Knjigica – Eine Libretto-Suchmaschine

Lukas Neef
Martin Luther University Halle-Wittenberg
Halle (Saale), Germany
lukas.neef@student.uni-halle.de

Jan Heinrich Reimer
Martin Luther University Halle-Wittenberg
Halle (Saale), Germany
jan.reimer@student.uni-halle.de

ZUSAMMENFASSUNG

Die Recherche und Suche von Libretti – Textvorlagen einer Oper – ist umständlich, da keine umfassende Suchmaschine dazu zur Verfügung steht. Allgemeine Websuchmaschinen können die vielen Details nur unvollständig erfassen, sodass sie als Suchmaschine für Libretto schlecht geeignet sind. Wir entwerfen eine umfassende Suchmaschine für Libretti, die alle im Libretto enthaltenen Informationen abbildet, und dadurch sowohl beim Besuch einer Oper, als auch in der Literaturwissenschaft der Librettologie, Bedeutung findet.

KEYWORDS

search engine, libretto, opera

1 EINLEITUNG

Die Textvorlage einer Oper, das Libretto, ist eine wichtige Quelle, sowohl für Besucher einer Oper, sowie für die Literaturwissenschaft. Für die schnelle Recherche von Libretti steht bisher keine Suchmaschine zur Verfügung. Wir entwerfen daher eine Suchmaschine, mit der sich Libretti schnell anhand ihres Textes oder ihrer Eigenschaften finden lassen.

Durch diese neue Suchmaschine, sind zahlreiche Verbesserungen bei der Arbeit mit Libretto zu erwarten: So ist es beispielsweise hilfreich, wenn der Text aus anderen Sprachen übersetzt wird, um ein besseres Verständnis für die Inhalte der Handlung zu bekommen. Auch bekommen Besucher die Möglichkeit, mit Hilfe des Textes der Handlung besser folgen zu können und nicht nur dem Geschehen auf der Bühne zuzuschauen. Dies ist auch der Fall, wenn während der Aufführung ein bestimmter Teil nicht verstanden wurde, weil die Präsentation oder Aussprache undeutlich war, oder durch starke Betonung unverständlich wurde.

Mit Hilfe der Suche im Libretto können des Weiteren auch zusätzliche Informationen zum Autor oder dem Jahr der Erstaufführung recherchiert werden, was in der Vor- und Nachbereitung eines Besuches einer Opernaufführung von Bedeutung ist.

Die Möglichkeiten der Benutzung dieser Suchmaschine sind also vielfältig und können sogar der Forschung in der Librettologie zugutekommen.

2 DATENQUELLEN

Im ersten Schritt der Umsetzung ist eine umfassende Recherche nach Libretto und deren Texten im Internet erfolgt. Als Ergebnis der Suche wurden die in Tabelle 1 aufgeführten Datenquellen ausgemacht.

Zeitgeschichtlich wurde der überwiegene Teil der in den Corpora enthaltenen Libretti im Zeitraum vom 17. bis 20. Jahrhundert uraufgeführt. Die meisten der Corpora enthalten Libretti verschiedener Sprachen, überwiegend jedoch italienische, französische, englische, deutsche oder russische. Auch werden bei einigen Corpora Übersetzungen zur Verfügung gestellt, so werden beispielsweise die Libretti von *Opera Glass* durch 389 Übersetzungen ergänzt. Bei *Kareol* sogar die Übersetzung dem Originaltext gegenübergestellt.

