

Finalreport - Universound

Steve Aumüller(1204280)

Daniel Lentsch (1109310)

Christian Muck (1246269)

1 Motivation

Unser Projekt basiert auf einem Auszug aus dem last.fm Dataset. Dabei haben wir eine für die Lehrveranstaltung bearbeitete Version verwendet, welche auch von dieser zur Verfügung gestellt wurde. Die Daten enthalten die Attribute SongID, SongName, Listeners, Playcount, TagID, TagName und TagWeight. Wie wir diese Daten für unser Projekt aufbereitet haben und was die einzelnen Attribute genau aussagen wird in dem Kapitel 1.1 (S. 1) näher beschrieben. Für jetzt genügt aber das Wissen, dass für die Lieder in dem Datensatz mehrere Tags gespeichert wurden. Die Tags beinhalten Strings, welche von Usern von last.fm vergeben wurden. Da alle diese Daten für Jedermann frei zugänglich sind, haben wir ein Tool erstellt, mit welchem man die Daten visuell untersuchen und erforschen kann. Daher ist unsere Zielgruppe, jeder der Interesse an den Daten hat und diese verwenden will oder sich für diese interessiert. Deswegen haben wir in unserem Projekt versucht, alle Zusammenhänge aufzuzeigen, die wir in den Daten gefunden haben oder die wir vermutet haben. Dabei haben wir uns auf drei Hauptbereiche gestützt. Zuerst wollten wir den Zusammenhang zwischen der Anzahl der Hörer und wie oft ein Lied gehört wurde zeigen. Anschließend wollten wir die Verteilung dieser zwei Attribute zeigen und als letzten Bereich haben wir die Tags eingefügt. Hier zeigen wir wie oft Tags generell vorkommen und wie sie im Zusammenhang mit der Hörerzahl und der Anzahl, wie oft ein Lied abgespielt wurde, zusammenhängen.

1.1 Dataset

Unser Dataset wurde uns in der VU Visualisierung, in Form einer .csv Datei, zur Verfügung gestellt und beinhaltet folgende Attribute: SongID, SongName, Listeners, Playcount, TagID, TagName und TagWeight. Dabei ist jede Zeile durch eine Kombination aus SongID und TagID identifiziert und weißt zu mit den Liedern mehrere Tags zu. Listeners speichert zusätzlich die Anzahl der Leute ab, welche das Lied gehört haben, Playcount ist die Anzahl, wie oft das Lied gehört wurde und TagWeight ist eine von last.fm vergebenen Gewichtung für die Kombination aus Lied und Tag. Für unsere Implementation, welche genauer in dem Kapitel 3 (S. 2) haben wir das Tool Tableau verwendet. Da wir einen Scatterplot verwenden und bei diesem automatisch für jede Zeile ein Punkt erstellt wird, mussten wir, zur Verbesserung der Performance das Dataset anpassen. Um für jedes Lied wirklich nur einen Punkt anzuzeigen, haben wir die Daten in drei Teilgruppen aufgeteilt, welche wir "subset_songonly.csv", "subset_songsandtags.csv" und "subset_tagonly.csv" genannt haben. In "subset_songonly.csv" wurden die Attribute SongID, SongName, Listeners, Playcount und NoOfTags gespeichert, wo bei das letzte von uns selbst erstellt wurde und die Anzahl der Tags für ein Lied beinhaltet. "subset_tagonly.csv" enthält TagID, TagName und TagCount. Letzterer wurde wieder von uns berechnet und beinhaltet

die Anzahl, wie oft der Tag in dem Dataset vorkommt. In "subset_songsandtags.csv" wurden die SongID und die TagID gespeichert, um weiterhin die Zugehörigkeit der Tags zu den Liedern abgespeichert zu haben. Zusätzlich waren einige Tags redundant und wiesen eine unterschiedliche Tag ID auf, wodurch Fehler in unserer Visualisierung zustande kamen. Auch dieser Fehler wurde per Algorithmus behoben.

