Hate Speech Detection in Songtexten mit logistischer Regression

Abschlussprojekt im *Projektseminar: Hate Speech Detection [230013]*

Chiara Larissa Matte
chiara.matte@uni-bielefeld.de
Nina Marianne Meier
n.meier@uni-bielefeld.de

28. März 2022

1 Einführung

Im Rahmen des *Projektseminar: Hate Speech Detection [230013]* haben wir uns mit der Verwendung von *Hate Speech* in Sprache beschäftigt. Wobei wir zunächst in den Begriff *Hate Speech* eingeführt wurden und dann anhand von vier Texten verschiedene Methoden der *Hate Speech Detection* kennenlernen konnten. Zu dem Begriff *Hate Speech* existieren viele Definitionen, von denen einige weniger streng und andere sehr viel strenger sind. Meistens geht es aber darum, dass bestimmte Gruppen das Ziel sind. So ist es auch bei der Definition von [Davidson et al., 2017]:

> „[W]e define hate speech as language that is used to expresses hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group. In extreme cases this may also be language that threatens or incites violence, but limiting our definition only to such cases would exclude a large proportion of hate speech."

Im Folgenden verwenden wir die deutsche Auffassung der Definition von [Davidson et al., 2017]. Wir definieren *Hate Speech* also als eine Sprache, die verwendet wird, um Hass gegen eine bestimmte Gruppe auszudrücken, oder die darauf abzielt, die Mitglieder der Gruppe herabzusetzen, zu demütigen oder zu beleidigen. In extremen Fällen kann es sich auch um Sprache handeln, die mit Gewalt droht oder dazu aufruft. Im nächsten Schritt haben wir uns dann mit einem etwas technischeren Teil beschäftigt, in dem es um die Aufbereitung von Daten und die Analyse ging. Danach haben wir uns um die Annotation sowie die Evaluation gekümmert.

1.1 Themenfindung und verwandte Studien

Als Erstes haben wir uns dann mit der Findung von eigenen Projektideen auseinandergesetzt. Dafür haben wir uns mit Themen wie der Analyse von *Hate Speech* in Instagram-Posts (vgl. [Kruk et al., 2019]), der Erstellung eines eigenen Datenkorpus zum Thema *Hate Speech* (vgl. [Mathew et al., 2020]) oder dem Vergleich von verschiedenen Modellen zur Einordnung von *Hate Speech* beschäftigt. Wir haben dazu vier verschiedene Texte gelesen, die sich alle mit dem Thema *Hate Speech Detection* beschäftigen. Da sich diese maßgeblich an unserer Themenfindung beteiligt haben, möchten wir diese im Folgenden noch einmal kurz zusammengefasst wiedergeben.
Im ersten Artikel von [Mathew et al., 2020]

wird *HateXPlain* vorgestellt, ein Benchmark-Datensatz für die Annotation von *Hate Speech* auf Word- und Phrasenebene, der menschliche Begründungen für die Zugehörigkeit zur Kategorie *Hate Speech* mit einbezieht. Motivation dazu war die immer weiter steigende Verwendung von *Hate Speech* in den sozialen Medien und damit das Interesse daran, diese zu erkennen und zu regulieren. In den bisherigen Erkennungssystemen wird *Hate Speech* oft mit Beleidigungen zusammengefasst und die Texte werden häufig fehlerhaft als hasserfüllt deklariert. Deshalb wollen die Autor*innen sich auf interpretierbare Modelle konzentrieren. Datengrundlage bilden dafür Texte von Twitter und Gab, die zuvor manuell in die drei Kategorien

- *Hate Speech*,
- beleidigend, aber nicht *Hate Speech*,
- keines von beidem

eingeteilt wurden. Zusätzlich wird geschaut, gegen wen genau sich der Hass richtet. Der neu entstandene Datenkorpus enthält drei Annotationstypen:

1. die Einordnung in die drei Kategorien *Hate Speech*, *Offensive* und *None*,
2. die Zielgruppe und
3. ob der Beitrag von mehr als der Hälfte als beleidigend oder hasserfüllt empfunden wird.

In der Hauptstudie wurde jeder Beitrag von drei verschiedenen Personen annotiert und das, was von der Mehrheit gewählt wurde, wurde als Kategorie anerkannt. Die Beiträge, wo alle drei Personen etwas anderes gewählt haben, wurden ausgeschlossen. Zielgruppe für *Hate Speech* waren vor allem die afrikanische, islamische und jüdische Gesellschaft. Die Wörter, die am häufigsten als Begründung für die Entscheidung, dass ein Beitrag *Hate Speech* ist, markiert wurden, waren *„n*gger“*, *„k*ke“* und *„moslems“*. Für die Beiträge, die als *Hate Speech* oder beleidigend gekennzeichnet wurden, haben die Autor*innen die markierten Abschnitte in „Aufmerksamkeitsvektoren“ (boolesche Vektoren) umgewandelt, um die *ground truth attention* zu erzeugen.
Es wurden außerdem verschiedene Modelle mit unterschiedlichen Messgrößen verwendet. Die Messgrößen waren unter anderem Makro-F1-Score und AUROC-Score, um die Leistung des Classifiers bei den Unterscheidungen der drei Gruppen zu messen und BPSN-AUC um die Falsch-Positiv-Rate im Zusammenhang mit der Voreingenommenheit des Classifiers zu bestimmen. Zusätzlich wird noch die Plausibilität, also wie genau der Denkprozess abgebildet wird, gemessen. Alle Modelle werden in zwei Varianten trainiert; eine, die nur mit den Kennzeichnungen *Hate Speech*, *Offensive* und *None* trainiert wird und eine, die mit diesen und dem Modell des *ground truth attention* trainiert wird. Das Ergebnis zeigt, dass die Modelle, die mit der markierten Begründung trainiert wurden, eine bessere Leistung im Bezug auf die Voreingenommenheit gegenüber Zielgruppen aufweisen. Dabei scheint es auch zu helfen, wenn die Zielgruppe innerhalb der markierten Textabschnitte auftaucht. Die Modelle, welche die besten Leistungen in Bezug auf die Leistungsfähigkeit zeigten, arbeiteten bei der Messung von *Precision* und *Recall* nicht so gut. Grundsätzlich konnten die Autor*innen feststellen, dass die Messung der Leistungsfähigkeit alleine nicht ausreicht, um zu beurteilen, wie gut ein Modell ist.

Der nächste Text von [Kruk et al., 2019] beschäftigt sich mit der Integration von Text und Bild in Instagram-Posts und der Bestimmung der Intention in solchen multimodalen

Daten. Dafür haben sich die Autor*innen drei Klassifikationsschemen überlegt. Diese sollen die Absicht des Verfassenden hinter dem Post untersuchen, die kontextuelle Beziehung zwischen den wörtlichen Bedeutungen von Bild und Text sowie die semiotische Beziehung zwischen den Bedeutungen von Text und Bild. Das Ziel der Studie ist es, einen computergestützten Rahmen für die Erforschung von Bild-Text-Zusammenschlüssen und der neuen Bedeutung, die sich daraus ergibt, zu schaffen.
Das erste Klassifikationsschema bezieht sich auf die Absicht des Verfassenden, die sich hinter einem Post verbirgt. Dazu haben sich die Autor*innen auf Grundlage vorheriger Untersuchungen folgende acht mögliche Absichten überlegt: **befürwortend**, **fördernd**, **selbstdarstellend**, **aussagekräftig**, **informativ**, **unterhaltend**, **provozierend/ kontrovers**.
Das zweite Klassifikationsschema bezieht sich auf die kontextuelle Beziehung zwischen den wörtlichen Bedeutungen von Bild und Text. In Anlehnung an eine andere Studie haben sich die Autor*innen folgende drei Kategorien überlegt:

- **minimale Beziehung**, d.h. die wörtlichen Bedeutungen von Text und Bild überschneiden sich nur wenig,
- **enge Beziehung**, d.h. die wörtlichen Bedeutungen zwischen Text und Bild überschneiden sich erheblich,
- **transzendente Beziehung**, d.h. die wörtliche Bedeutung des einen Elements greift die des anderen auf und erweitert sie.

Das dritte Klassifikationsschema untersucht die semiotische Beziehung zwischen den Bedeutungen von Text und Bild. Dazu übernehmen die Autor*innen folgende drei Kategorien aus Vorgängerstudien:

- **divergierende Beziehung**, d.h. die Semiotik von Bild und Text weichen voneinander ab und gehen in verschiedene Richtungen,
- **parallele Beziehung**, d.h. Bild und Text tragen unabhängig voneinander dieselbe Bedeutung bei,
- **additive Beziehung**, d.h. die Semiotik von Text und Bild verstärken sich gegenseitig.

