FRAUNHOFER INSTITUT FÜR KERAMISCHE TECHNOLOGIEN UND SYSTEM IKTS

VERBESSERUNG DER DIKTIER-/UNTERTITELFUNKTION FÜR DIE OBERSORBISCHE SPRACHERKENNUNG IN MEHREREN DOMÄNEN

Projektbericht

Ivan Kraljevski
Frank Duckhorn
Constanze Tschöpe
Matthias Wolff*

Fraunhofer Institut für Keramische Technologien und Systeme IKTS
Maria-Reiche-Straße 2, 01109 Dresden

**Brandenburgische Technische Universität Cottbus–Senftenberg, Cottbus*

Inhaltsverzeichnis

1 Einleitung

Die in diesem Projekt erstellten Sprachressourcen ergänzen die bereits vorhandenen Ressourcen und leisten einen wertvollen Beitrag zur Digitalisierung und Erhaltung der obersorbischen Sprache.

Die Erfahrungen, die entwickelten Verfahren und die Software bilden die Grundlage für weitere Anwendungen in der Computerlinguistik und Sprachtechnologie (Sprachsynthese, Dialogmanagement, Verstehen und Erzeugen natürlicher Sprache und maschinelle Übersetzung). Darüber hinaus werden sie die Gewinnung neuer Daten beschleunigen, die Sprachmodelle weiter verbessern und den Einsatz neu entstehender Sprachtechnologien ermöglichen.

Dieses Projekt konzentriert sich auf die weitere Verbesserung und Optimierung der Technologien durch das Sammeln und Organisieren von Sprachdaten.

Es erforscht auch die Einsatzmöglichkeiten des neuesten Stands der Technik (SdT) und untersucht neue und einfallsreiche Sprachanwendungen in den Bereichen Bildung, Büro- und Geschäftsverwaltung, Live-Untertitelung und Transkription von Medien, personalisierte virtuelle Assistenten und Chatbots sowie die Verbesserung der Lebensqualität von Menschen mit Behinderungen.

Die Hauptziele dieses Projekts lassen sich wie folgt zusammenfassen:

- Implementierung neuer Trainingsverfahren für akustische Modelle (monophon und triphon), einschließlich neuer Sprachaufnahmen, Änderungen im Graphem- und Phonem-Inventar und Ausspracheregeln.
- Optimierung des phonetischen Inventars. Weitere Ressourcen für die Erstellung des Lexikons wurden untersucht und integriert.
- Erstellung von anwendungsbezogenen Sprachmodellen auf der Basis der bereits erstellten normalisierten Texte.
- Normalisierung neuer anwendungsspezifischer Texte und deren Integration.
- Verbesserung des Ansatzes zur Verwendung morphologischer Einheiten, unter Verwendung alternativer Ansätze der Computerlinguistik (CL) zur Tokenisierung.
- Überprüfung des Codes für die Anpassung des Erkenners an die "VOSK"-API.
- Integration/Verbesserung/Parametrisierung der Spracherkennung ("VAD").
- Auf der Grundlage der derzeit verfügbaren Technologie und Ressourcen wird eine Strategie für die zukünftige Entwicklung der Spracherkennung für das Obersorbische entwickelt.

2 Bisherige Arbeiten

2.1 HSB-III: Vorbereitung der Spracherkennung für das Obersorbische für eine Diktierfunktion

Die bisherigen Projektziele war die Untersuchung der möglichen Entwicklung eines Anwendungs-unabhängigen Sprachdiktiersystems durch Erweiterung und Verbesserung der folgenden Technologien:

- Triphone-basierte akustische Modelle.
- Statistische Sprachmodelle auf der Grundlage von Teilwörtern (Silben- oder Morphem-Teilworteinheiten).

Dieses Projekt baut auf den Grundlagen auf, die in den beiden vorangegangenen Projekten (HSB-I und II) gelegt wurden, und zielt darauf ab, die Ressourcen und Werkzeuge weiter auszubauen und zu entwickeln, die notwendig sind, um in Richtung Large Vocabulary and Continuous Speech Recognition (LVCSR) im Obersorbischen voranzukommen.

Sie ist zunächst auf die Anwendung beschränkt (domänenabhängig) und kann perspektivisch auch in bereichsunabhängigen Anwendungen eingesetzt werden.

Die Aktivitäten waren in drei Arbeitspakete unterteilt, die auf die Hauptkomponenten eines traditionellen LVCSR-Systems abzielten: die Akustik (Arbeitspaket 1), die Sprachmodellierung (Arbeitspaket 2) und den Decoder innerhalb des Spracherkenners (Arbeitspaket 3).

Die akustische Modellierung unter Verwendung von Triphonen wurde entwickelt und erreichte einen Grad an Robustheit, der eine sprecherunabhängige Erkennung unter realen und ungünstigen Bedingungen gewährleistet. Das akustische Modellierungsverfahren muss nicht weiterentwickelt werden und die Modelle können mit weiteren Daten deutlich verbessert werden.

Die Spracherkennung für stark flektierte Sprachen (wie das Obersorbische) stellt aufgrund der vielen Wortformen, die in den Wortschatz aufgenommen werden müssen, eine Herausforderung für die Sprachmodellierung dar. Der Umfang des Vokabulars und die Qualität der Sprachmodi haben direkten Einfluss auf die Spracherkennungsleistung.

Im Arbeitspaket Sprachmodellierung wurden Werkzeuge und Verfahren für die Textverarbeitung und -normalisierung, die Wortklassenmodellierung mit Erkennung von benannten Entitäten und die Tokenisierung in Unterworteinheiten (z. B. Morpheme) entwickelt. Die Verfahren wurden evaluiert, und die leistungsfähigsten wurden für die domänenabhängige Spracherkennungskonfiguration verwendet.

Der Idealfall wäre der, bei dem das Sprachmodell auf den Anwendungs-abhängigen Texten mit möglichst kleinem Vokabular aufgebaut ist und die gesprochenen Sätze dem Anwendungsgebiet entsprechen.

Im dritten Arbeitspaket wurde der Decoder in der Spracherkennungsanwendung modifiziert, um die Wortklassenmodellierung zusammen mit den Teilwort-Token zu unterstützen. Der Erkenner ist konfigurierbar und flexibel und bietet eine robuste Echtzeiterkennung für die gegebenen akustischen und sprachlichen Modelle.

