Universität
Zürich UZH

Bachelor's thesis
presented to the Faculty of Arts and Social Sciences
of the University of Zurich
for the degree of
**Bachelor of Arts**

# Automatic Lexical Stress Detection in Isolated English Words

**Author: Vera Bernhard**

Student ID: 17-729-583

Supervisor: PD. Dr. Sandra Schwab

Co-Supervisor: Jean-Philippe Goldman

Department of Computational Linguistics

Submission Date: 01.06.2021

# Abstract

We propose a pipeline for automatic lexical stress detection in isolated English words. It is designed to be part of the computer-assisted pronunciation training application MIAPARLE that aims to improve stress production. The pipeline automatically segments audio input into syllables over which duration, intensity, pitch, and spectral information is calculated. Since the stress of a syllable is defined relative to its neighboring syllables, the values obtained over the syllables are complemented with differential values to the preceding and following syllables. The resulting feature vectors, retrieved from 1011 recordings of single words spoken by English natives, are used to train a Voting Classifier composed of four supervised classifiers, namely a Suppor Vector Machine, a Neural Net, a K Nearest Neighbor, and a Random Forest classifier. The approach classifies stress patterns of a single word with an F1 score of 94% and an accuracy of 96%.

# Zusammenfassung

In dieser Arbeit wird eine Pipeline zur automatischen Betonungserkennung in isolierten englischen Wörtern vorgestellt. Sie ist als Teil der computergestützten Aussprachetrainingsanwendung MIAPARLE konzipiert, die darauf abzielt, die Aussprache von Betonungen zu verbessern. Die Pipeline segmentiert Audio-Inputs automatisch in Silben, über welche Informationen über Dauer, Lautstärke, Tonhöhe sowie Spektralinformationen berechnet werden. Da die Betonung einer Silbe in Relation zu ihren benachbarten Silben definiert ist, werden die über die Silben erhaltenen Werte mit Differenzwerten zu der vorangehenden und nachfolgenden Silben ergänzt. So werden Feature-Vektoren von 1011 Aufnahmen von einzelnen Wörtern, die von englischen Muttersprachler:innen gesprochen wurden, gewonnen. Diese werden wiederum verwendet, um einen Voting Classifier, bestehend aus einer Support Vector Machine, einem Neuronalen Netz, einem Random Forest und einem K Nearest Neighbor Klassifikator, zu trainieren. Mit dieser Vorgehensweise werden Betonungsmuster einzelner Wörter mit einem F1-Score von 94% und einer Accuracy von 96% erkannt.

# Acknowledgement

I would like to thank my supervisors PD Dr. Sandra Schwab and Jean-Philippe Goldman for their continuous guidance and advice over the course of this thesis.

Especially, I am thankful to PD Dr. Sandra Schwab for organizing a presentation slot at the lunch meeting of the Phonetic Laboratory where I received valuable input on my work.

I would also like to express my gratitude to my writing group for the weekly discussions and motivation boosts, as well as for sharing tips and tricks.

Special thanks go to Kilian Werder, Martina Stüssi, Shirin Bär, and Hanno Bertle for proofreading this thesis and their constructive feedback.

Finally, I would also like to thank my friends and family for reassuring me during difficult times and their continuous mental and practical support.

# Contents

# List of Figures

# List of Tables

# List of Acronyms

| | |
|---|---|
| CAPT | Computer-Assisted Pronunciation Training |
| CALL | Computer-Assisted Language Learning |
| dB | Decibel |
| F0 | Fundamental Frequency |
| FN | False Negative |
| FP | False Positive |
| Hz | Hertz |
| L1 | Native Language |
| L2 | Target Language |
| MFCC | Mel Frequency Cepstral Coefficient |
| ST | Semitone |
| SVM | Support Vector Machine |

# 1 Introduction

## 1.1 Motivation

Although language learning websites and apps are very popular, they often lack prosodic training. The term prosody comprises suprasegmental features of speech, meaning phenomena that extend over a single sound such as intonation or stress (Meng et al., 2009). The web application MIAPARLE, developed by Goldman and Schwab (2018), aims to address the latter and hence provides a range of tools to train the perception and production of lexical stress. Several reasons can be named why prosody, or more specifically lexical stress, cannot be disregarded in language education:

Firstly, suprasegmental features are equally affected by negative language transfer as segmental features (Meng et al., 2009). This means that the way prosody is produced in a learner's mother tongue (L1) leads to inaccurate or erroneous production in the target language (L2). Lexical stress, in particular, poses difficulty for speakers whose L1 is a fixed-stress language like French when learning a free-stress language such as English (Goldman and Schwab, 2018). In fixed-stress languages, stress placement is determined by a fixed rule for all words, whereas in free-stress language, stress is assigned according to the complexity of the syllables (van Heuven, 2019). Thus, stress patterns in languages like English vary from word to word and can convey important grammatical information such as a difference in word class (e.g. *'import* - noun vs. *im'port* - verb [1])(Meng et al., 2009).

Apart from contributing to the grammaticality of a word, correct stress patterns also have a considerable impact on the intelligibility (Ferrer et al., 2015) and perceived fluency of a speaker (Li et al., 2017). A study by Anderson-Hsieh et al. (1992), where native English speaker judged the pronunciation of nonnatives, even showed that deviances in prosody in comparison to segmentals and syllable structure was the most likely factor to decrease the perceived fluency of a L2 speaker.

[1] The apostroph signals that the following syllable is stressed.

Lastly, prosody training is proving difficult to be included in a classroom setting as the derivation of easily understandable pronunciation rules is challenging, and hence it would require a teacher to correct the pupils' pronunciation individually (Meng et al., 2009). On the CAPT (computer-assisted pronunciation training) website MIAPARLE, the correction of stress production is conducted by a lexical stress detector. It requests the L2 learner to pronounce a certain word and returns which syllable is stressed and whether this stress pattern is correct. However, at the current state, this procedure is only implemented for Spanish.

Due to the importance of training stress production elaborated above and due to the lack of English language support in the production exercise on MIAPARLE, the aim of this Bachelor thesis is the development of a pipeline to detect stress patterns in isolated English words. Therefore, a data set provided by Schwab and Goldman consisting of isolated English words shall be used to train an automatic stress detector. Consequently, this work will largely focus on feature engineering, in other words, the effective representation of stress, and on exploring different supervised machine learning approaches. The resulting pipeline and the included lexical stress detector should satisfy the following requirements:

- The pipeline should receive an audio file as an input.
- The pipeline should represent the stress pattern of the input by meaningful features.
- The pipeline should include a lexical stress detector trained on the provided data.
- The lexical stress detector should detect stress patterns with comparable success to state of the art approaches.
- The lexical stress detector should employ the features and the machine learning algorithm most successful for its specific task.

## 1.2 Outline

The remaining part of this thesis is structured as follows: Chapter 2 provides the theoretical background for this thesis. It discusses the phenomenon of stress in English from a linguistic perspective, thematizes how stress manifests acoustically and, finally, summarizes common automatic approaches to detect stress. Chapter 3 examines the training data and methods used to, on the one hand, train a lexical stress detector with supervised machine learning and, on the other hand, develop the entire pipeline in which the lexical stress detector will be embedded. In Chapter 4, the best-performing algorithms and features are presented, and Chapter 5 summarizes the results and findings of this thesis. Appendix A lists the software used for developing the pipeline and Appendix B includes the parameters used for grid search.