Der Datensatz besteht aus 1299 Instagram-Posts, dabei wurde Wert darauf gelegt, dass es zu jeder der acht Absichten passende Posts gibt. Anhand der Hashtags wurde entschieden, mit welcher Absicht ein Post erstellt wurde. Der erste Schritt bestand darin, zu prüfen, ob ein Post überhaupt sowohl Text, als auch Bild enthält. Für Posts, wo das der Fall war, wurden die drei Klassifikationsschemen angewandt. Auf die Daten wurde ein *Convolutional Neural Network* angewandt, um sie zu encodieren. Das Modell nimmt als Input entweder einen Text oder ein Bild oder beides zusammen. Das Modell sollte dann Vorhersagen treffen, welche Merkmale aus den drei Klassifikationsschemen auf den Text, das Bild und beides zusammen zutreffen. Die Ergebnisse zeigen, dass für das Klassifikationsschema mit der Absicht Bilder informativer als Texte sind, zumindest unter Verwendung des *word2vec-Encodierers*. Bei dem Text, der mit *ELMo* encodiert wurde, war es andersherum. Auch bei der Klassifizierung der kontextuellen Beziehung hat *ELMo* die Leistung verbessert, bei der semiotischen Beziehung allerdings nicht. Das könnte daran liegen, dass bei *ELMo* der Satzkontext mit einbezogen wird und dieser zur Klassifizierung der semiotischen Beziehung nicht wichtig ist.
Die Ergebnisse stützen die Annahme, dass die Bedeutungen von Text und Bild in

den sozialen Medien häufig voneinander abweichen und somit als Ganzes eine neue Bedeutung schaffen.

In dem dritten Artikel *The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes* von [Kiela et al., 2020] wird ein Datensatz vorgestellt, der dabei helfen soll, Hass in multimodalen Inhalten zu erkennen. Dazu ist es notwendig, Text und Bild als Einheit zu betrachten und nicht separat. Der Datensatz besteht aus 10.000 „Hateful Memes“, d.h. es gibt sowohl Bild als auch Text. Die Bilder der Memes wurden durch ähnliche Bilder ersetzt. In den Fällen, wo das nicht möglich war, weil es z.B. keine passenden Bilder gab, wurden die Memes aussortiert. Dass diese Memes Hass ausdrücken, wurde von speziell darauf trainierten Leuten unter Verwendung einer Definition von *Hate Speech* entschieden. Unter den Memes im Datensatz gibt es aber auch welche, die ähnlich zu „Hateful Memes“ sind, aber keinen Hass ausdrücken. Die Existenz dieser Memes soll dabei helfen, Voreingenommenheit und falsch-positive Ergebnisse zu verhindern. Multimodale Inhalte sind für Algorithmen schwerer zu erkennen, weil sie meistens einen realen Kontext und so etwas wie einen gesunden Menschenverstand erfordern. Deshalb müssen die verschiedenen Elemente zu einem möglichst frühen Zeitpunkt im Klassifizierungsprozess wieder zusammengeführt werden. Dazu gibt es *early-fusion-systems*, welche beide Elemente zusammenführen, bevor sie versuchen diese zu klassifizieren und *late-fusion-systems*, die erst beide Elemente für sich klassifizieren und dann versuchen beides zusammen zuführen.
Zum Schluss wird noch zu der „Hateful Memes Challenge“ aufgerufen, die darin besteht, dass Interessierte Modelle entwickeln können, die anhand des „Hateful Memes“-Datensatzes trainiert werden.

Im letzten und für uns interessantesten Artikel von [Davidson et al., 2017] wird ein Modell zur automatisierten Erkennung von *Hate Speech* beschrieben, das sich auch mit dem Problem der beleidigenden Sprache befasst. In den meisten Studien zum Thema *Hate Speech Detection* wird *Hate Speech* mit beleidigender Sprache gleichgesetzt. Das möchten die Autor*innen aber nicht tun, deshalb wollten sie ein Modell entwickeln, das zwischen diesen beiden Kategorien unterscheiden kann. Ein Problem bei der Erkennung von *Hate Speech* in Texten ist, dass Abschnitte häufig als *Hate Speech* angesehen werden, weil dort Schimpfwörter auftreten oder beleidigende Sprache verwendet wird. *Hate Speech* ist aber nicht gleichzusetzen mit beleidigender Sprache. Ein weiterer Punkt, der die Erkennung erschwert, besteht darin, dass bestimmte Wörter in bestimmten Kontexten als beleidigend wahrgenommen werden, in anderen Kontexten aber nicht. Als Beispiel wird hier „gay“ genannt. Um genauere Ergebnisse zu erhalten, wurden Features entwickelt, die den Kontext mit einbeziehen.
Als Grundlage diente den Autor*innen eine Zusammenstellung von Wörtern und Phrasen, die von Internetnutzer*innen als *Hate Speech* festgelegt wurden. Der nächste Schritt bestand darin, Tweets auf Twitter zu durchsuchen und diejenigen Tweets zu sammeln, welche Wörter oder Phrasen aus der Zusammenstellung beinhalten. Jeder Tweet aus dieser Sammlung wurde daraufhin von mehreren Personen begutachtet und einer der drei Kategorien

- *Hate Speech*,
- beleidigend, aber nicht *Hate Speech*,
- keines von beidem

zugeordnet. Nur 5% der Tweets wurden dabei *Hate Speech* zugeordnet, obwohl

diese Sammlung nur Tweets beinhaltet, die auf Grundlage der Zusammenstellung von Ausdrücken, die Internetnutzer*innen als *Hate Speech* festgelegt haben, ausgewählt wurden.
Bei dem maschinellen Prozess wurden zuerst alle Wörter gesäubert. Danach wurden Bigramme, Trigramme und Unigramme, also Zusammenschlüsse aus zwei bzw. drei aufeinanderfolgenden Wörtern oder einem einzelnen Wort erstellt und anhand ihres *Tf-idf-Maßes* gewichtet. Als nächstes haben die Autor*innen *Part-of-Speech-Tagging* angewandt, um den Wörtern ihre grammatischen Kategorien zuzuordnen. Außerdem haben sie die Qualität der Tweets anhand ihrer Lesbarkeit überprüft. Dann wurde die Stimmung der Tweets unter Verwendung eines *sentiment lexicons* erfasst. Zusätzlich wurden Hashtags und Retweets gezählt sowie die Wörter und Buchstaben in den einzelnen Tweets.
Die Autor*innen haben logistische Regression verwendet, um vorherzusagen, zu welcher der drei Kategorien ein Tweet gehört. Dann haben sie für jede der drei Kategorien einen eigenen *Classifier* gebaut. Die Ergebnisse zeigen, dass fast 40% der *Hate Speech* falsch eingestuft wurde. Das Modell scheint Tweets als weniger hasserfüllt oder beleidigend einzustufen, als die Personen, die die Tweets zuvor zuordnen sollten. Nur bei ganz wenigen Tweets war es genau andersherum. 5% der beleidigenden Tweets wurden fälschlicherweise als *Hate Speech* eingestuft und 2% der Tweets als solche, die weder Beleidigungen noch *Hate Speech* enthalten. Tweets, denen die größte Wahrscheinlichkeit zugesprochen wurde, zur Kategorie *Hate Speech* zugeordnet zu werden, beinhalten meistens rassistische oder homophobe Beleidigungen. Tweets, bei denen die Wahrscheinlichkeit am höchsten eingestuft wurde, dass sie zur Kategorie Beleidigungen gehören, waren allgemein weniger hasserfüllt. Tweets, die eigentlich unter die Kategorie *Hate Speech* fallen und fälschlicherweise als weder beleidigend, noch hasserfüllt eingestuft wurden, enthalten meistens keine Wörter, die für Hass oder Beleidigung stehen. In vielen anderen Studien bestand das größte Problem darin, dass beleidigende Sprache fälschlicherweise als *Hate Speech* eingestuft wurde. Das war bei dieser Studie nicht so, nur 5% der beleidigenden Sprache wurde falsch eingestuft. Tweets, denen eine positive Stimmung und höhere Lesbarkeitswerte zugeordnet wurden, gehörten eher zur Kategorie „Weder Beleidigung, noch Hate Speech“.
Zusammenfassend konnten die Autor*innen feststellen, dass es bei der Erkennung von *Hate Speech* nicht reicht, die Tweets nur mit einem Lexikon, das bestimmte Wörter und Ausdrücke enthält, abzugleichen. Dann wird Vieles als *Hate Speech* eingestuft, was gar keine *Hate Speech* ist. Es gibt zwar Wörter, die allgemein nur im Bereich *Hate Speech* angewendet werden, aber auch viele, wo der Blick auf den Kontext erst entscheidet, ob es *Hate Speech* ist oder nicht. Um *Hate Speech* noch besser zu erkennen, kann es hilfreich sein, Daten zu sammeln, in denen *Hate Speech* ohne die Verwendung einschlägiger Wörter stattfindet. Außerdem könnte man die Verfasser*innen noch betrachten und z.B. mit einbeziehen aus welchem sozialen Umfeld sie stammen.
Besonders der Text von [Davidson et al., 2017] hat uns bei der Themenfindung sehr geholfen. Inspiriert durch die Texte haben wir beschlossen, uns näher mit dem Thema *Hate Speech Detection* zu befassen. Die meisten anderen Studien befassen sich mit *Hate Speech* in den sozialen Medien, aber auch in anderen Bereichen kann *Hate Speech* auftreten, zum Beispiel in Songtexten. Wir interessieren uns dafür, ob im Bereich der Musik ähnliche Untersuchungen von *Hate Speech* möglich

sind. Deswegen möchten wir uns einige Songs anschauen und unter Verwendung eines geeigneten Modells prüfen, ob diese Songs hasserfüllt, beleidigend oder keines von beidem sind.