Corpus	Anzahl Libretto
Kareol	385
Opera Lib	434
Opera Folio	512
Opera Glass	170
Aria	177
Collection Ulric Voyer	220

Tabelle 1: Libretto-Corpora

3 STRUKTUR

3.1 Literarische Struktur von Libretto

Im geschichtlichen Verlauf der Zeit haben sich Libretti stetig weiterentwickelt. Daher ist es schwierig, für ein Libretto eine feste Struktur festzulegen. Allgemein sin jedoch einige Bestandteile in jedem Libretto vorhanden. So wird anfangs der Titel und die Autoren genannt. Hier werden auch oftmals weitere Informationen, wie Daten zur Premiere, sowie weitere Notizen oder Widmungen des Autors genannt. Nachfolgend werden die im Libretto vorkommenden Rollen mit Namen, teils mit Beschreibung, sowie mit der vorgesehenen Stimmlage, aufgeführt. Anschließend folgt die Handlung des Libretto. So werden die Überschriften, sowie Unterüberschriften genannt. Textpassagen werden, meist mit dem Namen der vortragenden Rolle versehen, einfach aufgelistet. Zwischen Textpassagen stehen gelegentlich Regieanweisungen zur Betonung, Informationen zum Bühnenbild, sowie weitere Instruktionen. Mit der Handlung endet das Libretto.

3.2 Hilfsdatenstruktur

Aus der allgemeinen Struktur von Librettos leiten wir für die Entwicklung der Suchmaschine ein standardisiertes und maschinell lesbares Format ab. Dieses dient dem effizienten Speichern der gerawlten Rohdaten und der Vorbereitung sowie Standardisierung für die spätere erfolgende Indizierung durch die Suchmaschine.

Neben einer Datenstruktur, die ein vollständiges Libretto abbildet, entwerfen wir Klassen, die einzelne Teile der Struktur speichern. In einer Datenstruktur werden alle Informationen zum Autor gespeichert, in einer zweiten Informationen über Widmungen und Notizen des Autors, in einer dritten Informationen zu einer Rolle bzw. einem Darsteller, und in einer letzten werden alle die Handlung betreffenden Informationen abgespeichert. Dabei wird diese letzte in sich in drei Datenstrukturen unterteilt, für Textpassagen, Instruktionen, sowie für Überschriften.

4 TECHNISCHE IMPLEMENTIERUNG

Die Implementierung der Libretto-Suchmaschine unterteilt sich in mehrere Bausteine. Während der Crawler vor allem auf Linux-Bash-Programmen aufbaut, wird in den anderen Bausteinen, so

auch im Parser, Indexer und im Benutzerprogramm auf objektorientierte Programmierung mittels Java/Kotlin gesetzt. Als Framework wird das praxiserprobte *Elasticsearch*-Framework verwendet. Im Folgenden werden die einzelnen Komponenten der Suchmaschine näher erklärt.

4.1 Crawler

Der von uns entwickelte Crawler hat die Aufgabe, die Corpora in 2 aus dem Internet herunterzuladen, und für die weitere Verarbeitung zu speichern.

Zuerst werden dazu die URLs aller Libretti aus dem Register eines jeden Corpus extrahiert und als Liste zwischengespeichert. Um Probleme mit relativen URLs auszuschließen, werden diese auf die absolute Adresse erweitert.

Danach lädt ein weiteres Skript systematisch die HTML-Dokumente aller erfassten URLs herunter und speichert diese zunächst unverändert ab. Hierbei werden auch verlinkte, verschachtelte Seiten zunächst mit übernommen.

Beide Vorgänge wurden mit Hilfe von Bash-Skripten umgesetzt, die die Linux-Funktionen `wget`, `curl`, sowie die `html-xml-utils` verwenden.

4.2 Parser

Die Inhalte der Internetseite liegen nach dem Crawlen als HTML-Quelltext vor. Damit in der Folge nach enthaltenen Informationen gesucht werden kann, werden die Inhalte mittels Java bzw. Kotlin in ein Objekt geschrieben. Hierbei wird die Bibliothek *JSoup* verwendet, welche die Extraktion und Bearbeitung von HTML-Inhalten ermöglicht. Entsprechend werden die gecrawlten Dokumente je nach Quelle so verallgemeinert, dass die Hilfsdatenstruktur anhand der Inhalte spezifischer HTML-Elemente erzeugt wird.