# 2 Previous Literature

## 2.1 The Phenonemon of English Lexical Stress

Lexical stress[2] describes the phenomenon that in many languages not all syllables within an uttered word are perceived as equally prominent (Cutler, 2005). How this saliency of certain syllables is realized and what its function is, varies depending on the language concerned (Cutler, 2015). In English, as briefly touched upon in the introduction, stress can carry a grammatical function and, according to Cutler (2015) even more importantly, stress is also a crucial perceptual cue to detect word boundaries in speech. In multi-syllabic English words, one syllable is always particularly salient and hence it carries so-called primary stress (Cutler, 2015; van Heuven, 2019). The other syllables also differ in prominence, but the differences are more subtle. Where the primary stress is placed in English depends, as already mentioned, on the syllable structure rather than on a fixed rule, which marks English as a free-stress language (Cutler, 2015; van Heuven, 2019). Furthermore, one must distinguish lexical stress from so-called sentence stress[3]. While lexical stress makes a syllable protrude, sentence stress emphasizes a specific word in a phrase which "[...] contributes new and contextually unpredictable information to the discourse [...]" (van Heuven, 2019). Usually, the acoustic realization of sentence stress falls on the stressed syllable of the word in focus, which means that lexical stress and sentence stress coincide in that word (van der Hulst, 2014; van Heuven, 2019). Depending on the experimental setup, this renders it impossible to discuss one phenomenon separate from the other (cf. Chapter 2.2 and Chapter 3.4).

[2] Also called *Word Stress* in the work of van Heuven (2019).

[3] Also called *Pitch Accent* in the works by Li et al. (2017) and van der Hulst (2014).

## 2.2 Perceptual and Accoustic Correlates of Stress

How stress manifests acoustically is an intensively debated question. However, research agrees that “[S]tress [...] is never marked by a single acoustical property [...]” (van Heuven, 2019). Rather, stress is signaled by a complex interplay of relative changes in several acoustic dimensions (Jenkin and Scordilis, 1996). From the physiological perspective, a stressed syllable stands out from its context by being articulated with “greater physiological effort” (van Heuven, 2019). This increase in effort, according to van Heuven, results in more extreme articulatory movements, which acoustically translates to longer syllable duration. Furthermore, more effort causes rapid changes in the vibration rate of the vocal cords, which corresponds to a rise or fall of vocal pitch. Lastly, the increased air pressure on the glottis is the origin of increased loudness in stressed vowels.

Those physiologically caused acoustic changes in duration, pitch, and loudness were tested in perceptual experiments from the early stages of stress research. Often cited are Fry’s “Experiments in the Perception of Stress” (1958) where duration, intensity (as a measure of loudness), and fundamental frequency (as a measure of pitch) were altered to determine their effect on the perception of stress by English natives. He concluded that longer duration and higher intensity are both reliable cues for a listener to identify a stressed syllable, with duration showing better performance overall. Furthermore, he also confirmed that the relative difference in fundamental frequency (F0) is a similarly effective cue as duration with the stress falling on the syllable with the higher pitch. Succeeding perceptual research predominantly agrees with Fry’s findings that duration and pitch, followed by intensity are the most effective cues (van Heuven, 2019). However, Sluijter and Van Heuven (Sluijter et al., 1997) challenge the view of intensity as a weaker cue by introducing a new correlate of loudness. Instead of measuring the overall intensity, they tested the intensity levels above 500 Hz and discovered that this spectrally filtered measure is more reliable to signal lexical stress.

Apart from the suprasegmental features, duration, fundamental frequency, intensity, and spectral measure, in English particularly, vowels are known to differ in vowel quality depending on stress (Cutler, 2015). Vowels located in a stressed syllable are exclusively in their ‘full’ form, whereas vowels of unstressed syllables are often ‘reduced’, meaning that they tend to be more centrally articulated, but they can also be ‘full’ (Xie et al., 2004). This means vowel quality does not correspond one-to-one to stress but nevertheless has been explored as a correlate of stress (Xie et al., 2004). Opinions on the importance of this segmental distinction for stress judgment are divided: Cutler (2015) defends the opinion that vowel quality is in fact the most significant clue for listeners, whereas research by van Heuven (2019) identifies spectral expansion[4] as one of the weakest perceptual cues.

The results of perceptual studies have motivated many attempts of automatic approaches to detect stress. Overall, the perceptual correlates of stress, duration, pitch, loudness, spectral information, and also vowel quality, were explored in different combinations over the years. One has to note that the most relevant clues for humans might not be the same for automatic, computational approaches (van Heuven, 2019). In the early attempts, simple correlates of duration, loudness, and pitch were applied (Lieberman, 1960; Aull and Zue, 1985; Freij et al., 1990). The work of Aull and Zue (1985) and of Freij et al. (1990) additionally explored some spectral features, namely spectral change in the former and spectral envelope and slope in the latter. Yet, both concluded that loudness and fundamental frequency were their most successful feature, agreeing with Lieberman (1960), and that they outperformed the spectral features. In the research of Jenkin and Scordilis (1996), they first introduced context-aware features by considering durational, loudness, and pitch features from the previous syllable. The idea of contextual features has become popular in the close past (Deshmukh and Verma, 2009; Ferrer et al., 2015; Li et al., 2017).

Including spectral features gained momentum again when Sluijter and van Heuven (1996b) proposed a new spectral measure: They investigated why intensity is classified as a weaker perceptual cue in human stress judgment but is commonly considered a strong correlate of stress with automatic approaches. The reason for this, according to Slujiter and van Heuven, lies in the experimental setup: The algorithms are often trained on isolated words in focus-position, and hence the words also carry sentence stress besides lexical stress. By analyzing words out of focus-position, they found the performance of intensity being significantly less successful. Thus, they came to the conclusion that a higher intensity is mostly caused by sentence stress. Nonetheless, they still defend the view that stressed syllables are louder but instead of affecting the entire spectrum, the increased intensities are mainly located above 500 Hz. This finding was operationalized as the intensity difference in three frequency bands above 500 Hz, which is called spectral balance, and which was found to be close in reliability to duration measures. Spectral balance is adopted in several following works (e.g., Deshmukh and Verma (2009) or Zhao et al. (2011)) and has motivated similar spectral measures (Ferrer et al., 2015).

In another study by Sluijter and van Heuven (1996a) where the relative strengths

[4] A synonym for vowel quality.

of stress correlates were tested on English words, fundamental frequency was also unmasked as an effect of sentence stress rather than lexical stress. Moreover, in the same study, vowel quality was classified as a weak correlate no matter whether sentence stress coincided with lexical stress or not. This finding was also supported by the results of Xie et al. (2004). They compared and combined prosodic and vowel quality features. A combination of loudness, namely amplitude, and duration outperformed vowel quality features. In recent years, spectral information is commonly provided in the form of mel-frequency cepstral coefficients (MFCC), which are reported to outperform the traditional features such as duration or intensity (Ferrer et al., 2015; Tuhola, 2019).

## 2.3 The Task of Automatic Lexical Stress Detection

In the context of CALL, lexical stress detection is defined as identifying the stressed syllable within a spoken word (Tepperman and Narayanan, 2005). This differs from earlier research intended to improve automatic speech recognition, where the decision took place on a single syllable without considering its context within a word (Aull and Zue, 1985; Freij et al., 1990; Jenkin and Scordilis, 1996; Ying et al., 1996). A common approach to classify the stress pattern of an entire word is the following: Classification is still executed on the level of a syllable, but the classifier assigns probabilities to each syllable and chooses the one with the highest score as the location of primary stress. This approach is adopted in the works of Tepperman and Narayanan (2005), Deshmukh and Verma (2009) and Zhao et al. (2011).

