MACHINE LEARNING APPROACHES TO IMPROVING MISPRONUNCIATION DETECTION ON AN IMBALANCED CORPUS

BY

XUESONG YANG

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Adviser:

Professor Mark Hasegawa-Johnson

ABSTRACT

This thesis reports the investigations into the task of phone-level pronunciation error detection, the performance of which is heavily affected by the imbalanced distribution of the classes in a manually annotated data set of non-native English (Read Aloud responses from the TOEFL Junior Pilot assessment). In order to address problems caused by this extreme class imbalance, two machine learning approaches, cost-sensitive learning and over-sampling, are explored to improve the classification performance. Specifically, approaches which assigned weights inversely proportional to class frequencies and synthetic minority over-sampling technique (SMOTE) were applied to a range of classifiers using feature sets that included information about the acoustic signal, the linguistic properties of the utterance, and word identity. Empirical experiments demonstrate that both balancing approaches lead to a substantial performance improvement (in terms of `f1_score`) over the baseline on this extremely imbalanced data set. In addition, this thesis also discusses which features are the most important and which classifiers are most effective for the task of identifying phone-level pronunciation errors in non-native speech.

To my wife, parents, and grandparents
for their love and support.

ACKNOWLEDGMENTS

I would express my sincere gratitude to my advisor Professor Mark Hasegawa-Johnson for the continuous support of my PhD study and related research, for his motivation and immense knowledge. I would like also thank Mark for his patience especially when I had a hard time understanding the concepts and ideas during our individual meetings. His guidance helped me in all aspects of researching and writing of this thesis.

Besides my advisor, I would like to thank my mentors, Dr. Anastassia Loukina and Dr. Keelan Evanini, who provided me an opportunity to work as an intern in the Speech and NLP research group in Educational Testing Service (ETS). I am indebted to them for their support of the flexible research topics of my interests during my internship at ETS, for the discussions and insightful comments that helped me to focus on this topic, and for the encouragement that boosted my confidence in continuing to pursue the PhD degree. My sincere thanks also goes to my fellow interns, Keisuke Sakaguchi (Johns Hopkins University), Noura Farra (Columbia University), and Nils E. Murrugarra Llerena (University of Pittsburg) for many stimulating discussions that gave me incentives to broaden my visions other than speech signal processing by practical machine learning tricks and natural language processing in education, and for all the fun we had together during that 12 weeks. I also thank Su-Youn Yoon, Daniel Blanchard, Nitin Madnani, and Lei Chen for their help in using ETS internal tools, open sourced Python library SciKit-Learn Laboratory (SKLL), and valuable comments on my workshop paper.

Last but not the least, I appreciate that ETS agreed to share data with me for the purpose of completing my master thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Computer-assisted pronunciation training system (CAPT), as an interactive medium for non-native speakers to learn the second languages (L2), has attracted considerable attention from research communities of speech signal processing and applied linguistics in recent years [1]. This multidisciplinary research aims to develop such an effective way of enhancing the speaking skills of L2 learners that provides an accurate assessment of the pronunciation proficiency of L2 learners as well as detailed diagnostic information on segmental mispronunciations such as insertion, substitution or deletion of a specific pronunciation unit (phoneme), and suprasegmental errors such as pitch, duration, syllable stress.

In the context of pronunciation training systems, most previous work has focused on pronunciation error detection. Specifically, the goal of such systems would be to identify persistent errors in a non-native speaker's speech and to suggest directions for further training. In this thesis, on the other hand, we consider the task of pronunciation error detection in the context of large-scale assessment of English proficiency. While both assessment and training ultimately pursue similar aims, there are several restrictions posed by language assessment that need to be taken into account while designing the error detection system. For example, fairness considerations require that the system should not apply different criteria to test-takers with a different native language (L1). This means that assessment systems may not use the information about test-takers' L1 to determine prior probabilities of error patterns as is frequently done in pronunciation training systems.

For our system, we cast the task of pronunciation error detection as a binary classification problem based on a set of features consisting of acoustic information, word identity and linguistic information. In nearly all cases, the number of phones labeled as pronunciation errors in the corpus is very small in comparison to the number of phones labeled as correct; this heavily skewed class distribution leads to challenges in modeling and evaluation. We investigate two common approaches (cost-sensitive learning and sampling) to mitigate problems caused by the extremely imbalanced distributions. We also analyze the robustness of classifiers across different phones and discuss which classifiers would be most effective in the context of a practical CAPT or language assessment system. Besides the performance of phone-level error detection, we concern ourselves with word-level pronunciation errors as well, which would be triggered when any one of the phones in canonical pronunciation of the word is identified as an error.

The remainder of this thesis is organized as follows: Chapter 2 describes related works and state-of-the-art techniques in pronunciation error detection research, and introduces the problems of imbalanced learning. Chapter 3 describes the corpus with manual annotations of pronunciation errors that is used in this study. Chapter 4 presents the features that are extracted for identifying pronunciation errors, details of imbalanced learning approaches, and a searching algorithm of best feature selection for individual phones. Experiments of pronunciation error detection at both phone-level and word-level are illustrated in Chapter 5. This thesis concludes with discussions and future works in Chapter 6.

CHAPTER 2

RELATED WORKS

With the recent advance in automatic speech recognition (ASR) research, pronunciation error identification techniques based on ASR frameworks have been proposed for computer-assisted language learning (CALL) systems. These systems typically detect segmental mispronunciations (or say phone-level errors) from the second language (L2) learner's read speech of prompted texts with the help of an ASR decoder, and then they provide detailed diagnostic feedbacks to L2 learners [2].

For the pronunciation training purposes, two types of ASR-based pronunciation error detection techniques have been widely applied recently. The *rule-based approach* uses extended pronunciation confusion networks that include both canonical pronunciations and their mispronounced variants to capture the possible error types [3–7]. The other is the *confidence score-based approach* that measures the similarity between the realization of a given phone by L2 learners and its canonical pronunciation by native speakers [8–12].

It is well known that hidden Markov models (HMMs) are not powerful enough to discriminate sounds that are spectrally similar and differ mainly in duration. Besides, HMMs are also not quite suitable to distinguish fricatives from plosives, for example, since the difference between these two sounds is subtle in the amplitude envelope of the sound [13]. Therefore, another line of this research is casting the identifying pronunciation errors as a binary classification task [14–17].

2.1 Rule-Based Approach

One common *rule-based approach* is to utilize prior knowledge of mispronunciation patterns extracted from a large corpus of L2 speech (see [18]). Based on rules and statistical generalizations extracted from these mispronunciation patterns, the system's pronunciation dictionary can be extended to include mispronunciations, and prior probabilities can be added for their variants. The pronunciation error detection task then consists of identifying which realization of a given word occurred in a speaker's response. This can be done by using algorithms built into the automatic speech recognition engine to select the most probable variant [2]. For this method with extended pronunciation networks, the performance largely depends on the capability of capturing all possible errors made by L2 learners. If the coverage of the network is too small, any error that is not included in the lexicon will never be detected. Studies also show that the detection results by simply comparing the acoustic likelihoods among pronunciation variants do not have good consensus with human evaluators [19, 20]. Therefore, too large a network could significantly increase phone recognition errors and thus causes many false alarms that could confuse L2 learners. Nevertheless, this approach can still produce the best results when the predicted error patterns are customized to a specific combination of L1 and L2. However, as mentioned in Chapter 1, in case of a global and large-scale language assessment with many diverse L1 populations, such a customization may not be possible or desirable.

2.2 Confidence Measure

Another line of research is to identify pronunciation errors based on the similarity between a speaker's realization of a given phone and its target pronunciation based on a native-speaker acoustic model. This method mitigates the effects of confusions raised in the *rule-based approach* because it simply computes a score for each pronunciation to judge if this realization of the sound is correct or not. In many cases, the similarity metrics are based on ASR confidence scores: the lower the confidence score is, the higher the chance that the sound was mispronounced. Various types of confidence measures have been studied in [8]. For instance, the most widely used measure is probably the goodness of pronunciation (GOP) algorithm [11, 21], which has also been adopted in other studies [22, 23]. GOP algorithm calculates the duration-normalized log of the posterior probability that a speaker uttered a specific phone given the acoustic observations.