1.2 Users

Da die Daten, welche die Grundlage für unser Projekt bilden, kostenlos auf der Homepage von last.fm zu finden sind, haben wir unsere Zielgruppe als sehr breit angenommen. Und zwar waren all jene Menschen, welche sich für diesen Datensatz interessieren, potentielle Benutzer. Daher kann man diese Menschenmasse, unserer Meinung nach, nicht wirklich eingrenzen und wir haben uns zum Ziel gesetzt unsere Applikation leicht verständlich und intuitiv zu halten. Da wir uns unter dem großen Spektrum dieser möglichen User nur schwer etwas vorstellen konnten, haben wir für uns ein paar Beispiele überlegt. Einige dieser Beispielbenutzer wollen wir hier kurz vorstellen:

-) Ein DJ weiß, dass last.fm beliebt unter Musikhörern ist und sucht deswegen auf der Homepage Anhaltspunkte für seine zukünftigen Playlists. Hierbei stößt er auf unsere Visualisierung und erfährt durch diese etwas über die Tags, welche oft gehört wurden.
-) Ein Musikstudent will in einem seiner Projekte auf den Zusammenhang zwischen Hörerzahl und Anzahl wie oft ein Lied gehört wurde und wie die Verteilung dieser zwei Attribute aussieht.
-) Ein musikbegeisterter Mensch hat ein neues Lieblingslied und möchte, unter Zuhilfenahme von Tags, Lieder finden welche seinem Geschmack am ehesten entsprechen.

1.3 Tasks

Für unser Projekt haben wir uns folgende Aufgaben überlegt, welche Benutzer durchführen möchten:

- Ein User möchte wissen wie die Hörerzahl mit der Anzahl, wie oft ein Lied gehört wurde, zusammenhängt.
- Ein User möchte sich diesen Zusammenhang nur in einem gewissen Bereich der Datenmenge ansehen.
- Ein User möchte wissen wie die Anzahl der Hörer verteilt ist.
- Ein User möchte wissen wie die Anzahl, wie oft ein Lied gehört wurde, verteilt ist.
- Ein User hat einen Punkt gefunden, welcher für ihn interessant erscheint und möchte den Namen des Lieds und weitere Informationen über ein Lied erfahren.
- Ein User hat einen Punkt gefunden, welcher für ihn interessant erscheint und möchte alle Tags wissen, welche für dieses Lied vergeben wurden.
- Ein User möchte weitere Songs finden die mit den Tags seiner Lieblingslieder versehen sind.

- Ein User möchte die Top fünf oder Top 10 der Tags allgemein oder eines ausgewählten Bereichs wissen.
- Ein User möchte nach einem bestimmten Lied suchen.

1.4 Szenario

Zum Testen und Vorzeigen unserer Applikation haben wir uns folgendes Szenario ausgedacht:

Rainer wohnt in Wien und studiert Musik. Durch sein Studium ist er natürlich auch Privat sehr an Musik interessiert und möchte seine Musiksammlung ständig erweitern. Da er sehr wählerisch ist und seine Zeit nicht mit Songs vergeuden möchte die nicht seinem Geschmack entsprechen, nutzt er die Universound-Applikation. Mittels Texteingabe sucht er einen seiner Lieblingssongs, Stronger. Wählt in per Mausclick aus und kann sich nun durch dessen Tags arbeiten. Weil Rainer nur gute Lieder interessieren, wählt er einen Tag aus den Top 5 aus. Per Klick auf den für ihn interessantesten Tag Rock, kann er die Übersicht nach diesem Tag filtern. Auf einen Blick kann er nun gleich getaggte Songs betrachten die sich im hohen bzw. niedrigen Hörerbereich befinden und somit seinen musikalischen Horizont erweitern.