3 Arbeitspaket 1: Unterstützung beim akustischen Training

3.1 Übersicht

Einführung neuer Trainingsverfahren für akustische Modelle (monophon und triphon) einschließlich neuer Sprachaufnahmen und Änderungen in der Phonetik (siehe Arbeitspaket 2).

3.2 Zielstellungen

Die Ziele dieses Arbeitspaketes sind die Erstellung neuer trainierter akustischer Modelle unter Verwendung des erweiterten Sprachkorpus mit neuen Aufnahmen.

Die Ergebnisse des Arbeitspakets sind die Lieferung von:
- Monophone akustische Modelle für die Verwendung mit dLabPro (z.B. Smart Home Anwendung).
- Akustisches Triphon-Modell zur Verwendung mit dem eigenen Erkennungsprogramm (recIKTS) für Diktier- und Untertitelungsfunktionen.
- Trainingsumgebung mit entsprechenden Konfigurationen.

Zusätzlich werden abgestimmte, fehlerkorrigierte Sprachressourcen bereitgestellt:
- Augmentierte Sprachaufnahmen
- Phonetische Beschriftungen zur Erstellung des Merkmalsextraktionsobjekts (feainfo.object)

3.3 Ergebnisse

3.3.1 Sprachkorpus

Der Korpus wird aus den neu bereitgestellten Aufnahmen und Transliterationen erstellt, die nicht übereinstimmenden Dateien werden aus den Dateilisten entfernt.
Alle Originalaufnahmen werden ergänzt, die endgültige Dauer des Korpus beträgt 74:58:47 (hh:mm:ss).

Die Aufnahmen, die Beschriftungen für die Vorgabe, das phonemreduzierte Set und die Transliterationen wurden archiviert und in einem gemeinsamen Ordner in der Cloud bereitgestellt.

Einige der Aufnahmen wurden aufgrund fehlender Transliterationen oder schlechter Qualität nicht für Training und Tests berücksichtigt (etwa 3,5 %).
Insgesamt wurden 58008 von 60133 Dateien in den Trainings-, Entwicklungs- und Testdatensatz aufgenommen.

1. Training (131 Sprecher; 62,32 Stunden; 52895 Aufnahmen) inklusive augmentierte Versionen:

Name	Bezeichnung	Dauer (hh:mm:ss)
	In den vorangegangenen Projekten erstellte Sprachressourcen	
HSB-1	Recordings from the HSB-I project	3:33:50
HSB-2	Recordings from the HSB-I project	3:22:30
HSB-3	Recordings from the HSB-I project	4:38:34
SCF_9A	speech_corpus_film_9_a_pol	0:08:29
SCF_G	speech_corpus_film_gilles	0:53:12
SCF_KK	speech_corpus_film_karla_a_katrina	0:40:15
SCF_MI	speech_corpus_film_mpz_insekten	0:28:08
SCF_MR	speech_corpus_film_mpz_reise	0:26:38
SCF_MW	speech_corpus_film_mpz_wjedro	0:31:25
SCF_P	speech_corpus_film_peeweeje	0:48:07
SCF_SW	speech_corpus_film_syn_winnetouwa	1:43:09
	Im Rahmen dieses Projekts erstellte Sprachressourcen	
SCF_MWJERA	speech_corpus_film_mala_wjera	1:34:28
SCM_R1	speech_corpus_mic_recordings_1	4:24:18
SCM_R2	speech_corpus_mic_recordings_2	1:41:41
SCM_R3	speech_corpus_mic_recordings_3	3:10:03
SCM_R4	speech_corpus_mic_recordings_4	3:09:00
SCM_R5	speech_corpus_mic_recordings_5	3:28:14

Fig. 1 Korpus für das Training.

2. Entwicklungskorpus (Kreuzvalidierung) (17 Sprecher; 2,28 Stunden):

- CV Gemeinsamer Sprachdatensatz Version 5.1

3. Test (8 Sprecher; 6.63 Stunden):

- AABT
- HSB_1_0001
- HSB_1_0010
- HSB_2_0001
- HSB_2_0011
- HSB_3_0001
- HSB_3_0006
- HSB_3_0007

4. Verschiedene Aufnahmen:
- ADP Anpassungsdateien (0:11:02)
- ADP_B Anpassungsdateien (0:11:02)
- GRM Grammatiktests (0:01:30)

5. Anwendungsspezifische Aufnahmen

Die Aufnahmen werden aus YouTube-Videos von Sonntagsgottesdiensten zusammengestellt und dienen der Leistungsbewertung.

- MISA speech_corpus_boze_mse_chroscicy_1 (3:23:10)

3.3.2 Trainingsumgebung und Konfiguration

Um neue Aufnahmen zum Trainingsset hinzuzufügen, befolgen Sie die Schritte, die in den Benutzerhandbüchern des **HSB-III-Projektberichts** beschrieben sind.

Es wird eine allgemeine Schritt-für-Schritt-Erklärung gegeben, weitere technische Details finden Sie in der README.md der AP1-Lieferung:

1. Neue Sprachdaten in 16 kHz und 16-Bit-Mono-WAV-Format umwandeln.
2. Augmentieren der Daten mit `scripts/augument.py`.
3. Das Vorhandensein von Transliterationsdateien überprüfen und Dateilisten erstellen.
4. Die Phonemmodelle im Standard-Akustikmodell "3_8" entsprechend den Ergebnissen von AP2 reduzieren, um das Akustikmodell "hsb_redux.hmm" zu erhalten.
5. Das Vokabular aus den Transliterationsdateien erstellen.
6. Für jedes akustische Modell die Konfigurationen mit Lexika/Grammatiken aus dem Vokabular erzeugen.
7. Forced-Alignment durchführen, um Phonem-Labels zu erzeugen, indem die erstellten Konfigurationen in entsprechenden Experimenten verwendet werden (siehe unten).
8. Erzeugen Sie die Kaldi-Experimentdateien und die entsprechenden dLabPro/UASR-Konfigurationsdateien.
9. Führen Sie Kaldi-Experimente aus, die neu trainierten Modelle werden entsprechend im lokalen Modellordner der dLabPro-Experimente gespeichert.
10. Die Kaldi-Modelle (mono, tri1, tri2) in ein recIKTS (eigener Spracherkenner)-kompatibles Format konvertieren.