For the purposes of providing detailed diagnosis information on mispronunciations such as insertion, substitution or deletion of a specific phone, some methods have been proposed to improve the performance of mispronunciation detection with both *rule-based approach* and *confidence measures* together. For example, Harrison et al. [4] use context-sensitive phonological rules to generate an extended lexicon that better characterizes language transfer from L1 to L2. Doremalen et al. [24] improve *confidence score-based* misappropriation detection by taking consideration of non-native error patterns.

2.3 Pronunciation Error Detection as a Classification Task

Recent work on pronunciation error detection has approached the task as a supervised learning problem by training a classifier based on various acoustic parameters which are likely to differ between predicted realizations [25, 26]. While earlier work on GOP algorithms [2, 11, 21] aimed to establish appropriate thresholds for different phones and speakers, subsequent studies have re-cast it as a classification task in which GOP-like measures were combined with additional acoustic and suprasegmental features using different machine learning algorithms. Some of the machine learning methods used in previous studies include logistic regression [27], linear discriminant analysis [25], support vector machines [28], and decision trees [29]. These studies mostly focused on comparing the performance of the same machine learning algorithm on different feature sets, and seldom provided insights into how to choose the most appropriate classifier for the specific task. Yet other work in the field of machine learning has demonstrated that the choice of which classifier to use may have substantial effects on the system's performance due to task-specific strengths and weaknesses of various classifiers [30]. In this thesis, we compare different machine learning algorithms using the same set of features to evaluate whether and to what extent the choice of classifier may affect the performance of the pronunciation error detection models.

2.4 The Problem of Imbalanced Learning

Classification problems involving real-world data are often imbalanced, and have a highly skewed distribution of classes. Despite this, most standard learning algorithms assume a balanced class distribution or equal misclassification costs. Once such algorithms are applied to (extremely) imbalanced data sets, the false acceptance rate tends to increase, because the model does not adequately estimate the true distribution of the classes [31]. Two common approaches that have been investigated to address the problems caused by extremely imbalanced data sets are cost-sensitive learning and sampling methods [31]. Cost-sensitive learning methods assign a relatively high cost to misclassifications of minority class instances and minimize the overall cost, while sampling methods attempt to balance the class distributions by adjusting the relative proportion of instances in the distribution. These sampling-based methods, however, have some potential drawbacks: specifically, under-sampling (removing instances from the majority class) may cause the classifier to ignore important information pertaining to the majority class, whereas an over-sampling approach (duplicating instances from the minority class) may cause "tied" issues and lead to overfitting [32]. The synthetic minority over-sampling technique (SMOTE) [33] was proposed to overcome these issues by generating artificial data based on similarities in the feature space across instances in the minority class.

Pronunciation error detection, cast as a binary classification task, also faces the standard problems that are caused by an imbalanced distribution of classes in the corpus. The number of phones which are labeled as errors by expert annotators is extremely small in relation to the overall number of phones. For example, Doremalen et al. [27] note that they were not able to train classifiers well for a number of phones because of the low percentage of annotated errors for those phones. In this thesis, we investigate the two previously mentioned approaches to mitigate the affects raised by imbalanced data sets, and compare the performance of assigning weights inversely proportional to the class frequencies (Auto-Weighting) and the SMOTE over-sampling methods across a range of different classifiers. In the end, we also empirically evaluate the quality of different classification decisions and suggest the most effective classifier for the task of identifying mispronounced phones.

CHAPTER 3

DESCRIPTION OF THE DATA

This thesis work utilizes the data provided by Educational Testing Service (ETS) that is derived from non-native spoken responses to Read Aloud items during the Test of English as a Foreign Language (TOEFL®) Junior pilot administration.[1] In this chapter, we will describe the development of this corpus and corresponding statistics.

3.1 Corpus Preparation

The corpus of spoken responses used in this study was collected during the pilot administration of an international assessment of English proficiency targeted at middle-school students aged from 11 to 15. It consisted of 178 responses and included native speakers of Korean, Arabic, Spanish and Vietnamese. All speakers were learners of English as a foreign language and resided in non-English speaking countries. Each speaker was asked to read one of the four texts out loud (45 responses for each text). The responses were manually transcribed and were automatically aligned with human transcriptions using the HTK-based Penn forced aligner [34].

This corpus was annotated for pronunciation errors by two linguists who annotated pronunciation errors following the approach from [35] in which raters were asked to identify "the most serious errors to be corrected in the subject's speech". The annotators used their own judgment about what errors should fall under this category; they were provided with the phonetic dictionary (ARPABET symbols [36] are used) transcriptions of each word and were asked to modify the transcriptions for errors that they considered serious enough to break communication. For each text, six files were selected

[1] Disclaimer: The opinions set forth in this publication are those of the author(s) and not ETS. Copyright © 2014-2015 ETS. www.ets.org

for double annotation to test inter-annotator agreement. The remaining files were split between the two annotators using stratified sampling so that each annotator was assigned a similar number of responses for each L1. The files selected for double annotation were interspersed with the other responses, and the annotators were not aware which responses were selected for double annotation.

3.2 Human Inter-Annotator Agreement

On average the annotators corrected about 7% of all phones. This number varied between the phones: for example, 29% of all occurrences of /ð/(DH)[2] were marked as mispronounced, while for /m/(M) this number was just 0.01%.

For the doubly annotated responses, we aligned the transcriptions using edit distance and computed the absolute agreement (% of matching values) and Cohen's kappa (κ) on the phone level for each response. The inter-annotator agreement on the localization of errors varied between items with an average $\kappa = 0.52$ and an average absolute agreement 92%. In addition, the two annotators agreed strongly on the relative number of mispronounced phones in each response with Pearson's $r = 0.9$ ($p = 3 \times 10^{-6}$, $N = 24$) for the number of phones corrected by each annotator per response.

These results compare favorably with inter-annotator agreement results reported in previous studies. For example, [29] reported 80.2% agreement for the localization of phone-level pronunciation errors in a corpus of Spanish. For English, Bonaventura et al. [37] reported 67% agreement on the localization of phone-level errors. Intraclass Correlation Coefficients (ICC) values between 0.29 and -0.56 were reported in [38]. The annotation procedure used in this study consistently produced agreement above these reported values.

To evaluate the validity of our annotations, we computed correlations between the number of words corrected by the annotator and the holistic proficiency score assigned by the first human rater (this holistic score was based on an evaluation of several aspects of English speaking proficiency, including

[2] Phone symbols are IPA (CMUdict in parens), where CMUdict: `http://svn.code.sf.net/p/cmusphinx/code/trunk/cmudict/cmudict-0.7b.phones`

delivery, vocabulary, grammar, and content, and was not limited solely to an evaluation of pronunciation). For responses that were annotated by both annotators we used the mean value of the number of corrections from the two annotations. The overall correlation between the number of corrections and the holistic proficiency score was $r = -0.57$ ($p = 3.02 \times 10^{-22}$, $N = 175$).

CHAPTER 4

METHODOLOGY

In this chapter, we will describe the details of feature extraction including acoustic information, linguistic information and word identity. Machine learning methods consisting of both linear and non-linear classifiers will be introduced. In order to address the issues raised by the imbalanced nature of the training data set, two balancing approaches (Auto-Weighting and SMOTE) are also illustrated. At the end of this chapter, an exhaustive search algorithm of feature selection will explain how to find the optimal feature subset for individual phones in order to maintain similar or better performance of classification.

4.1 Automatic Speech Recognition

Automated assessment of non-native speech relies on automatic speech recognition (ASR) to convert the spoken response to a text transcription [39]. For this study, we used a hidden Markov models (HMMs) based ASR system to recognize the speech from the target corpus. The triphone acoustic models are trained on approximately 800 hours of non-native adult speech and adapted to 137.2 hours of children's speech. The language model adaptation is also applied using in-domain data [40]. The word error rate (WER) on the read aloud evaluation set[1] is 9.7%.