2 Related work

Beim planen und gestalten unserer Visualisierung, haben wir uns kein konkretes Vorbild genommen. Sämtliche Ideen entstanden aus eigener Erfahrung, sowie den LV-Folien die uns im Kurs bereitgestellt wurden. Als ähnliches Tool kann Last.fm selbst genannt werden, welches allerdings minimalistische Diagramme und dabei eher auf historische Daten eingeht als auf Beziehungen zwischen Tags. Nichtsdestrotz lassen sich einige von uns bestimmten Tasks auch direkt auf Last.fm realisieren. [2]

Auch konnten wir eine Visualisierung von Google finden, welche sich aber ebenfalls mit historischen Daten befasst und damit eher uninteressant für unsere Fragestellung ist. [1]

3 Approach

In der linken, oberen Ecke unserer Applikation sieht man den Scatterplot, welcher das Herz unserer Visualisierung darstellt. In diesem werden auf der X-Achse das Attribut Playcount und auf der Y-Achse Listeners dargestellt. Die einzelnen Punkte sind verschieden eingefärbt. Jeder Eintrag der Top 5 Tags hat eine eigen Farbe und die restlichen werden in der gleichen angezeigt. Wir haben die Punkte eingefärbt, damit der User gleich auf die ersten Blicke sehen kann wie die beliebtesten Tags verteilt sind und wo sie sich befinden. Die Wertebereiche der zwei Attribute, welche angezeigt wird, kann durch zwei Schieberegler eingegrenzt werden. Man kann auch die Grenzen direkt als Wert eingeben, falls man den Bereich genau einstellen will. In diesem Fenster haben wir auch ein Suchfeld angelegt, mit welchem nach einem Songnamen gesucht werden kann. Durch die dynamische Anpassung des Bereichs, wollen wir es dem User ermöglichen das er sich zuerst einen Überblick über die Daten und durch filtern und zoomen einen genaueren Einblick holen kann. Wenn er einen Punkt gefunden hat, welcher ihn interessiert, kann er sich über einen Rollover noch die Detailinformationen anzeigen lassen. Durch diese Technik kann er sich den die Attribute Listeners, Playcount, SongID, Listeners(Partition), Playcount(Partition) und NoOfTags anzeigen lassen. Die Partitionseinträge zeigen an, in welchem Bucket des Histogram sich das Lied befindet und über NoOfTags sieht man sofort, wie viele Tags dem Song zugewiesen wurden. Diesen Ansatz haben wir aus dem Prinzip Overview First, Zoom and Filter, Detail on Demand aus [3] abgeleitet. Rechts neben dem Scatterplot befindet sich eine Auflistung

der Top 10 Lieder der momentanen Auswahl im Scatterplot. Wenn man keine Punkte ausgewählt hat, wird die Liste an die gesamten Daten angepasst. Hier werden auch wieder die zwei Attribute Listeners und Playcount angezeigt, damit soll ein Vergleichen der Lieder ermöglicht werden. Wir haben uns dafür entschieden die Werte in der Liste als Balken anzuzeigen um den Vergleich zu erleichtern. Die Bereiche in welchen die Informationen visuell dargestellt werden haben immer die selbe Breite und symbolisieren den maximalen Wert dieses Attributs. Dadurch soll es dem User zusätzlich ermöglicht werden, das er die Werte mit den Maximum vergleichen kann. Zusätzlich wird die Anzeige im Scatterplot angepasst, wenn man Lieder in dieser Liste markiert. Am rechten Rand unserer Applikation befindet die Liste der Tags. Diese ist Sortiert, nach der Anzahl der Lieder in welchen jeder Eintrag vorkommt. Der User kann wählen, ob er die Top 5, die Top 10 oder alle Tags angezeigt haben will, von dem Bereich an Punkten, die er gerade ausgewählt hat. Wenn er keine Auswahl markiert hat, wird die Liste von allen Liedern in dem Datensatz angezeigt. Hier sieht man die Farben, in welchem die Punkte im Scatterplot eingefärbt sind. Man kann auch Tags in der Liste markieren und dadurch die Auswahl im Streudiagramm beeinflussen. Unten links beginnend haben wir zwei Histogramme eingefügt. Das linke ist dabei für das Attribut Listeners und das rechte für Playcount. Sie zeigen die Verteilung der Daten, damit der User einen Überblick hat wo die meisten Lieder liegen. Da das Aussehen der Histogramme und somit die Informationen welche sie übermitteln stark von der Gewählten Bucketgröße abhängen, haben wir diese dynamisch wählbar gemacht. Man kann mit den Schieberegler unter den Diagrammen, aus einer von uns festgelegten Anzahl von Buckets wählen. Rechts neben den Reglern kann haben wir auch Pfeile eingefügt, mit dem man die Buckts einzelnen weiter schalten kann. Diese Option haben wir eingebaut, falls der User die Möglichkeiten durchsuchen will, um die für ihn beste Variante zu finden. Über das Anlicken von Buckets kann man die Auswahl im Scatterplot beeinflussen. Dadurch kann sich ein User die für ihn interessanten Bereiche anzeigen lassen. Die Anzeige im Histogramm wird, auf der anderen Seite, auch an der Markierung im Histogramm angepasst. Somit kann man sich auch Verteilungen von selbst gewählten Gruppen Anzeigen lassen. In [3] haben wir über das Prinzip Brushing and Linking gelesen und deshalb haben wir auch alle unsere Diagramme durch die Auswahl im Scatterplot beeinflussen lassen und auch umgekehrt.