1.2 Erste Schritte

In einem ersten Versuch suchten wir nach einem geeignten Datensatz, den wir verwenden konnten. Wir mussten überprüfen, ob die von uns gefundenen Datensätze für unser Vorhaben geeignet waren. Wir überlegten auch kurz, ob es vielleicht möglich wäre, einen eigenen Datensatz mit circa 200 Wörtern zu erstellen, verworfen diese Idee jedoch wieder, da es uns weniger kompliziert vorkam einen fertigen Datensatz zu verwenden.
Bei unserer Suche stießen wir auf das *Million Song Dataset* von [Bertin-Mahieux et al., 2011], in dem ein Datensatz namens *musiXmach* zur Verfügung gestellt wurde. Bei diesem handelt es sich um eine Songtextsammlung im *Bag-of-Words*-Format. Dieser Datensatz bietet eine Liste mit den 5000 am häufigsten in Songtexten vorkommenden Wörtern, diese Wörter sind bereits gestemmed. Bezeichnet werden diese Wörter als *Music-Top-Words*. Diese 5000 *Music-Top-Words* machen etwa 92% des Gesamtvorkommens aus. Es gibt eine Liste mit allen Wörtern und deren Anzahl in der gesamten Songtextsammlung. Außerdem gibt es eine Liste mit Songtitel und den Wörtern, die im jeweiligen Songtext vorkommen und zu einem der *Music-Top-Words* zählen. Bei diesen Wörtern wird zusätzlich die Anzahl angegeben, allerdings nur in Bezug auf den jeweiligen Songtext. Hier wurden bewusst nicht alle im Songtext vorkommenden Wörter angegeben, sondern nur die *Music-Top-Words*, weil die Angabe des kompletten Songtexts zu Urheberrechtsproblemen führen könnte. Die Songtitel stehen nicht ausgeschrieben als *string* in der Liste, sondern mit ihrer ID. Der Grund dafür ist die möglicherweise einfachere Verwendung dieses Formats. Um nachverfolgen zu können für welchen Songtitel die jeweilige ID steht, gibt es eine weitere Liste, in der die Zuordnung sichtbar wird. In der Liste wird zusätzlich noch der Interpret angegeben. Auch die im Songtext vorkommenden *Music-Top-Words* stehen nicht in ihrer Stringform in der Liste, sondern mit ihrem Index, der sich auf die Liste bezieht, die es für die *Music-Top-Words* gibt. Der Index startet bei 1 und nicht bei 0. Wie sich der *musiXmatch-Datensatz* zusammensetzt ist nochmal in Abbildung 9 dargestellt.
Unser Plan war es, aus dem Datensatz einen eigenen Datensatz anzulegen und wir hatten praktisch schon angefangen, einen kleinen Teil davon zu erstellen. Das Problem an diesem Datensatz war jedoch das Zurückordnen von Zahlen und Wörten. Da wir ein Format brauchten, in dem die *Music-Top-Words* ausgeschrieben sind, hätte unsere Vorgehensweise darin bestehen müssen, die Indexe wieder in Wörter umzuwandeln. Mit diesen Wörtern hätten wir dann eine Tabelle erstellt, in der es eine Spalte für den Songtitel, eine für den Interpreten, eine für die Wörter und deren Anzahl und eine für die Klasse gibt. Dargestellt ist das nochmal in Abbildung 10.
Auch mit dieser Vorarbeit hätten wir nicht alle Probleme gelöst. Wir hätten mit diesem Datensatz und diesem Format schließlich nicht leicht voraussagen können, welches Wort zu welchem Songtext gehört. Dafür hätten wir umständlich eine neue Funktion bauen müssen, die durch diverse Umwandlungen zu dem passenden Song geführt hätte. Wir hätten also nur vorhersagen können, dass sich in dem *Bag-of-Words*-Datensatz Wörter befinden, die als *Hate Speech* angesehen werden, aber weder den Kontext noch die Häufigkeit in die Klassifizierung miteinbeziehen können.

Deswegen haben wir uns letztendlich gegen diesen Datensatz entschieden, obwohl wir schon einiges an Zeit in die Bearbeitung gesteckt haben. Unsere Suche führte uns schließlich auf die Seite *Kaggle*, wo wir den Datensatz **Audio features and lyrics of Spotify songs** von [Nakhaee, 2020] fanden.

2 *Hate-Speech*-Datensatz

Der Datensatz *Audio features and lyrics of Spotify songs* besteht aus 18.454 Songs. Zu jedem Song werden Informationen bereitgestellt, darunter Interpret, Album, Audiomerkmale (z. B. Lautstärke), Liedtexte, die Sprache der Liedtexte, Genres und Sub-Genres. Da wir uns in unserem Projekt ausschließlich auf englischsprachige Texte beziehen wollen, haben wir aus dem Datensatz von [Nakhaee, 2020] einen neuen Datensatz, der besser zu unserem Vorhaben passt, gebaut. Dafür haben wir zunächst die Indexe derjenigen Songs, die nicht das Attribut englischsprachig haben, rausgefiltert und in eine Liste geschrieben. Als Nächstes haben wir eine Funktion geschrieben, die uns eine beliebige Spalte aus dem Datensatz in Form eines Tupels, bestehend aus Index und Inhalt wiedergibt, z.B. für die Spalte artist [(0, 'Barbie's Cradle'), (1, 'Steady Rollin'),...(18452, 'Father MC'), (18453, 'Moonstar88')]. Unsere nächste Funktion nimmt diese Tupel entgegen, prüft ob die Indexe in der Liste mit den Songs, die nicht das Attribut englischsprachig haben, auftauchen und - sollten die Indexe nicht in der Liste auftauchen - schreibt die Inhalte in eine neue Liste. Unter Verwendung dieser Listen haben wir dann einen neuen Datensatz erstellt. Dieser rein englische Datensatz besteht aus 15.405 Songs und enthält nur noch folgende auserwählte Informationen zu jedem Song:

1. den Songtitel,
2. den Interpret,
3. den Lyrics,
4. das Genre der Playlist,
5. das geschätzte Gesamttempo des Songs in Beats pro Minute (BPM),
6. die Energie als ein Maß zwischen 0,0 und 1,0, dieses Maß ist ein Wahrnehmungsmaß für Intensität und Aktivität,
7. die Gesamtlautstärke des Songs in Dezibel (dB).

Um zu überprüfen, ob in dem Datensatz überhaupt Songs vorkommen, die möglicherweise hasserfüllt sind, sind wir alle Songtexte durchgegangen und haben geprüft, ob sie n-Gramme aus dem *Hate-Speech*-Lexikon (vgl. Abschnitt 3) enthalten. Zuerst hatten wir die Idee, Künstler und Songtexte zu suchen, die auf der Zensurliste stehen und damit sehr wahrscheinlich als *hateful* angesehen werden können. Nach einer wenig erfolgreichen Suche bemerkten wir schließlich, dass in einem Spotify-Datensatz sehr wahrscheinlich keine zensierten Künstler oder Songtitel auftauchen würden, weil diese eventuell gar nicht auf Spotify gespielt werden dürften. Aus diesem Grund haben wir uns dann für das Lexikon als Indikator entschieden. Von den Songtexten, die eines oder mehrere der n-Gramme aus dem Lexikon enthalten, wurden wieder die Indexe in eine Liste geschrieben. Da die Indexe der Songtexte, die mehr als ein n-Gramm enthalten, auch mehrmals in der Liste vorkamen, mussten wir die doppelten Zahlen aus der Liste entfernen. Danach bestand die Liste aus 901 Zahlen, also 901 Songs aus dem Datensatz enthalten n-Gramme aus dem *Hate-Speech*-Lexikon. Der rein englische Datensatz war für den Rahmen unseres Projekts noch zu groß, deshalb haben wir uns entschieden einen kleinen Datensatz zu bauen, bestehend

aus den 901 Songs mit den n-Grammen und weiteren 901 zufällig ausgewählten Songs. Dazu haben wir uns alle Zahlen zwischen 0 und 15.404 ausgeben lassen und die Zahlen entfernt, die schon in der Indexliste stehen. Aus diesen Zahlen haben wir uns dann 901 zufällige Zahlen ausgeben lassen. Nun haben wir uns zu jeder Spalte aus dem englischen Datensatz den Inhalt zu der jeweiligen Indexzahl ausgeben lassen. Daraus wiederum haben wir unseren endgültigen Datensatz erstellt, der aus 1766 Songs besteht. Ein Ausschnitt dieses Datensatzes ist in Abbildung 1 zu sehen.

3 Lexikon

Wir verwenden das Lexikon von [Davidson et al., 2017]. Das Lexikon enthält 178 n-Gramme, die als Anzeichen für *Hate Speech* betrachtet werden. Um dieses Lexikon zu erstellen, nahmen [Davidson et al., 2017] die n-Gramme der Länge 1-4, die in ihren eigenen gelabelten Twitter-Daten enthalten waren, und berechneten für jedes n-Gramm den Anteil der Tweets, die es enthielten und von den menschlichen Codierern als *Hate Speech* eingestuft wurden. Zum Beispiel ist das n-Gramm *„you a lame b*tch“* in 55,6% der Tweets enthalten, die von den Codierern als *Hate Speech* eingestuft wurden. Ein Ausschnitt des Lexikons ist in Abbildung 2 zu sehen.