In this study, we only used the words where the ASR hypothesis was in agreement with the human transcription (14,302 out of 20,772 words). The purpose of this procedure is to ensure that our system should be capable of identifying actual pronunciation errors rather than ASR systematic errors. The final training corpus consists of 50,261 phones (error: 3,665, correct:

[1] The test-taker reads one of 684 paragraphs (containing approximately 90 - 100 words) presented on the screen out loud.

46,596), and evaluation corpus consists of 24,346 phones (error: 1,755, correct: 22,591). The distributions of two class labels ("error" and "correct") for individual phones (39 unique phones) on the training set and evaluation set are illustrated in Figure 4.1(a) and 4.1(b), and details of numbers are listed in Appendix A.1(a) and A.1(b), respectively.

4.2 Feature Extraction

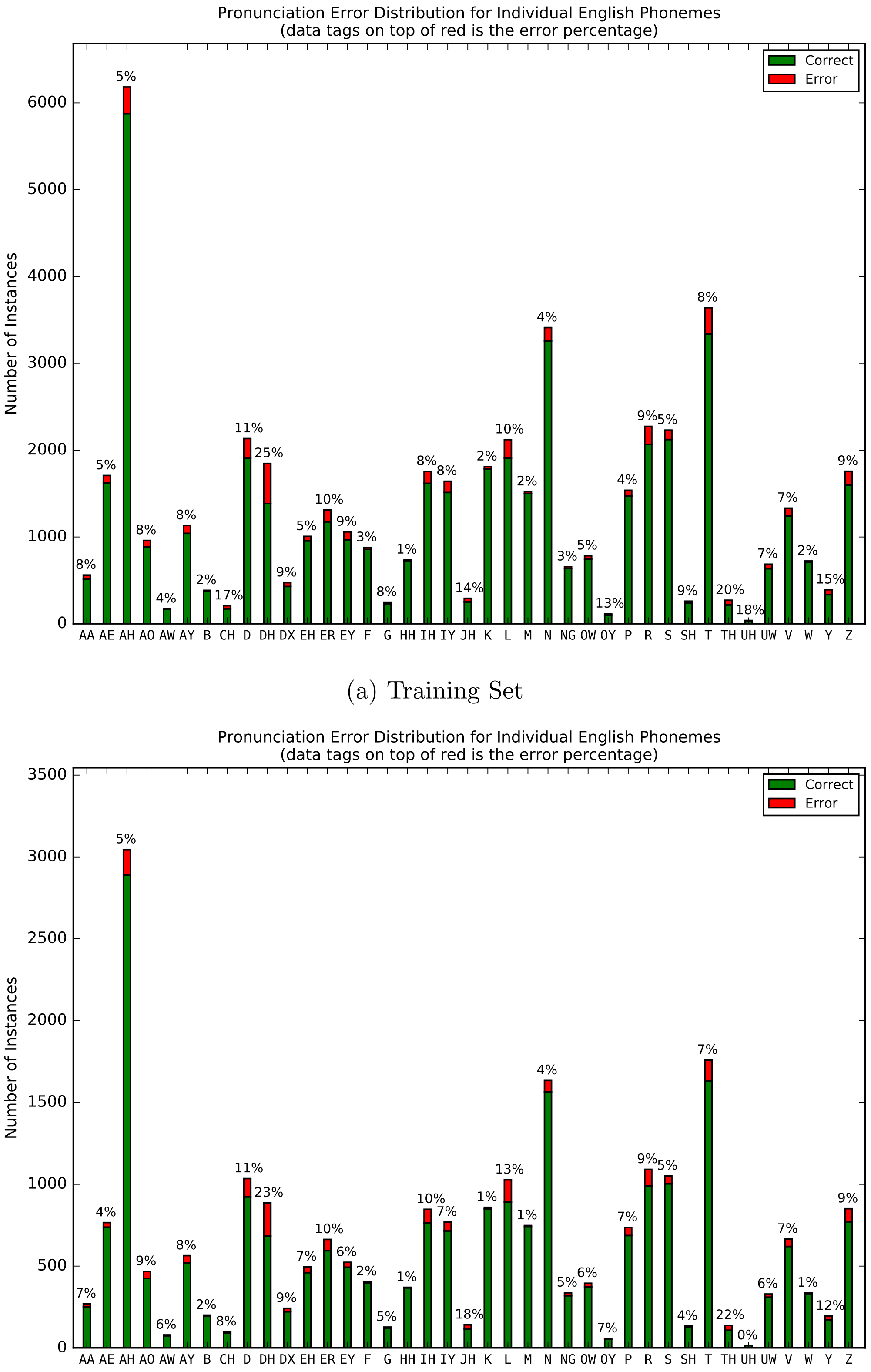
We select three different kinds of feature representations for potential pronunciation errors—acoustic features, word identity, and linguistic features. Individual features as well as their combinations are applied to train prediction models. (See Table 4.1 for details.)

Table 4.1: Feature candidates.

Model	Feature	Description
ac	*am_raw*	acoustic likelihood
	am_dur	duration-normalized acoustic likelihood
	cs_raw	acoustic confidence score
	cs_dur	duration-normalized confidence score
	logcs	log-scale confidence score
	logcs_dur	duration-normalized log-scale confidence score
ling	*str*	primary stressed syllable vs unstressed syllable
	onset	syllable structures: (onset, nucleus) vs. coda
	wrdIn	further segment between onset and nucleus: word-initial or word-medial syllable
	full	syllable with full vowel or reduced vowel
wi	word identity	

4.2.1 Acoustic Features

For each phone we extracted the six acoustic features listed in Table 4.1. Acoustic likelihood scores correspond to raw likelihoods [41]. Confidence scores are raw posterior probabilities computed based on the phone lattice. For each measure we also used a duration-normalized version computed by



(a) Training Set

(b) Evaluation Set

Figure 4.1: Pronunciation error distributions for individual phonemes.

dividing the raw or posterior probability for each phone by the number of frames in the phone.

4.2.2 Word Identity

Each speaker in this study read one of four texts. Preliminary analysis of human annotations showed that some words were more likely to contain errors than others [17]. Similar patterns have also been observed in previous work on pronunciation error detection [38]. For instance, 75% of speakers mispronounced the word "barley". To ensure that our models identify actual mispronunciations rather than simply learn difficult words, we also trained models using a word identity feature and used these models as a baseline.

4.2.3 Linguistic Features

We may still need to consider several types of phonological knowledge to further improve the performance for each phone.

Vowel reduction (full) is relevant to changes in stress, duration and position in the word. Usually vowels are uttered shorter for L2 English learners. So full vowel or reduced vowel may be a good representation of error pronunciations.

Lexical stress (str) is the stress placed on a given syllable within a word. The stress is always on a vowel. Here we only choose primary stress as the indicator of stress. Native speakers of English use lexical stress naturally, while for non-native speakers, it is often a challenge to imitate the force or strength of each syllable. Some languages, such as Japanese and French, pronounce each syllable with equal stress. This may result in the error pronunciations for those L2 learners. For example, the vowels in "photograph", "photographer" and "photographic" have different pronunciations caused by the stress.

Syllable structures consists of three segments: onset, nucleus and coda. Most often, onset and coda are consonants, while nucleus segments are vowels. Meanwhile, the segment between onset and nucleus can be further divided into initial and medial.

4.3 Classifiers and Imbalanced Learning

We trained and evaluated separate models for each one of the 39 phones rather than general models for all phones together, since previous work has found that the distribution of acoustic features differs across phones [21, 42]. Six classifiers[2] are selected to distinguish pronunciation errors on the imbalanced corpus—decision trees, random forest, gradient boosting, support vector machines with linear kernel (LinearSVC) and radial basis function kernel (SVC), and binomial logistic regression.