4 Implementation

Ursprünglich war es geplant das Projekt mithilfe von Javascript und der Library D3 (Data-Driven Documents) umzusetzen, da dies uns den größtmöglichen Grad an Freiheit erlauben würde. Jedoch wurde uns schnell bewusst, dass diese Implementierung aufgrund unserer fehlenden Erfahrung mit dem Tool nur in einem sehr beschränkten und für den Komplexität des Projektes möglicherweise nicht auszureichenden Umfang möglich wäre. Aus diesem Grund wurde im Team die einstimmige Entscheidung getroffen das Projekt mittels Tableau umzusetzen.

Tableau ist ein professionelles Datenvisualisierungstool das eine breite Anzahl an Möglichkeiten bietet Daten visuell darzustellen, mit denen Benutzer in Echtzeit interagieren können. Im Folgenden wird versucht zu erklären wie die Implementierung des Projektes mit Hilfe des gewählten Tools erfolgt ist.

4.1 Dashboard

Ein Dashboard ist in Tableau eine aus einem oder mehreren Arbeitsblättern bestehende Oberfläche, ähnlich einer Präsentationsfolie in Office-Programmen, die es erlaubt durch sogenannte Aktionen zwei oder mehr dieser Arbeitsblätter zu verbinden und so Interaktion zu erlauben. Unser Dashboard besteht aus fünf Arbeitsblättern, die mit mehreren Aktionen und Schnellfiltern sowie Parametern erlauben die angezeigten Daten in Echtzeit zu manipulieren, und detaillierte Aussagen über deren Inhalt zu treffen. Im folgenden Abschnitt werden die Bedeutung der einzelnen Blätter und deren zustandekommen näher beschrieben.

4.2 Blatt 1: Scatterplot

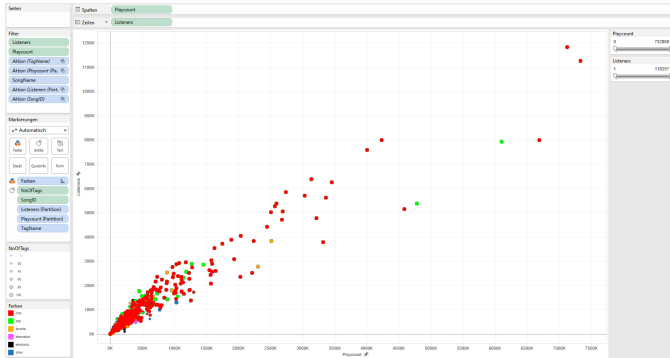


Abbildung 1: Scatterplot

Der Scatterplot soll einen Überblick über alle Songs im Datensatz geben, und deren Listener und Playcount anzeigen. Um Punkte auf den ersten Blick unterscheiden zu können, wurde deren Größe von der Anzahl der Tags (NoOfTags) die einem jeden Song beigefügt wurden, abhängig gemacht. Eine weitere visuelle Abgrenzung ist die Farbe der einzelnen Punkte (Songs). Diese wurden anhand der Top 5 vorkommenden Tags eingefärbt, die im Datensatz vorkommen. Der Rest (Songs die keinen dieser fünf Tags enthalten) bekamen die Standardfarbe. Da ein Song mehr als nur einen der Top fünf Tags besitzen kann, wurde die Reihung manuell nach dem Top Tag sortiert (zB. Ein Song hat den die Tags "rock" und "pop", die beide Top fünf Tags sind. Der Song wird nur in der Farbe des öfter vorkommenden Tags, "rock" eingefärbt). Anhand eines Schnellfilters können sowohl Listeners als auch Playcount weiter eingegrenzt werden.