Both in the context of speech recognition and of CALL, lexical stress detection is commonly addressed as a supervised machine learning problem. As a consequence, the truth labels, the correct stress patterns, as well as the data, usually syllables, represented as feature vectors need to be provided to the machine learning algorithm. Thus, the first step of preprocessing the training data includes the detection of syllables within the speech recordings. Syllable alignment has not been an easy task and hence was responsible for a good portion of mistakes in early models (Aull and Zue, 1985). In fact, early research often works with data where the syllable boundaries have been annotated manually (Lieberman, 1960; Freij et al., 1990). More recent works rely on state-of-the-art syllable aligners (see for example, EduSpeak in the research of Ferrer et al. (2015)). For the truth label, each syllable also has to be annotated with its stress value. In case that the speech data is produced by native speakers, it is often assumed that they produce the correct stress pattern, and hence the truth labels can be retrieved from a dictionary (Tepperman and Narayanan, 2005; Ferrer et al., 2015). Otherwise, the speech data has to be manually annotated for stress, which is a time-consuming and therefore costly task (Ferrer et al., 2015).

As already mentioned, each syllable is represented by a feature vector which is calculated by means of diverse measures (cf. Chapter 2.2). Even though stress is considered a phenomenon of the entire syllable (van Heuven, 2019), in all previous literature considered in this thesis, the measures are calculated on the vowel portion - the nucleus of the syllable - only. The linguistic reason for this is that "[...] large part of prosodic stress information is carried by the vocalic nucleus [...]" (Silipo and Greenberg, 1999). In practical terms, it is an easier task to detect the boundaries of the syllable nucleus than of the entire syllable (Tepperman and Narayanan, 2005; Li et al., 2017) and it simplifies the normalization procedure (Ying et al., 1996). Normalizing the feature vectors is crucial as "[a]ll of these prosodic features may vary for reasons other than signaling stress" (Xie et al., 2004). The cause for variation might also lie in speaker differences such as speech rate or pitch range, recording setup, or intrinsic differences of speech sounds.

Many different machine learning algorithms have been trained on the feature vectors. Common classifiers include Naive Bayes (Deshmukh and Verma, 2009), Logistic Regression (Deshmukh and Verma, 2009), Decision Trees (Lieberman, 1960; Xie et al., 2004; Deshmukh and Verma, 2009), Gaussian Mixture Models (Tepperman and Narayanan, 2005; Ferrer et al., 2015), Support Vector Machines (Xie et al., 2004; Deshmukh and Verma, 2009; Zhao et al., 2011) and different Neural Network architectures (Jenkin and Scordilis, 1996; Ferrer et al., 2015; Li et al., 2017; Tuhola, 2019). Defining the most successful approach is a difficult task, as the works differ widely in what training data they use, what and in which manner the features are calculated and how the models are evaluated. Only a few studies actually compare different algorithms: Xie et al. (2004) concluded that Support Vector Machines outperform Decision Trees. Decision Trees are also challenged in the work of Ferrer et al. (2015) where Decision Trees and also Neural Networks were exceeded by Gaussian Mixture Models. In contrast, Decision Trees perform well in the work of Deshmukh and Verma (2009), where they achieved better results than Naive Bayes, Logistic Regression, and Support Vector Machines.

# 3 Materials and Methods

## 3.1 Training Data

The training data was collected by Schwab and colleagues[5] and comprises 1012[6] recordings of isolated English nouns spoken by 6 English natives (three female and three male). The 92 different nouns were chosen because, on the one hand, their stress pattern does not differ between English varieties (i.e. British English vs. American English), and, on the other hand, they vary between two to four syllables in length. Following the examples of Tepperman and Narayanan (2005) and Ferrer et al. (2015), it was assumed that natives produced the words correctly, and the stress patterns were retrieved from a dictionary. In total, the training data consists of 3054 syllables of which 67% are unstressed, and 33% are stressed syllables. This means that the two classes are not represented equally, which is an intrinsic issue with stress annotated English data as only one syllable in multisyllabic words carries primary stress (Cutler, 2015; van Heuven, 2019). As will be discussed in Chapter 3.5.1, the resulting imbalance of the data has to be addressed when evaluating a trained model.

## 3.2 General Pipeline Description

Following the previous research on automatic lexical stress detection, in this thesis, a supervised machine learning model was trained on the provided data. As elaborated in Chapter 2.3, several preprocessing steps, namely syllabification, normalization and scaling, and one postprocessing step, ensuring a single stress per word, must be taken, not only when training the model but also when classifying. In Figure 1, an overview of the pipeline, which is intended to be included on MIAPARLE, is depicted: The pipeline receives an audio recording of a single word spoken by a

[5] The collection was financed by the Lehrkredit of the University of Zurich.

[6] 1011 of the recordings were used excluding the recording *S5_LOC_4_4_secretary_2.wav*which was not processable by Prosogram, see Chapter 3.4.

language learner together with the orthographic transcription of the word the learner was asked to produce. Afterwards, with the help of the phonetic transcription, syllables are detected in the speech signal. Each syllable is then represented by a feature vector consisting of measures calculated on the syllable nucleus. To account for variances that are not due to stress placement, the feature vectors are then normalized and scaled before being sent to the classifier. The classifier considers syllable by syllable, assigns each a probability score, and finally detects the primary stress on the syllable with the highest probability. In the following subsections, the preprocessing steps, the feature engineering, and the training of the classifier is examined. An overview of all technical tools used in the pipeline and a link to the code to reproduce the results is provided in Appendix A.

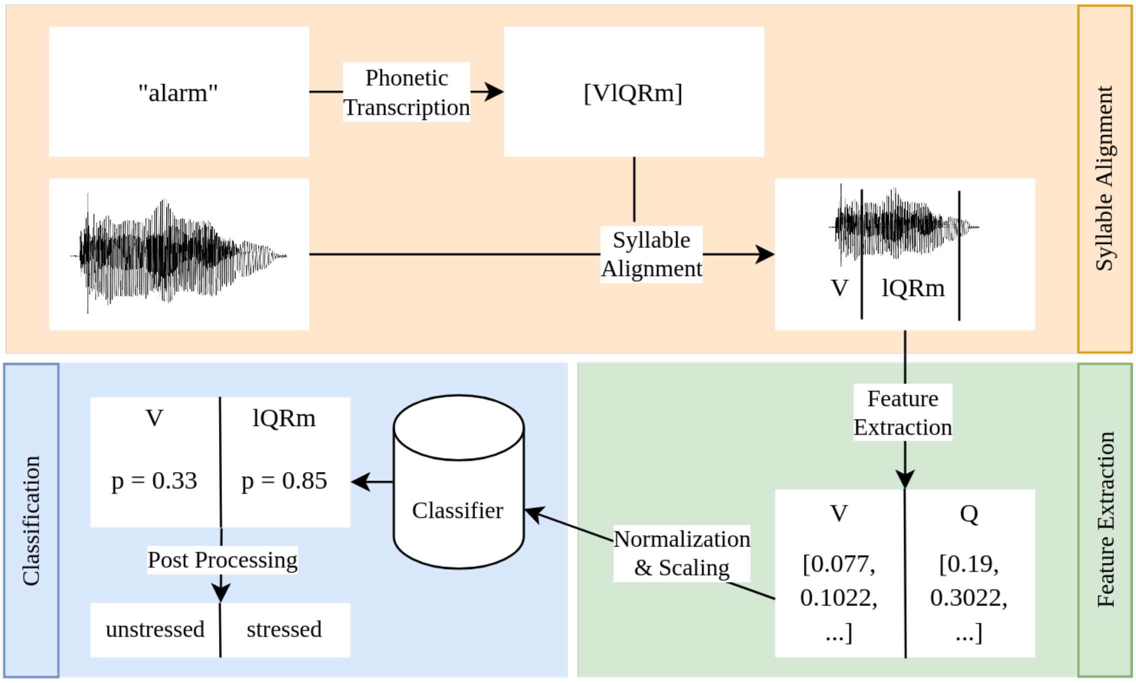

Figure 1: Overview of the pipeline to automatically detect stress in isolated words.