To address the issues caused by the imbalanced distribution of class labels in the training corpus, we performed experiments using the two approaches as described in Section 2.4. First, we increased the cost of misclassifying instances from the minority class by assigning weights inversely proportional to class frequencies (Auto-Weighting). Due to practical limitations, this procedure was only explored for three classifiers: SVC, LinearSVC and Logistic Regression. Second, we also applied the synthetic minority oversampling method (SMOTE) to all six classifiers. Table 4.2 summarizes these available combinations.

Table 4.2: Available classifiers and approaches for balancing the data.

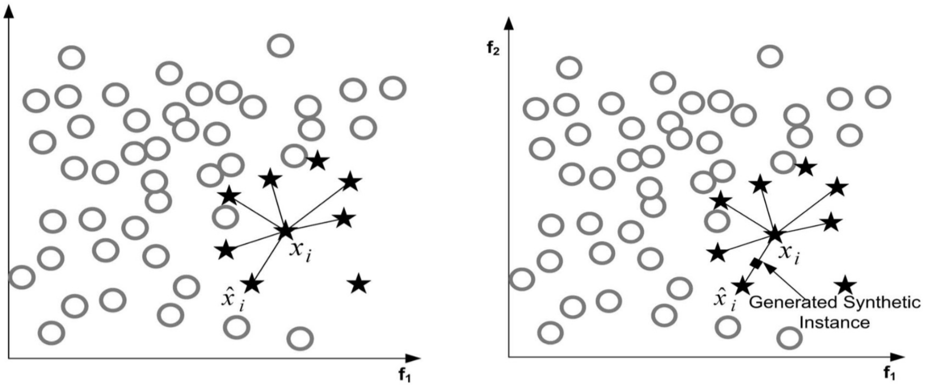
Classifier	Auto-Weighting	SMOTE-3	SMOTE-6
SVC	✓	✓	✓
LinearSVC	✓	✓	✓
Logistic Regression	✓	✓	✓
Decision Tree			✓
Random Forest			✓
Gradient Boosting			✓

The SMOTE algorithm creates artificial data based on the feature space similarities between existing minority examples. More formally, for subset $S_{minority} \in S$, consider the K-nearest-neighbors (KNN) for each example $x_i \in S_{minority}$, where $i \in [1, |S_{minority}|]$ and K is a predefined integer. In n-dimensional feature space X^n, KNN is defined as the K examples in $S_{minority}$ that hold the Top-K smallest Euclidean distances between itself and x_i. Figure 4.2 (captured from [31])demonstrates the procedure of creating a new example whose target label belongs to minority class. One of K examples is

[2] SKLL package: `https://github.com/EducationalTestingService/skll.git`

selected randomly, which is denoted as $\hat{x}_i$ and then a new instance is inserted between two vectors x_i and $\hat{x}_i$ by multiplying some random number $\delta \in [0, 1]$, namely,

$$x_{new} = x_i + (\hat{x}_i - x_i) * \delta \tag{4.1}$$



(a) K-near-neighbors of x_i

(b) x_{new} is generated

Figure 4.2: Demonstration of synthesizing a new example in original feature space. x_i is the example in $S_{minority}$; $\hat{x}_i$ is one of the K-near-neighbors; x_{new} is the synthesized example; $K = 6$.

All models were evaluated using stratified 10-fold cross validation. For the balanced data sets, the oversampling was applied only to the folds used for training while the fold used for evaluation remained unchanged. Our goal is to analyze differences in mean scores under more than three conditions, therefore, we use a repeated measures analysis of variance (rANOVA) [43] for the studies of comparing the performances of the classifiers and models for individual phones.

4.4 Feature Selection

Section 4.2 creates new features from prior knowledge of acoustic-phonetics, phonology, and word identities, however, from the feature engineering point of view, these features are commonly either redundant or irrelevant for the tasks of machine learning [44]. For example, one relevant feature may be redundant when there exists another relevant feature that is strongly correlated with it [45, 46]. Therefore, those features may need to be removed without incurring much loss of information. In our study, we applied multiple acoustic likelihood scores to distinguish the pronunciation errors from corresponding canonical pronunciations. Although these score features utilized different normalization techniques, the redundant information remains due to the possible correlations between their nature characteristics of speech sounds.

More formally, the algorithm of feature selection is predefined as a problem of searching an appropriate subset of features, such that the score on this feature subset is maximized in terms of some evaluation metric. The most straightforward solution is an exhaustive search in the hypothetical subset space, such that we can find one of them that minimizes the classification errors. This algorithm is computational intractable for large feature set, however, it is tractable for our small feature set as mentioned in Table 4.1.

CHAPTER 5

EXPERIMENTS

In this chapter, we will illustrate details of experiments on four different tasks: cross validation on the training set, evaluation on the held-out test set, *1-best* feature subset selection and word-level pronunciation error identification. In order to make clear explanation, this chapter will make use of several variables defined in Chapter 4, including three different features (word identity, acoustic information, linguistic information and their combinations), six classifiers (LinearSVC, SVC, logistic regression, decision tree, random forest, and gradient boosting), and two approaches (Auto-Weighting and SMOTE) for balancing the training data (see Table 4.2 and Table 5.1). All experiments are performed using the tool of SciKit-Learn Laboratory (SKLL) [47] that is a wrapper library of the Python machine learning toolkit Scikit-learn [48].

5.1 Cross Validation on Training Set

We performed 10-fold cross validation on both the original training set (imbalanced data) and its balancing improvements (balanced data). In experiments on imbalanced data, we compared the performance between acoustic features (`ac`), word identity (`wi`) and their combination (`ac+wi`), and found that the correlation between average `f1_score` and pronunciation error rate of individual phones revealed that insufficient training data in a minority class might be one of reasons leading to bad performance. Motivated by this observation, further experiments on two approaches of balancing the data are conducted.