3.3.3 Verzeichnis der Standard-Phoneme

Der dLabPro-Experimentierordner heißt "HSB-P4DF" und der entsprechende Name "p4df" wird für die Modelle und Konfigurationsdateien verwendet.

3.3.4 Verzeichnis der reduzierten Phoneme

Der dLabPro-Experimentierordner heißt "HSB-P4RF" und der entsprechende Name "p4rf" ist für die Modelle und Konfigurationsdateien.
Das akustische Modell für das Forced-Alignment ist gegenüber dem Standardmodell "3_8.hmm" reduziert, indem die Vokale e->E, o->O, u->U zusammengefasst werden.

3.3.5 Kaldi-Rezepte

Die gelieferten Rezepte sollten im Ordner "${KALDI_ROOT}/egs" abgelegt bzw. verlinkt werden und die symbolischen Links (unter Linux) erstellt werden. Die symbolischen Links sollten den vollständigen Pfad enthalten und die folgenden Pfade sollten entsprechend der Ordnerstruktur in den folgenden Skripten angepasst werden:

- kaldi_recipes/p4df/run.sh
- kaldi_recipes/p4rf/run.sh

die zum Trainieren der neuen Mono- und Triphonmodelle für den Standard- ("3_8") und den reduzierten ("hsb_redux") Phonemsatz verwendet wurden.

3.3.6 Modellkonvertierung

Die mit Kaldi trainierten Modelle (mono, tri1, tri2) werden im "model"-Ordner des entsprechenden Experiments gespeichert und entsprechend ausgegeben:

Für das HSB-P4DF-Modell:
- default (3_8.hmm with corresponding feainfo.object)
- feainfo.object (new one created by *fea_uasr2kaldi.py*)
- hsb_p4df_1000_40000_sb125p025_nfea_tri2
- hsb_p4df_20000_sb125p025_nfea_mono
- hsb_p4df_500_20000_sb125p025_nfea_tri1

Für das HSB-P4RF-Modell:
- default (hsb_redux.hmm with corresponding feainfo.object)
- feainfo.object (new one created by *fea_uasr2kaldi.py*)
- hsb_p4rf_1000_40000_sb125p025_nfea_tri2
- hsb_p4rf_20000_sb125p025_nfea_mono
- hsb_p4rf_500_20000_sb125p025_nfea_tri1

Jeder Modellordner enthält das "*.hmm"-Format, das mit dem eigenen "recikts_1.0.3l"-Erkennungsprogramm und Konfigurationen verwendet werden kann. Die Unterordner enthalten auch die Dekodierungsordner mit den Leistungsbewertungen, die während des Trainings und der Evaluierung des Test- und Entwicklungsteils des Korpus erzielt wurden.

3.3.7 Performanz-Ergebnisse

Die Leistung wurde im Rahmen des Kaldi-Trainingsverfahrens bewertet, indem die optimalen Werte für die Worteinfügungsstrafe und das Sprachmodellgewicht ermittelt wurden. Das Sprachmodell wird aus dem kombinierten Korpus (train, dev, test) erstellt, das den Namen "kaldi_v4" trägt. Daher sind die erzielten Ergebnisse zu optimistisch, da der Inhalt der bewerteten Sprache bereits beim Training des Sprachmodells gesehen wurde.
Sie geben jedoch einen Eindruck über die relative Verbesserung oder Verschlechterung der akustischen Modelle während des Trainings und der Parameteroptimierungen.

Fig. 2 Performanz-Ergebnisse

Wortfehlerrate (%)	Entwicklungsdatensatz[1]			Testdatensatz		
	mono	tri1	tri2	mono	tri1	tri2
HSB-III default[2]	10.40	6.91	7.94	12.52	10.83	10.14
HSB-IV default	4.84	3.81	5.21	**2.46**	**1.91**	1.83
HSB-IV reduced	**4.10**	**3.48**	**5.03**	2.88	2.20	**1.66**

Aus der Tabelle geht hervor, dass das Modell mit weniger Phonemen ("reduced" - p4rf) auf dem Entwicklungsdatensatz (dem Common-Voice-Korpus) bessere Ergebnisse erzielte, die im Training nicht zu sehen waren, was auf ein robusteres akustisches Modell auf ungesehenen Daten hindeutet, während das Standardmodell (p4df) bessere Ergebnisse auf den Testdaten erzielte, die den Trainingsdaten ähnlich sind. Im Durchschnitt erzielte das Modell "triphone 1" die beste Leistung, weshalb es als akustisches Modell gewählt wurde, das in den anderen Arbeitspaketen (AP 3) für die Implementierung der Diktier-/Transkriptionsfunktion verwendet wird.

3.4 Zusammenfassung

Mit den neu erstellten Sprachressourcen, den überlieferten Sprachkorpora und ihren erweiterten Gegenstücken, konnte der Umfang der Sprachressourcen im Obersorbischen auf über 70 Stunden erhöht werden.
Das Korpus wurde verwendet, um neue und robustere akustische Modelle mit dem ursprünglichen (Standard-) Phoneminventar und Ausspracheregeln sowie mit einem reduzierten Phoneminventar mit den Vokalzuordnungen e->E, o->O, u->U zu trainieren.
Das erweiterte Korpus mit den bearbeiteten Transliterationen und der augmentierten Sprache wurde zusammen mit der Trainings- und Konfigurationsumgebung und den Skripten wie in den Zielen des Arbeitspakets angegeben geliefert.

[1] Common-Voice-HSB-Korpus

[2] HSB_III_Report_2023_EN_full.pdf

4 Arbeitspaket 2: Lexikon / Phonetisches Inventar

4.1 Übersicht

In diesem Arbeitspaket wurden die Optimierung des phonetischen Inventars und weitere Quellen zur Erstellung des Lexikons untersucht und integriert.

4.2 Zielstellungen

Die wichtigsten Punkte, die untersucht und bewertet wurden, sind:

- Neudefinition der Symbole des Inventars (insbesondere der Phoneme "jn" und "ji") nach Vergleich mit dem Aussprachewörterbuch der obersorbischen Sprachsynthese.
- Berücksichtigung/Integration von fremdsprachlichen Phonemen ("ä" "ö" "ü" "~n" ...).
- Vergleichende Messung zur Bewertung der Modelle für die Klänge "e", "o" und "jn" und ggf. Integration in bestehende Modelle: "E", "O", "n".
- Verbesserungen bei der Erstellung von Wörterbüchern:
  - Verbesserungen des regelbasierten Ansatzes.
  - Integration von manuell erstellten Ausspracheausnahmen und von Hand erstellten Lexika.
  - Integration/alternativer Ansatz unter Verwendung des Aussprachewörterbuchs der obersorbischen Sprachsynthese.
  - Bewertung des angebotenen alternativen Ansatzes für die statistische Erstellung des Lexikons (Phonetisaurus).