4.3 Blatt 2: Tags

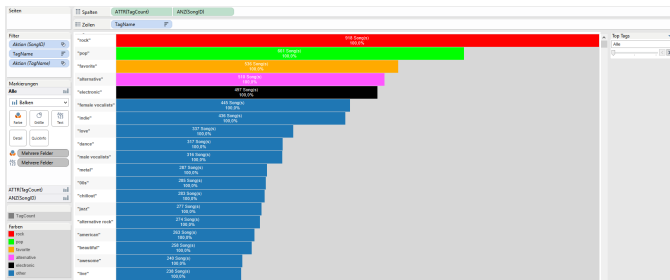


Abbildung 2: Tags

Das Arbeitsblatt Tags bietet einen Überblick über alle im Datensatz vorkommenden Tags und reiht sie nach de-

ren Vorkommen in einem gewählten Bereich (SongID). Zur Realisierung wurden zwei Kennzahlen herangezogen. Zum einen die Anzahl an vorkommenden SongIDs, die sich anhand des im Scatterplot markierten Bereiches ändert, und zum anderen TagCount, die das Vorkommen des Tags im gesamten Datensatz darstellt. Anschließend wurde eine Doppelachse geschaffen, anhand derer sofort zu erkennen ist wie oft ein Tag im gewählten Bereich im Verhältnis zum gesamten Datensatz vorkommt. Die Beschriftung ergibt sich durch die Anzahl der SongIDs die festlegt wie vielen Songs dieser Tag beiliegt, sowie das Verhältnis zum Gesamtvorkommen in Prozent. Um eine klare Verbindung zwischen der Einfärbung im Scatterplot und den Top Tags herzustellen, wurden die Top fünf Tags ebenso eingefärbt. Weiters lässt sich dieses Arbeitsblatt durch einen sogenannten Parameter (eine Variable) auf Top 5 und Top 10 begrenzen, was nur die fünf bzw. zehn am häufigsten vorkommenden Tags anzeigt.

4.4 Blatt 3: Top 10 Songs

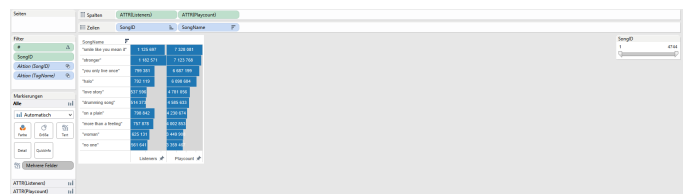


Abbildung 3: Top 10 Songs

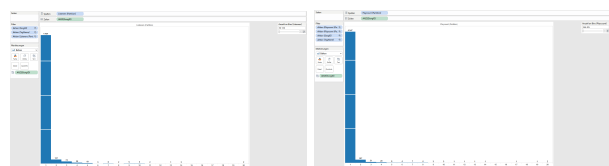
Hier werden die Top zehn Songs in einem Bereich mit Name, sowie Listener und Playcount angezeigt. Hierfür wurde mit der INDEX-Funktion jedem Eintrag ein Index zugewiesen und dieser auf die Zahl 10 beschränkt.

4.5 Blatt 4/5: Histogramm (Listener)/(Playcount)

In dem Histogramm werden die Listener/Playcount in sogenannte Buckets (Klassen mit einem Wert von, bis) eingeteilt um deren Häufigkeit feststellen zu können. Hierfür wurden mit der Partitionsfunktion von Tableau Klassen von Listener/Playcount geschaffen. Deren Vorkommen im Datensatz wird mit der eindeutigen Anzahl der SongIDs berechnet. Beim Erstellen einer Partition ist in Tableau eine Bucketgröße zu wählen, die hier mittels eines Parameters dynamisch berechnet wird. Der Parameter lässt eine Anzahl zwischen 2 - 20 Buckets auswählen.

