

# Algorithmen zur Umsetzung von Text in Töne

**Anne Peter**

Bauhaus-Universität Weimar  
Fakultät Medien  
anne.peter@uni-weimar.de  
Matrikelnr.: 114421

**Norina Marie Grosch**

Bauhaus-Universität Weimar  
Fakultät Medien  
norina.marie.grosch@uni-weimar.de  
Matrikelnr.: 114456

## KURZFASSUNG

Wir haben das Programm *Text to Music* entwickelt, das die Emotionen in einem gegebenen Text analysiert und darauf basierend algorithmisch Musik generiert. *Text to Music* nutzt dafür die Zusammenhänge der Emotionen, die im Text auftauchen und wie Emotionen in der Musik verdeutlicht werden können.

Das Programm könnte in der Zukunft genutzt werden, um passende Musik zu einem Buch aufgrund der Textanalyse vorzuschlagen oder im Bereich der Musikgenerierung für Drehbücher und audiovisuelle E-Books Anwendung finden.

## 1 PROBLEMSTELLUNG

Wir haben es uns zur Aufgabe gemacht einen Text - wir haben uns auf Romane und Märchen beschränkt - algorithmisch und vollkommen automatisch in Musik umzuwandeln. Dafür muss man zuerst eine Verbindung zwischen dem Text und der darauf basierenden Musik finden. Die empfundenen Emotionen beim Lesen eines Textes beziehungsweise beim Hören von Musik sind hierfür die Grundlage. Die Beschränkung der Texte auf Romane und Märchen haben wir auf Grund der Dramaturgie, die in diesen Genres einem ähnlichen Muster folgt, getroffen. Es gibt immer einen Anfang, einen Mittelteil und ein Ende. Dieser Aufbau lässt sich gut auf ein Musikstück übertragen.

Um den Text auf die Verteilung der Emotionen zu untersuchen, haben wir das *NRC Word-Emotion Association Lexicon* [17] benutzt. Dieses hat eine große Anzahl von Wörtern vieler Sprachen die acht häufigsten Emotionen (Wut, Erwartung, Abneigung, Angst, Freude, Trauer, Überraschung, Vertrauen) zugeordnet und die Wörter nach positiv, negativ oder neutral klassifiziert.

Im Abschnitt 2 werden vorherige und verwandte Arbeiten zu unserer Arbeit beschrieben. Im Abschnitt 3 werden Hintergründe, die zum Verstehen unserer Arbeit benötigt werden, erklärt. Danach wird im Abschnitt 4 beschrieben wie unser Programm funktioniert und im Abschnitt 5 wird die grafische Benutzeroberfläche erklärt. Im Abschnitt 6 und 7

werden die Ergebnisse diskutiert und ein Ausblick auf zukünftige Arbeiten gegeben.

## 2 VERWANDTE ARBEITEN

In der wissenschaftlichen Arbeit *TransProse* [7] von Hannah Davis und Saif Mohammad wird das gleichnamige System vorgestellt, dass automatisch Musik auf Basis eines Textes erstellt. Es bedient sich musikalischer Elemente wie Oktaven, Tonhöhe und Tempo, die es gezielt einsetzt, um die allgemeine Gefühlslage und die Stimmungswechsel des Werks wiederzugeben.

Es beginnt mit einer Textanalyse, die durch das *NRC Word-Emotion Association Lexicon* [17] unterstützt wird. Die einzelnen Emotionen des Textes sowie die Häufigkeit dieser und aller Wörter wird herangezogen, um den Text zu charakterisieren. Das oben genannte Wörterbuch beinhaltet Wörter, die in acht Emotionen (Wut, Erwartung, Abneigung, Angst, Freude, Trauer, Überraschung, Vertrauen) und positiv, neutral und negativ unterteilt sind.

Zu Beginn der Textanalyse wird der Text in vier Abschnitte mit je vier Unterabschnitten eingeteilt. Jeder Abschnitt steht für eine ganze Note im späteren Musikstück. Das Verhältnis aller im Werk vorkommenden als positiv eingestuft Wörter wird den negativ eingestuft gegenübergestellt, um das überwiegende Gefühl festzustellen. Des Weiteren wird die Relation zwischen allen Wörtern und emotionalen Wörtern betrachtet sowie zwischen einem Abschnitt und seiner Unterabschnitte, um die Emotionalität des Textes anzugeben. Das am häufigsten und am zweithäufigsten vorkommende Gefühl gibt die dominanteste beziehungsweise zweitdominanteste Emotion an.

In jedem Abschnitt und Unterabschnitt werden alle Wörter und die emotionalen Wörter, die mit Hilfe des Wörterbuchs im jeweiligen Abschnitt gefunden wurden, gezählt. Diese Beziehung spiegelt die Emotionsdichte wider. Dabei wird nach der generellen Emotionsdichte und die bestimmter Emotionen unterschieden. So gibt es zum Beispiel das Verhältnis zwischen traurigen Wörtern und allen Wörtern des

Textes, das die Traurigkeitsdichte über den gesamten Text bestimmt. Dieses setzt man außerdem in Relation zur Freudigkeitsdichte, um beide Gefühle einander gegenüber zustellen.

Nachdem die Gefühle im Text erkannt und analysiert wurden, werden diese Erkenntnisse in eine Spezifikation überführt, die von der Java-Programmierschnittstelle *Jfugue* [16] in Musik umgewandelt werden kann. Sie erhält Angaben zu den einzelnen Tönen, den Tonhöhen, Oktaven, Notenwerten, Tempo sowie Tonart und generiert daraufhin drei Melodien, die zusammen angehört werden können. Auf der Internetseite von *TransProse* [6] sind verschiedene Hörbeispiele von solchen Transformationen zu hören, darunter die englischen Versionen von *Peter Pan*, *Alice im Wunderland* und *Herr der Fliegen*.

Wir haben uns diese Arbeit als Vorbild genommen, sie nachimplementiert und uns mit Möglichkeiten der Weiterentwicklung beschäftigt.

### 3 THEORIE

In diesem Abschnitt beschäftigen wir uns mit der Theorie, auf der wir unser Programm aufgebaut haben. Dazu haben wir uns mit der Theorie der Musik und wie die einzelnen Elemente zum Klang beitragen, beschäftigt, sowie mit dem Zusammenhang von Wörtern und Emotionen.

#### 3.1 Musik

Um Musik zu generieren, muss man zuerst Musik als solches definieren. Musik im physikalischen Sinne sind Frequenzen, die in aller Regel abgespielt werden, um einem Zuhörer zu gefallen. Die Frequenzen treffen im Ohr auf das Trommelfell, „dort findet eine Umwandlung der Luftwellen in mechanische Energie statt.“ [10] Über Hammer, Amboss und Steigbügel wird diese an das Innenohr weitergegeben und letztlich in Nervenimpulse umgewandelt, die im Gehirn verarbeitet werden können.

In der Musik bestimmt die Frequenz die Tonhöhe und die relative Zeit die Tondauer. Die relative Tondauer wird als beispielsweise als ganze Note oder halbe Note beschrieben. Ein C als ganze Note klingt entsprechend wie eine 65,4 Hz [11] ganze Note, ein a als halbe Note klingt wie eine 220 Hz [11] Frequenz, die nur halb solange dauert. Mehrere Tonhöhen mit relativen Tondauern können in Takten zusammengefasst werden und auch als Melodie bezeichnet werden und somit als Musik. Ein Musikstück kann aus einer Melodie bestehen, aber auch aus mehreren, die zur gleichen Zeit gespielt werden. Das eröffnet unendliche Möglichkeiten Musik zu gestalten. Eine unserer Herausforderungen ist es, nun diese Möglichkeiten zu nutzen, um angemessene beziehungsweise anhörbare Musik zu generieren.