Die folgenden Ressourcen wurden geliefert:

- Optimierte Phoneminventare und Aussprachеregeln.
- Verbesserte und angepasste Skripte und Konfigurationen für die automatische Erstellung von Lexika.

4.3 Ergebnisse

4.3.1 Korpus-Ersteller

Neue Funktionalitäten des Skripts "corpus creator" (früher "BAS_generator") wurden eingeführt:

- Die Auswahl von Sätzen und die Erstellung von Aufzeichnungskörpern funktioniert wie das alte BAS-Skript.
- Verbindliche und alternative Aussprachеregeln.
- "OR"-Operator "|" pro Regel, der alle Kombinationen erzeugt.
- Einbeziehung von handgefertigten Lexika (Lexika oder UASR grm-Dateien).
- Definition des Ausgabeordners, um ein Überschreiben bei unterschiedlichen YAML-Konfigurationen zu vermeiden.
- Fehlerbehebungen (z.B. doppelte Phoneme).
- Phonem-Zuordnung vom größeren zum kleineren Bestand für die handgefertigten Lexika.

Weitere technische Einzelheiten sind in der Datei WP3/README.md der Lieferungen enthalten.

4.3.2 Vergleich der Spracherkennungs-Performanz (Standard vs. reduziert)

Die Datei "ASR_results_default_vs_reduced.log" enthält die Ergebnisse der Spracherkennung von KALDI auf dem Test- und Entwicklungsset unter Verwendung von "default" und "reduced" Phonem-Sets.
Die Ergebnisse werden auch im AP1-Abschnitt dargestellt, wo die Verbesserungen des reduzierten Phonembestandes innerhalb der statistischen Sicherheit liegen, die Verwendung eines reduzierten Satzes jedoch die Erkennungsgeschwindigkeit verbessert.

4.3.3 Vergleich mit MARY-TTS-Lexikons

Die Datei "mary_tts_phoneme_inventory.xlsx" enthält das ursprüngliche MARY-TTS-Lexikon mit den handgefertigten Aussprachen und einem wesentlich größeren Phoneminventar sowie das mit der MARY-TTS.yaml-Konfiguration generierte Lexikon für das gleiche Vokabular.
Beide Lexika werden pro Wort und dessen Aussprache verglichen.

4.3.4 Statistische G2P-Trainingsexperimente

Das Quelllexikon und das Phoneminventar (inventory.txt) stammen aus dem Projekt "MARY-TTS-HSB".[1]
Die Originaldatei hsb.txt wurde konvertiert und alphabetisch sortiert in hsb_sorted.txt.

Die generierten Aussprachen sind in der Excel-Tabelle "mary_tts_phoneme_inventory.xlsx" auf der Registerkarte "Phonetisaurus Comparison" aufgeführt und mit denen verglichen, die mit der "corpus creator" HSB-P4DF.yaml-Konfiguration generiert wurden.
Der Glottalstopp "Q" wird entfernt und alle Aussprachevarianten bleiben erhalten.

Ein direkter Vergleich ist nicht möglich, da das phonetische Inventar von "mary-tts-hsb" viel größer ist, aber viele der Wörter die gleiche Aussprache haben.

4.4 Zusammenfassung

Das Skript "corpus creator" wird neben der Korpusnormalisierung und der Erstellung von Sprachaufzeichnungsexperimenten auch zur automatischen Lexikonerstellung verwendet. Es werden neue Funktionalitäten eingeführt und die Benutzerfreundlichkeit wird verbessert.
Nach Prüfung der Möglichkeiten zur Reduzierung des Phoneminventars kam man zu dem Schluss, dass die Artikulationsposition (offen, geschlossen) einiger Vokale bei der Spracherkennung keine so große Rolle spielt wie bei der Sprachsynthese.
Daher wurde das Standard-Phoneminventar um die Zuordnungen der Vokale reduziert: e->E, o->O und u->U. Es wurde auch nicht in Betracht gezogen, das Phonem-Inventar entsprechend dem "MARY-TTS-HSB"-Projekt signifikant zu verändern, da die Zielsetzungen sehr unterschiedlich sind. Hörer des Obersorbischen konnten leichte Aussprachefehler im Text-To-Speech (TTS)-System leicht erkennen, während im Falle der Spracherkennung solche Nuancen weniger häufig vorkommen und statistisch gesehen für die akustische Modellierung nicht wichtig sind.
Die "MARY-TTS-HSB"-Aussprachen wurden qualitativ mit denen verglichen, die automatisch mit dem Standard-Phoneminventar erstellt wurden, und man kam zu dem

[1] https://github.com/marytts/marytts-lexicon-hsb/tree/master/modules/hsb/lexicon

Schluss, dass es bei der Spracherkennung im Allgemeinen keine großen Unterschiede bei den Aussprachen gibt.

5 Arbeitspaket 3: Unterstützung bei der Erstellung von Sprachmodellen

5.1 Übersicht

Dieses Arbeitspaket dient der Erstellung von anwendungsspezifischen Sprachmodellen auf Grundlage der bereits gesammelten und normalisierten Texte, mit zusätzlichen neuen anwendungsspezifischen Texten und deren Normalisierung und Integration. Mögliche Verbesserungen wurden auch für den bereits verwendeten Ansatz untersucht, der tokenisierte morphologische Einheiten verwendet, wobei auch andere Algorithmen zur Tokenisierung natürlicher Sprachverarbeitung (Natural Language Processing, NLP) berücksichtigt wurden.