Zur besseren Speicherung der so erhaltenen Objekte für die nächsten Bearbeitungsschritte, werden zusätzlich ein Parser und ein Formatter entwickelt, mit dem die in der Hilfsdatenstruktur enthaltenen Informationen in das JSON-Format umgewandelt, sowie aus diesem wieder geparst werden können.

4.3 Suchmaschine

Als Suchmaschine verwenden wir die Open-Source-Software *Elasticsearch*[1]. Diese beruht auf dem *Lucene*-Framework[2] von Apache und realisiert die Kommunikation mit Klienten über ein RESTful-Webinterface.

4.4 Indizieren

Da das *Elasticsearch*-Framework keine verschachtelten Arrays aus Objekten handhaben kann, werden stattdessen angepasste Datenstrukturen verwendet, die durch Verflachen aus der umfassenden Datenstruktur entsteht. So wird beispielsweise in einem Handlungs-Objekt noch der Titel der Oper mit gespeichert, und die Information zur aktuellen Überschrift.

Auch für diese Strukturen werden Formatter in das JSON-Format erstellt.

Die aus den zwischengespeicherten JSON-Daten geparsten Dokumente werden nun aufgespalten, sodass aus einem Libretto eine Vielzahl von Teilobjekten entsteht. Diese werden dann jeweils nach Typ in einzelne Indices von *Elasticsearch* über das REST-Interface des Clusters indiziert.

[1] https://www.elastic.co/products/elasticsearch
[2] http://lucene.apache.org/

4.5 Anfrageverarbeitung

Im Anschluss liegen die Dokumente aufgelistet in Elasticsearch vor und können über den Index von der Suchmaschine gefunden werden.

Eine Suchanfrage wird, wie die Inhalte zuvor, im JSON-Format über das REST-Interface an das Elasticsearch-Cluster gesendet. Die Anfrage kann verschiedene Parameter enthalten, welche die Resultate mit Filtern versehen und Spezifikationen wie eine bestimmte Ordnung der Inhalte und Eingrenzung auf bestimmte Stichworte erlauben.

Für die Libretto-Suchmaschine werden mehrere Indices, nämlich die der verschiedenen Dokumenttypen, gleichzeitig angefragt. Dabei werden die Ergebnisse je nach Typ unterschiedlich gewichtet, sodass bei Konflikten beispielsweise Resultate, den Titel der Oper betreffend, stärker gewichtet werden können, als solche, die in einer Textpassage vorkommen.

Durch die Kombination aller Indices kann über das selbe Suchfeld nach Titel, Autor, Rollennamen, Regieanweisungen, Akt, Szene, Text, Widmung und Premiere gesucht werden.

5 EVALUATION

Die Evaluation wurde anhand des Vorbildes der „Text Retrieval Conference“ (TREC) durchgeführt. In diesem Rahmen wird die Suchmaschine anhand von 25 Topics auf ihre Funktionalität untersucht.

Ein Topic besteht etwa aus einer zentralen Suchanfrage, aus der weitere Unteranfragen resultieren können. Um wichtige Aspekte der Suchmaschine abzudecken, werden die Topics nach den Typen „Autor“, „Struktur“ und „Handlung“ eingeordnet.

Die Relevanz der Suchanfragen werden anhand einer Skala von 0 bis 1 erfasst: Hierbei steht ein Wert von 0 für ein Suchergebnis, welches überhaupt nicht relevant oder nicht hilfreich ist, ein Wert von 1 für ein Ergebnis, das das Informationsbedürfnis des Benutzers aus der Suchanfrage in vollem Umfang erfüllt.

5.1 Durchführung

Einem unabhängigen Dritten wurden die Topics präsentiert, und dieser konnte selbstständig über die Suchbox eine Anfrage – oder die vorgeschlagene Beispielanfrage – eingeben.