Das Tempo der Musik wird in *bpm* (Beats per minute) angegeben, was übersetzt Schläge pro Minute heißt. Oft wird eine Viertelnote als Schlag angenommen, das heißt die *bpm* geben dann die Anzahl der Viertelnoten an, die in einer Minute gespielt werden. Als Beispiel nehmen wir an, unsere Melodie ist eine Sequenz, die aus 261,6 Hz Viertelnote, 440

Hz ganze Note, 329,6 Hz Viertelnote, 440 Hz halbe Note, 261,6 Hz halbe Note besteht. Die Schläge in Viertelnoten sind dann:  $1 + 4 + 1 + 2 + 2 = 10$  und werden in 10/120 Minuten abgespielt, was bedeutet, dass die Sequenz 5 Sekunden dauert.

Um zu definieren wie viele Schläge pro Minute in einem Takt sind, gibt es die Taktangabe. Sie definiert auch welche Tondauer in einem Takt repräsentiert wird. Beispielsweise definiert ein  $\frac{3}{4}$  Takt 3 Schläge pro Takt und 4 Noten pro Schlag. Der  $\frac{4}{4}$  Takt ist die weitverbreitetste Taktangabe.

Töne, die Faktoren oder Vielfache von einander sind, werden von Menschen als konsonant und angenehm wahrgenommen, weil sie überlappende Höhen und Tiefen haben. Die Intervalle, in denen sich die Frequenzen überlappen, werden Oktaven genannt. Eine Oktave besteht theoretisch aus sieben Tönen, endet aber mit dem ersten Ton aus der nächst höheren Oktave und umfasst somit acht Töne.

Basierend auf dieser Frequenzeinteilung entstehen Tongeschlechter, Dur und Moll. Dur ist eine Reihe von Tonhöhen in folgendem Muster: Ganztonschritt - Ganztonschritt - Halbtonschritt - Ganztonschritt - Ganztonschritt - Ganztonschritt - Halbtonschritt. Moll folgt diesem Muster: Ganztonschritt - Halbtonschritt - Ganztonschritt - Ganztonschritt - Halbtonschritt - Ganztonschritt - Ganztonschritt. In der Musik definiert man ein Tongeschlecht immer mit einer Tonart, zum Beispiel ist C-Dur „eine Tonart des Tongeschlechts Dur, die auf dem Grundton c aufbaut.“ [4] Die Tonleiter von C-Dur lautet C, D, E, F, G, A, B, C und von C-Moll C, D, D#, F, G, G#, A#, C. <sup>1</sup> Dur klingt dabei fröhlicher und angenehmer als Moll, welches eher düsterer erscheint.

Wie angenehm oder stabil dabei zwei Töne klingen, die nacheinander gespielt werden, nennt man Konsonanz. Wenn sie unangenehm zusammen klingen, spricht man von Dissonanz. Zu Konsonanz und Dissonanz in der Musik gibt es diverse Theorien, die vor allem auf einen kulturellen Unterschied hinweisen. Davis entscheidet sich in ihrer Arbeit die Tonleitern in konsonante Folgen zu ändern, C-Dur wird zu C, G, E, A, D, F, B und C-Moll zu C, G, D#, G#, D, F, A#. Damit klingt die später erzeugte Musik angenehmer und kann die Emotionen des Textes besser repräsentieren.

#### 3.2 Lexikon

Das *NRC Word-Emotion Association Lexicon* [17] ist eine Tabelle bestehend aus 11 Spalten und 14183 Zeilen. Es ist in verschiedenen Sprachen, zum Beispiel Deutsch, Englisch, Russisch, Französisch, Japanisch und Arabisch, verfügbar. Den 14182 Wörtern werden acht Emotionen (Wut, Erwartung, Abneigung, Angst, Freude, Trauer, Überraschung, Vertrauen) und zwei Stimmungen (positiv und negativ) zugeordnet. Die Verknüpfung erfolgt über boolesche Werte. Wird ein Wort zum Beispiel mit *Freude* assoziiert, steht in der entsprechenden Zelle eine Eins. Wird es damit nicht in Verbindung gebracht, findet man an jener Stelle eine Null. Wird nun ein

<sup>1</sup> Wir benutzen die Tonleitern aus der Arbeit von Davis [7], im Gegensatz zur deutschen Schreibweise wird hier B anstatt H verwendet.

Wort weder mit *positiv* noch mit *negativ* verknüpft, kann man es als *neutral* betrachten.

Das erste Wörterbuch dieser Art wurde in Englisch verfasst. Alle weiteren Sprachen - über zwanzig - wurden durch schlichtes Übersetzen der Emotionswörter mit Hilfe des Übersetzers von Google hinzugefügt. Es wird darauf hingewiesen, dass einige Übersetzungen von Google inkorrekt oder simple Transliteration des englischen Originals sein können [17].

Als wir uns das Lexikon genauer angesehen haben, haben wir bemerkt, dass 7714 Zeilen (im Deutschen wie im Englischen) nur mit Nullen gefüllt sind und daher mit keiner Emotion oder Stimmung verknüpft sind. Dadurch sind sehr viele Einträge für unsere Textanalyse nicht nutzbar, da sie keine relevanten Informationen enthalten.

Uns ist außerdem aufgefallen, dass 2370 Zeilen des deutschen Wörterverzeichnisses leer sind und nicht übersetzt wurden. Somit werden im Deutschen weniger Wörter mit Hilfe des Wörterbuchs klassifiziert als im Englischen, obwohl auf der Webpräsenz des *NRC Word-Emotion Association Lexicon* [17] das deutsche Wörterbuch mit gleich vielen Einträgen wie das englische beworben wird. Dem englischen Vokabular wurden tatsächlich 14182 Ausdrücke den acht Emotionen und den zwei Stimmungen zugeordnet.

Wir haben daher für unser Programm *Text to Music* beide Wörterbücher in Betracht gezogen. Dadurch ist es uns möglich Texte der jeweiligen Sprachen zu analysieren und aufgrund dessen Musik zu generieren.

## 4 KONZEPT

Unser Programm *Text to Music* wandelt einen gegebenen Text in Musik um, indem es drei essentielle Schritte befolgt. Zuerst analysiert es den Text auf die vorkommenden Emotionen und berechnet die Verteilungen dieser, dann werden mit Hilfe dieser Berechnungen die Spezifikationen für die Musik generiert. Im letzten Schritt geben wir die Spezifikationen im entsprechenden Format an die Bibliothek *JFugue* [16] weiter, die dann hörbare Musik produziert. Die Details der einzelnen Schritte werden in diesem Abschnitt genauer beschrieben.

### 4.1 Programmablaufplan

In Abbildung 1 sieht man den Programmablaufplan von *Text to Music*, der die wesentlichen Schritte zur Erstellung einer Musik auf Basis eines Textes zusammenfasst.

### 4.2 Textanalyse

Um mit der Textanalyse beginnen zu können, lesen wir die übergebene Datei ein. Diese muss im *txt*-Format vorliegen und die *UTF-8* (Abkürzung für *8-Bit UCS Transformation Format*) Zeichenkodierung besitzen, um sie optimal untersuchen zu können. Wir entfernen Zahlen, Sonderzeichen, doppelte Leerzeichen und wandeln alles in Kleinbuchstaben um, damit wir einen zuverlässigen Abgleich mit dem Wörterbuch gewährleisten können.