5.2 Zielstellungen

In Anbetracht der Voraussetzungen der bereitgestellten Texte für die Normalisierung und Integration sowie der Einigung über die Anzahl und den Umfang der Anwendungsbereiche wurden die Hauptziele des Arbeitspakets definiert:

- Sammeln und Normalisieren neuer Textdaten für eine bestimmte Domäne.
- Training von verschiedenen domänenabhängigen Sprachmodellen.
- Erstellung entsprechender Skriptkonfigurationen für die Sprachmodellgenerierung
- Implementierung alternativer Tokenisierungsalgorithmen zur Erzeugung von morphologieähnlichen Einheiten

5.3 Ergebnisse

5.3.1 Definition des Anwendungsbereichs

Der Anwendungsbereich wurde auf die Untertitelung von Audioaufnahmen von Gottesdiensten ("Boža mša z Chróśćic") mit möglicher Verwendung bei Live-Übertragungen und
Offline-Transkriptionen, um kompatible Untertitel für YouTube bereitzustellen.
Live-Übertragungen sind aufgrund der ungünstigen akustischen Bedingungen eine Herausforderung. Es gibt auch Zeiten, in denen nicht gesprochen wird, z. B. Orgelmusik und Chorauftritte.
Darüber hinaus sind die Textdaten aus diesem Anwendungsbereich immer noch spärlich. Auch wenn sich der Großteil der Sprache auf religiöse Texte bezieht, gibt es irgendwo angekündigte zukünftige Ereignisse, die die korrekten Bezeichnungen von Personen, Orten, Zeiten und Daten enthalten.

5.3.2 Korpus

Das domänenspezifische Korpus enthält verifizierte und normalisierte Texte der aufgenommenen Gottesdienste (mit Datum):

- misa221030
- misa221106
- misa221113
- misa221225
- misa230129
- misa230319
- misa230326
- misa230402
- misa230406
- misa230409
- misa230430

Das "Master"-Korpus, das als Quelle für die Sprachmodellierung verwendet wird, setzt sich aus den folgenden bestehenden und neuen Textdaten zusammen:

- misa (aktualisiert mit neuen Transkriptionen)
- mcn
- spw,und
- hsb_bible_btu.

Der "Extra"- oder "Out-of-the-Domain"-Korpus ist eine Kombination aus:

- kaldi_v4 (aktualisiert mit neuen Transkriptionen) und
- Wortklassen (Wortklassenwörter, Zahlen (1-1000), Zeit, Datum, Orte, Namen).

"Kaldi_v4" enthält die Transkriptionen "train", "dev" und "test", die für das Training des akustischen Modells mit Kaldi verwendet werden. Alle Texte sind Teil der Auslieferung des Arbeitspakets und wurden in die entsprechenden Repositories in Unterordnern in einem separaten Zweig verschoben.

Die Menge an In-Domain-Texten ist noch sehr gering; daher enthält das Quellkorpus auch Out-of-Domain-Texte (kaldi_v4, Wortklassen). Durch die Modellierung von Teilwörtern wird jedoch sichergestellt, dass unbekannte Wortkontexte bis zu einem gewissen Grad abgedeckt sind.

Es ist wichtig zu betonen, dass das Hauptproblem immer noch darin besteht, dass das Korpus nicht mit den ungesehenen Aufnahmen übereinstimmt. Die Lösung ist die kontinuierliche Verbesserung der Sprachmodelle durch Transkription möglichst vieler Sendungen.

Ein gutes Sprachmodell kann bei schlechter Audioqualität (Nachhall) robustere Ergebnisse liefern und umgekehrt kann eine gute Audioqualität eine gute Leistung bei einem schlechten Sprachmodell liefern.

Spezifikation des "Master"-Korpus (Wordpiece Tokenizer):

Input: corpus/master.corp
- Sentences: 95025
- Tokens: 1236601
- Unique Tokens: **42961**
- Avg.Length U.Tokens: 7.87

Output: corpus/master_wrdpc_hsb.corp
- Sentences: 95025
- Tokens: 4154165
- Unique Tokens: **15991**
- Avg.Length U.Tokens: 5.96
- Tokens Increase (%): 235.93
- Unique Tokens Decrease (%): -62.78
- Average Token Length Decrease (%): -24.20

Spezifikation des "Extra"-Korpus (Wordpiece tokenizer):

Input: corpus/extra.corp
- Sentences: 302574
- Tokens: 1832909
- Unique Tokens: **34178**
- Avg.Length U.Tokens: 8.08

Output: corpus/extra_wrdpc_hsb.corp
- Sentences: 302574
- Tokens: 6070818
- Unique Tokens: **17035**
- Avg.Length U.Tokens: 6.21
- Tokens Increase (%): 231.21
- Unique Tokens Decrease (%): -50.16
- Average Token Length Decrease (%): -23.15

5.3.2.1 Text-Augmentiertung

5.3.3 Das Python-Modul "NLPAUG" wurde verwendet, um die Menge der Textdaten zu erhöhen, indem die Wortstellen in den Sätzen zufällig vertauscht werden. Dies wird in zwei Iterationen durchgeführt. Das Verhältnis zwischen dem Originaltext und den ergänzten Texten ist 1:2.

5.3.4 Werkzeuge

Die folgenden Tools für die Tokenisierung, Text- und Audioerweiterung, Formatkonvertierung und andere Hilfsprogramme wurden entwickelt:

- ir_augument.py, für die Impulsantwortvergrößerung.
- bpe_tokenizer.py, BPE Tokenizer von huggingface/tokenizers.
- wordpiece_tokenizer.py, Wortstück-Tokenizer von huggingface/tokenizers.
- compile_wordclasses.py, aktualisiertes Skript für die Kompilierung.
- print_wordclass_sequences.py, Ausgabe aller möglichen Wortklassenfolgen.
- Morphesor.py, morfessor tokenizer (wie in der vorherigen Version HSB-III).
- evlres.py, Auswertung der Erkennungsausgabe.
- log2srt.py, Konvertierung der Erkennungsausgabe in das SubRip-Format.
- text_augment.py, zufällige Worttausch-Text-Ergänzung.
- iasrrdc.py, Spracherkennungsmodule.
- iasrfst.py, Zustandsautomat-Module.
- idlabpro.py, dLabPro-Funktionen.
- hsb_helper.py, verschiedene Dienstprogramme.

5.3.5 Kreuzvalidierung

Um die Auswirkungen von Wörtern außerhalb des Vokabulars (OOVs) zu bewerten, wurde eine Kreuzvalidierung mit "Morfessor"-Tokenisierung von Teilwörtern (3-Gramm) durchgeführt und mit Vollwörtern (3-Gramm) verglichen. Die gewählte Strategie beinhaltete die Auslassung eines Ordners (z.B. misa221030) zum Testen und die Erstellung der Erkennungskonfiguration (Voll- und Teilwort-LMs) auf den übrigen Ordnern. Dies wurde für jeden Ordner des "misa"-Korpus wiederholt.

