Richtlinie zum „Software-Sprint“

TRANSKRIPT

Schlussbericht

Zuwendungsempfänger:

Matthias Hannich

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S57 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Qualitativ hochwertige Transkripte gesprochener Sprache sind im Moment nur mit hohem finanziellen und personellem Aufwand erzeugbar. Das Projekt TRANSKRIPT vereinheitlicht und vereinfacht den Erzeugungsprozess eines Transkripts, indem es eine Web-Plattform bereitstellt, auf der Menschen ohne Expertenwissen Transkripte erzeugen und korrigieren können. Dieser Prozess wird durch Bereitstellung automatischer Spracherkennung ergänzt.

Dadurch haben unkommerzielle Medienproduzent*innen, Podcaster*innen und andere Initiativen aus der Zivilgesellschaft die Möglichkeit, die Reichweite ihrer aufgezeichneten Sprachaufnahmen (z.B. für Menschen mit Hörbeeinträchtigungen) zu erhöhen und Transkripte verfügbar zu machen.

Das Projekt teilte sich in zwei wesentliche Arbeitspakete: Erstellung eines Editors zum Transkribieren und Korrigieren bzw. Anbindung automatischer Spracherkennungssoftware, die Hypothesen für das Transkript generieren kann. Letztere sollen im Editor korrigierbar sein.

Der Editor-Teil wurde unter Nutzung bzw. Erweiterung des kollaborativen Texteditors Etherpad-Lite realisiert. Für die Anbindung der Spracherkennungssoftware Kaldi wurde kaldi-gstreamer-server genutzt.

Zur Umsetzung des Vorhabens mussten die einzelnen Workflows (Segmentierung der Audiodatei zur Integration in den Editor, manuelles Transkribieren bzw. Import vorhandener Transkripte, Korrektur und Vorgaben für lesbare Verschriftlichungen, Integration automatisch generierter Hypothesen) klar voneinander getrennt werden. Aus den Anforderungen der einzelnen Workflows wurden die notwendigen, zu implementierenden Features ermittelt.

Beitrag des Projektes zu den Zielen der Förderinitative „Software-Sprint“

Zielgruppe sind alle, die Sprachaufnahmen selbst erzeugen oder aufzeichnen. Diese können durch Unterstützer*innen Transkripte erzeugen lassen oder selbst erzeugen und die Reichweite ihrer Themen erhöhen. Tondokumente können dadurch nicht nur angehört, sondern gelesen werden.

Darüber hinaus können Forscher*innen die zugänglich gemachten Audio-Aufnahmen zum Training eigener Modelle nutzen. Neue, in Modellen bisher unberücksichtigte Aufnahmen (z.B. freie-radios.net mit unter by-nc-sa Creative Commons-Lizenz veröffentlichten Beiträgen) können dafür genutzt werden. Diese sind in großem Umfang Spontanaufnahmen – im Gegensatz zu den bereits verfügbaren aus Büchern oder Wikipedia vorgelesenen Datensätzen. Die daraus entstehenden Modelle sind demzufolge besser für die Erkennung spontaner Sprache geeignet.

Ausführliche Darstellung der Ergebnisse

Es wurde ein Plugin für den kollaborativen Editor Etherpad-Lite erstellt, mit dem einzelne Segmente einer Tondatei in einem ersten Schritt transkribiert und in einem zweiten Schritt durch andere Nutzer*innen korrigiert werden können. Das Plugin enthält grundlegende Funktionalität, um Audio-Dateien innerhalb des von Etherpad-Lite bereitgestellten Editors abspielbar zu machen und dabei segmentweise vorzugehen.

Die Software kaldi-gstreamer-server, mit der mittels Kaldi fertig trainierte Modelle genutzt werden können, wurde im Anschluss so eingebunden, dass diese für die einzelnen Segmente Hypothesen generiert. Diese Komponente erscheint in Etherpad-Lite als normaler Autor, weshalb aus Sicht der Nutzer*innen kein wesentlicher Unterschied zwischen den von Menschen erzeugten Transkripten und den durch die Spracherkennungssoftware erzeugten Hypothesen besteht.

Ergänzt wurde der an die Spracherkennungssoftware angebundene Webeditor durch einen prototypischen Crawler, mit dem mehrere Audioaufnahmen segmentiert und in den Editor eingepflegt werden können.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Unkommerzielle Medienproduzent*innen erhalten durch das Tool die Möglichkeit, ohne Kosten automatisch Transkripte ihrer Aufnahmen zu erstellen bzw. diese selbst oder durch Unterstützer*innen transkribieren und korrigieren zu lassen.

Im Rahmen des Projekts wurden vorhandene Aufnahmen indiziert, die als Creative Commons auf freie-radios.net lizenziert veröffentlicht wurden. Daraus ergeben sich neue Rohaufnahmen für die Integration in vorhandene Korpora. Sprachwissenschaftler können daraus neue Modelle trainieren.

Eine Weiterentwicklung der Plattform zur Umsetzung einer Volltextsuche für Audiodateien wäre denkbar. Der Schwerpunkt würde sich dann verlagern von einzelnen Dateien, die Nutzer*innen selbst hochladen bzw. verlinken können, hin zu einzelnen Segmenten, die zum Transkribieren bzw. Korrigieren vorgeschlagen werden.