Anschließend zählen wir alle Wörter im abgeänderten Text und speichern das Ergebnis. Es müssen mindestens 16

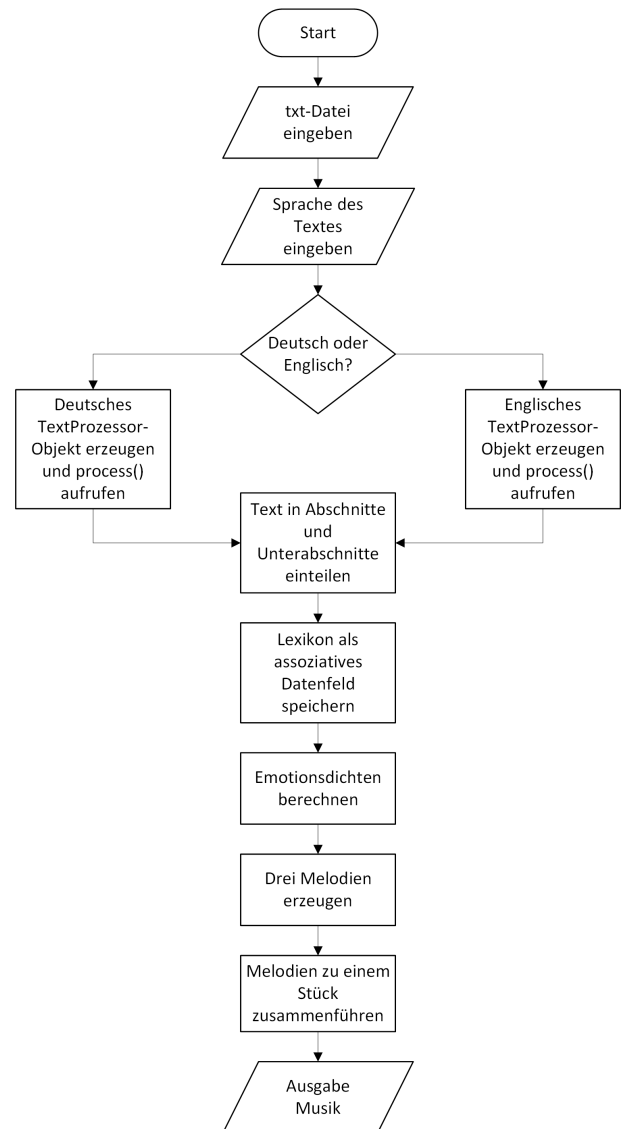


Abbildung 1: Programmablaufplan von *Text to Music*

Wörter vorhanden sein, um den nächsten Schritt vollziehen zu können.

Wir teilen den Text in vier Abschnitte mit je vier Unterabschnitten ein, während wir darauf achten, dass ganze Wörter erhalten bleiben. Jeder dieser sechzehn Unterabschnitte steht für einen Vierteltakt beziehungsweise für eine ganze Note im zu erzeugenden Tonstück. Jede Musik, die mit unserem Programm generiert wird, besteht aus genau 16 Takten, egal wie lang der Text ist. Das ist der Grund, weshalb die Datei nicht weniger als 16 Wörter beinhalten darf.

Danach lesen wir je nach Text das deutsche oder das englische *NRC Word-Emotion Association Lexicon* [17] als assoziatives Datenfeld ein, um es mit dem Text abgleichen zu können. Dieses Datenfeld speichert die emotionalen Wörter und deren zugeordnete Gefühle ab.

Dafür haben wir verschiedene Methoden entwickelt:

- In der einfachsten Variante unserer Textanalyse betrachten wir jeweils ein Wort für sich und gleichen es mit dem Wörterbuch ab. Gibt es einen Treffer, speichern wir diesen.
- In der zweiten Variante betrachten wir zusätzlich die nächsten drei Wörter im Text und suchen im Wörterbuch nach möglichen Treffern von mehrteiligen Wörtern. Es wird immer der längste Treffer gespeichert.
- In der dritten Variante betrachten wir zusätzlich zu mehrteiligen Wörtern auch noch deren Verneinung. Finden wir vor dem Wort eines, das auf Verneinung hinweist, negieren wir die Stimmung. Wenn die Stimmung neutral ist, belassen wir sie dabei.

Die Verneinungen, nach denen wir suchen, sind: *nicht, kein, keine, keiner, keinen, keinem* oder *keines*. Im Englischen beachten wir folgende Ausdrücke: *not, none, dont, doesnt, isnt, arent, wasnt, werent, wont, hasnt, havent, hadnt, didnt, cant, couldnt, neednt, shouldnt* und *musnt*. Hier ist zu beachten, dass wir *dont* anstatt von *don't* schreiben, da Sonderzeichen wie ' vorher aus unserem Originaltext entfernt wurden.

So ergibt sich für jeden Unterabschnitt eine absolute Anzahl an Wörtern jeder Emotion und Stimmung. Mit diesen Summen können wir dann die Dichte berechnen. Die emotionalen Wörter im gesamten Text werden durch die Anzahl aller Wörter geteilt, um die generelle Emotionsdichte des Werkes zu bezeichnen. Die Dichte einer bestimmten Emotion ergibt sich aus der Division der Anzahl der gefundenen Wörter, die diesem Gefühl zugeordnet sind und der Anzahl aller Wörter der entsprechenden Passage.

### 4.3 Musikgenerierung

Für jedes Musikstück generieren wir drei separate Melodien, zwei klingen laut Davis zu dünn und vier oder mehr scheinen weniger kohäsiv. [7] Im Folgenden werden die einzelnen Eigenschaften der Melodien erklärt.

#### 4.3.1 C Dur oder C Moll

Das Tongeschlecht, das wir dem Text zuordnen, wird durch das Verhältnis von positiven und negativen Wörtern berechnet. Enthält der Text mehr positive als negative Wörter ist es ein fröhlicher Text und wird dem Tongeschlecht Dur zugeordnet. Wenn der Text aber mehr negative Wörter beinhaltet, wird er Moll zugeordnet. Es gibt allerdings noch keine Forschungen, die eindeutige Aussagen darüber treffen, welche Tonart positiv klingt und welche negativ. Deshalb haben wir uns, wie Davis auch, dazu entscheiden nur zwischen C-Dur und C-Moll zu entscheiden, da dies die beiden gängigsten Tonarten sind. Die festgelegte Tonart gilt für das gesamte Musikstück.

Für zukünftige Arbeiten wäre es allerdings interessant mit anderen Tonarten zu experimentieren und Wahrnehmungsstudien durchzuführen. Beispielsweise wäre es denkbar den Höhepunkt eines Textes mit einem Wechsel der Tonart zu betonen oder auch einen Wechsel von Moll zu Dur, wenn der

Text positiver wird oder umgekehrt, ist denkbar. Auch könnte man über verschiedene Tonarten für die einzelnen Melodien nachdenken.

#### 4.3.2 Melodien

Um die Emotionen im Text möglichst gut abzubilden, generieren wir drei Melodien. Die erste Melodie beschreibt die gesamte Emotionsdichte. Wir nennen diese Melodie im folgenden  $M_0$ . Die anderen beiden Melodien ( $M_{e1}, M_{e2}$ ) beschreiben jeweils die dominanteste Emotion ( $e1$ ) und die zweitdominanteste Emotion ( $e2$ ) des Textes.