5.1.1 Imbalanced Data

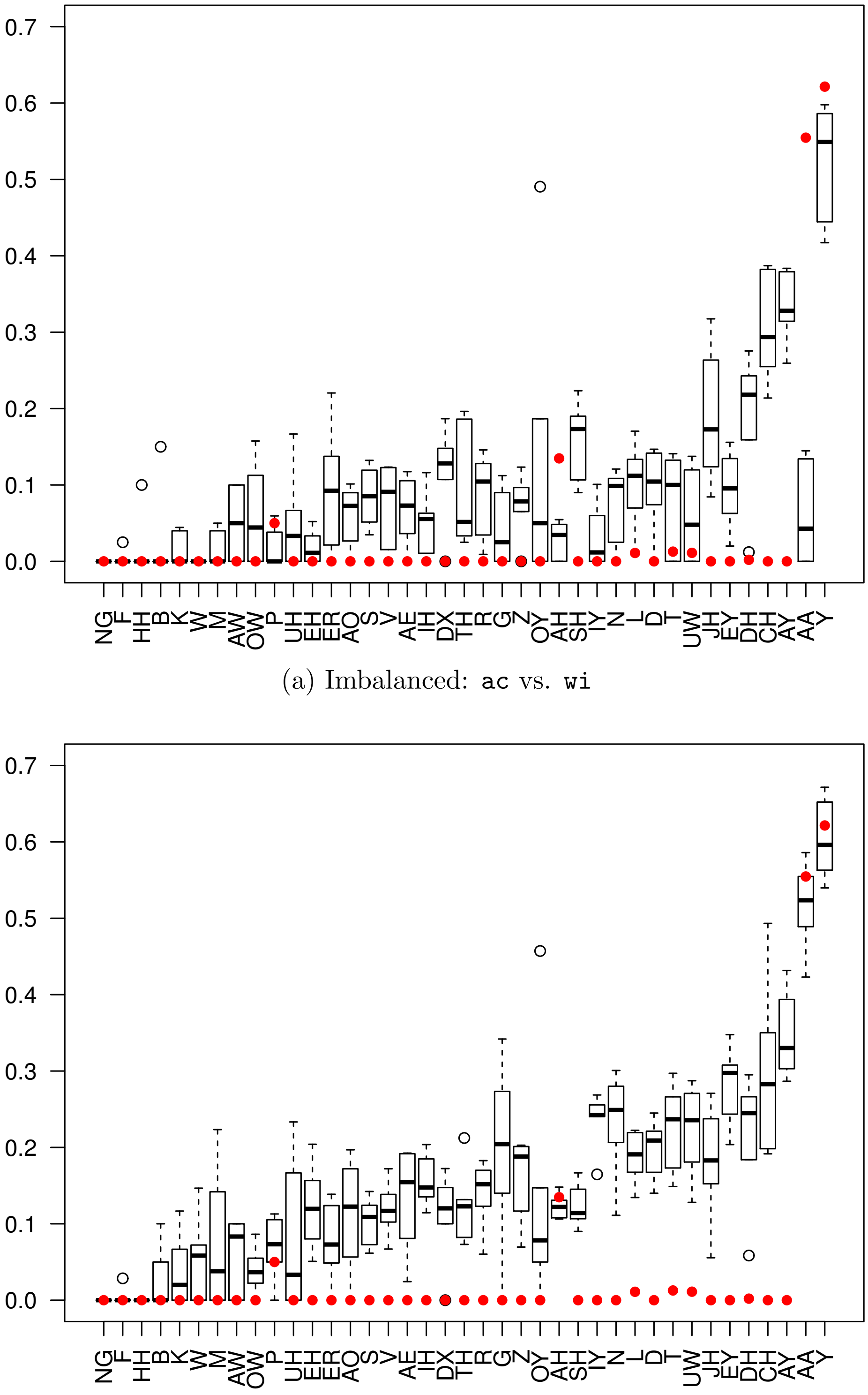
Figure 5.1(a) shows the box plot of f1_score over all classifiers for each phone for ac model. Red dots indicate the average performance for the baseline wi model over all classifiers. For two phones /ɑ/(AA) and /j/(Y), the baseline wi achieved an f1_score greater than 0.55. In most other cases the baseline model performed at chance level, whereas the ac model generally performed above chance level. However, for four phones /p/(P), /ʌ/(AH), /ɑ/(AA), and /j/(Y), the ac model did not outperform the baseline wi. The performance for most phones remained relatively low: f1_score was below 0.1 for 32 out of the 39 phones with an average f1_score of 0.09 across all phones. (See the first row in Table 5.1.)

Table 5.1: Mean value of f1_score for different models (columns) and methods of balancing the data (rows). SMOTE-3 includes the three classifiers where we also applied Auto-Weighting, and SMOTE-6 includes all six classifiers. (See Table 4.2.)

	wi	ac	ac+wi
Imbalanced	0.05	0.09	0.16
SMOTE-6	0.22	0.20	0.23
SMOTE-3	0.22	0.22	0.25
Auto-Weighting	0.21	0.24	**0.27**

Combining together both acoustic features and word identity ac+wi leads to a further increase in f1_score as shown in Figure 5.1(b). The performance across all phones, however, remains low, with an average f1_score of 0.16 (see Table 5.1).

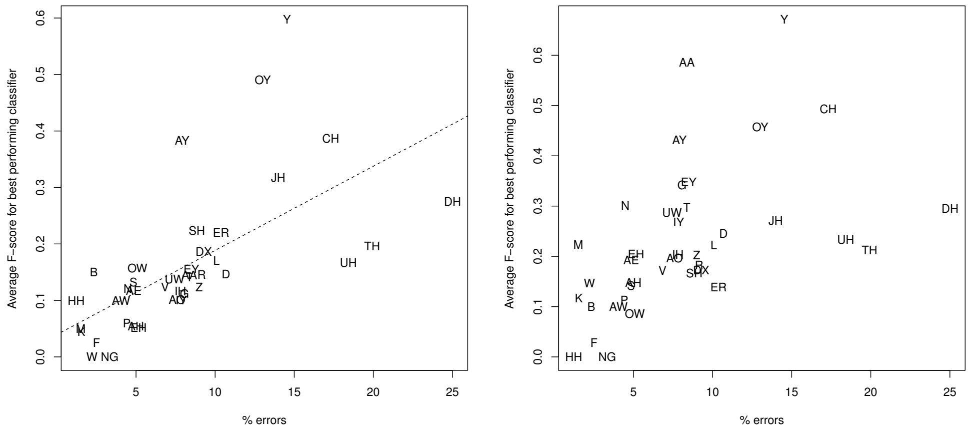
Besides, from Figure 5.2(b) we also observed that the average f1_score for ac+wi model was correlated with the percentage of errors for each phone: $r = 0.509$ ($p < 0.001$). Model ac also held similar correlation with error rates although it did not outperform the combination model. In other words, both models performed better for phones with a larger percentage of errors, thus confirming our initial intuition that the imbalanced nature of the training data affects the performance of binary classifications for identifying pronunciation errors.



(a) Imbalanced: `ac` vs. `wi`

(b) Imbalanced `ac+wi` vs. `wi`

Figure 5.1: `f1_score` over all six classifiers for different models on original imbalanced data. Boxes show `f1_score` across six classifiers for individual phones given a model (`ac` or `ac+wi`); red dots indicate the mean `f1_score` across six classifiers for the baseline model (`wi`) on word identity.



(a) ac model

(b) ac+wi model

Figure 5.2: Linear correlations between average f1_score and error rates on training set. For ac+wi model, $r = 0.509$ ($p < 0.001$).

5.1.2 Balanced Data

In this section, we conducted a study on balancing the training data by leveraging two approaches, Auto-Weighting and SMOTE, in order to alleviate the affects of imbalance nature of data. Similar with previous section, the combined model ac+wi was compared with baseline model wi.

(a) Auto-Weighting and SMOTE over All Classifiers

Figure 5.3 shows the ranked f1_score over all phones and all classifiers for different combinations of features and imbalanced learning approaches. We see large increases in performance over the baseline model wi.Imbalanced when the two approaches wi.SMOTE and wi.Autoweighting are applied; and similarly, moderate increases are observed in performance over ac+wi.Imbalanced model in original imbalanced data set ($p = 6 \times 10^{-7}$ for model, $p = 2 \times 10^{-16}$ for the sampling method, $p = 2 \times 10^{-16}$ for interaction) when ac+wi.SMOTE and ac+wi.Autoweighting are applied. Table 5.1 shows that the mean value of f1_score achieved by the Auto-Weighting approach increased from 0.16 on the imbalanced data to 0.27.

Considering a more fine-grained comparison across all 39 phones, we also illustrate f1_score over all six classifiers for each phone with respect to both ac and ac+wi models. Figure 5.4 and Figure 5.5 show the box plot for two approaches, respectively. The sequence of phones in these two figures are

