-Out und Leave-Children-Out verglichen.

## 5 Diskussion

Der im Originalartikel propagierte Leave-N-Out Ansatz schafft es im Vergleich zum Leave-One-Out Ansatz, kontextuell abhängige Wörter bei der Visualisierung besser in Bezug zu setzen. Die Frage die sich stellt, ist jene, ob die dadurch in Bezug gesetzten Wörter einen besseren Einblick in die Klassifikation eines Neuronalen Netzwerkes ermöglicht ? Natürlich ist es bei Sätzen wie *One of the best movies ever created*. richtig, dass die Wörter *best* und *movie* durch ihren kontextuellen Zusammenhang eine gleichwertige Wichtigkeitsgewichtung erhalten sollten. Dennoch wird durch den Leave-One-Out Ansatz deutlich, dass das Neuronale Netzwerk das Wort *best* stärker für die Klassifizierung benutzt als das Wort *movies*. Dies gibt eine stärkere Einsicht in die Wichtigkeit eines einzelnen Wortes, wohingegen beim Leave-N-Out Ansatz unklar ist ob durch das Weglassen beider Wörter die Klassifizierung schlechter geworden ist oder nur durch das Weglassen des stärker klassifizierten Wortes.

Der Originalartikel war trotz fehlenden Codes replizierbar. Die Aussagen der Autoren gaben einen guten Einblick in die verwendeten Strukturen und Techniken, welche mit Hilfe von state-of-the-art Modellen gut umgesetzt werden konnten. Dennoch sind leichte Abweichungen, durch fehlende Angaben zum BERT Modell, vorhanden. Ebenso ist eine abweichende Farbskala ein ebenso möglicher Grund für leicht verschiedene Darstellungen.

Es wurde deutlich, dass der Originalansatz im Vergleich zu den selbst dargestellten Methodiken, eine bessere Visualisierung von kontextuell abhängigen Wörtern ermöglicht.

### **5.1 Was war einfach ?**

Aufgrund des recht einfachen, propagierten Ansatzes, waren nahezu alle Umsetzungen der Leave-N-Out Methodik nicht schwer zu implementieren. Durch klare Package Benennung und einer einfachen Algorithmusbeschreibung, konnten alle Schritte von der Klassifikation, dem Produzieren neuer Text, der Filterung der Neuklassifikation und der Visualisierung problemlos repliziert werden. So benötigte das Einlesen in die API des jeweiligen Packages eine gewisse Zeit, dennoch handelte es sich dabei nie um zu umfängliche Beschreibungen.

Das angegebene spacy Package ist ein sehr gutes Tool für die Erstellung von Abhängigkeitsgraphen auf denen sich recht einfach navigieren lässt, wodurch die Abhängigkeitsbestimmung dadurch auch nicht besonders schwer viel. Und die Bearbeitung auf den neu generierten Texten viel ebenfalls, durch das sehr umfängliche pandas Package, sehr leicht aus.

### **5.2 Was war schwer ?**

Die Implementierung des Klassifikationsmodells, sowie der allgemeine Umgang mit diesem, war nicht sehr einfach. Durch fehlende Angaben der Autoren zum Modell war beginnend eine gewisse Recherche notwendig, welches BERT Modell nun genau benutzt werden soll. Es wurde zwar grob eine Implementierung der verschiedenen Layer erwähnt, diese war aber nicht ausreichend für die eigenständige Erstellung. Angaben der Daten waren ebenfalls recht sporadisch, wodurch erst eine geeignete Implementationsstruktur erdacht werden musste. Auch die Tatsache, dass keine Angaben zu Hyperparametern gemacht wurden, trug erschwerend bei. Das Finden der richtigen Batch Size und das Einstellen des Trainings auf der Grafikkarte erwies sich als äußerst umständlich, da zu große Batch Sizes für einen `Out Of Memory Error` sorgten und das Training auf einer CPU (8 Kerne) dennoch zu langsam war.

### **5.3 Empfehlungen für die Replizierbarkeit**

Natürlich wäre ein, von vornherein beigereicherter, Code sehr praktisch gewesen. Wobei sich auch hier sagen lässt, dass das Verstehen von fremden Code eine Problematik in sich ist. Genauere Angaben zur Implementierung würden hier schon reichen. Eine genaue Angabe des verwendeten BERT Modells sowie eine Bereicherung der verwendeten Hyperparameter und einer Hardwarebeschreibung würde viel Testen obsolet machen. Zusätzlich dazu, auch wenn nicht unbedingt von Nöten, wäre eine Angabe von verwendeten Funktionen der Packages, welche zur Realisierung des Algorithmus verwendet wurden, sehr bequem.

## **6 Kommunikation mit den Autoren**