Um die Oktaven zu berechnen, betrachten wir das Verhältnis von Freude zu Trauer, im Folgenden  $FT$  genannt. Zuerst berechnen wir die Oktave für  $M_0$ , dafür benutzen wir die Formel aus der Arbeit von Davis:

$$Oct(M_0) = 4 + r \left( \frac{(FT - FT_{min}) \cdot (6 - 4)}{FT_{max} - FT_{min}} \right)$$

Die Gleichung bestimmt eine Oktave aus der Menge  $\{4, 5, 6\}$ , dabei ist  $r$  eine Funktion, die das Ergebnis auf eine ganze Zahl rundet.  $FT_{min}$  und  $FT_{max}$  sind Werte, die Davis in ihrer Arbeit mit Hilfe einer Analyse von mehreren Romanen bestimmt hat. Die genauen Werte sind allerdings nicht in ihrer Arbeit zu finden und wir haben sie deshalb mit  $FT_{min} = 0$  und  $FT_{max} = 1$  provisorisch festgelegt. Folglich haben sie keinen Einfluss auf die Bestimmung der Oktave. Um hier zuverlässige Werte zu finden müsste man eine Großzahl an Texten analysieren und vergleichen, für zukünftige Arbeiten ist das durchaus von Interesse.

Die Oktave für  $M_{e1}$  wird wie folgt berechnet:

$$Oct(M_{e1}) = \begin{cases} Oct(M_0) + 1, & \text{falls } e1 \text{ Freude oder} \\ & \text{Vertrauen ist} \\ Oct(M_0) - 1, & \text{falls } e1 \text{ Wut, Angst, Trauer} \\ & \text{oder Abneigung ist} \\ Oct(M_0), & \text{sonst} \end{cases}$$

- $M_{e1}$  wird eine Oktave höher als die von  $M_0$  gesetzt, wenn  $e1$  eine positive Emotion ist
- $M_{e1}$  wird eine Oktave tiefer als die von  $M_0$  gesetzt, wenn  $e1$  eine negative Emotion ist
- Wenn  $e1$  eine neutrale Emotion ist, also Überraschung oder Erwartung, wird die selbe Oktave wie bei  $M_0$  benutzt

Der Grund dafür ist, das höhere Oktaven einen fröhlicheren und positiveren Eindruck machen als tiefere Oktaven. Für  $M_{e2}$  wiederholen wir die Berechnung der Oktave von  $M_{e1}$  mit der zweitdominatesten Emotion.

#### 4.3.3 Struktur und Noten

Die Struktur und Noten der drei Melodien ( $M_0, M_{e1}$  und  $M_{e2}$ ) werden nach dem selben Prinzip generiert, deshalb wird im Folgenden die allgemeine Berechnung erklärt. Die Emotionsdichte für die Berechnungen bezieht sich immer auf die Emotion, die der jeweiligen Melodie zugeordnet ist.

Alle Melodien werden über die vier Abschnitte des Textes, die jeweils wiederum in vier Unterabschnitte eingeteilt sind, kalkuliert. Das bedeutet, dass jeder Abschnitt durch vier Takte repräsentiert wird. Jeder Takt hat die Länge einer ganzen Note. Wie in Abschnitt 3.1 definiert, ist ein Takt eine Sequenz von Tönen. Die Anzahl an Noten, die Tonhöhe und die relative Tondauer werden so berechnet, dass sie die Emotionen an der dazugehörigen Stelle im Text abbilden.

Zuerst berechnen wir die Anzahl der Noten und damit die relative Tondauer. Wir wählen dabei zwischen einer ganzen Note, zwei halben Noten, vier Viertelnoten, acht Achtelnoten oder sechzehn Sechzehntelnoten aus. Eine hohe Emotionsdichte in einem Unterabschnitt bilden wir auf viele, sehr kurze Töne ab. Ist die Emotionsdichte gering, bilden wir auf wenige, dafür aber längere Töne ab. Um zu entscheiden, ob die Emotionsdichte hoch oder niedrig ist, suchen wir die maximale und minimale Emotionsdichte der Unterabschnitte und bilden darauf eine lineare Verteilung. Je nachdem in welches der fünf Intervalle in dieser Verteilung die Emotionsdichte des aktuellen Unterabschnitts fällt, entscheiden wir uns für eine der Tondauern und damit auch für die Anzahl der Noten. Diese müssen immer zusammen die Dauer einer ganzen Note ergeben. Das Ergebnis sind kurze Noten an den Textabschnitten mit einer hohen emotionalen Aktivität und lange Noten bei weniger emotionalen Abschnitten. Das lässt das Musikstück je nachdem aktiver oder passiver klingen.

Danach berechnen wir die Tonhöhen der einzelnen Töne. Nehmen wir an, wir haben vorher eine Anzahl  $n$  an Tönen berechnet, die in diesem Takt vorkommen, dann teilen wir die Wörter des Unterabschnitts in  $n$  äquivalente Teile auf. Wir berechnen nur wieder die minimale und maximale Emotionsdichte auf diesen  $n$  Teilen. Daraus erstellen wir erneut eine lineare Verteilung, um dann die einzelnen Teile auf diese abzubilden und ihnen Tonhöhen zuzuordnen. Für die Zuordnung benutzen wir die in 3.1 genannten konsonanten Tonfolgen, je nachdem ob wir vorher C-Dur oder C-Moll ausgewählt haben. Dabei bilden wir hohe Emotionsdichten auf das Ende der Folge und niedrige Dichten auf den Anfang ab. Das hat zur Folge, dass eine hohe emotionale Aktivität dissonanter klingt als eine geringe emotionale Aktivität, die eher konsonant klingt.

Davis führt in ihrer Arbeit an dieser Stelle die Wiederholung jeden Abschnitts ein, um „ein strukturiertes und melodisches Gefühl zu erzeugen“ [7]. Wir haben diese Repetition auch in unserem Programm implementiert, jedoch klang sie unserer Meinung nach zu merkwürdig und verfehlte den Zweck die Konsonanz zu verstärken. Davis meint hingegen, dass „das Stück ohne Wiederholung weniger kohäsiv klingt“ [7]. Wir haben uns, nachdem wir mehrere Musikstücke mit sich wiederholenden Abschnitten angehört hatten, dafür entschieden die Wiederholung entfallen zu lassen.

#### 4.3.4 Tempo

Wir haben uns auf einen  $\frac{4}{4}$  Takt festgelegt, da er der weitverbreitetste ist. Das heißt, dass jeder Takt vier Schläge hat. Um das Tempo, also die Schläge pro Minute, im Text zu be-

stimmen, berechnen wir zuerst den Aktivitätswert des gesamten Textes. Jede der acht Emotionen ist entweder aktiv, passiv oder neutral. Der Aktivitätswert ( $AW$ ) berechnet sich aus der Differenz der aktiven und passiven Emotionen. Freude und Wut sind aktive Emotionen. Trauer ist dahingegen passiv. Die anderen fünf Emotionen (Erwartung, Überraschung, Angst, Abneigung und Vertrauen) sind neutral oder nicht klar zuzuordnen, weshalb sie nicht betrachtet werden. [7]

Wie in der Arbeit von Davis berechnen wir das Tempo zwischen 40 und 180 Schlägen pro Minute. Diese Grenzen hat sie subjektiv festgelegt. Das Tempo wird mit folgender Formel berechnet:

$$tempo = 40 + \frac{(AW - AW_{min}) \cdot (180 - 40)}{AW_{max} - AW_{min}}$$

Damit wird eine hohe Aktivität auch einem hohen Tempo zugeordnet und eine geringe Aktivität einem langsameren Tempo. Die Grenzwerte für den Aktivitätswert sind hierbei festgelegt,  $AW_{min} = -0.002$  und  $AW_{max} = 0.017$ . Davis hat diese Werte aus ihrer Kollektion an Texten gemessen.