## 3.3 From Audio to Syllable Alignment

Since the MIAPARLE website tests a selected number of words, the phonetic transcriptions were computed once and after that were only retrieved if a given recording is to be classified. The transcription was performed by the 'G2P' (Grapheme to Phoneme) service included in the BAS (Bavarian Archive of Spech Signals) Web Services[7] (Reichel, 2012). All web services of BAS can be accessed via an API. The

[7] https://clarin.phonetik.uni-muenchen.de/BASWebServices/interface

syllable alignment was performed by a pipeline consisting of two other BAS web services: 'MAUS' automatically segmented the audio file into phones given the phonetic transcription (Kisler et al., 2017). Having had the phones aligned, 'PHO2SYL' (Phones to Syllable) grouped the segments into syllables (Reichel, 2012). These preprocessing steps mentioned above resulted in a file in .TextGrid format, which encompassed all syllables phonetically transcribed, accompanied by the timestamps they occur in the recording.

## 3.4 Representing Syllables with Feature Vectors

Each syllable retrieved from the preprocessing was represented by a feature vector. The first measure chosen to be calculated was duration, as it has been undisputedly considered a correlate of stress in previous literature. Additionally, loudness and pitch features were extracted from each syllable even though Sluijter and van Heuven (1996b) showed both to be rather a result of sentence stress than lexical stress. This is justified by the fact that the words in the training data and later on MIAPARLE are pronounced in isolation, hence are in-focus position and, according to Sljuiter and van Heuven's argumentation, carry sentence stress as well. Lastly, motivated by the statement of Sluijter and van Heuven (1996b) "[...] spectral balance is a clear acoustic correlate of stress [...]", spectral information in the form of spectral balance was included.

Spectral information gained by MFCCs was disregarded because their performance drastically decreases with the level of noise (Bhattacharjee et al., 2016) - a complication that cannot be ruled out when recording on MIAPARLE with simple laptop microphones. Moreover, this work did not consider any features representing vowel quality as neither the research of Sluijter and van Heuven (1996a) nor Xie et al. (2004) showed convincing results. In addition, according to Xie et al. (2004) deriving an adequate measure to depict vowel quality is a challenging task in itself. The four feature groups, duration, loudness, pitch, and spectral information, were all calculated on the syllable nucleus[8] for reasons outlined in Chapter 2.3. Subsequently, each feature was passed through a normalization process.

Since a stressed syllable is defined by its differences in acoustic dimensions to the neighboring syllables, this thesis followed the works by Jenkin and Scordilis (1996) and Zhao et al. (2011) and calculated for each feature a set of differential values to the previous and succeeding syllable. In case the syllable stands at the first or last

[8] One exception is the syllable duration which was experimentally included, see Chapter 3.4.1.

position in the word and thus misses a left or right neighbor, the approach of Zhao et al. (2011) was copied: The mean value over all syllables of the feature replaced the missing neighboring feature. It also needs to be mentioned that the normalized values were chosen to calculate the differences between the syllable and its neighbor.

Most of the features were calculated using the Praat[9] extension Prosogram (Mertens, 2020). Prosogram analyses pitch variations in speech and can measure prosodic features per syllable. For this purpose, it places great importance on applying stylization on the pitch to reflect human perception as accurately as possible. Unfortunately, this modeling of human perception posed difficulties: Even though the syllable alignments previously retrieved via the BAS web services could be given to Prosogram as input, it still applied its own heuristics in the given syllable boundaries based on pitch and intensity to detect the vowel nucleus. This resulted in Prosogram not recognizing syllables that were indeed already found in the preprocessing step. In total, in only about 50% of the files, Prosogram detected all syllables. To improve the pitch detection of Prosogram, a Praat script[10] was devised, which applies a strategy by Hirst (2007) to recalculate the pitch with the maximum and minimum F0 values of the 1st and 3rd quartile of the initially calculated pitch range. The portion of files in which Prosogram detected all syllables could be improved to 73%. Recalculating the pitch according to the specific pitch ranges of the six speakers in the training data did not further increase the number of files with correctly aligned syllables. Having still a quarter of files with undetected syllables, a fallback strategy was devised. By means of Praat scripts, all measures that Prosogram calculated and which could easily be emulated were retrieved from the undetected syllables with the help of the syllable boundaries known from the syllable alignment. Praat scripts were also used for extracting features that Prosogram did not provide. It is interesting to note that all the undetected syllables were unstressed. Thus, a feature encoding whether Prosogram detected the syllable or not was added to the feature vector. Based on the fact "[...] that there is a highly significant tendency for stress in English words to fall on the initial syllable [...]"(Cutler, 2015), the position of the syllable within the word was appended to the features vectors as well.

In Table 1, one can see an overview of all features calculated. The features which initially were not calculated by Prosogram are marked in grey. The columns 'Prosogram' and 'Fallback' describe which features were computed by Prosogram and for which a fallback strategy could be implemented. Lastly, the right column encodes for which features the neighboring syllables were taken into account. In the follow-

[9] Praat is open-source software for speech analysis (Boersma and Weenink, 2021).

[10] Based on a script by Sandra Schwab and the 'Extract pitch' script from the Praat Vocal Toolkit Corretge (2020).

ing subsections, a brief overview of the implementation of duration, loudness, pitch, and spectral features is provided.

| | | Prosogram | Fallback | Context |
|---|---|---|---|---|
| Duration Features | Syllable Duration | x | x | x |
| | Nucleus Duration | x | x | x |
| Loudness Features | Root Mean Square Amplitude | ▨ | ▨ | x |
| | Peak Intensity | x | x | x |
| Spectral Features | Mean Intensity 500-1000 Hz | ▨ | ▨ | x |
| | Mean Intensity 1000-2000 Hz | | | x |
| | Mean Intensity 2000-4000 Hz | | | x |
| Pitch Features | Trajectory | x | | |
| | Max F0 in Hz | x | x | x |
| | Max F0 stylized in HZ | x | | x |
| | Mean F0 in Hz | x | x | x |
| | Mean F0 in ST | x | x | x |
| | Intersyllab | x | | |
| Other Features | Position of Syllable | ▨ | ▨ | |
| | Detected by Prosogram | | | |

Table 1: Overview of all features calculated per syllable.

### 3.4.1 Durational Features

As durational features, the nucleus and syllable duration were extracted. To reduce the effects of the individual speech rate and to counteract the fact that the final syllable of an utterance tends to be longer[11], the nucleus duration and the syllable duration were normalized by dividing it through the mean nucleus respectively syllable duration of the given word (Zhao et al., 2011; Sluijter and van Heuven, 1996b). Vowels that constitute the syllable nucleus vary in length according to intrinsic differences, as pointed out by Ying et al. (1996) and Xie et al. (2004) (e.g. diphthongs are longer than ordinary vowels regardless of stress placement). Therefore, the average duration of each phoneme was calculated on the entire dataset, and each nucleus duration was normalized further by dividing it by the corresponding mean vowel length. Both values, normalized for speech rate and normalized for vowel type, were added. Feature engineering during the training of the machine learning model decides which normalization strategy is more successful.