4 Weiteres Vorgehen

Nachdem wir uns für den *Spotify-Datensatz* entschieden haben, mussten wir unser Vorgehen noch einmal etwas umplanen. Wir wollten jetzt nicht mehr mit dem *Bag-of-Words*-Format arbeiten, sondern nun das *Tf-idf*-Format verwenden. Außerdem mussten wir unsere Fragestellung noch einmal klarer defnieren, auch wenn sich diese im weiteren Projektverlauf immer wieder leicht veränderte und unseren Ergebnissen und Möglichkeiten anpasste. Wir wollen nun herausfinden, ob sich das Format „Songtext“ mit dem logistischen Regressions-Modell von [Davidson et al., 2017] gut klassifizieren lässt. Der Hauptunterschied zwischen den Daten von [Davidson et al., 2017] und unseren Daten besteht darin, dass bei uns viel längere Texte enthalten sind. Uns interessiert also, wie gut sich das Modell auf längere Texte trainieren lässt. Schließlich mussten wir uns noch um die bevorstehende Annotation des Datensatzes kümmern und dafür alles vorbereiten. Bei unserem Vorgehen überlegten wir, welches die beste Möglichkeit wäre, die Daten zu annotieren und ob es möglicherweise schon Tools für das automatische Teilen von Songtexten gibt, die wir nutzen könnten. Da wir bei unserer Suche nicht wirklich fündig wurden, einigten wir uns auf den im nächsten Abschnitt beschriebenen Ablauf.

5 Annotation

Damit sich das Modell von [Davidson et al., 2017] auf unseren Datensatz anwenden lässt, muss jeder Songtext mit einem der drei Label

- 0 = *Hate Speech*,
- 1 = *offensive language*,
- 2 = *neither*

versehen werden. Um das zu bewerkstelligen haben wir zunächst jeweils 50 Songtexte zusammen mit dem Interpret und dem Songtitel in ein PDF-Dokument geschrieben. Diese PDF-Dokumente, insgesamt 36 Stück, haben wir dann jeweils an eine Person aus unserem Bekanntenkreis gegeben. Zusätzlich zu dem Dokument haben die Personen eine Anleitung, zu sehen in Abbildung 11, bekommen. Diese

	Unnamed: 0	track_name	track_artist	lyrics	genre	tempo	energy	loudness
0	0	Baby It's Cold Outside (feat. Christina Aguilera)	CeeLo Green	I really can't stay Baby it's cold outside I'v...	r&b	118.593	0.378	-5.819
1	1	Changes	2Pac	I see no changes, wake up in the morning and I...	rap	111.115	0.657	-6.722
2	2	Laps	Zotiyac	I'mma make your CXRPSE dance Ugh, hop in that ...	rap	140.132	0.453	-9.965
3	3	Hot	Confetti	(Yeah) I'm the new truth, the crypto Erry'body...	pop	168.015	0.908	-3.883
4	4	Love Sosa	Chief Keef	Fuckers in school telling me, always in the ba...	rap	131.965	0.413	-8.193
...	...	...	...	...	...	...	...	...
1761	1761	Coastin'	Cali Life Style	Mike G, why don't you drop us somethin' else? ...	latin	169.892	0.473	-11.115
1762	1762	I'm A G	Rick Ross	Uh I wear a gun like a girdle Bullet proof car...	rap	103.239	0.685	-6.835
1763	1763	Slutty Girls	Mr. Knightowl	I was riding in my rola That's when I spotted ...	latin	114.992	0.555	-6.367
1764	1764	Work REMIX (feat. A$AP Rocky, French Montana, ...	A$AP Ferg	I gotta close the window before I record 'Caus...	rap	130.009	0.733	-5.077
1765	1765	I'll Do 4 U (Re-Recorded / Remastered)	Father MC	(Would you do for me) Sweetheart (Would you do...	r&b	109.536	0.666	-4.920

Abbildung 1: Unser endgültiger Datensatz

	ngram	prophate
0	allah akbar	0.870
1	blacks	0.583
2	chink	0.467
3	chinks	0.542
4	dykes	0.602
...	...	...
173	nigga you a lame	0.556
174	niggers are in my	0.714
175	wit a lame nigga	0.556
176	you a lame bitch	0.556
177	you fuck wit a	0.556

Abbildung 2: Das n-Gramm-Lexikon aus [Davidson et al., 2017]

Anleitung enthält unsere Definition zu dem Begriff *Hate Speech*, eine Tabelle mit möglichen Zielgruppen und schrittweise die Aufgabenstellung.
Die Aufgabenstellung bestand darin, sich die Songtexte kurz durchzulesen und dann anhand der Definition zu entscheiden, zu welcher Kategorie der Song gehört. Jeder Songtext wurde nur von einer Person annotiert und das Label, das diese Person ausgewählt hat, haben wir übernommen. Um einen kleinen Einblick in die Zuverlässigkeit dieser Annotationsmethode zu bekommen, haben wir beide jeweils in einem PDF-Dokument alle Songtexte annotiert und unser Label mit dem bereits vergebenen verglichen. Dabei sind wir auf rund 71% bzw. 68% Übereinstimmung gekommen. Es traten mehrere Fälle auf, in denen die Annotierenden sich nicht auf ein Label festlegen wollten. In diesen Fällen haben wir die stärkere Kategorie als Label festgelegt. Das bedeutet in den Fällen, wo die Entscheidung zwischen *neither* und *offensive* lag, haben wir *offensive* ausgewählt und in den Fällen, wo es um *offensive* und *Hate Speech* ging, haben wir *Hate Speech* ausgewählt. Wir nahmen an, dass die Annotierenden einen Grund dafür hatten, auch die stärkere Kategorie in Betracht zu ziehen und wenn es einen Grund gibt, der für das stärkere Label spricht, dann sollte auch dieses Label gewählt werden. Außerdem gibt es Songs, bei denen in der Spalte für den Songtext nicht der Songtext, sondern ein Bemerkung steht. Diese Bemerkung besagt meistens „This music does not contain words.“ oder „Lyrics for this song have yet to be released. Please check back once the song has been released.“ An den Stellen, wo diese Bemerkungen auftreten, haben wir so getan, als wäre der Text der Songtext und haben das Label auf den Wortlaut der Bemerkung bezogen. In der Regel wurden die Bemerkungen also als *neither* eingestuft. Des Weiteren bestehen nicht alle Songtexte ausschließlich

aus englischen Wörtern. In vielen Songtexten kommen rhythmische Laute, wie „Rat-a-tat-tat-tat-tat“, „Wee-ooh-wee-ooh-wee“ oder „badada-bam-bam“ vor. Zusätzlich gibt es Songtexte, in denen auch Wörter oder Phrasen aus anderen Sprachen verwendet werden, z.B. diese Phrase, „Eko dun mawole Toba ti lowo lapo“, die wahrscheinlich aus einer afrikanischen Sprache stammt. Bei Songs mit vermischten Sprachen haben wir die Annotation nur auf den englischen Teil bezogen und anhand dessen das Label vergeben.

6 Modell

Wir haben das finale Modell nach [Davidson et al., 2017] verwendet. Das Modell besteht aus logistischer Regression mit L2-Regularisierung und sagt für jeden Songtext die wahrscheinlichste Kategorie vorher.

6.1 Features

So wie [Davidson et al., 2017] auch, haben wir jeden Songtext kleingeschrieben und gestemmed. Ebenso haben wir viele Leerzeichen durch eine Instanz ersetzt und Interpunktion, sowie überflüssige Leerzeichen entfernt. Aus den Merkmalen haben wir dann Unigramme, Bigramme und Trigramme erstellt, die nach ihrer **Term Frequency Inverse Document Frequency** (TFIDF) [1] gewichtet sind, also nach ihrer Vorkommenshäufigkeit in einem Songtext gemessen an ihrer Vorkommenshäufigkeit in allen Songtexten. Ein weiteres Feature, das wir von [Davidson et al., 2017] übernommen haben, ist die *Sentiment Analysis* mit *vaderSentiment*. Das ist ein lexikon- und regelbasiertes Analyse-Tool, das speziell auf die Stimmungen in sozialen Medien abgestimmt ist. Das Tool gibt Auskunft darüber, wie positiv, negativ oder neutral eine Stimmung ist. Mit dem *Compound-Score* wird dann die Summe aus den drei Eigenschaften berechnet und die endgültige Stimmung als ein Wert zwischen -1 und +1 festgelegt.
Außerdem haben wir auch die Silben in den Wörtern gezählt und die durchschnittliche Silbenanzahl in einem Songtext berechnet. Zusätzlich haben wir uns die Anzahl der Buchstaben, der insgesamt vorkommenden Wörter in einem Songtext und der verschiedenen Wörter ausgeben lassen. Um die Qualität der Texte zu beurteilen, haben wir uns auch wie [Davidson et al., 2017] den Lesbarkeits-Index unter Verwendung des Flesch-Kincaid-Lesbarkeitstests [2] ausgeben lassen. Um dann auch noch syntaktische Informationen zu bekommen, haben wir *Part-of-Speech-Tagging* mit *NLTK* angewandt.