Der Editor muss im Funktionsumfang erweitert werden, damit Funktionalitäten (Rechtschreibung, Grammatik-Korrektur, Abkürzungen, usw.) abgedeckt werden, die vorhandene (nicht browserbasierte) Anwendungen wie LibreOffice bereits enthalten.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Ursprünglich kannte ich nur oTranscribe als eine Möglichkeit browser-basiert Transkripte zu erstellen, weshalb es naheliegend schien, dieses Tool zu erweitern. Im Zuge anfänglicher Recherchen und durch Interviews mit potenziellen Nutzer*innen bzw. Medienproduzent*innen wurde schnell deutlich, dass es verschiedenste Tools bereits gibt bzw. dass es eine Vielzahl an Wünschen gibt, die durch eine solche Plattform abgebildet werden sollen. oTranscribe ist als privacy-preserving WebApp konzipiert – Dateien verlassen niemals den Browser. Diese Eigenschaft wird durch die kollaborative Plattform prinzipiell verletzt, weil Dateien entweder bereits online veröffentlicht sind oder direkt in die Plattform geladen werden bzw. andere Nutzer*innen die Transkripte einsehen können.

Das wavesurfer.js-Projekt kann hingegen alternativ als Bibliothek genutzt werden und bietet einen ähnlichen Funktionsumfang wie oTranscribe.

Die Planung des gesamten Projektzeitraums in einzelnen Arbeitsschritten war herausfordernd. Aufgabenverteilung im Team hätte den Entwicklungsprozess vereinfacht.

Vor allem die Anbindung externer Spracherkennungssoftware ist komplizierter, als in der Planungsphase erwartet. Durch Nutzung von kaldi-gstreamer-server konnten verschiedene Modelle integriert und deren Hypothesen verwendet werden.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Die Softwarekomponenten (Editor-Plugin, Crawler für Audio-Ressourcen, Schnittstelle zu kaldi-gstreamer-server) und die Serversoftware zum Aufsetzen der Plattform werden unter https://github.com/transkription veröffentlicht. Der Release der Plattform ist bis zum Jahresende 2019 geplant, da insbesondere Bugs im Zusammenhang mit der Korrektur-Funktionalität und die bisher nicht abgeschlossene Integration mehrerer Modelle eine sinnvolle Nutzung blockieren.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Aus ersten Gesprächen bzw. Interviews mit potenziellen Nutzer*innen ergab sich der Bedarf, bereits vorhandene Transkripte integrieren zu können, auch wenn diese fehlerhaft sind oder nur in Form von Sendungsskripten vorliegen. Die Möglichkeit bereits vorhandene, außerhalb der Plattform erzeugte Textdateien mit den Sprachaufnahmen zu verknüpfen ist sinnvoll, um auch Nutzer*innen erreichen zu können, die nicht zwangsweise mit der Plattform transkribieren möchten, diese aber zum Beispiel zur Korrektur von Transkripten nutzen wollen. Der Zeitaufwand für eine prototypische Umsetzung dieser Import-Funktionalität lag bei ca. drei Wochen, jedoch konnten daraus Erkenntnisse in den Meilenstein „Anbindung eines Spracherkenners“ fließen, wodurch dieser entsprechend verkürzt wurde.

Ursprünglich geplant war, auch den Trainingsprozess für neue Modelle abdecken zu können. Die Komplexität dieses Arbeitspakets wurde unterschätzt, so dass unter Berücksichtigung des Zeitrahmens nur die automatische Erweiterung um eigene Wörter bzw. Wortgruppen, die von der Spracherkennungssoftware im Anschluss erkannt werden können, implementiert wurde und nicht die Erweiterung der akustischen Modelle. (Meilenstein 5)

Desweiteren war geplant nicht nur mit einen, sondern mit mehreren Modellen verschiedene Hypothesen zu generieren, so dass Nutzer*innen zwischen den besten auswählen können. Das ist eine Herausforderung unter dem Gesichtspunkt der Übersichtlichkeit innerhalb des Editors, weshalb für den Prototyp vorübergehend nur ein Modell gleichzeitig angebunden, jedoch zwischen verschiedenen Modelle gewählt werden kann. (Meilenstein 4)

Die Arbeitsstunden teilten sich innerhalb des Projektzeitraums im Wesentlichen auf die zwei großen Arbeitspakete Entwicklung des Editors und Integration bzw. Anbindung der Spracherkennungssoftware gleichmäßig auf (Meilensteine 1-3 bzw. 4-6).

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Der Bereich Spracherkennung ist durch eine Vielzahl aktueller Entwicklungen gekennzeichnet, die Lösungsansätze für Teilprobleme liefern oder ganze Workflows im Umgang mit Transkripten abbilden sollen. Es existieren vorhandene Ansätze zur Korrektur automatisch generierter Transkripte bzw. zur Umsetzung von Voting-Verfahren, um das beste Transkript zu ermitteln (z.B. NYPL Transcript Editor oder BBC Transcript Editor).

Auch für das Einpflegen der Transkripte unter Zuhilfenahme unterschiedlicher Editoren gibt es bereits Weblösungen (z.B. OCTRA von der Uni München).

Der Vielzahl an Ansätzen wurde innerhalb des Projekts insofern Rechnung getragen, als dass die einzelnen Bestandteile optional in Form von Komponenten über eine möglichst generische API integrierbar sein sollen. Es soll in Zukunft möglich sein, einzelne Editoren auszutauschen, sowohl in Hinblick auf den Transkriptionsprozess, als auch auf den Korrekturprozess.

Da die vorhandenen Tools meist als Einzelwerkzeuge und nicht als modulare Komponenten entwickelt werden, ist deren (Code-)Komplexität relativ groß, weshalb sie nicht von vorn herein integriert werden konnten. Eine punktuelle Anpassung der vorhandenen Lösungen konnte nicht

erfolgen, weil die Einarbeitungszeit im zur Verfügung stehenden Projektzeitraum einen unverhältnismäßigen Schwerpunkt gesetzt hätte.

Die Zielgruppe vieler Tools ist noch zu sehr auf ein Fachpublikum aus dem linguistischen oder phonetischen Bereich zugeschnitten.