Die Auswirkung von OOV und ungesehenen n-Grammen aufgrund der begrenzten Anzahl von anwendungsbezogenen Transkriptionen zeigte sich in einer niedrigen Erkennungsleistung, mit einer durchschnittlichen Genauigkeit von ~32% mit Vollwort-Sprachmodellen. Es wurde gezeigt, dass die Modellierung von Teilwörtern die Erkennungsleistung verbessert, indem auch Wörter außerhalb des Vokabulars (OOV) erkannt werden und eine durchschnittliche Genauigkeit von 51% erreicht wird.

5.3.6 Modellierung von Teilwörtern

Außerdem haben wir den "Morfessor" mit dem "Wordpiece"-Tokenizer für die Modellierung von Teilwörtern verglichen:

- "Morfessor" wird für die Tokenisierung mit dem HSB-Modell verwendet (laut den Ergebnissen des HSB-III-Projekts die beste Qualität).
- "Wordpiece" von Huggingface Tokenizers[1] wurde ebenfalls auf allen verfügbaren HSB-Daten mit Ausnahme der anwendungsspezifischen Daten trainiert und für die Tokenisierung von Teilwörtern verwendet.

"Wordpiece" lieferte etwas bessere Ergebnisse mit qualitativ besseren Tokens und wurde für die Tokenisierung der Textdaten für die Sprachmodellierung verwendet.

[1] https://github.com/huggingface/tokenizers

5.3.7 Sprachmodelle

Statistische Teilwort-Sprachmodelle mit 3-, 4- und 5-Gramm-Modellen wurden an den "misa"-Aufnahmen getestet. Der 4-Gramm-Tokenizer "Wordpiece" lieferte die beste ausgewogene Leistung in Bezug auf Wortfehlerrate und Echtzeitfaktoren.
Die aus "Master" und "Extra" erstellten Sprachmodelle wurden im Verhältnis 3 zu 2 interpoliert. Das daraus resultierende "merged. ARPA"-Sprachmodell wurde für die Konfiguration des Erkenners verwendet.

5.3.8 Augmentiertung des MISA-Korpus mit nachhallender Sprache

Beim Testen der Erkennung auf neuen, ungesehenen Aufnahmen fiel auf, dass sich die akustischen Bedingungen im Vergleich zu den "misa"-Aufnahmen deutlich unterscheiden. Die neuen, ungesehenen Aufnahmen haben einen viel stärkeren Nachhall (Echo). Dies macht die Erkennung für Live-Transkriptionen undurchführbar, auch Offline-Untertitelung ist aufgrund der schlechten Erkennungsleistung schwierig.

Das "misa"-Korpus wurde mit Hilfe des Python-Moduls "audiomentati-ons"[1] und der Funktion "RoomSimulator" um echoreiche Aufnahmen erweitert.
Die "Raum"-Abmessungen (x, y, z, in Metern) wurden so weit wie möglich an den Ort angepasst, an dem die Aufnahmen gemacht wurden.

```
min_size_x=19,
max_size_x=23.0,
min_size_y=25,
max_size_y=50.0,
min_size_z=6,
max_size_z=12.0,
max_order=12,
min_absorption_value=0.01,
max_absorption_value=0.05.
```

5.3.9 Nachhall-Entfernung

Das Python-Modul "Voicefixer"[2] wurde für die Enthallung der echolastigen Audioaufnahmen verwendet. Die resultierenden Dateien wurden rekonstruiert und es waren einige Nicht-Sprach-Artefakte zu erkennen. Als Nebeneffekt wurden die Musiksegmente und der Gesang größtenteils aus den kompletten Sendeaufzeichnungen herausgefiltert. Die Ergebnisse sind jedoch je nach Audioqualität sehr unterschiedlich.

[1] https://github.com/iver56/audiomentations

[2] https://github.com/haoheliu/voicefixer

5.3.10 IKTS Erkenner

Die neue Version **recIKTS v1.0.3l** des Erkenners wird ausgeliefert. Sie enthält folgende Verbesserungen und neue Funktionalitäten:

- Verwendung von "YAML"-Konfigurationsdateien für den "tesbld"-Konfigurator.
- Verwendung von Dateilisten als Eingabe für den "testrec"-Erkenner.
- Konfigurationsmöglichkeit für minimale und maximale aktive Zustände im Decoder (smin, smax)
- Neue Schnittstelle für "recikts"-Bibliothek (kann veröffentlicht werden).
- Fixe Anpassung für große dLabPro-Datenobjekte: Dekodierung mit Mischungen und Neuzuordnung der resultierenden Symbole.
- Erneuerung der Wortgrenzenmarkierung für Kaldi-Sprachmodelle.
- Ausgangstriggerpositionen und Signalamplitude.Verschiedene Fehlerkorrekturen (UPFA-Merkmalsberechnung, Subword-Tagging, Zustandsautomat-Bestimmung).

5.3.11 Anpassung

Das akustische Modell wird in zwei Iterationen angepasst. Zunächst an den Echo-augmentierten Aufnahmen und anschließend an den realen, halligen Aufnahmen (für einen Sprecher).

5.3.12 Erkennungs-Experimente

Die Ergebnisse der Erkennungsexperimente basierten auf dem angepassten akustischen Modell und dem
4-Gramm-Fusionsmodells auf dem "misa"-Augment "misa_A" (je 69 Sätze) sowie einer Auswahl von transkribierten echoischen Aufnahmen "vsegment" und "vsegment_fix" (je 10 Sätze).
Die vollständigen Logfiles ("test_") und die gefilterten Wortfehlerraten ("results_") befinden sich im Ordner "re-sults".

5.3.13 Transkription von unbekannten aufgezeichneten Übertragungen

Während der Erkennung wird die Ausgabe des Erkenners in eine Protokolldatei umgeleitet. Die Protokolldatei wird weiter in das SubRip-Format (*.srt) für Untertitel konvertiert, die zusammen mit der aufgezeichneten Sendung auf YouTube hochgeladen werden können. Die entsprechenden SRT-Dateien befinden sich im selben Verzeichnis wie die Audiodateien.
Wir empfehlen die Software SubtitleEdit[1] für die Bearbeitung von SRT-Dateien, um die kurzen Segmente zu verketten und die Erkennungsfehler zu korrigieren. Die Software kann eigenständig ausgeführt werden und hat eine benutzerfreundliche Oberfläche. Beispiele für die Untertitelung von zwei aktuellen Sendungen vom 24.09.23 und 01.11.23 finden Sie im Ordner "examples" der WP3-Lieferung.