#### 4.4 Spezifikationen in Musik umwandeln

Damit die berechnete Musik auch hörbar ist, haben wir die Java-Programmierschnittstelle *JFugue* [16] benutzt. Diese ermöglicht es, verschiedene Instrumente, Noten, Oktaven, Tondauern, etc. in einem Java-Programm anzugeben und abzuspielen. Dafür benötigt *Jfugue* eine spezielle Formatierung der Zeichenkette, die übergeben wird. Beispielsweise kann diese so aussehen:

```
KCmaj V0 T160 A6/0.125 C6/0.5 F6/0.25
F6/0.125 G6/1 D6/0.25 D6/0.25 A6/0.5 ...
```

$K$  steht für die Tonart und  $Cmaj$  für C-Dur. Mit  $V0$  gibt man die erste Melodie (oder Stimme) an, weitere kann man dann mit  $V1$  und  $V2$  usw. angeben.  $T$  referenziert das Tempo, welches danach in Schlägen pro Minute, im Beispiel 160, angegeben wird. Darauf folgen die einzelnen Töne, die im Muster *Tonhöhe Oktave / Tondauer* angegeben werden. Die Tonhöhe wird immer als Gleitkommazahl des zugehörigen Bruchs dargestellt, also ist beispielsweise 0.5 eine halbe Note und 0.25 eine Viertelnote. Diese Zeichenkette übergibt man an *JFugue* und kann sie abspielen oder als Midi-Datei speichern, welche man wiederum in MP3 umwandeln kann.

#### 5 GRAFISCHE BENUTZEROBERFLÄCHE

Wir haben uns dazu entschieden unseren Algorithmus zur Musikgenerierung in eine Grafische Benutzeroberfläche (GBO) zu integrieren. Die GBO bekommt als Eingabe eine Text-Datei mit der Endung .txt, wie in Abbildung 2 [a] zu sehen. Als nächstes kann der Benutzer bei [b] ein Lexikon auswählen, mit diesem wird dann der Text analysiert. Es gibt die Auswahlmöglichkeiten *German* für das deutsche Lexikon und *English* für das englische Lexikon.

Bei [c] kann der Benutzer die Analysemethoden auswählen, *NRC Lexicon simple*, *NRC Lexicon advanced* und *NRC*

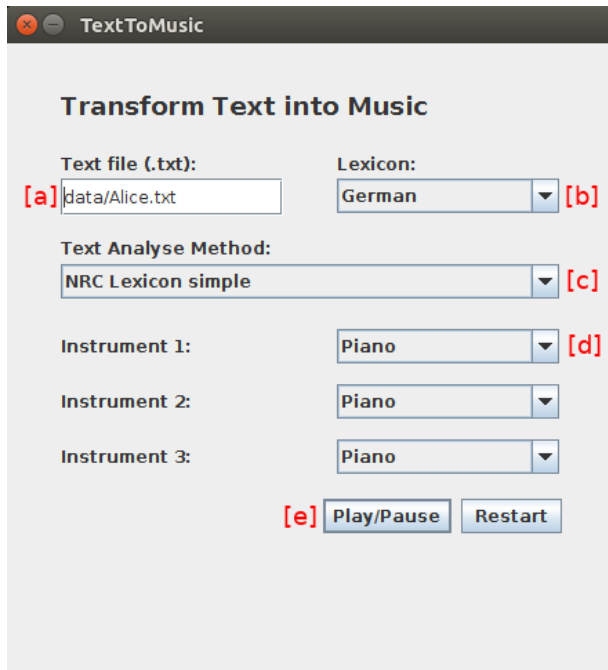


Abbildung 2: GBO nach dem Starten

*Lexicon advanced with negation* stehen zur Verfügung. *NRC Lexicon simple* analysiert den Text primitiv Wort für Wort und erkennt nur Wörter aus dem Lexikon, die nicht durch ein Leerzeichen getrennt sind. *NRC Lexicon advanced* hingegen kann auch mehrteilige Wörter aus dem Wörterbuch im Text erkennen, *NRC Lexicon advanced with negation* erkennt zusätzlich zu mehrteiligen Wörtern auch eine vorangegangene Negation des Wortes und ändert dessen Stimmung ins Gegenteil.

Nun kann der Nutzer bei [d] für jede der drei Melodien, die generiert werden, ein eigenes Instrument auswählen. Standardmäßig ist hier Piano eingestellt, allerdings stehen insgesamt 123 verschiedene Instrumente aus der JFugue-Bibliothek zur Verfügung.

Wenn alle Optionen ausgewählt wurden und eine Text-Datei angegeben wurde, kann der Nutzer auf *Play/Pause* drücken und das Musikstück wird generiert und abgespielt. Bei erneutem Drücken auf diesen Knopf pausiert das Musikstück.

Nachdem die Musik generiert wurde und anfängt zu spielen, erscheinen wie in Abbildung 3 [f] die Spezifikationen der Musik. Hinter *Key* steht die Tonart des Stücks (*C Major* für C-Dur und *C Minor* für C-Moll). *Highest Emotion* und *Second Highest Emotion* geben jeweils die dominanteste Emotion und die zweitdominanteste Emotion des Textes an. Als letztes wird bei *Tempo* das Tempo des Textes in Schlägen pro Minute (bpm) angegeben.

Falls der Nutzer nun die generierte Musik speichern möchte, kann er auf *Save* drücken. Der Knopf bei [g] speichert das Stück als Midi-Datei an der selben Stelle wie die angegebene Text-Datei ab. Als letztes ist noch der *Restart*-Knopf

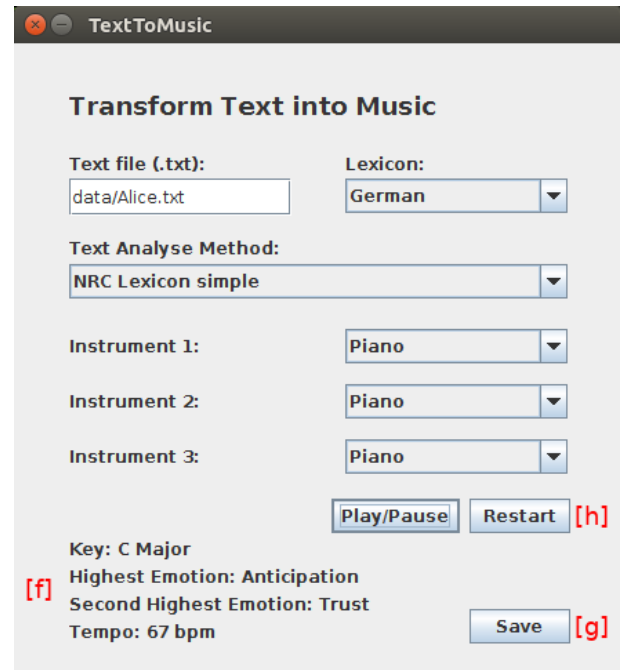


Abbildung 3: GBO nach der Musikgenerierung

bei [h] zu finden. Mit ihm kann man unter Ubuntu das Programm neu starten, wenn man die Jar-Datei von der Konsole aus gestartet hat. Diese Option ist leider nötig, da ein *JFugue-Player* nicht zum zweimaligen Abspielen von Musik mit mehreren Melodien genutzt werden kann. Darauf wird in Abschnitt 6 genauer eingegangen.

Sollten Fehler auftreten, zum Beispiel gibt der Nutzer keine oder eine nicht existierende Datei an, wird bei [f] eine Fehlermeldung anstatt der Spezifikationen ausgegeben und der Nutzer wird zum Neustarten aufgefordert.