6.2 Ausführen des Modells

Wir haben zunächst 30% der Daten als Testset zurückgehalten und nur 1236 Songs zum trainieren verwendet. In dem Modell werden zwei Matrizen erstellt. X beinhaltet die Features und y die wahren von unseren Bekannten vergebenen Label. Dann wird zunächst logistische Regression mit L1-Regularisierung angewendet, um Überanpassung des Modells zu verhindern. Denn durch die L1-Regularisierung wird der Koeffizient des weniger wichtigen Merkmals auf Null geschrumpft und somit einige Merkmale vollständig entfernt. Dies funktioniert also gut

[1] Bei der TFIDF bekommen Wörter, die in einem der betrachteten Texte häufig vorkommmen, einen höheren Rang, als Wörter, die häufig in allen betrachteten Dokumenten vorkommen.

[2] Der Flesch-Kincaid-Lesbarkeitstest besteht aus zwei verschiedenen Tests, die anzeigen sollen, wie schwierig ein Text zu verstehen ist. Die beiden Tests, die Flesch Reading-Ease und die Flesch-Kincaid Grade Level nehmen zwar dieselben Parameter, Wort- und Satzlänge, entgegen, haben aber unterschiedliche Gewichtungsfaktoren.

für die Feature-Auswahl. Bei der logistischen Regression wird die Kategorie vorhergesagt, die am wahrscheinlichsten ist, keine konkreten Werte, wie bei der linearen Regression. Um die Klassifizierung durchzuführen, wird außerdem die SVC-Methode (*Linear Support Vector Classifier*) angewandt. Diese beiden Methoden schnitten bei der Untersuchung von [Davidson et al., 2017] am besten ab. Um die beste Methode zu finden, wurde zuvor eine fünffache Kreuzvaliderung mit *GridSearchCV* durchgeführt. *GridSearchCV* ist nützlich, wenn nach dem besten Parameter für das Zielmodell und den Datensatz gesucht wird. Bei dieser Methode werden mehrere Parameter durch Kreuzvalidierung getestet und die besten Parameter können extrahiert werden, um sie für ein Vorhersagemodell zu verwenden. Die anderen drei Kandidaten waren, basierend auf den Modellen aus früheren Arbeiten, *Naive Bayes*, Entscheidungsbäume und *Random Forests*.
Um Informationen aus dem Modell zu erhalten, werden danach verschiedene Abfragen gemacht. Zum Beispiel lassen wir uns die finale Feature-Liste ausgeben. In dieser Liste tauchen n-Gramme, wie *'afraid'* oder *'get outta'* auf. Außerdem taucht das POS-Tag *'NN WP'* auf, also ein Singular-Nomen in Verbindung mit einem wh-Pronomen, wie *who* oder *what*. Zusätzlich sind einige der weiteren Features aus Abschnitt 6.1 gelistet, unter anderem die Silbenanzahl und die Wortanzahl.

7 Ergebnisse

Wie in Abbildung 3 zu sehen ist, konnte das Modell die Kategorie *Neither* mit 81% korrekt vorhergesagten Labels am besten erkennen. Anders als bei den Ergebnissen von [Davidson et al., 2017] wurde die Kategorie *Hate Speech* mit 55% ein bisschen besser erkannt, als die Kategorie *Offensive* mit 50%. Das Modell hat 46% der Kategorie *Hate Speech* falsch zugeordnet, davon zu drei Vierteln zur Kategorie *Offensive*. Hinsichtlich der Kategorie *Offensive* hat das Modell 23% falsch als *Hate Speech* eingestuft und 27% falsch als *Neither*. Bei der Kategorie *Neither* ist der Anteil der falsch zugeordneten Kategorien mit 19% am geringsten. Der Übergang zwischen den einzelnen Kategorien scheint für das Modell schwierig zu sein. Bei den Kategorien *Hate Speech* und *Neither* beläuft sich die falsche Zuordnung eher auf die Kategorie *Offensive*.
Die *Precision* [3] für das Modell insgesamt

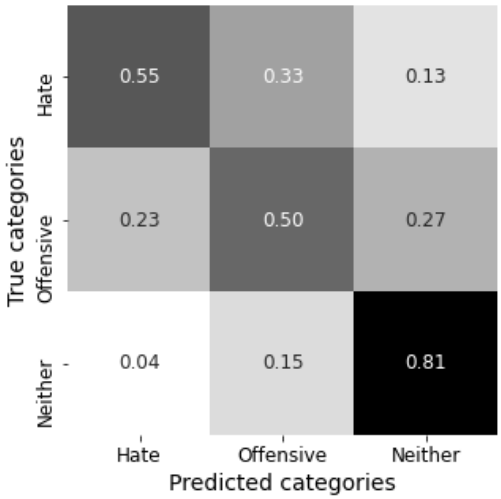

Abbildung 3: Wahre versus vorhergesagte Label

	precision	recall	f1-score	support
0	0.58	0.55	0.57	95
1	0.39	0.50	0.44	102
2	0.87	0.81	0.84	333
accuracy			0.70	530
macro avg	0.61	0.62	0.61	530
weighted avg	0.73	0.70	0.71	530

Abbildung 4: *Precision*, *Recall* und F1-Score

[3] Die *Precision* beschreibt, wie viele der vom Modell erkannten Label tatsächlich wahr sind.

beträgt 61%. Der *Recall* [4] beträgt insgesamt 62% und der F1-Score [5] beträgt auch 61% (vgl. Abbildung 4). Betrachtet man die *Precision* liegen die Prozentzahlen für die Kategorien *Hate Speech* und *Neither* etwas höher, als für den *Recall*. Die *Precision* für die Kategorie *Offensive* liegt mit 39% allerdings deutlich unter der Zahl für den *Recall*. Das Modell erkennt also bezüglich der Kategorie *Offensive* mehr wahre Label, aber nur rund ein Drittel der erkannten Label sind tatsächlich wahr. Bei den beiden anderen Kategorien ist es anders herum. Da erkennt das Modell etwas weniger wahre Label, aber dafür sind mehr der erkannten Label tatsächlich wahr.
Betrachtet man die Abbildungen 5 und 6 ist zu sehen, dass sowohl bei den wahren Labels im Trainingsset, als auch bei den vorhergesagten Labels im Testset die Kategorie *Neither* mit rund 61% bzw. rund 63% am stärksten vertreten ist. Im Trainingsset sind die beiden anderen Kategorien *Offensive* und *Hate Speech* mit rund 20% bzw. rund 19% zu einem gleichen Anteil vertreten. Im Testset ist die Kategorie *Hate Speech* mit rund 18% ebenfalls fast zum selben Anteil vertreten, wie die Kategorie *Offensive* mit rund 19%. Da der Datensatz zur Hälfte aus Songtexten besteht, die *hateful* Wörter aus dem n-Gramm-Lexikon enthalten, liegen sowohl die menschlichen Annotationen, als auch die Vorhersagen des Modells bei deutlich weniger *Hate Speech*.

Die Songtexte im Testset gehören größtenteils dem Genre *rap* an. Weitere häufiger vertretene Genres sind *pop*, *r&b* und *rock* (vgl. Abbildung 12). Im Testset sind einige Interpreten mehrmals vertreten (vgl. Abbildung 13). Mit jeweils sechs Songs kommen *Rick Ross*, *The Game*, *Queen* und *Logic* am häufigsten vor, dicht gefolgt von *Future*, *50 Cent* und *Lil*

[4]Der *Recall* beschreibt, wie viele der wahren Label vom Modell erkannt werden.

[5]Der F1-Score ist eine gewichtete Durchschnittsgröße von *Precision* und *Recall*.

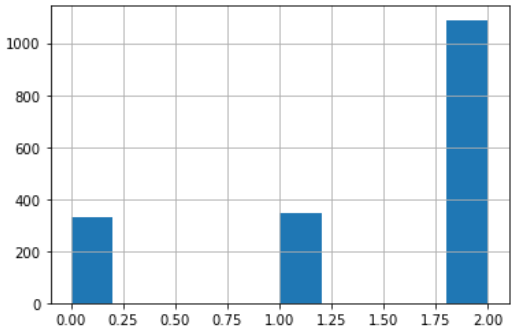

Abbildung 5: Verteilung der wahren Label im Trainingsset

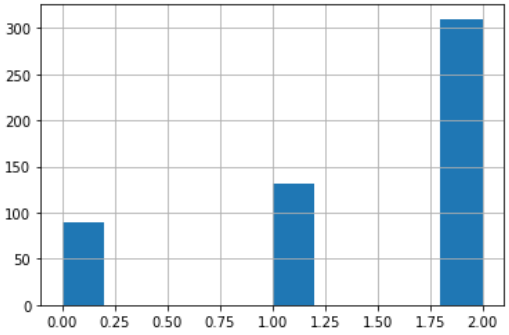

Abbildung 6: Verteilung der vorhergesagten Label im Testset

Wayne mit fünf Songs, sowie *DMX*, *Khalid* und *5 Seconds of Summer* mit jeweils vier Songs. In Abbildung 7 ist das geschätzte Gesamttempo in Beats pro Minute der Songs aus dem Testset zu sehen. Die Werte für das Tempo belaufen sich auf einen Bereich zwischen 60 und 200 Beats pro Minute. Die Analyse der Stimmung im Testset unter Verwendung des *Compound-Score* ergab, dass ca. 54% der Songtexte einen negativen Wert haben. Bezogen auf den gesamten Datensatz haben ca. 52% einen negativen Wert. Vergleicht man das mit der Anzahl der Songtexte, für die ein negatives Label vorhergesagt wurde, stimmt das nicht ganz überein, denn nur 37% der Songtexte wurden der Kategorie *Hate Speech*