[11] A phenomenon called ‘preboundary lengthening’.

### 3.4.2 Loudness Features

To depict loudness, measures describing the energy in Decibel (dB) and amplitudes in Pascal have been used in previous literature. This thesis applied an amplitude feature, namely the root mean square amplitude motivated by Silipo and Greenberg (1999) and Xie et al. (2004), and peak intensity motivated by van Heuven (2019). The mean values for both measures retrieved from all nuclei of a word were subtracted from the value per syllable to normalize for speaker differences and varying recording setup.

### 3.4.3 Pitch Features

Following several previous studies, peak and mean fundamental frequency were calculated over the syllable nucleus (Aull and Zue, 1985; Jenkin and Scordilis, 1996; Silipo and Greenberg, 1999; Xie et al., 2004; Tepperman and Narayanan, 2005; Deshmukh and Verma, 2009; Zhao et al., 2011). Since Prosogram also provided maximal F0 stylized and mean F0 in semitones (ST), these features were incorporated as well. The selection of a measure in semitones was also motivated by Zhao et al. (2011). The maximum and mean F0 features were normalized by subtracting the corresponding average value over all nuclei of a word. Apart from average and peak F0, several different measures related to the movement of F0 in a syllable were proposed in previous literature. For example, Freij et al. (1990) used derivates of the F0 curve over a syllable or Li et al. (2017) used a pair of dynamic pitches, describing rises and falls of tone. Inspired by this, Prosogram's 'trajectory' features describing "the sum of absolute pitch interval of tonal segments [...] after stylization" (Mertens, 2020) was added as well. Apart from default context-aware features (cf. Chapter 3.4), the difference in semitones between the end of the previous nucleus and the start of the current nucleus was calculated using Prosogram (called 'intersyllab').

### 3.4.4 Spectral Features

As described in Chapter 3.4, this thesis adopts spectral balance inspired by Sluijter and van Heuven (1996b). Therefore, the mean intensity in three frequency bands, namely between 500-1000 Hz, 1000-2000 Hz, and 2000-4000 Hz was determined.

## 3.5 Developing a Lexical Stress Detector with Supervised Machine Learning

### 3.5.1 Training the Classifiers

Since a single best algorithm could not be identified in previous literature (cf. Chapter 2.3), eight different classifiers were tested on the training data. Inspired by previous literature, Decision Trees, Logistic Regression, Support Vector Machines, Naive Bayes and a simple Neural Network architecture, more precisely a Multilayer Perceptron, were applied. Since the Python library Scikit-Learn (Pedregosa et al., 2011) was used for implementing all machine learning models, a K Nearest Neighbor classifier and two Ensemble classifiers, namely Random Forest and AdaBoost, were also trained as recommended in the Scikit-Learn Documentation (2021). Since most machine learning algorithms are sensitive to different scales of the data, the training data was scaled with Z-score normalization, which means that the features were rescaled, such that their mean is zero and their standard deviation is one (Géron, 2019).

To evaluate the classifiers, 10-fold cross-validation was performed. This means that the training data was split ten times into two non-overlapping sets (75/25 ratio). In each case, on the larger part - the training set - the classifiers were trained, and on the other part, they were evaluated. The average performance of the classifiers over all splits was considered. This procedure makes sure that one yields reliable performance measures as it rules out that the division into test and training set is by chance more favorable to the classifier and hence responsible for the good performance (Géron, 2019). When classifying the test data to evaluate the trained model, a postprocessing step as in previous research, such as the works of (Tepperman and Narayanan, 2005; Deshmukh and Verma, 2009; Zhao et al., 2011), was included to ensure that only one stressed syllable per word is detected (cf. Chapter 2.3). Both when no syllable or several syllables are recognized as stressed, the syllable with the highest probability is classified as stressed. In this thesis, the evaluation relied on F-Score (also called F1 score) instead of the common accuracy as the evaluation metric. This decision was made on the basis of the following two considerations: On the one hand, as discussed in Chapter 3.1, stress annotated English data is always imbalanced because there can be one syllable carrying primary stress at most. With our data, 2/3 of the syllables are unstressed. This means that if a classifier disregards the class ‘stressed’ entirely and only assigns the class ‘unstressed’, the accuracy would still reach 60%. For the task of lexical stress detection, however, it is essential to ensure that the class ‘stressed’ is reliably detected. On the other hand, since the lexical stress detector devised in this thesis is part of a CALL application, it should avoid false corrections (Ferrer et al., 2015). Hence the model should minimize false negatives (stressed syllables that are detected as unstressed) and false positives (unstressed syllables that are detected as stressed). If we look at the definitions of F1 and accuracy in Figure 2 Géron (2019), F1 increases as false negatives (FN) and false positives (FP) decrease, whereas accuracy does not depend on the two counts. Accordingly, F1 is the more adequate evaluation measure to evaluate models in this thesis.

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} = \frac{TP}{TP + \frac{FN+FP}{1}}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Figure 2: Definition of F1 score and accuracy.

### 3.5.2 Improving the Classifiers

From the eight classifiers, the four achieving the highest F1 score were chosen to be further investigated. This included, on the one hand, feature engineering - to discover the best performing feature combinations - and, on the other hand, optimizing the hyperparameters of the remaining four models. In a scenario of unlimited time and resources, both dimensions, features and hyperparameters, should be investigated parallelly as the chosen features might influence the optimal choice of hyperparameters and the other way round (Kuhn and Johnson, 2019). However, considering that time was limited for this project and that four models had to be optimized, the procedure was simplified: Since feature engineering is known to have a bigger impact on the model performance, it preceded hyperparameter optimization (Lee, 2019). Therefore, the performance of different feature group combinations was analyzed by training the four algorithms with only a certain feature group. Within a feature group, it was manually experimented whether removing potentially correlating features (e.g. maximum Pitch in ST and Hz) and different normalization strategies improved the model performance. After that, a more systematic grid search on the four models was performed to retrieve the best hyperparameters. Again, F1 was the evaluation measure for which the grid search was optimized. As a last optimization experiment, the four improved classifiers were combined into a Voting Classifier. The idea of a Voting Classifier comprises that all four classifiers assign probabilities to an item, in our case a syllable, and the average overall probability defines which class is associated with the item (Géron, 2019). Voting Classifiers are known to outperform the individual classifiers they are composed of (Géron, 2019).