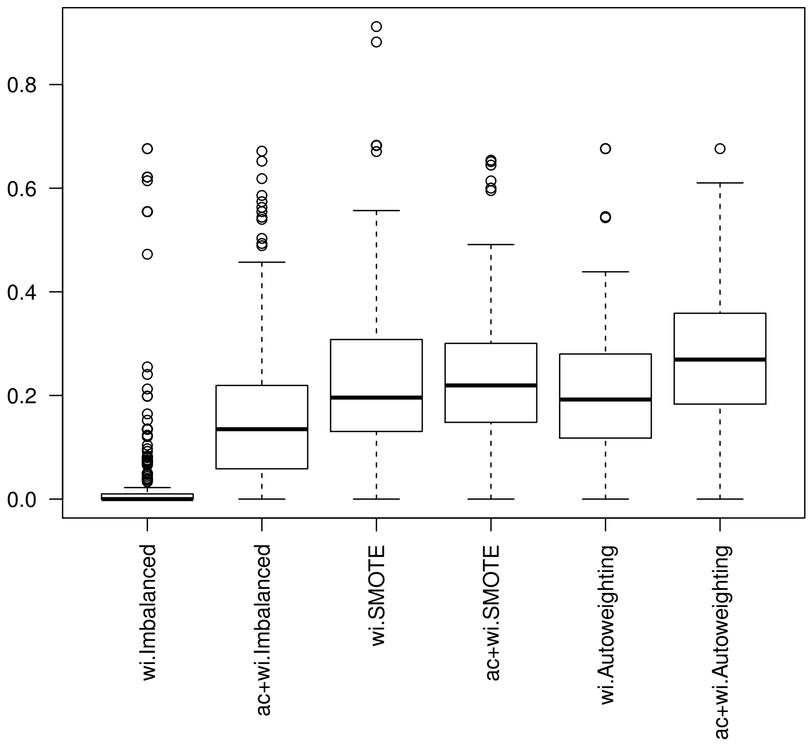
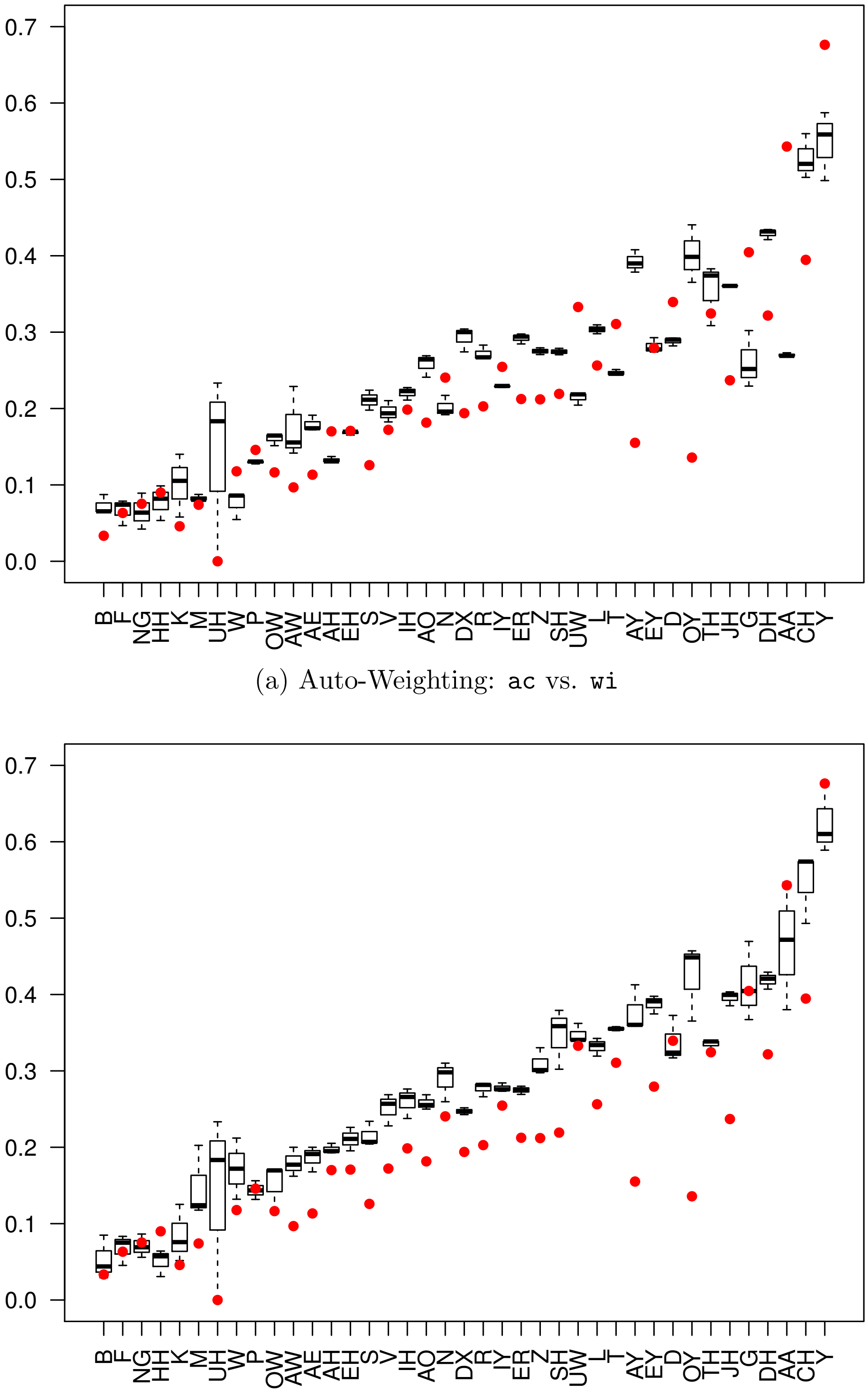


Figure 5.3: Ranked `f1_score` across all classifiers and all phones. Segment lines inside boxes denote median.

sorted by the average $f1\_score$ over all six classifiers. In comparison to Figure 5.1, both approaches help to increase the performance by a large margin for all models of `wi`, `ac`, `ac+wi`. Besides, Auto-Weighting approach holds relatively more narrow variations of training samples for individual phones than SMOTE approach. This observation indicates that this over-sampling method indeed helps to distinguish the pronunciation errors in comparison to the experiment on original data, however, it also brings more noisy samples that could affect the general performance in comparison to sample-weighting approaches.

(b) Comparison between Classifiers

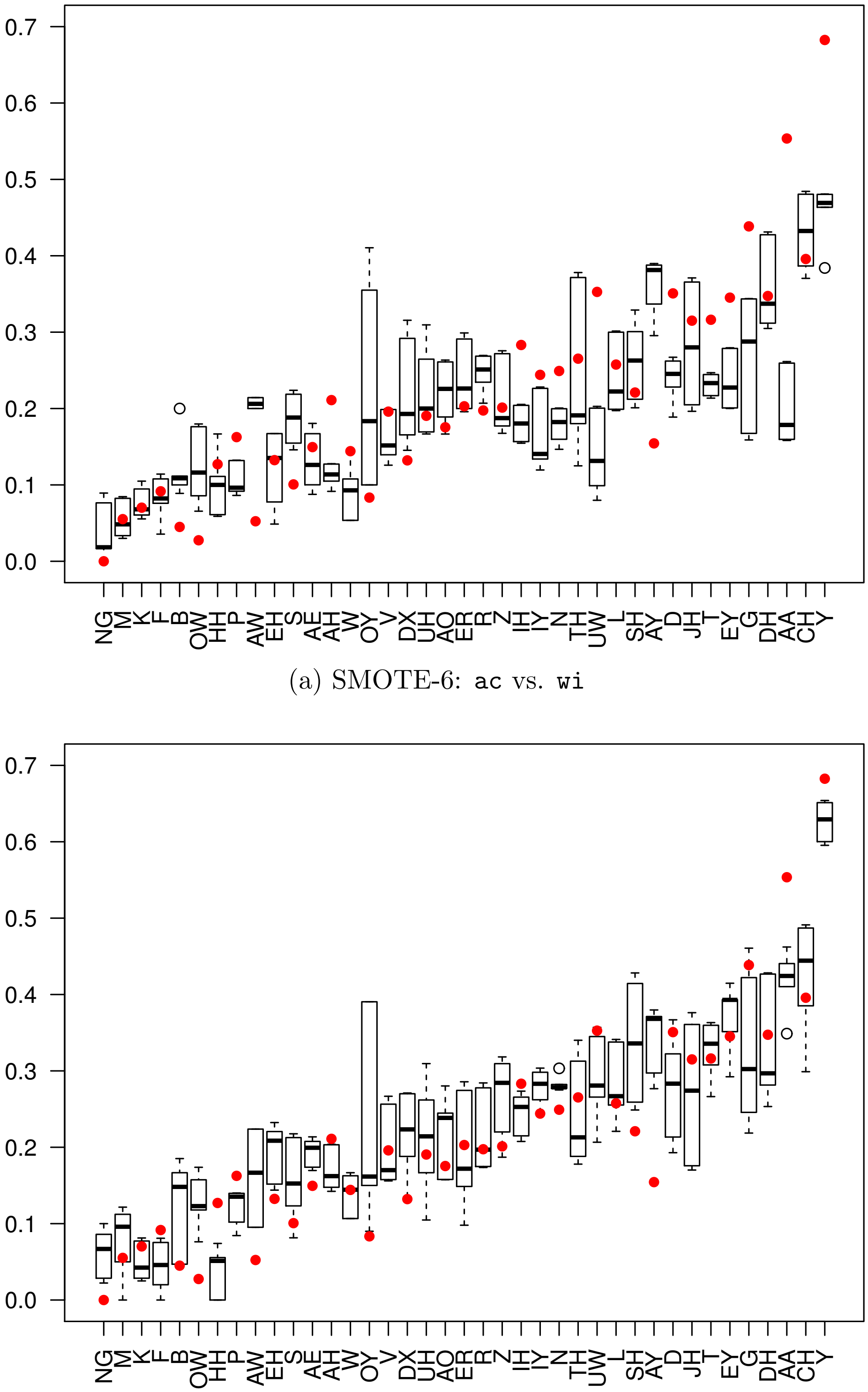
We further explored the difference among the six classifiers by comparing the average `f1_score` over all phones for the combined model `ac+wi`. Figure 5.6 illustrates that on imbalanced data, decision trees and a support vector machine with a non-linear kernel (SVC) outperformed the two linear classifiers (logistic regression and linearSVC), but linear classifiers outperformed