4.6 Dashboardaktionen

Wie oben bereits kurz erwähnt erlauben so genannte Aktionen Interaktion zwischen zwei oder mehreren Arbeitsblättern. Da unser Dashboard eine Vielzahl von Interaktionsmöglichkeiten gibt, wird hier ein kurzer Überblick geschaffen.



(a) Histogramm Listeners (b) Histogramm Playcount

Abbildung 4: Titel unterm gesamten Bild

4.6.1 Scatterplot

Der Scatterplot erlaubt das Markieren einer oder mehrerer Punkte (Songs) entweder durch Drücken und Halten der STRG-Taste oder durch aufziehen eines Bereichs mit der Maus. Dieses Eingrenzen der angezeigten Auswahl hat Einfluss auf alle anderen Arbeitsblätter.

In Tags werden nur mehr alle Tags angezeigt, die auch wirklich bei den gewählten Songs vorkommen. Das Vorkommen wird durch die Anzahl der Songs bestimmt in denen der Tag vorkommt.

In Top 10 Songs werden die zehn (oder weniger, wenn im gewählten Bereich weniger als zehn Songs vorkommen) besten Songs im gewählten Bereich angezeigt.

In Histogramm (Listener)/(Playcount) werden nur mehr die Buckets angezeigt, die im gewählten Bereich vorkommen und deren Anzahl wird ebenfalls durch die Auswahl beschränkt.

4.6.2 Tag

Wählt man einen oder mehrere Tags werden im Scatterplot nur mehr die Lieder angezeigt, die diese(n) Tag(s) auch besitzen.

In Top 10 Songs werden die zehn (oder weniger, falls ein Tag gewählt wurde der bei weniger als zehn Songs vorkommt) besten Songs dieser/dieses Tags angezeigt.

4.6.3 Top 10 Songs

Im Scatterplot werden nur mehr die gewählten Songs angezeigt.

In Tags werden alle Tags der/des gewählten Songs angezeigt. In Histogramm (Listeners)/(Playcount) wird nur der Bucket angezeigt in dem sich der/die gewählte(n) Song(s) befinden.

4.6.4 Histogramm (Listeners)/(Playcount)

Im Scatterplot werden die Songs angezeigt die sich in den gewählten Buckets befinden.

5 Results

Für unser Projekt haben wir uns ein Szenario ausgedacht, welches wir im Kapitel 1.4 (S. 2) vorgestellt haben. Wenn Rainer die Applikation öffnet sieht er als erstes einen Überblick, siehe Abbildung 5 (S. 4, über alle Ansichten).



Abbildung 5: Überblick über unsere Applikation

Er bemerkt das Suchfeld unter dem Scatterplot und sucht sein Lieblingslied mit dem Namen Stronger und klickt das Ergebnis bei der Suche an. Da er gute Songs sucht, entschließt sich nur in den Top 5 Tags zu suchen. Nach der Auswahl erhält er die Ansicht auf Abbildung 6 (S. 4).

In diesem Schritt sieht er auch das das Lied, in den beiden Histogrammen im 20 Bucket liegt und somit eher in den