# 4 Results and Discussion

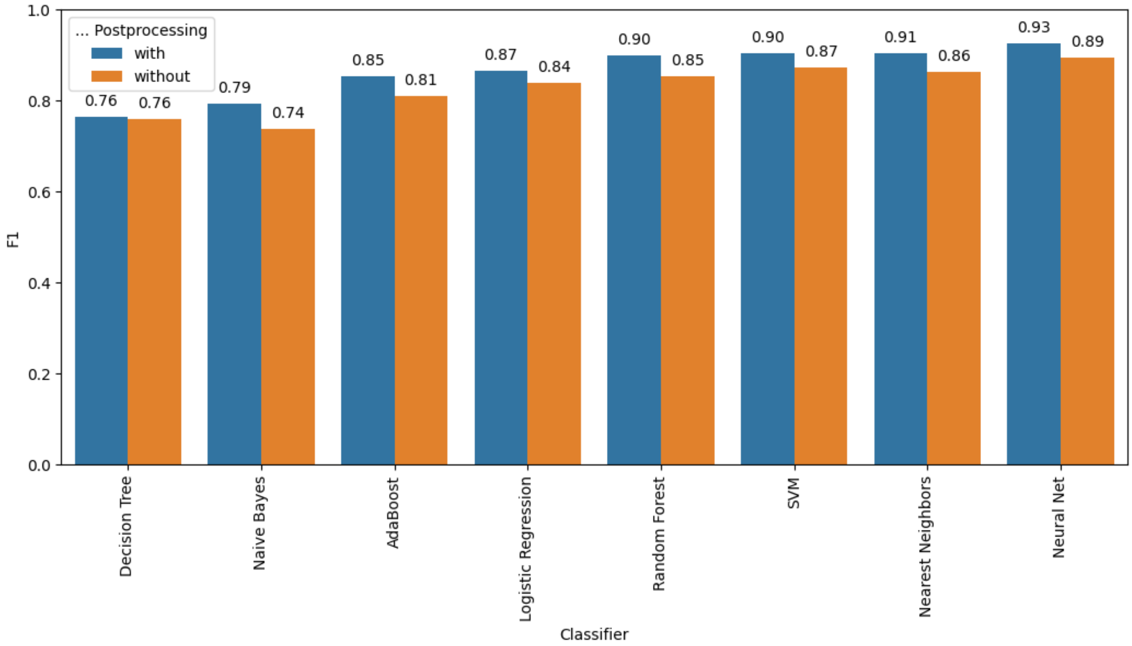

Figure 3: Baseline performance of all classifiers.

As can be seen in Figure 3, the Neural Network, Support Vector Machine, K Nearest Neighbor and Random Forest classifiers performed the best when implemented out of the box with all features. These results contradict the results of Deshmukh and Verma (2009) where Decision Trees outperformed Naive Bayes, Logistic Regression, and Support Vector Machine; with the presented work, Decision Trees show the worst performance. Thus, it rather confirms the conclusion of Xie et al. (2004) that Support Vector Machines surpass Decision Trees in detecting lexical stress. The success of the postprocessing strategy can be seen in the same figure, as each algorithm reached a higher F1 score when postprocessing was applied, which reflects the findings of Tepperman and Narayanan (2005).

## 4.1 Performance of Different Features

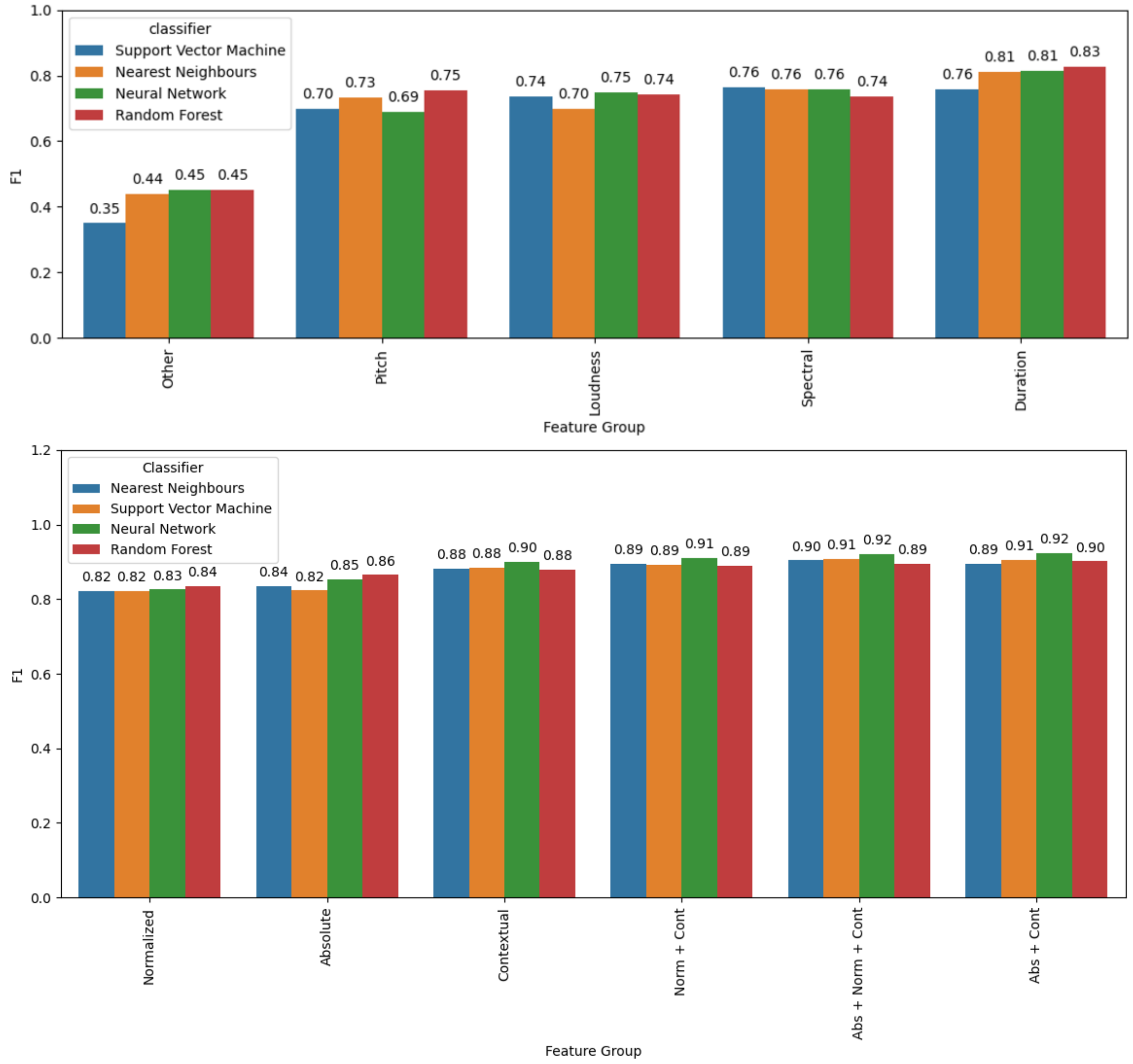

Figure 4: Performance of feature groups. Top: Features are grouped according the acoustic dimensions they cover. ‘Other’ includes ‘position of syllable’ and ‘detected by prosogram’. Bottom: Features are grouped according to how they are calculated.

From all features, the durational features proved to be the most successful (see Figure 4 top). This ties in with the fact that durational measures were the uncontested feature in previous literature. The exploratory features, ‘position of syllable’ and ‘detected by prosogram’ (both summarized in ‘Other’), are the weakest correlates of stress. Hence, they were removed, which did not have a negative effect on the performance metric. The strength of pitch, loudness, and spectral features depends on which classifier is considered. For the Support Vector Machine, K Nearest Neighbor, and Neural Network, spectral features led to a better performance than pitch and loudness features. With the Random Forest, the pitch features surpassed loudness and spectral features. Nevertheless, the combination of pitch, loudness, spectral and, duration features yielded the best performance with all classifiers.