(a) Auto-Weighting: `ac` vs. `wi`

(b) Auto-Weighting: `ac+wi` vs. `wi`

Figure 5.4: `f1_score` over all six classifiers for different models when applying Auto-Weighting. Boxes show `f1_score` across six classifiers for individual phones given a model (`ac` or `ac+wi`); red dots indicate the mean `f1_score` across six classifiers for the baseline model (`wi`) on word identity.



(a) SMOTE-6: `ac` vs. `wi`

(b) SMOTE-6: `ac+wi` vs. `wi`

Figure 5.5: `f1_score` over all six classifiers for different models when applying SMOTE. Boxes show `f1_score` across six classifiers for individual phones given a model (`ac` or `ac+wi`); red dots indicate the mean `f1_score` across six classifiers for the baseline model (`wi`) on word identity.

the other classifiers after applying SMOTE. When using Auto-Weighting approach, three classifiers, SVC, LinearSVC and logistic regression, achieved similar performance (the classifier effect on `f1_score` within each phone: $p = 0.00162$ after controlling for model and oversampling method).

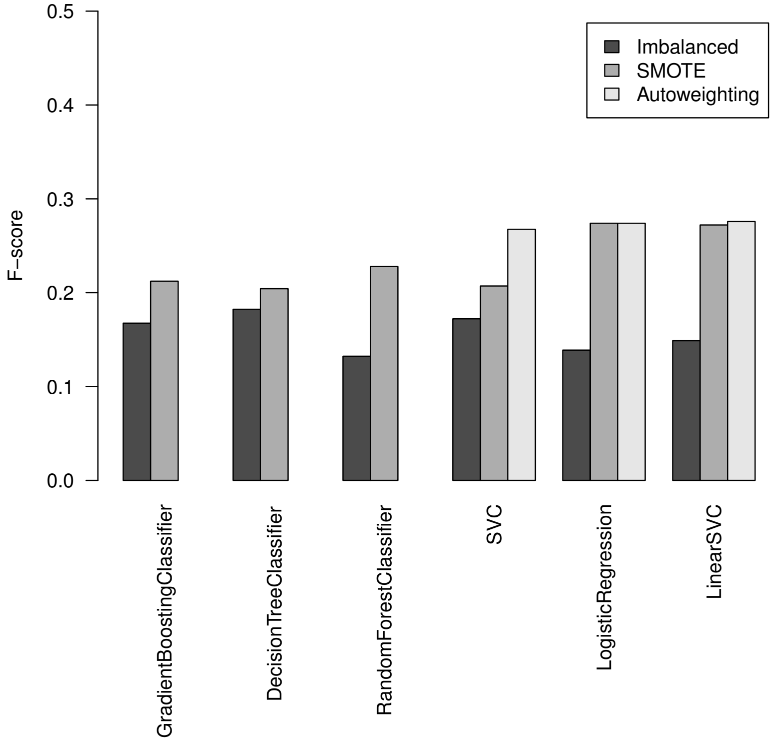
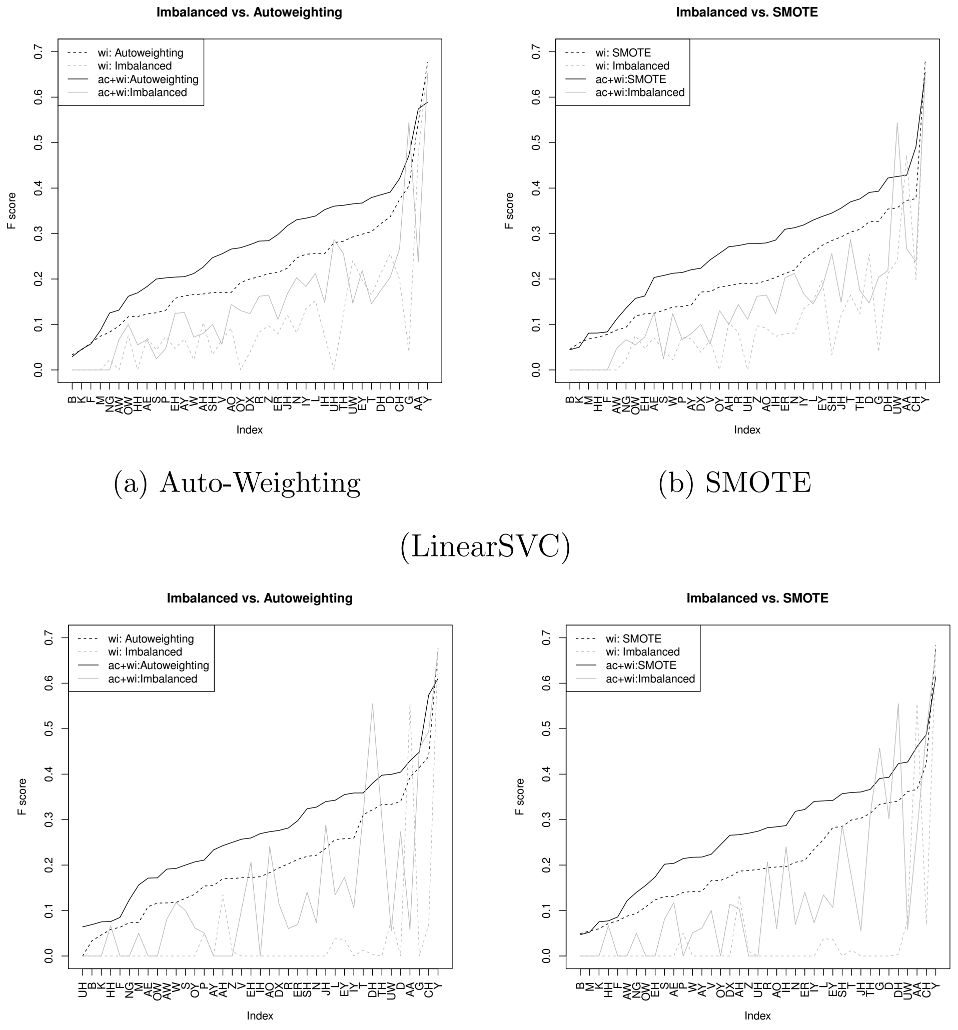


Figure 5.6: Classifier performance for the `ac+wi` model.

In a fine-grained comparison, the performance of classifiers varied substantially across individual phones. We choose LinearSVC and logistic regression to make analysis. Even these two classifiers achieved almost the same overall performance, the performances for the models on imbalanced data perturbed severely across phones (see Figure 5.7(a) and 5.7(c)). We observed that there is a big difference in the `f1_score` for /ð/ (DH) when applying these two classifiers.