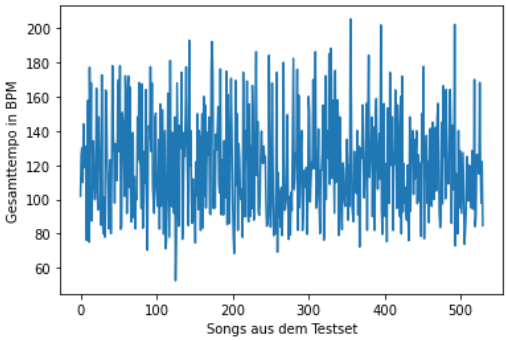

Abbildung 7: Tempo im Testset

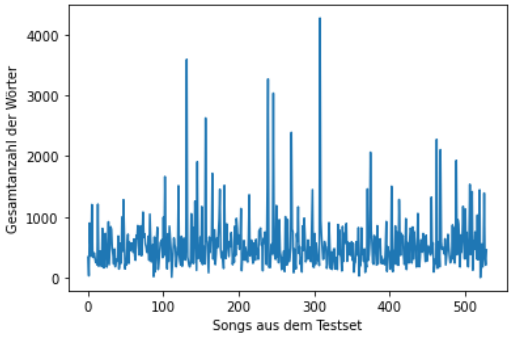

Abbildung 8: Gesamtanzahl der Wörter im Testset

oder *Offensive* zugeordnet. Durchschnittlich enthalten die Songtexte im Testset 548 Wörter. Im Vergleich dazu enthalten die Songtexte im gesamten Datensatz durchschnittlich 532 Wörter. Schaut man sich die Buchstaben an, enthalten die Songtexte im Testset durchschnittlich 2738 Buchstaben, während es im gesamten Datensatz 2675 sind.

8 Diskussion

Wir haben in die Suche nach einem geeigneten Datensatz einige Zeit investiert. Unser erster Datensatz *Million Song Dataset* von [Bertin-Mahieux et al., 2011] stellte sich im Nachhinein als nicht geeignet für unser Vorhaben heraus. So mussten wir, trotz der bereits investierten Zeit, die Arbeit daran abbrechen und neu beginnen. Der Datensatz war nämlich nur in einem *Bag-of-Words*-Format, das nicht zu dem Modell gepasst hätte, welches wir darauf anwenden wollten. Das Modell bezieht die Positionen der Wörter im geschriebenen Text mit ein, das bedeutet es betrachtet Gruppen von Wörtern und nicht nur Einzelwörter. Bei dem *Bag-of-Words*-Datensatz hätten wir den Kontext nicht mit einbeziehen können, dieser erscheint aber für die Einordnung relevant.
Eine weitere Schwierigkeit ergab sich bei der Annotation der Songtexte in dem Datensatz. Da das Labeln der Songtexte durch Bekannte von uns erfolgte, die keine englischen Muttersprachler*innen sind und die Songtexte größtenteils nur von einer Person gelabelt wurden, ist unser Datensatz möglicherweise nicht besonders aussagekräftig. Es ist wahrscheinlich, dass die Songtexte nur aufgrund des Auftretens bestimmter Wörter kategorisiert wurden und der Kontext nicht immer mit einbezogen wurde. In unserem Datensatz kommen viele Songs aus dem Bereich *rap* vor. Dort werden oftmals Wörter wie *„n*gga“* verwendet, die aber gar nicht als diskriminierend oder beleidigend zu verstehen sind, weil sie von Interpreten/ Interpretinnen verwendet werden, die sich mit diesem Begriff identifizieren und damit sich selbst oder ihre Freunde meinen. Bei solchen Wörtern kommt eine andere Bedeutungsebene hinzu, die von den Annotierenden möglicherweise nicht beachtet werden konnte, da das nötige sprachliche Hintergrundwissen fehlt. Die Möglichkeit, Hintergrundwissen, das nichts mit Sprache zu tun hat, über einen Song mit einzubeziehen, bestand allerdings, weil wir den Interpreten/ die Interpretin und den Songtitel mit in die zu annotierenden Dokumente geschrieben haben. Vereinzelt wurde zurückgemeldet, dass dies auch getan wurde, doch bei der Mehrheit wissen wir nicht, ob die Möglichkeit genutzt wurde und halten es für unwahrscheinlich, da es auch nicht explizit in der Aufgabenstellung stand. Bei

Begriffen wie *„b*tch“* oder *„h*e“* ist eventuell weniger sprachliches und somit Kontextwissen nötig, weil es hier vor allem darauf ankommt, von welchem Geschlecht die Begriffe benutzt werden. Wenn sie von Männern verwendet werden, die damit Frauen bezeichnen, stellen sich diese Begriffe mehr hasserfüllt dar, als wenn eine Frau sie benutzt. Hier könnte es also hilfreich sein, das Geschlecht des Interpreten/ der Interpretin mit einzubeziehen. Auch diese Möglichkeit war den Annotierenden gegeben, allerdings ist es auch hier unwahrscheinlich, dass sie genutzt wurde. Außerdem sind Interpreten/ Interpretinnen nicht immer nur eine Person, sondern häufig auch eine Gruppe aus mehreren Leuten verschiedenen Geschlechts. Zusammenfassend lässt sich sagen, dass bei den meisten Begriffen vermutlich nicht der Kontext mit einbezogen wurde. Die Annotierenden haben wahrscheinlich eher auf das Vorkommen bestimmter markanter Begriffe geachtet und dies als Grundlage für die Kategorisierung verwendet.

Wie bereits in Abschnitt 5 erwähnt, haben wir 100 der 1766 Songtexte ein zweites Mal gelabelt, um zu prüfen, wie hoch die Übereinstimmung zwischen den Labels ist. Die Übereinstimmung lag insgesamt bei 69%. Um aufschlussreichere Daten zu bekommen, sollten die Songtexte jeweils von mehreren Personen annotiert werden, damit die Einordnung nicht zu subjektiv ausfällt. Uns war dies jedoch nicht möglich, da wir zu großen Teilen die Songtexte selbst annotieren mussten. Wir hatten zwar genügend Leute, die sich bereit erklärten, die Annotation zu übernehmen, jedoch hielten sich einige von ihnen nicht an unser Zeitlimit oder wir bekamen kurzfristig Absagen, aufgrund von anderen Dingen wie Krankheit oder Prüfungen. Letzten Endes mussten wir etwas weniger als die Hälfte selbst übernehmen.

Wir haben für die Sentiment-Analyse das Tool *vaderSentiment* verwendet. Dieses Tool ist allerdings für die Analyse von Daten aus den sozialen Medien gedacht und für Songtexte vielleicht nicht so gut geeignet. Schließlich handelt es sich bei Tweets oder anderen Social-Media-Beiträgen in der Regel nur um kurze Texte, die eine begrenzte Anzahl von Zeichen aufweisen. Die Songtexte hingegen, die wir verwendet haben, bestehen teilweise aus nur wenigen kurzen Sätzen, vielen langen Sätzen oder richtig langen Abschnitten, die sich über eine gesamte Seite erstrecken. Wir haben dieses Tool trotzdem angewendet, können jedoch nicht differenzieren, wo die Vor- oder Nachteile liegen und welche Unterschiede es gemacht hätte, ein anderes Tool zu verwenden.

Einige der speziellen Features in dem Modell von [Davidson et al., 2017] bezogen sich auf Hashtags, Verlinkungen zu anderen Tweets, Markierungen von Personen oder die Verlinkung zu anderen externen Webseiten, die für unsere Daten nicht verwendbar waren. Aus diesem Grund haben wir die Features entfernt. Allerdings fehlen dem Modell dadurch möglicherweise Merkmale für die Kategorisierung. In Songtexten könnte man anstelle von Hashtags oder Retweets die Wiederholungen bestimmter Strophen zählen oder sich wiederholende Zeilen entfernen. Außerdem sollte es Features geben, die sich speziell auf die Bereinigung von Songtexten beziehen, denn bei dieser Textart treten andere für das Modell nicht relevante Passagen auf, als bei Social-Media-Beiträgen. Wie sich bei der Annotation herausstellte, beinhalten die meisten Songtexte rhythmische Laute, die keine Bedeutung für den Kontext tragen. Diese Laute könnten aus den Daten entfernt werden. Außerdem sollten Wörter, die nicht aus der englischen Sprache stammen, rausgenommen werden, da das Modell für andere Sprachen nicht ausgelegt ist.

Im Vergleich zu den *Recall*-Werten, die [Davidson et al., 2017] bei der Untersuchung der Tweets erhalten haben, fallen unsere

Werte hier deutlich schlechter aus. Man könnte also sagen, dass das Modell von [Davidson et al., 2017] auf längeren Texten nicht so gut funktioniert, allerdings spielen hier auch die oben erwähnten Faktoren eine Rolle, die dazu geführt haben könnten, dass das Modell auf unseren Daten nicht so gut funktioniert. Es wäre möglich, dass das Modell auf anderen längeren Daten besser funktioniert. Vielleicht funktioniert es auf unseren Daten auch besser, wenn wir Überflüssiges entfernen und andere Features komplementieren.
Schließlich hatten wir Probleme dabei, die Fehlermeldungen des Notebooks zu interpretieren. Die meisten Funktionen ließen sich ohne Probleme ausführen, für andere Funktionen fehlten jedoch wichtige Teile oder der vorangegangene Code war nicht passend. Wir haben also sehr viel Zeit damit verbracht, Fehler zu suchen und zu versuchen diese zu beheben. Eine weitere Schwierigkeit dabei war, dass sich die Zahlen der Matrizen in dem Notebook teilweise bei jeder erneuten Ausführung geändert haben und es für uns keinen ersichtlichen Grund dafür gab.