Nach Durchführung einer jeden Suchanfrage sollte die Testperson die Relevanz anhand der oben beschriebenen Skala einschätzen und zusammen mit einem optionalen Kommentar erfassen, welcher Aufschluss über die Einordnung der vorgegebenen Skala gibt.

Die Resultate wurden zusätzlich mit der aktuellen Version der Suchmaschine annotiert, sodass ein Vergleich verschiedener Versionen der Suchmaschine möglich ist.

5.2 Ergebnis

Im Mittel ergibt sich über alle Topics eine Relevanz von 0,344, was eine verhältnismäßig geringe Erfüllung der Suchanfragen bedeutet. Jedoch fällt die Relevanz bei den unterschiedlichen Topic-Typen verschieden aus. Hierbei sind die Mittelwerte für den Typen „Autor“ 0,51, für „Struktur“ 0 und für „Plot“ 0,4. Die Topics „Autor“ und „Plot“ liefern also beide bessere Suchergebnisse, was durch den derzeitigen Stand der Umsetzung zu begründen ist.

Die Suchmaschine ist in der Lage, nach spezifischen Informationen zu suchen, berücksichtigt jedoch noch keine Operatoren oder Suchanfragen, welche eine Umwandlung der Inhalte anhand einer bestimmten Fragestellung enthält. Dies führt auch dazu,

dass die Intention einer Suchanfrage nicht vollständig korrekt erkannt wird und die eigentlich richtig eingeordneten Suchergebnisse mit falschen Parametern ausgegeben werden.

6 ZUSAMMENFASSUNG UND AUSBLICK

Im derzeitigen Umsetzungsstand der Suchmaschine können innerhalb einer Kommandozeilenanwendung Anfragen gestellt werden, welche die bereits vollständig indizierten Dokumente nach passenden Inhalten durchsucht. Dabei werden als Resultate nur übereinstimmende Inhalte ausgegeben, welche in den Kategorien „Titel“ und „Sprache“ (3), „Autor“ (1,5), „Rollen“ (1,2), „Bemerkungen“ und „Widmungen“ (0,7), „Plot“ (0,15) mit absteigenden Gewichtungen festgelegt sind.

Im Suchprozess wird auf eine Datenbank zugegriffen, welche Dokumente mit einer von uns festgelegten Datenstruktur enthält. Diese wurde im jetzigen Stand lediglich aus einer Datenquelle heraus befüllt.

6.1 Corpus

Obwohl wir bisher nur für den „Opera Lib“-Corpus einen Parser geschrieben haben, lässt sich das entwickelte Framework problemlos auf weitere Corpora erweitern. So planen wir beispielsweise, die im Gliederungspunkt „Datenquellen“ angegebenen Seiten zu crawlen und in die Hilfsdatenstruktur zu transferieren. Dies setzt auch die Zusammenführung mit den bereits vorhandenen Dokumenten voraus, welche als Folge eventuell doppelt und nicht mehr redundanzfrei vorliegen würden. Als Folge dieses Prozesses dürften Libretto teilweise in mehreren Sprachen vorliegen und könnten auch in vollständigerer Form hinterlegt sein.

Eine weitere Erweiterungsmöglichkeit besteht darin, Kunstlieder mit in die Suchmaschine einzubeziehen. Der „Kareol“-Corpus stellt dazu 687 Lieder zur Verfügung.

6.2 Query Understanding

Ein weiteres Ziel in folgenden Versionen soll es sein, die Suchanfragen zu analysieren und somit auch auf eventuelle Spezifizierungen eine angepasste Ausgabe geben zu können. Ein wichtiger Punkt wäre hierbei zunächst das sogenannte „Natural Language Processing“, welches mit Hilfe der gegebenen Quellen Sprachen erkennt, deren Inhalte segmentiert und im Anschluss Funktionen und Bedeutungen der Wörter herausfiltert. Mit diesem Vorgehen würde ein hauptsächlicher Grund für die schlechten Ergebnisse der Evaluation entfallen.