[1] https://www.nikse.dk/subtitleedit

5.4 Zusammenfassung

Weitere Verbesserung der Erkennung im Bereich "misa":

- Es wurden neue Transkriptionen aufgenommen.
- Akustische Verstärkung mit simulierter Raumimpulsantwort.
- Text-Augmentierung mit zufälliger Wortvertauschung.
- Anpassung des akustischen Modells an die nachhallenden Sprachaufnahmen (echte und erweiterte).
- Interpolation eines 4-Gramm-Sprachmodells innerhalb und außerhalb von Anwendungs-Korpora.

Wenn die Teilwortkontexte in den Trainingsdaten zu finden sind, ist die Erkennung im Allgemeinen sehr genau und praktisch brauchbar.

Um eine zuverlässige On- und Offline-Spracherkennung zu erreichen, muss das Sprachmodell dem erwarteten Sprachinhalt entsprechen:

- Weitere Transkriptionen und Textinhalte (z. B. Lesehinweise) hinzufügen.
- Verwendung robuste Sprechpausenerkennung, um sprachfremden Ton herauszufiltern.
- Verwendung eines Mikrofonarrays (ReSpeak[1]) für die Aufnahmen, um bessere Audioaufnahmen zu erhalten.
- Optional kann die Impulsantwort des Kirchensaals aufgezeichnet werden, um weitere Aufnahmen für das Training/Anpassung zu ergänzen.

[1] https://wiki.seeedstudio.com/ReSpeaker_Mic_Array_v2.0/

6 Arbeitspaket 4: Unterstützung bei der Anwendungsintegration

6.1 Übersicht

Das zuvor gelieferte "recIKTS" ASR-System wurde in die "Jitsi Meet"-Anwendung integriert, wobei die Überprüfung und Optimierung des Codes noch aussteht. Die Lieferung dieses Arbeitspakets besteht aus Korrekturen, Anpassungen und Verbesserungen an der Erkennungsanwendung und der "VOSK"-API und wird über Pull-Requests von Code-Änderungszweigen auf den entsprechenden GitHub-Repositories durchgeführt.

6.2 Zielstellungen

Die folgende Leistung wird erbracht:

- Überprüfung des Codes für die Anpassung des Erkenners an die "VOSK"-API.
- Integration, Verbesserung und Parametrisierung der Spracherkennung ("VAD").

6.3 Ergebnisse

Einen detaillierten Überblick über die geleistete Arbeit bei der Überprüfung und Optimierung des Codes gibt die Datei "pull.pdf" im WP4-Ordner der Lieferung.
Die Zusammenfassung der Pull Requests mit Codeänderungen des Repositorys ZalozbaDev/docker_recikts_vosk[1]:

- Behebung der Compiler-Warnung bei unpassender Zuordnungsfunktion.
- Festlegen der maximalen Array-Länge.
- Array-Länge in acceptWaveform korrigieren.
- Aktivierung der Konfiguration zur Kompilierungszeit für das Präfix-Verzeichnis und den reciktslib-Dateinamen.
- Konfigurierbare PREFIX und RECIKTSLIB.
- Wiederverwendung von übriggebliebenen Samples im VADWrapper auch für kürzere Segmente als nrVADSamples & Korrektur der Offset-Berechnung für memcpy.
- Verwendung von WebRtcSpl_Resample48khzTo16khz für das Resampling in acceptWaveform.
- Der erste Audioblock wird nicht verworfen.

Die Zusammenfassung der Pull-Requests mit Code-Änderungen des Repositorys ZalozbaDev/vosk_server[2]:

- Beenden der Zeichenkette vor der Ausgabe.

[1] https://github.com/ZalozbaDev/docker_recikts_vosk

[2] https://github.com/ZalozbaDev/vosk-server

6.4 Zusammenfassung

Der Code wurde geprüft und optimiert, und es wurde ein Dokument mit einer detaillierten Beschreibung der Pull Requests erstellt.

7 Arbeitspaket 5: Beratung

7.1 Übersicht

Auf der Grundlage der derzeit verfügbaren Technologie und Ressourcen wird ein Strategiepapier für die zukünftige Entwicklung der Spracherkennung für das Obersorbische erstellt.

7.2 Zielstellungen

Dabei wurden die folgenden Hauptpunkte berücksichtigt:

- Bewertung des Bedarfs an weiteren Sprachressourcen (Text, transkribiertes Audio) und Beschaffungsstrategie.
- Zukünftige technologische Verbesserungen (KI, Wortklassen ...).
- Open-Source-Spracherkennung (Kaldi-Decoder).Ende-zu-Ende Transfer Learning (Coqui, OpenAI Whisper, fairseq, word2vec, ...).

Es wurde ein Bericht mit entsprechenden Quellen und Referenzen erstellt.

7.3 Ergebnisse

Das Dokument skizziert den aktuellen Stand der obersorbischen Sprachtechnologien mit dem Fokus auf Spracherkennung und die verfügbaren Ressourcen. Es stellt den Stand der Technik (SdT) auf diesem Gebiet dar und vergleicht die klassischen, hybriden und Ende-zu-Ende Ansätze.
Im Kontext des Obersorbischen stellt sich die Frage, welcher Ansatz am effizientesten einen praktisch nutzbaren Spracherkenner liefert, der in realen Anwendungen eingesetzt werden kann und Sprechern bei ihren täglichen oder beruflichen Aufgaben hilft. Jeder Ansatz hat seine Vor- und Nachteile, wenn man die benötigten Sprach- und Rechenressourcen für das Training und auch für die Nutzung berücksichtigt.

7.3.1 Klassische und hybride Systeme

Die firmeneigene Software, der recIKTS-Spracherkenner, der den klassischen Ansatz verwendet, hat den Vorteil, dass die beobachteten Probleme in jeder Komponente, sei es die akustische oder die Sprachmodellierung, korrigiert werden können. Jedes der Module kann separat optimiert werden, und das System kann schrittweise verbessert werden, wenn mehr Daten zur Verfügung stehen, wie es in den früheren HBS-I-III-Projekten gezeigt wurde.