The juxtaposition of absolute, normalized and, context-aware features (see Figure 4 bottom) revealed that in agreement with the work of Zhao et al. (2011), the context-aware feature group outperformed the normalized and absolute features if the groups are considered on their own. This finding is supported by the fact that stress is defined as a phenomenon on the level of the syllable in relation to its neighboring syllables. It is, however, surprising that normalized features performed worse than the absolute features. This may be due to the fact that spectral features, which were a strong correlate of stress, are not part of the normalized feature groups since this paper did not adapt any normalization strategy for them. The best F1 score was reached with either the combination of absolute, normalized, and context-aware features or absolute and context-aware features. Correspondingly, both combinations were tested when further optimizing the classifier.

Finally, removing some similar (e.g. vowel normalized vs. speech rate normalized duration features) or collinear features (e.g. F0 mean in Hz vs. F0 mean in ST) was explored. The removal of any feature led to a decrease in performance. As a consequence, all features were considered (apart from 'position of syllable' and 'detected by prosogram' which had been removed earlier).

## 4.2 Optimization of Classifiers

Absolute + Context-Aware Features

| | Before | After |
|---|---|---|
| Random Forest | 0.90 | 0.90 |
| SVM | 0.91 | 0.93 |
| Nearest Neighbor | 0.89 | 0.93 |
| Neural Net | 0.92 | 0.92 |
| Voting | 0.94 | |

Absolute + Nomalized + Context-Aware Features

| | Before | After |
|---|---|---|
| Random Forest | 0.89 | 0.89 |
| SVM | 0.91 | 0.93 |
| Nearest Neighbor | 0.90 | 0.93 |
| Neural Net | 0.92 | 0.93 |
| Voting | 0.94 | |

Table 2: Performance in F1 scores of classifiers before and after grid search with different feature combinations. The Voting Classifier is composed of all four optimized classifiers but was not grid searched itself (no before/after).

Most classifiers improved slightly after grid search (see Appendix B for an overview of the grid search parameters and Table 2 for the performance results) both when trained on absolute and context-aware features, as well as absolute, normalized and context-aware features. As described in Chapter 3.5, the optimized classifiers were combined into a Voting Classifier. The resulting Voting Classifiers, as predicted by Géron (2019), outperformed the single classifiers. Unfortunately, no definite conclusion between the two feature sets could be drawn; both resulting classifiers reached an F1 score of 0.94. Since a model with less features is usually preferred because less time for training is required, the Voting Classifier trained on absolute and context-aware features was chosen to work in the lexical stress detection pipeline. Therefore, the final model achieves an F1 score 94% and an accuracy of 96%. As shown in Figure 5, the classifier performs slightly better at detecting unstressed than stressed syllables (97% of unstressed and 94% of stressed syllables are detected correctly). This can probably be accredited to the fact that unstressed syllables are overrepresented in the training data, as described in Chapter 3.1.

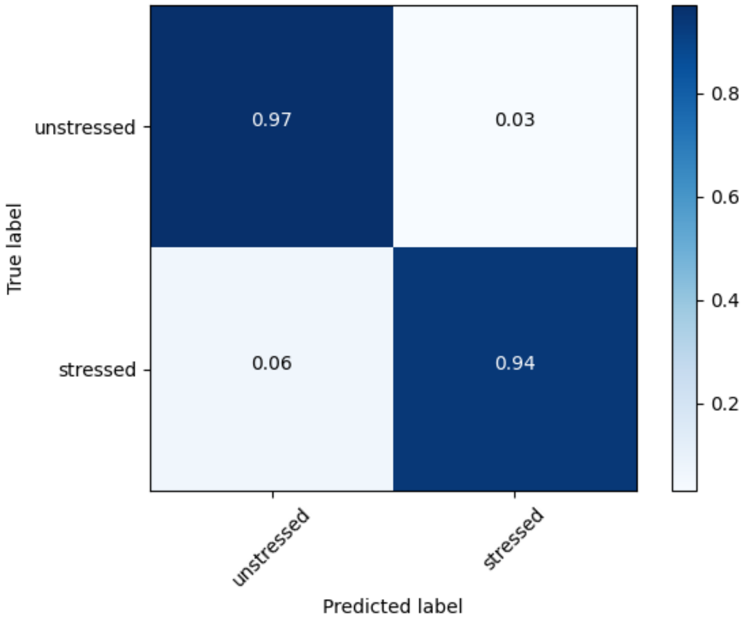

Figure 5: Normalized confustion matrix of the Voting Classifier.

# 5 Conclusion and Outlook

This thesis proposes a pipeline that successfully detects syllables in recordings of isolated English words and identifies the syllables carrying primary stress. For syllable alignment, three different tools from BAS Web Services were exploited. The lexical stress detector was trained with a supervised machine learning algorithm: A Voting Classifier composed of four different classifiers, including a Support Vector Machine, a Neural Net, a Random Forest and a K Nearest Neighbor classifier, achieved a 94% F1 score and 96% accuracy on the provided training data. With such high scores, the classifier exhibits comparable performance to recent approaches in lexical stress detection (e.g. Ferrer et al. (2015): 91.5% accuracy, Zhao et al. (2011): 88.6% accuracy[12]). The final classifier employed absolute and context-aware duration, pitch, loudness, and spectral features, which in combination led to the best performance. The durational features proved to be the most reliable feature group to detect lexical stress in this task setup. Apart from feature selection, a postprocessing step that ensured that only one stressed syllable per word is found contributes substantially to the quality of the classification. In sum, all requirements mapped out in Chapter 1.1 were met in the course of this thesis.

Future work should investigate how the proposed pipeline performs on non-native and noisy data since this was not evaluated in the presented thesis but is the use case scenario on MIAPARLE. Moreover, the proposed pipeline only detects primary stress. Therefore, the detection of different degrees of stress could be a topic of future research. Furthermore, the pipeline could also be improved if mispronunciation of a language learner on MIAPARLE is taken into consideration. At the current state, if a vowel is mispronounced, it is still force aligned with the correct pronunciation in the preprocessing steps. Concerning the training, more time could be spent on systematic and parallel feature engineering and hyperparameter optimization to evaluate whether the performance of the system can be increased even further. Lastly, exploring Deep Neural Network architectures, which are able to represent time series[13], may constitute the object of future studies.

[12] Both works consider secondary stress as well, which is probably the reason for the lower accuracy.

[13] For example, recurrent neural networks (RNN) as in Tuhola (2019).

# References

Anderson-Hsieh, J., Johnson, R., and Koehler, K. (1992). The relationship between native speaker judgments of nonnative pronunciation and deviance in segmentais, prosody, and syllable structure. *Language Learning*, 42(4):529–555.

Aull, A. M. and Zue, V. W. (1985). Lexical stress determination and its application to large vocabulary speech recognition. In *ICASSP'85. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 10, pages 1549–1552. IEEE.

Bhattacharjee, U., Gogoi, S., and Sharma, R. (2016). A statistical analysis on the impact of noise on mfcc features for speech recognition. In *2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, pages 1–5. IEEE.

Boersma, P. and Weenink, D. (2021). Praat: doing phonetics by computer [Computer program]. Version 6.1.38, retrieved 2 January 2021 `http://www.praat.org/`.

Corretge, R. (2012-2020). Praat vocal toolkit. `http://www.praatvocaltoolkit.com`.

Cutler, A. (2005). Lexical stress. In Pisoni, D. B. and Remez, R. E., editors, *The handbook of speech perception*, Blackwell Handbooks in Linguistics, chapter 11, pages 264–289. Blackwell, Malden, Mass.