## 6 ERGEBNISSE UND EVALUIERUNG

Die unterschiedlichen Ergebnisse haben wir auf verschiedene Dinge untersucht, auch unter der Betrachtung der einzelnen Schritte bei der Generierung.

### 6.1 Musik

Wir haben mehrere Musikstücke aus bekannten Texten, zum Beispiel *Alice im Wunderland* und *Faust*, generiert und sie untersucht.<sup>2</sup>

#### 6.1.1 Tonart

Unser Programm fängt die generelle Stimmung des Textes ein und erkennt, ob der Text einen positiven oder negativen Ton anschlägt. Dies äußert sich im Bestimmen des Tongeschlechts. Ist der Text eher fröhlich und beschwingt geschrieben, wie zum Beispiel *Alice im Wunderland*, so wird der Musik das Tongeschlecht Dur zugewiesen. Dur hat einen hellen, klaren Charakter [4]. Das hat zur Folge, dass das Stück auch eher heiter und positiv klingt.

Demgegenüber wird aus Texten mit bedrückenden und dü-

<sup>2</sup>Hörbeispiele sind auf der CD im Ordner *Hörbeispiele* zu finden.

steren Themen, wie etwa dem alten Märchen *Schneewittchen*, eine Musik in Moll erzeugt. Dieses Tongeschlecht wirkt dunkel und weich [4]. Daraus resultieren betrubte, traurige und negative Musikstücke. So ist es uns möglich von Anfang an eine Musik zu generieren, die die Stimmung auslöst, die aus dem Text hervorgeht.

In der Tabelle 1 haben wir eine Auswahl an Texten in Musik überführt und deren Eigenschaften aufgelistet. Dort sieht man die oben beschriebene Gegebenheit, dass positiven Werken Dur und negativen Texten Moll zugeordnet wird.

Jedoch werden nicht allen Texten das zu erwarteten Tongeschlecht zugeordnet. So wird *Heart of Darkness* Dur zugeordnet, obwohl der Text ein finsternes Motiv trägt. Davis schreibt hingegen in ihrer Arbeit, dass *TransProse* diesem Werk Moll zuordnet. Warum das bei uns nicht der Fall ist, ist uns nicht ganz klar. Es ist möglich, dass sich das Ausgangsmaterial unterscheidet und wir verschiedene Versionen des Stückes benutzt haben. In unserem Beispiel überwiegen im Text die positiven Wörter die negativen, weshalb Dur gewählt wird.

Tabelle 1: Vergleich von Texten bzgl. Tempo, Oktave der Hauptmelodie und Tonart

Titel	Sprache	Tempo (bpm)	Oktave	Tonart
Alice im Wunderland	Deutsch	67	5	C-Dur
Alice's Adventures in Wonderland	Englisch	55	6	C-Dur
Das tapfere Schneiderlein	Deutsch	63	6	C-Moll
Faust Teil 1	Deutsch	70	5	C-Dur
Schneewittchen	Deutsch	98	5	C-Moll
Heart of Darkness	Englisch	78	5	C-Dur

### 6.1.2 Tempo

Texte mit vielen aktiven Emotionswörter (Wut und Freude) wie *Schneewittchen* generieren temporeiche Stücke. Das von *Schneewittchen* generierte Stück besitzt 98 Schläge pro Minute. Andererseits erzeugen Werke wie das englische *Alice's Adventures in Wonderland* mit mehr passiven (Trauer) als aktiven Gefühlswörtern eher langsame Musikstücke. *Alice's Adventures in Wonderland* hat 55 Schläge pro Minute.

Im Gegensatz zu den Beispielen aus Davis Arbeit sind

unsere Stücke relativ langsam. Ihr *Alice in Wonderland* hat ein Tempo von 150 Schlägen pro Minute und *Heart of Darkness* 122 Schläge pro Minute, wo hingegen unser erzeugtes Stück zu *Heart of Darkness* nur 78 Schläge pro Minute aufweist. Ihre Stücke sind generell schneller als unsere. Im Hörvergleich ist der Unterschied allerdings geringer, siehe Abschnitt 6.1.6.

Wir wissen nicht, welche Textvarianten der jeweiligen Geschichten sie genommen hat. Ein unterschiedlicher Ausgangstext könnte die Unterschiede erklären. Ansonsten benutzen wir die gleiche Formel wie sie zur Tempoberechnung siehe Abschnitt 4.3.4.

### 6.1.3 Emotionen

Die beiden Emotionen, die am häufigsten vorkamen, sind Vertrauen und Erwartung (Tabelle 2). Das hat uns, besonders im Vergleich mit den Ergebnissen aus Davis Arbeit, verwundert. Zum Beispiel sind bei *Heart of Darkness* die dominierenden Emotionen Vertrauen und Angst, in Davis Arbeit sind jedoch Angst und Trauer die dominanten Emotionen. Woher dieser Unterschied kommt, ist uns nicht ganz klar, es muss an der unterschiedlichen Textanalyse liegen. Obwohl wir das gleiche Lexikon für die Analyse benutzt haben, sind die Differenzen in den Ergebnissen deutlich zu sehen. Dadurch unterscheiden sich die Stücke auch in ihrer Tonart. Davis hat C-Moll berechnet und wir C-Dur (Tabelle 1). Dadurch entsteht auch ein deutlicher Klangunterschied zwischen den Ergebnissen.

### 6.1.4 Oktaven

Die meisten unserer Stücke sind in den Oktaven 5 und 6. Es gab auch während unserer Arbeit nur selten die Oktave 4 als Oktave der Hauptmelodie. Das ist in der Arbeit von Davis anders, bei *Heart of Darkness* ist die Oktave 4, bei uns, wie in Tabelle 1 zu sehen, 5. Auch bei *Alice's Adventures in Wonderland* ist unsere berechnete Oktave 6 und bei Davis nur 5.

Allerdings ist hier anzumerken, dass wir, wie in Abschnitt 4.3.2 erwähnt, das Freude/Trauer-Verhältniss ohne richtige minimal und maximal Werte berechnen. Davis hat diese in ihrer Arbeit nicht angegeben. Allerdings unterscheiden sich unsere generellen Werte, die im Bereich zwischen 0.6 und 1.0 liegen deutlich von Davis Werten, die zwischen -0.008 und 0.008 liegen. Der Grund hier für könnte sein, dass Davis die Werte in Verhältnis zur gesamten Wörterzahl des Textes setzt, wir jedoch mit dem absoluten Verhältnis rechnen.

### 6.1.5 Textanalysemethoden

Die verbesserten Textanalysemethoden, die wir in Abschnitt 4.2 behandeln, haben leider nur einen kleinen, kaum merklichen Einfluss auf die Ergebnisse gehabt. In der deutschen Version von *Alice im Wunderland* (Tabelle 3) hat sich bei einer gesamten Wörteranzahl von 29871 der absolute Unterschied in einer Differenz von nur zwei Wörtern geäußert.

Die Anzahl der veränderten Stimmungen war nie ausschlaggebend, um die Tonart des Stückes zu ändern oder Ähnliches. Auch die Betrachtung von mehrteiligen Wörtern

Tabelle 2: Vergleich von Texten bzgl. der Emotionen

Titel	Sprache	Emotion 1	Emotion 2	Freude/ Trauer
Alice im Wunderland [2]	Deutsch	Erwartung	Vertrauen	0.927
Alice's Adventures in Wonderland [3]	Englisch	Vertrauen	Erwartung	1.0
Das tapfere Schneiderlein [13]	Deutsch	Angst	Wut	1.0
Faust Teil 1 [12]	Deutsch	Vertrauen	Erwartung	0.888
Schneewittchen [14]	Deutsch	Trauer	Erwartung	0.733
Heart of Darkness [5]	Englisch	Vertrauen	Angst	0.692

aus dem Lexikon war nur geringfügig erfolgreich und die hinzugekommenen Wörter sind in Relation zur Masse an Wörtern, die schon mit der simplen Methode abgedeckt werden, zu gering. Dementsprechend hatten die unterschiedlichen Analysemethoden keinen nennenswerten Einfluss auf das Musikstück. Im *Hörbeispiele-Ordner* ist der Roman *Heart of Darkness* mit der Endung *simple* für die primitive Textanalysemethode zu finden und mit der Endung *Advanced with negation* für die erweiterte Textanalysemethode mit Negation. Das Beispiel zeigt keine hörbaren Veränderungen.