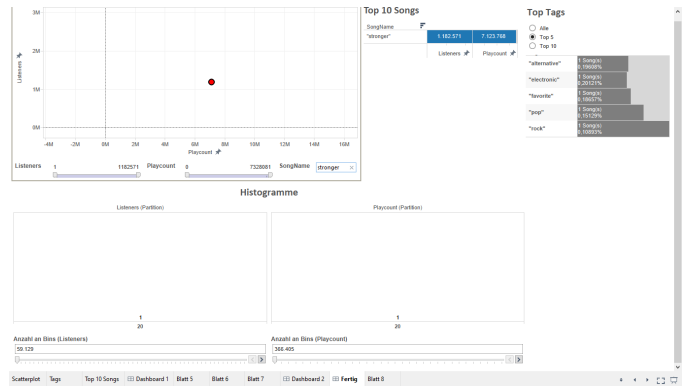


Abbildung 6: Textsuche nach Stronger und auswählen des Ergebnisses

Attributen Listener und Playcount höhere Werte hat. Des Weiteren sieht er in der Top 10 Song Ansicht sofort die zwei Werte für die eben genannten Attribute. Als er die Top 5 Tags durchforstet entschließt er sich dafür Lieder mit dem Tag Rock genauer zu betrachten. Dazu klickt er auf diesen und löscht die zwei vorher getroffen Auswahlen und erhält die Ansicht auf Abbildung 7 (S. 4).

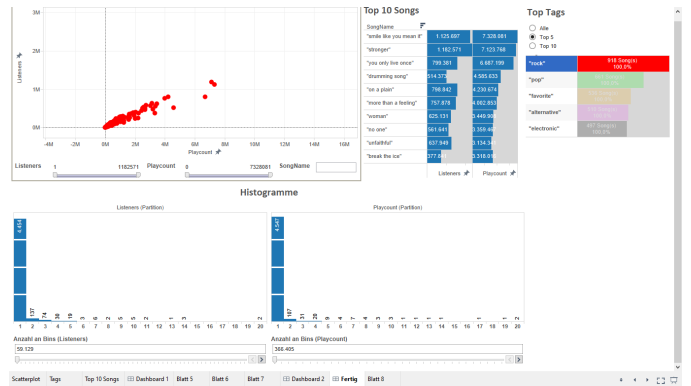


Abbildung 7: Filterung der Daten durch einen Tag

Nun sieht er alle Lieder die den Tag Rock besitzen und er kann sich neue Lieder suchen und seinen musikalischen Horizont erweitern.

Da sich andere Szenarios teilweise sehr, zu unserem jetzt vorgestellten, ähneln würden, möchten wir hier noch ein paar Screenshots von anderen Funktionen einfügen.

In der Abbildung 8 (S. 5) sieht man wie man mit den Histogrammen interagieren kann. Das Histogramm für das Attribut Playcount wurde so eingestellt das es nur 3 Buckets zeigt und hier wurde dann der erste gewählt. Für das Histogramm von Listeners wurde auf fünf Bucketes gestellt und der dritte gewählt.

In der Abbildung 9 (S. 5) haben wir den Bereich des Scatterplots mit Hilfe der Schieberegler eingegrenzt und das Diagramm weiter heran gezoomt. Desweiteren sieht man das Rollover-Fenster für einen Punkt.

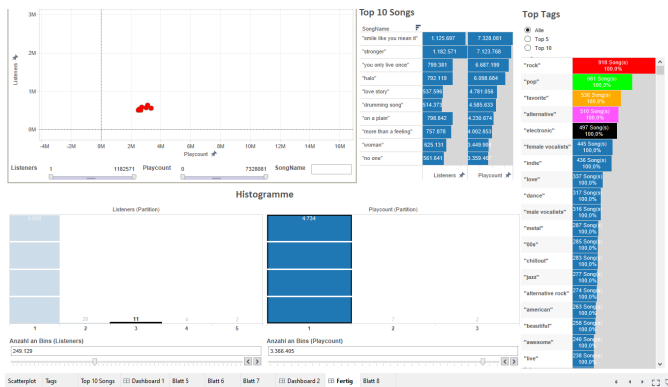


Abbildung 8: Histogramme: Anzahl der Buckets geändert und Buckets ausgewählt



Abbildung 10: Markierung mehrerer Punkte

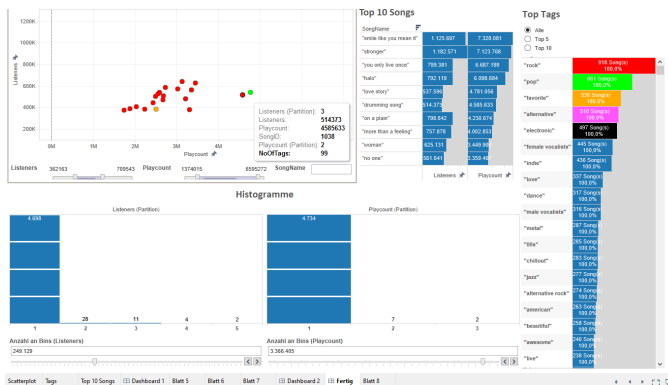


Abbildung 9: Eingrenzung des Bereichs des Scatterplot und Rollover-Fenster eines Punkts