9 Zusammenfassung

Zunächst haben wir uns überlegt, welches Projektthema wir bearbeiten könnten. Dabei galt unser Interesse sehr schnell dem Thema *Hate Speech* in Songtexten. Als das beschlossene Sache war, haben wir uns ein eigenes *Padlet* angelegt, in dem wir Texte ausgetauscht, unser weiteres Vorgehen festgehalten und die einzelnen Dateien zwischengespeichert haben. Unser *Padlet* haben wir im weiteren Verlauf der Projektarbeit immer weiter ergänzt, um einen Überblick über unseren Fortschritt und andere nützliche Dinge, die uns während unseres Projekts aufgefallen sind, zu haben. Das *Padlet* war außerdem sehr hilfreich um uns gegenseitig auf denselben Stand zu bringen und um bearbeitete Notebooks untereinander zu tauschen und dies nicht auf umständlicheren Wegen, z.B. über Emails zu machen.
Unser erster Schritt bestand darin, einen geeigneten Datensatz und ein passendes Modell zu finden, das wir verbessern oder auf unsere Daten zuschneiden könnten. Für unser Projekt haben wir beschlossen, das Modell von [Davidson et al., 2017] auf einen neuen Datensatz anzuwenden. Der Unterschied bestand darin, dass bei [Davidson et al., 2017] Tweets untersucht wurden, wir hingegen wollten Songtexte untersuchen, also sehr viel längere Texte. Bei unserer Arbeit ging es dann im Weiteren um die Erstellung eines geeigneten Datensatzes. Wir haben dafür im Internet nach passenden Datensätzen gesucht, die Songtexte enthalten. Ein Problem, das uns erst auffiel, nachdem wir uns schon einige Zeit mit dem Datensatz *Million Song Dataset* beschäftig hatten, war, dass wir für das Modell von [Davidson et al., 2017] keinen *Bag-of-Words*-Datensatz brauchten, sondern einen Datensatz bestehend aus Text und Labels. Im weiteren Suchverlauf stießen wir dann auf einen passenden Datensatz. Der Datensatz *Audio features and lyrics of Spotify songs* von [Nakhaee, 2020] war als Grundlage für unser Vorhaben besser geeignet.
In diesem Datensatz waren jedoch nicht nur englischsprachige Songtexte. Da das Modell von [Davidson et al., 2017] auf englischsprachige Texte trainiert ist, filterten wir alle englischsprachigen Songtexte heraus und erstellten einen neuen kleineren Datensatz. Im nächsten Schritt mussten wir dann überprüfen, ob überhaupt hasserfüllte Songtexte in dem von uns reduzierten Datensatz zu finden waren. Dafür haben wir das *Hate-Speech*-Lexikon von [Davidson et al., 2017] verwendet. Das Lexikon besteht aus n-Grammen der Länge 1-4. Nach diesen n-Grammen haben wir in unserem Datensatz gesucht. In 901 Texten waren

n-Gramme enthalten. Diese Tatsache zeigt zumindest, dass in den Texten Wörter vorhanden sind, die häufig in einem hasserfüllten Kontext verwendet werden. Unsere nächste Aufgabe bestand darin, diese Songtexte herauszufiltern und einen 50:50-Datensatz zu erstellen, der zu gleichen Anteilen Texte enthält, die anhand des Lexikons als *hateful* bezeichnet werden können und solche, die keine n-Gramme aus dem Lexikon enthalten. Damit entstand dann unser 50:50-Datensatz mit hasserfüllten Songtexten, der 1766 Songs enthält. Unter Verwendung dieses Datensatzes wollten wir dann schauen, wie gut das Modell von [Davidson et al., 2017] auf längeren Texten funktioniert. Wir wendeten also das Modell auf unsere Daten an, wobei wir auf einige Probleme stießen, die wir im Abschnitt 8 schon beschrieben haben und auf die wir hier nicht mehr weiter eingehen werden. Unser abschließendes Ergebnis fällt nicht besonders gut aus. Im Vergleich zu den Ergebnissen von [Davidson et al., 2017] erkennt das Modell bei unserem Datensatz deutlich weniger der wahren Labels. Eine Begründung dafür könnte der vergleichsweise kleine Datensatz sein, aber auch der längere Text könnte problematisch für das Modell sein, vor allem unter den Umständen, dass der Text nicht komplett bereinigt wurde und zum Teil nicht-englische Passagen und nicht-bedeutungstragende Laute enthält. Der Grund für das schlechtere Ergebnis könnte also auch unser Datensatz sein, der eventuell noch einmal überarbeitet werden müsste.

Um bessere Ergebnisse zu erzielen, könnte bei einer weiterführenden Untersuchung ein erster Schritt darin bestehen, einen größeren Datensatz zu verwenden auf dem das Modell trainiert werden kann. Außerdem könnte es hilfreich sein, die Struktur der Songtexte zu überarbeiten. Die Texte sind teilweise zu lang, vor allem, weil sie viele Wiederholungen durch den Refrain und die Strophen enthalten. Es müsste überlegt werden, wie die Texte aufgeteilt werden könnten, um sie für das Modell von [Davidson et al., 2017] zu optimieren. Man könnte Doppelungen entfernen oder hasserfüllten Parts, die mehrmals vorkommen eine höhere Gewichtung zukommen lassen. Eine alternative Möglichkeit würde darin bestehen, ein komplett neues Modell zu bauen, das speziell auf Songtexte zugeschnitten ist. Dieses Modell sollte in langen Texten und vor allem in Texten mit vielen sich wiederholenden Parts *Hate Speech* erkennen und unabhängig von der Vorkommenshäufigkeit der Wörter, die zu *Hate Speech* oder beleidigender Sprache gezählt werden, die Kategorisierung vornehmen.

Außerdem könnten in einer weiterführenden Studie noch weitere Eigenschaften aus dem Datensatz hinzugefügt werden. Der zugrunde liegende Datensatz *Audio features and lyrics of Spotify songs* von [Nakhaee, 2020] enthält neben den Informationen zu Interpret, Songtext, Titel und Genre noch weitere Informationen wie beispielsweise Sprachlichkeit, Lautstärke, Tempo und Dauer. Wir hätten diese Informationen bei unserer Analyse mit einfließen lassen können. Ebenso hätten wir die Herkunft oder das Geschlecht des Interpreten/der Interpretin mit einbeziehen können, um mehr Kontextwissen zu haben. Diese Informationen sind allerdings noch nicht in dem Datensatz enthalten, doch es könnte hilfreich sein, diese zu ergänzen.

Literatur

[Aulia and Budi, 2019] Aulia, N. and Budi, I. (2019). Hate Speech Detection on Indonesian Long Text Documents Using Machine Learning Approach. In *Proceedings of the 2019 5th International Conference on Computing and Artificial Intelligence*, ICCAI

'19, page 164–169, New York, NY, USA. Association for Computing Machinery.

[Bertin-Mahieux et al., 2011] Bertin-Mahieux, T., Ellis, D. P., Whitman, B., and Lamere, P. (2011). The Million Song Dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*.

[Davidson et al., 2017] Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. pages 512–515.

[Fell et al., 2019] Fell, M., Cabrio, E., Corazza, M., and Gandon, F. (2019). Comparing Automated Methods to Detect Explicit Content in Song Lyrics. In *RANLP 2019 - Recent Advances in Natural Language Processing*, Varna, Bulgaria.

[Kiela et al., 2020] Kiela, D., Firooz, H., Mohan, A., Goswami, V., Singh, A., Ringshia, P., and Testuggine, D. (2020). The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes. *arXiv preprint arXiv:2005.04790*.

[Kruk et al., 2019] Kruk, J., Lubin, J., Sikka, K., Lin, X., Jurafsky, D., and Divakaran, A. (2019). Integrating Text and Image: Determining Multimodal Document Intent in Instagram Posts. *arXiv preprint arXiv:1904.09073*.

[Mathew et al., 2020] Mathew, B., Saha, P., Yimam, S. M., Biemann, C., Goyal, P., and Mukherjee, A. (2020). HateXplain: A Benchmark Dataset for Explainable Hate Speech Detection. *arXiv preprint arXiv:2012.10289*.

[Nakhaee, 2020] Nakhaee, M. (2020). Audio features and lyrics of Spotify songs. A dataset of 18000 Spotify songs along with lyrics, audio features and language. `https://www.kaggle.com/imuhammad/audio-features-and-lyrics-of-spotify-songs`. Accessed: 2022-03-08.

[Rospocher, 2021] Rospocher, M. (2021). Explicit song lyrics detection with subword-enriched word embeddings. *Expert Systems with Applications*, 163:113749.