#### 6.1.6 Hörunterschiede

Für den Vergleich mit *TransProse* [7] stehen die Stücke *Heart of Darkness* und *Alice im Wunderland* (deutsche und englische Version) zur Verfügung. *Heart of Darkness* klingt bei uns nicht so düster und melancholisch wie bei Davis, allerdings sind hier auch die Tonarten verschieden. Was allerdings sehr interessant ist, ist dass Davis ein Tempo von 122 bpm angibt und wir nur 78 bpm, aber unsere Version deutlich aktiver und schneller klingt.

Bei *Alice in Wonderland* ist der Unterschied nicht so deutlich. Die Stücke klingen unterschiedlich aber sind viel eher vergleichbar. Beide sind vom subjektiven Empfinden aus gleich schnell, auch wenn die Tempopangabe eine Differenz von 98 bpm aufweist. Auch die deutsche Version folgt einem

Tabelle 3: Vergleich der Stimmungen in *Alice im Wunderland*

Analyse-methode	Positive Wörter	Negative Wörter	Emotionale Wörter Gesamt
primitiv	355	330	837
mehrere Wörter	355	330	837
mehrere Wörter und Negation	353	332	837

ähnlichen Klangmuster, jedoch hört sie sich etwas schneller an. Allerdings wissen wir nicht, welche Textversionen sie für ihre Musikstücke genutzt hat und uns fehlen andere Details, die nicht in ihrer Arbeit veröffentlicht wurden.

## 6.2 JFugue

Die von uns genutzte Java-Programmierschnittstelle *JFugue* bietet viele Möglichkeiten, verschiedene Instrumente zu nutzen und es ist einfach, sich ein generiertes Musikstück anzuhören. Jedoch sind manche der Instrumente subjektiv nicht wohlklingend und uns ist ein deutlicher Unterschied zwischen dem Klang unter Windows und Ubuntu aufgefallen. Unter Windows klingt die Musik wesentlich angenehmer, während bei Ubuntu gerade die höheren Töne sehr intensiv klingen.

Ein weiteres Problem ist beim Programmieren unserer Benutzeroberfläche aufgetaucht. Ein *Player* von *JFugue* lässt sich nicht zwei Mal mit verschiedenen Texten verwenden. Beim zweiten Verwenden spielt er nur noch die erste der drei Melodien ab. Sämtliche Versuche dieses Problem mit mehr oder weniger kreativen Ansätzen zu umgehen, schlugen leider fehl. Für jeden neuen Durchgang einen neuen Player einzufügen, brachte leider auch nicht den gewünschten Erfolg. Das ist der Grund warum unsere Benutzeroberfläche einen Neustarten-Knopf hat.

## 6.3 Textanalyse

Unsere Textanalyse erfasst viele emotionale Wörter und erkennt die Stimmung des Textes. Jedoch ist sie nicht genau genug. Das Wörterbuch, das wir benutzen, ist teilweise leer, besitzt Dopplungen und Zeilen, die nur Nullen beinhalten, also Wörter, die keine Emotion haben. Das liegt größtenteils daran, dass das Wörterbuch in über zwanzig Sprachen übersetzt und dafür nur der Übersetzer von Google genutzt wurde.

Es müsste überarbeitet werden, um bestmöglichst genutzt werden zu können. Dazu zählt die Prüfung und Ergänzung aller Übersetzungen vom englischen Lexikon ins Deutsche, die Entfernung der Zeilen, die nur Nullen enthalten und das Hinzufügen von weiteren emotionalen Wörtern.

Sprachliche Mittel wie Ironie und Sarkasmus werden nicht erkannt, obwohl sie einen großen Einfluss auf die Aussage und die Atmosphäre des Textes haben. Abschwächung und



Verstärkung von Ausdrücken werden im jetzigen Zustand ebenfalls nicht entdeckt und verändern daher auch nicht die Musik.

Wir haben uns nach Alternativen zum *NRC Word-Emotion Association Lexicon* [17] umgesehen, aber keine für uns passende gefunden. Folgende boten uns keinen Mehrwert und wurden daher nicht genutzt.

*SentiWordNet* [9] ist eine lexikalische Ressource für die Meinungsanalyse und teilt Wörter nur in Positivität, Negativität und Objektivität ein.

*Affective norms for English words (ANEW)* [1] stellt emotionale Bewertungen für eine Vielzahl von englischen Wörtern bereit.

*Linguistic Inquiry and Word Count (LIWC)* [18] liest einen Text ein und zählt den Prozentsatz von Wörtern, die eine bestimmte Emotion widerspiegeln. Diese Quelle schien gut geeignet zu sein, ist jedoch nur käuflich erwerblich.

Alle drei Alternativen sind zudem nur in Englisch verfügbar und sind somit für unsere deutschen Texte ungeeignet.

Um jedoch die Textanalyse wirklich zu verbessern, müsste man sich genauer mit der Theorie von Emotionen in der Sprache beschäftigen und auch mit der Theorie von Wahrnehmung von Emotionen in der Musik. Der nächste Schritt wäre dann die Parallelen zwischen diesen Theorien zu ziehen und sie zu verknüpfen.

#### 6.4 Grafische Benutzeroberfläche

Mit unserem präsentierten Programm *Text to Music* ist die Eingabe von Text über die Grafische Benutzeroberfläche, die Erzeugung von Musik auf Basis unserer Textanalyse und das Abspielen des Ergebnisses möglich. Jedoch wurden noch nicht alle Systemkomponenten und die Benutzerinteraktion systematisch getestet. Dadurch könnten sich Fehler unbemerkt eingeschlichen haben. Für eine Weiterführung des Projekts wäre es gut über automatische Tests und Nutzerstudien nachzudenken.

### 7 ZUSAMMENFASSUNG UND AUSBLICK

Aus dieser Arbeit ist ein Programm hervorgegangen, welches auf Grundlage der Emotionen in einem Text ein Musikstück generiert. Diese Musik spiegelt die emotionalen Höhen und Tiefen des Stückes wieder, klingt aber relativ simpel und bezieht keine fortgeschrittenen musikalischen Mittel wie eine sich verändernde Dynamik, wiederkehrende Motive oder Dur-Moll-Wechsel ein. Es bietet sich noch viel Freiraum für Verbesserungen und Weiterentwicklung:

Die Textanalyse kann verbessert werden, indem man die Einträge des *NRC Word-Emotion Association Lexicon* [17] überprüft. Zum einen müsste man die deutschen Übersetzungen kontrollieren, gegebenenfalls berichtigen, leere Zeilen vom Englischen ins Deutsche übersetzen und Dopplungen entfernen. Die Emotionen, die jedem Wort zugeordnet sind, müssten ebenfalls verifiziert werden. Weitere emotionale Wörter können dem Wörterbuch hinzugefügt werden. Ausdrücke ohne verknüpfte Stimmung oder Gefühl können entfernt werden, um die Performanz beim Abgleich des Textes mit dem Wörterbuch zu erhöhen.