7.3.2 Ende-zu-Ende-Systeme

Der Hauptvorteil der Ende-zu-Ende-Systeme besteht darin, dass sie den "Black-Box"-Ansatz anwenden, wenn die Anforderungen an die Datenmenge erfüllt sind. Sie könnten bei der Allzweckaufgabe robuster sein, da sie die Graphemsequenz von ungesehenen Wörtern erzeugen können. Für eine akzeptable Leistung ist jedoch immer noch eine Sprachmodellierung erforderlich.
Im Gegensatz dazu erfordert die Allzweck-Anwendungserkennung mit klassischen und hybriden Systemen die Sammlung repräsentativer Textdaten für die statistische Sprachmodellierung mit begrenztem Vokabular, in dem noch Wörter außerhalb des Vokabulars vorkommen können.
Mehr Daten für die Feinabstimmung der vortrainierten Modelle sind vorteilhaft, aber es ist nicht garantiert, dass die Leistung proportional mit der Zunahme der Datenmenge wie bei den klassischen und hybriden Systemen verbessert wird.

7.3.3 Empfehlungen

In Anbetracht der aktuellen Entwicklung der Ende-zu-Ende- und der hybriden Spracherkennungssysteme und in Anbetracht des geringen Umfangs der verfügbaren elektronischen Sprachressourcen sollte die Hauptstrategie für die weitere Entwicklung der obersorbischen Sprachtechnologien (insbesondere der Spracherkennung) die folgenden allgemeinen Empfehlungen berücksichtigen:

7.3.3.1 Sprachkorpus (akustische Modellierung)

- Kontinuierliche Sammlung neuer Sprachressourcen unter Verwendung verfügbarer Sprachtechnologien im Obersorbischen und semi-supervised Transkription von Audioaufnahmen.
- Sobald eine größere Menge neuer Daten gesammelt wurde, können die akustischen Modelle aktualisiert werden.

7.3.3.2 Lexikon (Modellierung der Aussprache)

- Gegenwärtig sind das Phoneminventar und die Ausspracheregeln gut definiert, und es sind keine weiteren wesentlichen Verbesserungen zu erwarten, wenn die Definitionen reduziert oder erweitert werden.

7.3.3.3 Textkorpus (Sprachmodellierung)

- Kontinuierliche Bemühungen um die Sammlung von Textinhalten durch Kontaktaufnahme mit einschlägigen Organisationen, bei denen solche Ressourcen erhältlich sind (Nachrichtenportale, Regierungsdokumente, Bildungsmaterialien).
- Anreicherung des Textkorpus mit Part-Of-Speech (POS)-Tags unter Verwendung von SdT NER.
- Auswahl von POS-Tags für die Modellierung von Wortklassen.
- Unabhängig von der verwendeten Technologie (N-Gramm, rekurrente neuronale Netze - RNNs, usw.) ist eine Sprachmodellierung erforderlich.

7.3.3.4 Sprachtechnologie

- Die Aufrüstung des eigenen Spracherkennungssystems auf andere Sprachmerkmale und hybride Zeitverzögerte- Neuronale-Netz-Modelle (DNN/HMM) wird die Erkennungsleistung erheblich verbessern.

7.3.3.5 Ende-zu-Ende-Systeme

- Untersuchung der Machbarkeit des praktischen Einsatzes der von HuggingFace empfohlenen Ende-zu-Ende-Frameworks, wie OpenAI Whisper und fairseq MMS.
- Einrichtung und kontinuierliche Aktualisierung von Versuchsrahmen auf der Grundlage der SdT Ende-zu-Ende-Spracherkennungssysteme für das Transferlernen unter Verwendung vorhandener Sprachressourcen.
- Bewertung der eingesetzten Ende-zu-Ende-Systeme im Vergleich zum klassischen oder hybriden Spracherkennern für allgemeine (Jitsi) und fachspezifische Anwendungsfälle (Misa, Smart Lamp, etc.)

7.3.4 Zukünftige Arbeiten

- Weitere Verbesserung des recIKTS Spracherkennungssystems durch robustere Merkmale und Einführung von DNN/HMM-Modellen.
- Unterstützung bei der Sammlung und Validierung von Textdaten für eine zuverlässige Sprachmodellierung.
- Untersuchung der Machbarkeit des praktischen Einsatzes und Evaluierung von geeigneten Ende-zu-Ende-Frameworks.
- Verbreitung des Wissens auf Konferenzen und unter relevanten Interessengruppen (Stiftung, Institut, Zeitungen, Medien ...).

7.4 Zusammenfassung

Es wurde ein 34-seitiges Strategiedokument mit den folgenden Hauptabschnitten erstellt.

- Einleitung, die die Herausforderungen in der Sprachtechnologie mit einem Überblick über Ansätze und Anwendungen sowie den aktuellen SdT skizziert. Der aktuelle Stand der obersorbischen Spracherkennung, Überblick und Beschreibung des Vorgängerprojekts und der veröffentlichten Ergebnisse. Strategiepapier mit Überlegungen zur möglichen Sprachanwendung, Lizenzierung, Transferlernen, Übersetzung mit LLMs und Bedarf an Sprachressourcen.
- Übersicht über die in den bisherigen Projekten gesammelten und erstellten Sprachressourcen in obersorbischer Sprache. Aktueller Stand mit den festgestellten Defiziten und dem Bedarf an zusätzlichen Sprachressourcen.
- Mögliche Technologieverbesserungen, der aktuelle Stand der HSB-ASRT-Technologie und welche SdT-Werkzeuge für die obersorbische Spracherkennung genutzt werden können, z. B. robuste Named Entity Recognition (Pre-, Post-Processing), verbessertes Feature-Engineering und Unterstützung für zeitverzögerte neuronale Netze (TDNNs) im recIKTS. Leitlinien für regelmäßige Technologie-Updates und -Upgrades
- Überblick über die Open-Source-Lösungen für klassische und hybride Spracherkennungsansätze.
- Detaillierte Übersicht über die Möglichkeiten des Ende-zu-Ende Transfer Learning.

Das Dokument schließt mit einer Zusammenfassung der Gründe für die Verwendung von klassischen, hybriden und Ende-zu-Ende-Ansätzen, allgemeinen Empfehlungen und einem Ausblick auf die zukünftige Arbeit und den möglichen Einsatz geeigneter ASR-Technologien für verschiedene Anwendungstypen.