Anhang

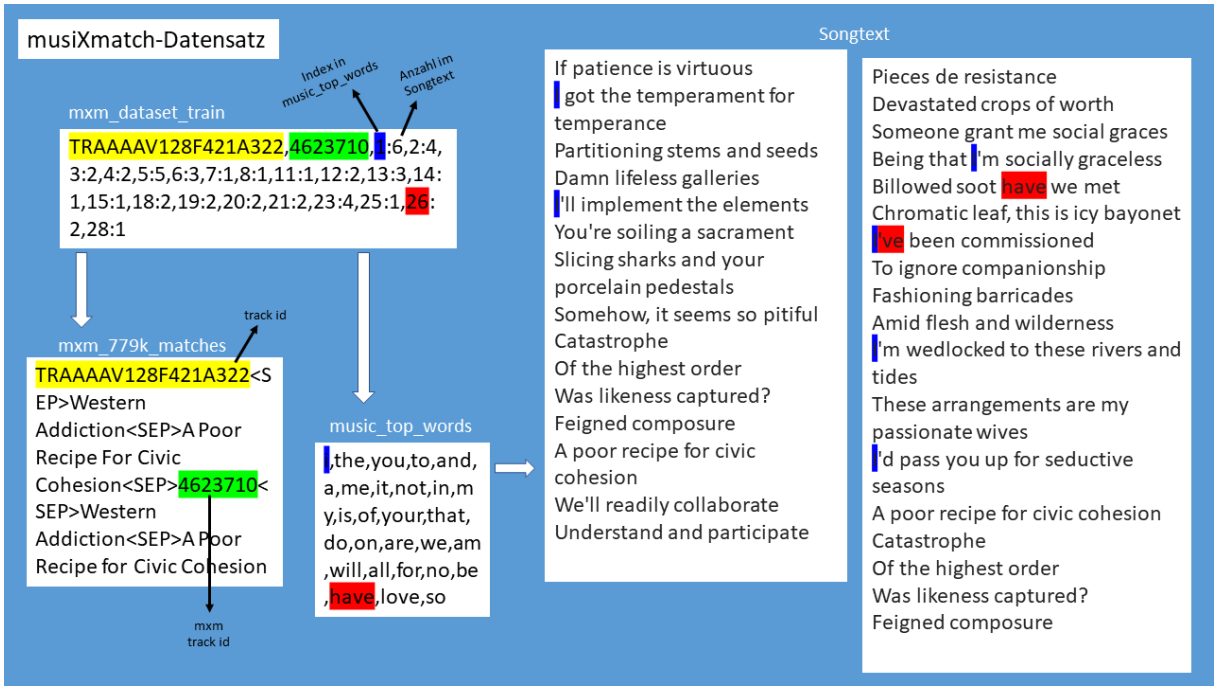

Abbildung 9: Der *musiXmatch-Datensatz*

Songtitel	Interpret	Top words - Anzahl	Klassifizierung
A poor Recipe for Civic Cohesion	Western Addiction	i – 6, the – 4, you – 2, to – 2, and – 5, ...	1 = hateful oder 0 = non-hateful?
...	...	...	...

Abbildung 10: Ausschnitt des zuerst angedachten Datensatzes

- **Definition „Hate Speech“ nach Davidson:**

  „Wir definieren „Hate Speech“ als Sprache, die verwendet wird, um Hass gegen eine bestimmte Gruppe auszudrücken, oder die darauf abzielt, die Mitglieder der Gruppe herabzusetzen, zu demütigen oder zu beleidigen. In extremen Fällen kann es sich auch um Sprache handeln, die mit Gewalt droht oder dazu aufruft.“ (Übersetzung: Davidson, 2017, S.12)

Categories	Example of possible targets
Race	nigga, black people, white people
Behavior	insecure people, sensitive people
Physical	obese people, beautiful people
Sexual orientation	gay people, straight people
Class	ghetto people, rich people
Gender	pregnant people, cunt, sexist people
Ethnicity	chinese people, indian people, paki
Disability	retard, bipolar people
Religion	religious people, jewish people
Other	drunk people, shallow people

- **Deine Aufgabe:**

  1. Lese dir die Definition zu „Hate Speech“ durch.

  2. Lese dir jeweils einen der Songtext-Blocks in dem PDF-Dokument durch. **TIPP:** Du kannst dir den Songtext auch über Spotify anhören oder dir die Übersetzung über DeepL (www.deepl.com) durchlesen. Du musst dir den Text nicht zu genau durchlesen, dein erster Eindruck zählt!

  3. Entscheide dich, ob du den Text zur Kategorie „Hate Speech“, beleidigende Sprache oder keins von beidem, also normale Sprache zuordnen würdest.

  4. Notiere dir die Anfangszahl des Blocks und die Kategorie für die du dich entschieden hast. (Bsp.: 1798 Hate speech, 2896 Beleidigung, 3075 keines,…)

  5. Wiederhole das für alle Songtext-Blocks.

  6. Schicke das Ergebnis an uns, egal ob abfotografiert, als Datei oder im Whats-App-Chat.

Vielen Dank für deine Hilfe!

Abbildung 11: Anleitung für die Annotation

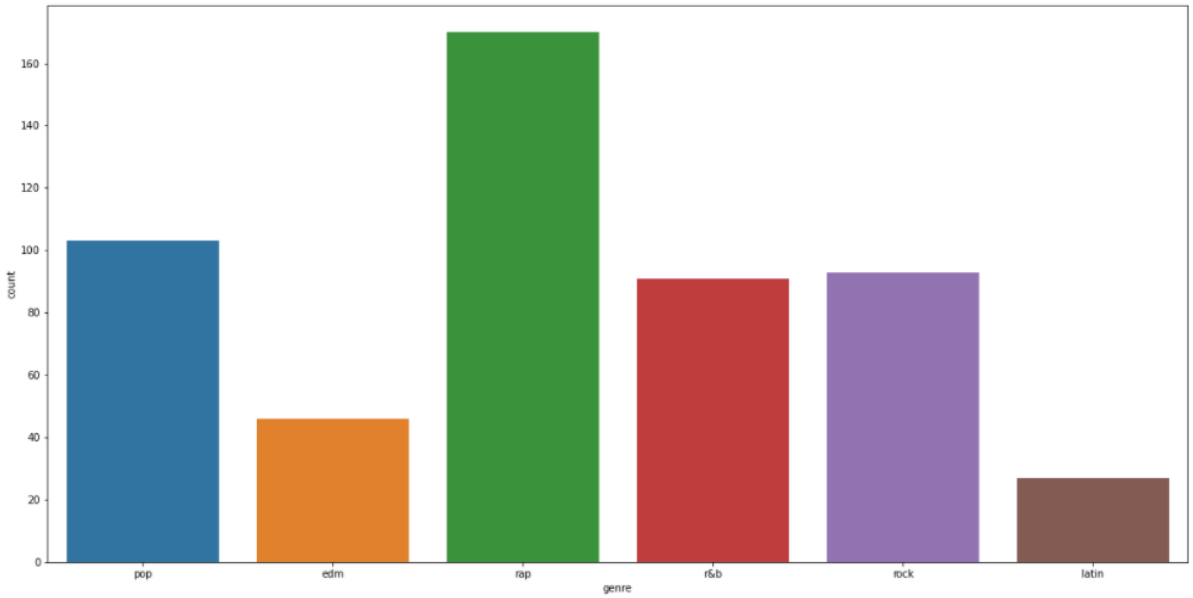

Abbildung 12: Genres im Testset

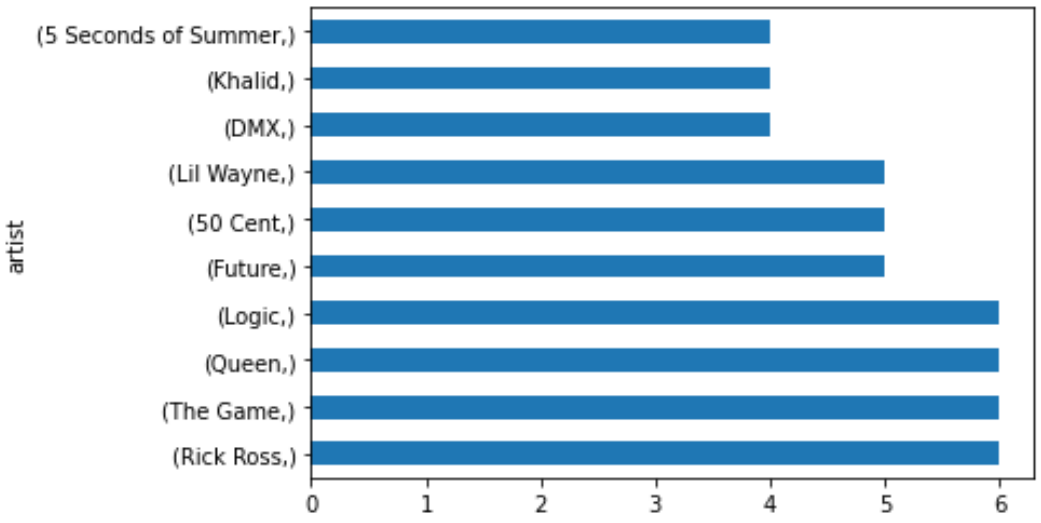

Abbildung 13: Die zehn häufigsten Interpreten im Testset