Cutler, A. (2015). Lexical stress in english pronunciation. In *The Handbook of English Pronunciation*, Blackwell Handbooks in Linguistics, pages 106–124. Wiley Blackwell, Malden, Mass.

Deshmukh, O. D. and Verma, A. (2009). Nucleus-level clustering for word-independent syllable stress classification. *Speech Communication*, 51(12):1224–1233.

Ferrer, L., Bratt, H., Richey, C., Franco, H., Abrash, V., and Precoda, K. (2015). Classification of lexical stress using spectral and prosodic features for computer-assisted language learning systems. *Speech Communication*, 69:31–45.

Freij, G., Fallside, F., Hoequist Jr, C., and Nolan, F. (1990). Lexical stress estimation and phonological knowledge. *Computer Speech & Language*, 4(1):1–15.

Fry, D. B. (1958). Experiments in the perception of stress. *Language and Speech*, 1(2):126–152.

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly Media.

Goldman, J.-P. and Schwab, S. (2018). Miaparle: Online training for the discrimination of stress contrasts. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018).*

Hirst, D. J. (2007). A praat plugin for momel and intsint with improved algorithms for modelling and coding intonation. In *Proceedings of the XVIth International Conference of Phonetic Sciences*, volume 12331236, pages 1223–1236.

Jenkin, K. L. and Scordilis, M. S. (1996). Development and comparison of three syllable stress classifiers. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, volume 2, pages 733–736. IEEE.

Kisler, T., Reichel, U., and Schiel, F. (2017). Multilingual processing of speech via web services. *Computer Speech & Language*, 45:326–347.

Kuhn, M. and Johnson, K. (2019). *Feature engineering and selection: A practical approach for predictive models.* CRC Press.

Lee, A. (2019). Why you should do feature engineering first, hyperparameter tuning second as a data scientist. `https://towardsdatascience.com/why-you-should-do-feature-engineering-first-hyperparameter-tuning-second-as-a-data-scientist-334be5eb276c`. Accessed: 2021-04-04.

Li, K., Wu, X., and Meng, H. (2017). Intonation classification for l2 english speech using multi-distribution deep neural networks. *Computer Speech & Language*, 43:18–33.

Lieberman, P. (1960). Some acoustic correlates of word stress in american english. *The Journal of the Acoustical Society of America*, 32(4):451–454.

Meng, H., Tseng, C.-y., Kondo, M., Harrison, A., and Viscelgia, T. (2009). Studying l2 suprasegmental features in asian englishes: a position paper. In *Tenth Annual Conference of the International Speech Communication Association*.

Mertens, P. (2020). *Prosogram User's Guide*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Reichel, U. D. (2012). PermA and Balloon: Tools for string alignment and text processing. In *Proc. Interspeech*, Portland, Oregon.

Scikit-Learn Documentation (2021). Choosing the right estimator. `https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html`. Accessed: 2021-05-29.

Silipo, R. and Greenberg, S. (1999). Automatic transcription of prosodic stress for spontaneous english discourse. In *Proc. of the XIVth International Congress of Phonetic Sciences (ICPhS)*, volume 3, page 2351.

Sluijter, A. M. and van Heuven, V. J. (1996a). Acoustic correlates of linguistic stress and accent in dutch and american english. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, volume 2, pages 630–633. IEEE.

Sluijter, A. M. and van Heuven, V. J. (1996b). Spectral balance as an acoustic correlate of linguistic stress. *The Journal of the Acoustical society of America*, 100(4):2471–2485.

Sluijter, A. M., van Heuven, V. J., and Pacilly, J. J. (1997). Spectral balance as a cue in the perception of linguistic stress. *The Journal of the Acoustical Society of America*, 101(1):503–513.

Tepperman, J. and Narayanan, S. (2005). Automatic syllable stress detection using prosodic features for pronunciation evaluation of language learners. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I–937. IEEE.

Tuhola, M. (2019). English lexical stress recognition using recurrent neural networks. Master's thesis, Tampere University.

van der Hulst, H. (2014). The study of word accent and stress: past, present, and future. In van der Hulst, H., editor, *Word Stress: Theoretical and Typological Issues*, pages 3–55. Cambridge University Press.

van Heuven, V. (2019). Acoustic correlates and perceptual cues of word and sentence stress: towards a cross-linguistic perspective. In Goedemans, R. G., Heinz, J., and van der Hulst, H., editors, *The study of word stress and accent: theories, methods and data*, pages 15–59. Cambridge University Press, Cambridge.

Xie, H., Andreae, P., Zhang, M., and Warren, P. (2004). Detecting stress in spoken english using decision trees and support vector machines. In *Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation-Volume 32*, pages 145–150. Citeseer.

Ying, G. S., Jamieson, L. H., Chen, R., Michell, C. D., and Liu, H. (1996). Lexical stress detection on stress-minimal word pairs. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, volume 3, pages 1612–1615. IEEE.

Zhao, J., Yuan, H., Liu, J., and Xia, S. (2011). Automatic lexical stress detection using acoustic features for computer assisted language learning. *Proc. APSIPA ASC*, pages 247–251.

# A Software and Code

The code to replicate the results and the finally trained model can be found on GitHub:

`https://github.com/vera-bernhard/stress-detector`.

In addition, the following software was used:

- Praat[14](Boersma and Weenink, 2021)
- Prosogram[15](Mertens, 2020)
- G2P(Reichel, 2012) and MAUS_PHO2SYL (WebMaus Basic(Kisler et al., 2017) + PHO2SYL(Reichel, 2012)) from the BAS Web Services[16]
- Praat Vocal Toolkit[17], adapted 'Extract pitch' script to stylize pitch (Corretge, 2020)
- Scikit-Learn (Pedregosa et al., 2011)

[14] `https://www.fon.hum.uva.nl/praat/`

[15] `https://sites.google.com/site/prosogram/home`

[16] `https://clarin.phonetik.uni-muenchen.de/BASWebServices/interface`

[17] `http://www.praatvocaltoolkit.com/index.html`

# B Hyperparameters for Grid Search

Table 3 provides an overview of all hyperparameters considered for grid search. The values marked in blue are the ones that achieved the best performance with feature set 1 (absolute and context-aware features), values marked in red with feature set 2 (absolute, normalized and context-aware features) and orange when they performed best for both feature sets.

| Classifier | Hyperparameter | Values |
|---|---|---|
| Neural Net | Hidden Layer Size | {(100), (50, 50), (100, 50), (50)} |
| | Activation Function | {identiy, logistic, tanh, relu} |
| | Optimizer (solver) | {lbfgs, sgd, adam} |
| | Learning Rate (alpha) | {0.001, 0.0001, 0.00001} |
| K Nearest Neighbor | Number of Neighbors | {3, 5, 9, 13} |
| | Weight Function | {uniform, distance} |
| | Algorithm | {auto, ball_tree, kd_tree, brute} |
| | Distance Metric | {euclidian, manhattan} |
| Support Vector Machine | Regularization Parameter (C) | {0.01, 0.1, 1.0, 10.0} |
| | Kernel | {linear, rbf, poly, sigmoid} |
| | Kernel Coefficient (gamma) | {scale, auto} |
| | Class Weight | {balanced, None} |
| Random Forest | Number of Trees (Nr or Estimators) | {50, 100, 200} |
| | Quality of Split Measure (criterion) | {gini, entropy} |
| | Maximal Depth | {10, 50, 100, None} |
| | Minimum Samples for Split | {2, 5, 10} |
| | Class Weight | {balanced, None} |

Table 3: Hyperparameters used in grid search for each classifier.