Einige Deklinationen von Substantiven, Adjektiven, Pronomen und Numeralen fehlen ebenfalls, sodass nicht alle Varianten eines Wortes erkannt werden, obwohl der ungebeugte Ausdruck im Wörterbuch steht. Diese kann man ergänzen, um noch mehr Wörter zu erkennen.

Zur Zeit wird nur das momentan untersuchte Wort, die drei nächsten und eine vorangegangene Verneinung mit Hilfe des Lexikons betrachtet. Diese vorausschauende Technik sollte man so verbessern, dass Höhepunkte und Stimmungswechsel vorzeitig erkannt werden, um in der Musik angekündigt werden zu können. Auf diese Weise kann man Spannung schon vor dem eigentlichen Ereignis erzeugen.

Die Analyse könnte so erweitert werden, dass Abschwächung und Verstärkung von Ausdrücken sowie sprachliche Mittel wie Ironie, Sarkasmus, Metapher, Symbole, Klimax und Euphemismus erfasst werden. Diese verändern die Aussage und die Stimmung des Textes, was sich auch auf die Musik auswirken sollte. Hinzukommend werden nur acht Emotionen abgebildet, obwohl unserer Meinung nach mehr betrachtet werden sollten. So klang für uns noch kein generiertes Musikstück gruselig, mutig oder kraftvoll, wenn gleich der Text dementsprechend wirkte. Man könnte auch in Betracht ziehen, mehrere unterschiedliche Wörterbücher bei der Textanalyse zu benutzen und somit ein besseres Ergebnis zu erzielen.

Jede Melodie hat in unserem Programm das gleiche Tongeschlecht. Man könnte ausprobieren wie sich die Musik verändert, wenn jede Melodie ein anderes Tongeschlecht oder eine andere Tonart hätte. Das Tongeschlecht oder die Tonart könnten sich ebenso innerhalb eines Musikstückes ändern.

Die Oktaven der drei Melodien bedienen sich gerade prosvisorisch festgelegter Werte für  $FT_{min}$  und  $FT_{max}$  (siehe Abschnitt 4.3.2). Diese könnte man durch eine Analyse mehrerer Romanen genauer bestimmen.

Es wäre interessant ein Stereosignal zu erzeugen und so zum Beispiel  $M_0$  in der Mitte,  $M_{e1}$  links und  $M_{e2}$  rechts zu hören. Ebenso spannend wäre eine sich verändernde Dynamik im Verlauf des Stückes. Das Tempo würde dann nicht für das gesamte Tonstück gelten, sondern zum Beispiel für jeden Unterabschnitt. Ein oder mehrere wiederkehrende Motive würden das Ergebnis verstärkt kohäsiv und einheitlich klingen lassen.

Um die generierte Musik für einen audiovisuellen E-Book-Reader zu nutzen, kann man die Anzahl der Abschnitte erhöhen, um längere Stücke zu erzeugen, die lang genug für das Buch sind. Mit einer Blickerfassungsmethode könnte man herausfinden, welchen Abschnitt des Buchs der Leser gerade liest und die Musik dahingehend anpassen. So würde sich die Musik auf die Lesegeschwindigkeit jeden Lesers einstellen können.

Hier sind drei JFugue Alternativen, die wir noch nicht testen konnten:

- Jmusic [19]
- abc4j [15]
- JASS [8]

Diese können ausprobiert werden, um herauszufinden, wel-

chen Mehrwert sie im Gegensatz zu JFugue besitzen. Es wäre möglich, dass sie JFugue ersetzen oder ergänzen.

Wenn man unserer Programm und unsere Grafische Benutzeroberfläche weiterentwickeln möchte, sollte man sie weiter testen und die Fehlerabfangung ausbauen.

Es gibt also noch viele Möglichkeiten um *Text to Audio* zu verbessern und an mögliche Anwendungsbereiche anzupassen.

## LITERATUR

- [1] Margaret M. Bradley und Peter J. Lang. *Affective norms for English words (ANEW): Stimuli, instruction manual and affective ratings*. Techn. Ber. University of Florida, 1999. URL: <http://csea.phhp.ufl.edu/media.html#bottommedia> (besucht am 01.07.2017).
- [2] Lewis Carroll. *Alice's Abenteuer im Wunderland*. URL: <http://www.gutenberg.org/ebooks/19778> (besucht am 18.06.2017).
- [3] Lewis Carroll. *Alice's Adventures in Wonderland*. URL: <http://www.gutenberg.org/ebooks/11> (besucht am 18.06.2017).
- [4] C-Dur. 2017. URL: <https://de.wikipedia.org/wiki/C-Dur> (besucht am 27.07.2017).
- [5] Joseph Conrad. *Heart of Darkness*. URL: <https://www.gutenberg.org/files/219/219-h/219-h.htm> (besucht am 18.06.2017).
- [6] Hannah Davis. *TransProse - The Music*. 2014. URL: <http://www.musicfromtext.com/novels.html> (besucht am 10.07.2017).
- [7] Hannah Davis und Saif M. Mohammad. "Generating Music From Literature". In: *In Proceedings of the EACL Workshop on Computational Linguistics for Literature, April 2014, Gothenburg, Sweden* (2014).
- [8] Kees van den Doel. *Real-time audio synthesis using JASS*. URL: <http://www.cs.ubc.ca/~kvdoel/jass> (besucht am 01.07.2017).
- [9] Andrea Esuli und Fabrizio Sebastian. *SentiWordNet*. URL: <http://sentiwordnet.isti.cnr.it> (besucht am 01.07.2017).
- [10] Manuel Fischer. *Wahrnehmung von Musik*. 2008. URL: [https://www.hdm-stuttgart.de/~curdt/Wahrnehmung\\_von\\_Musik.pdf](https://www.hdm-stuttgart.de/~curdt/Wahrnehmung_von_Musik.pdf) (besucht am 27.07.2017).
- [11] *Frequenz der gleichstufigen Stimmung*. 2017. URL: [https://de.wikipedia.org/wiki/Frequenzen\\_der\\_gleichstufigen\\_Stimmung](https://de.wikipedia.org/wiki/Frequenzen_der_gleichstufigen_Stimmung) (besucht am 27.07.2017).
- [12] Johann Wolfgang von Goethe. *Faust: Eine Tragödie*. URL: <http://gutenberg.spiegel.de/buch/-3664/1> (besucht am 18.06.2017).
- [13] Brüder Grimm. *Das tapfere Schneiderlein*. URL: <http://gutenberg.spiegel.de/buch/-6248/169> (besucht am 18.06.2017).
- [14] Brüder Grimm. *Schneewittchen*. URL: <http://gutenberg.spiegel.de/buch/-6248/150> (besucht am 18.06.2017).
- [15] Lionel Guéganton und Sylvain Machefert. *abc4j*. URL: <https://code.google.com/archive/p/abc4j/> (besucht am 01.07.2017).
- [16] David Koelle. *JFugue*. URL: <http://www.jfugue.org/> (besucht am 01.06.2017).
- [17] Saif M. Mohammad und Peter Turney. *NRC Word-Emotion Association Lexicon*. 2013. URL: <http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm> (besucht am 01.06.2017).
- [18] James W. Pennebaker. *Linguistic Inquiry and Word Count (LIWC)*. URL: <http://liwc.wpengine.com/> (besucht am 01.07.2017).
- [19] Andrew Sorensen und Andrew Brown. *jMusic - Music composition in Java*. URL: <http://explodingart.com/jmusic> (besucht am 01.07.2017).