In der Abbildung 10 (S. 5) sieht man die Applikation wenn mehrere Punkte ausgewählt wurden. In der Top 10 Songs Ansicht kann man Lieder direkt, in Bezug auf die Attribute Playcount und Listeners, vergleichen. Des Weiteren wurden die Histogramme angepasst, damit man die Verteilung der Auswahl sieht und in der Auflistung der Tags werden nur Tags angezeigt, welche in den markierten Punkten vorhanden sind.

Hinsichtlich der Performance gibt es keine großen Auffälligkeiten. Zwar treten bei bestimmten Auswahlmöglichkeiten teilweise Wartezeiten auf, jedoch halten sich diese in Grenzen und dauern in der Regel nicht länger als 10 Sekunden. Somit lässt sich mit dem Tool bequem arbeiten. Wie auch im nächsten Abschnitt erwähnt war unser erster Ansatz in D3 Performancemäßig der Tableau alternative weit unterlegen.

6 Discussion

In diesem Abschnitt gehen wir vor allem auf die Stärken und Schwächen unserer Implementation ein, aber auch welche Lektionen wir beim Implementieren gelernt haben, beziehungsweise was wir generell beim Ausführen des Projektes an Erfahrungen gemacht haben.

Eine Stärke der Visualisierung ist vor allem die Verwendung von Tableau, da dieses ein fertiges und getestetes Tool ist und alleine damit viele kleine Implementationsfehler ausgeschlossen werden können. Wir konnten uns somit voll und ganz auf die Visualisierung selbst konzentrieren. Im Ver-

gleich zu unseren Ansätzen mit D3 fiel vor allem die weitaus bessere Performance auf, die durch Tableau erreicht werden konnte.

Ebenfalls als Stärke sind die Diagrammart zu nennen, welche relative einfach und unkompliziert zu lesen sind und damit von jedermann gelesen werden können, wodurch eine größere Zielgruppe erreicht werden kann.

Obwohl für uns die Vorteile der Implementierung durch Tableau überwiegen, brachte dies auch einige Nachteile mit sich. Beispielsweise mussten wir einige Abstriche in Designentscheidungen hinnehmen. Man war auf den Rahmen angewiesen den Tableau einem bietet.

Wie auch bei jedem anderen größeren Projekt, konnten wir einiges an Wissen und Know-how mitnehmen. Vor allem im Zeitmanagement haben wir uns trotz relativ genauer Planung mehrmals grob verschätzt, weswegen viele Arbeitsschritte unter Zeitdruck durchgeführt werden mussten. Schwierig war zu Beginn vor allem eine grobe Vorstellung vom Gesamtbild zu bekommen. Wir änderten unsere Prototypen mehrmals ziemlich stark, und mussten somit sehr oft von neuem beginnen, was uns im Zeitplan immer stark zurückwarf. Dieser Fehler lag einerseits daran, dass wir die nötige Kreativität unterschätzten, die für eine Visualisierung notwendig ist, andererseits wäre dieser Zeitdruck durch bessere Planung und großzügigere Bemessung der einzelnen Arbeitsschritte mit Sicherheit zu vermeiden gewesen.

Auch haben wir einige Erfahrungen mit Tableau gemacht und können damit im Vergleich zu vorher, relativ routiniert umgehen. In einigen Teilbereichen haben wir die Grenzen des Programmes erreicht und können nun auch einigermaßen die Mächtigkeit von Tableau einschätzen.

Literatur

- [1] Google research music timeline. <https://research.google.com/bigpicture/music/>.
- [2] Last.fm. <http://www.lastfm.at>.
- [3] T. Munzner. *Visualization Analysis and Design*. CRC Press, 2014.