(c) Comparison between Auto-Weighting and SMOTE on LinearSVC

Since both the logistic regression and the LinearSVC obtained almost the same `f1_score` as shown in Figure 5.6, we will choose LinearSVC as a representative classifier to further explore the comparisons between two ap-



(a) Auto-Weighting

(b) SMOTE

(LinearSVC)

(c) Auto-Weighting

(d) SMOTE

(Logistic Regression)

Figure 5.7: `f1_score` comparison between LinearSVC and Logistic Regression across individual phones.

proaches in a fine-grained level. Figure 5.8 illustrates the performance of the support vector classifier with a linear kernel using two different imbalanced learning approaches across different phones for both the `wi` baseline and its improved model `ac+wi`. It shows that both approaches for the combined model `ac+wi` achieved almost similar `f1_score` in a fine-grained phone level that is consistent with the coarse-grained performance (see the last column in Figure 5.6), outperformed the baseline `wi` model for all phones except /ɑ/(AA) and /j/(Y) that were observed in Section 5.1.1.

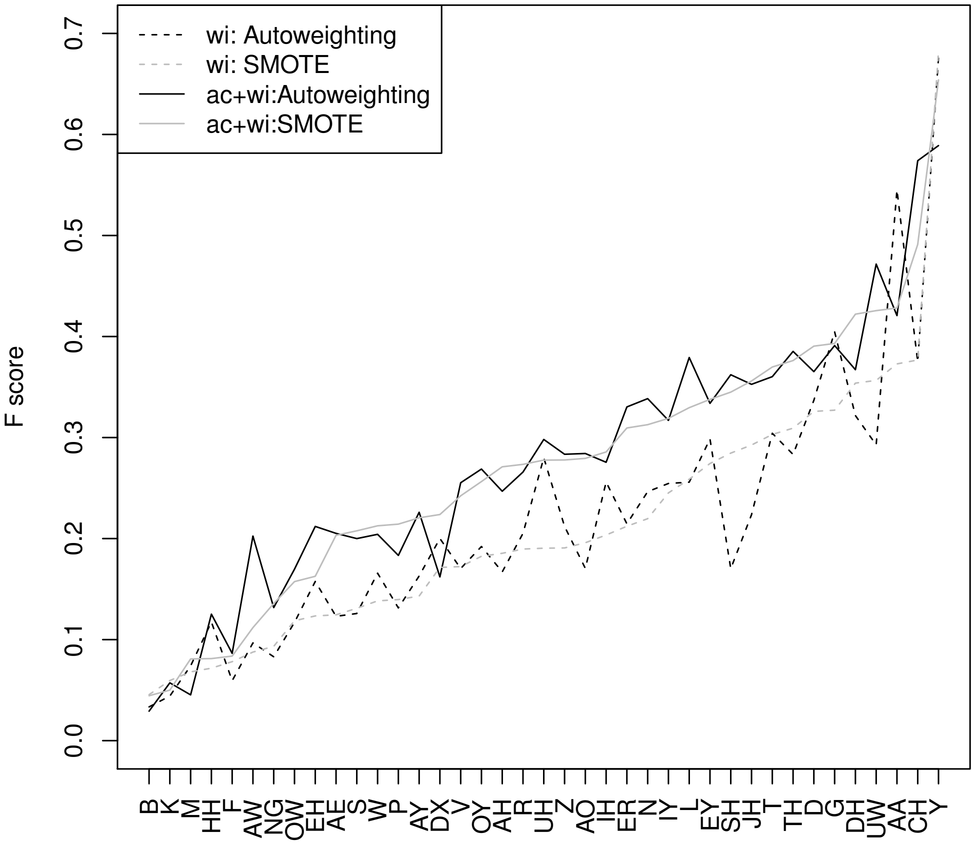


Figure 5.8: Comparison of Auto-Weighting and SMOTE imbalanced learning approaches for support vector machine with a linear kernel.

5.2 Evaluation on Test Set

In this section, we will further explore the generalization power of models by leveraging different combinations of two classifiers (LinearSVC and logistic regression) and two balancing approaches (Auto-Weighting and SMOTE) based on acoustic features and word identity (`ac`+`wi`). Figure 5.9 illustrates the `f1_score` for all different combinations of classifiers and balancing approaches across all phones, where the sequence of phonemes is ordered by error rates on training set.

5.2.1 Comparison between Classifiers

When applying SMOTE approach, LinearSVC and logistic regression achieved almost the same performance for each phone (see two dashed lines with colored circle markers), while Auto-Weighting approach did not obtained consistent `f1_score` between these two classifiers, particularly for the phones with error rate less than 4.48% (see solid red and blue curves between /h/(HH) and /p/(P)). Nevertheless, when the error rate of pronunciation goes beyond 4.48% (after /p/(P)), consistent performance between two classifiers is obtained, except for the vowel /ɑ/(AA). Generally, the combinations of `logisticRegression` and `SMOTE`, `logisticRegression` and `AutoWeighting`, `LinearSVC` and `SMOTE` maintained the same mean value of `f1_score` = 0.25, while `LinearSVC` and `AutoWeighting` achieved better performance. (See Table 5.2.)

5.2.2 Comparison between Balancing Approaches

When applying logistic regression classifier, two balancing approaches achieved consistent `f1_score` for all 39 phones (see blue line and dashed line with yellow circle markers). After applying LinearSVC (see red line and dashed line with purple circle marker), inconsistent performance falls in the vowel /ɑ/(AA) and phones /m/(M), /k/(K), /w/(W), /b/(B) with extremely low error rates (less than 2.38%). For the phones /m/(M), /k/(K), and /w/(W), synthesizing samples in the minority class did not improve the performance of classification, while weighting inversely proportional to the frequency of classes lead to a significant increase in `f1_score`. Table 5.2 shows the mean values of `f1_score` for different combinations of classifiers and approaches of balancing the training data. The values indicate that the combination of LinearSVC and Auto-Weighting is suggested for solving the task of general classification of pronunciation errors.

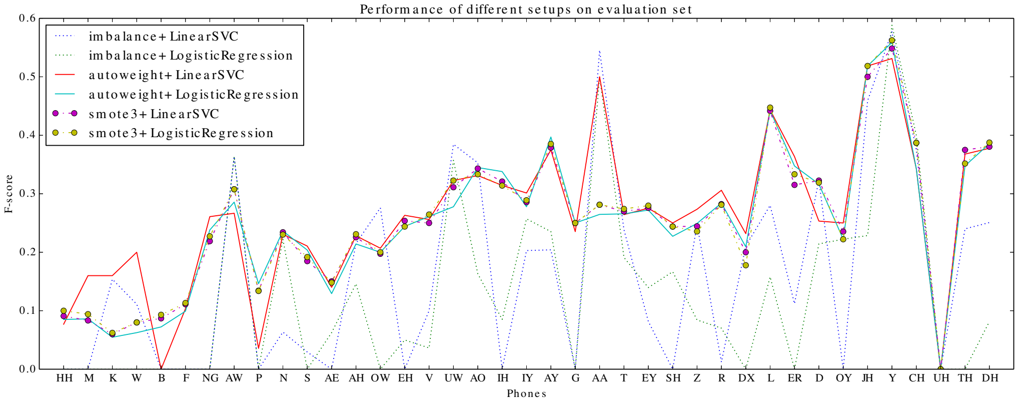


Figure 5.9: Performance of different setups on held-out test set: LinearSVC vs. Logistic Regression, Auto-Weighting vs. SMOTE, where the sequence of phones is ordered by error rates on training set.

Table 5.2: Mean value of `f1_score` over 39 phones for different classifiers and methods of balancing the data.

	Imbalance	SMOTE	Auto-Weighting
Logistic Regression	0.129	**0.254**	0.250
LinearSVC	0.164	0.253	**0.262**

5.3 Feature Selection

From Section 5.1.2(c), we see that there still exists a large range of `f1_score` for individual phones from 0.02 to 0.6, even though we applied the relatively best configuration (LinearSVC and Auto-Weighting) based on acoustic features and word identity (`ac`+`wi`) as mentioned in Table 4.1. All current configurations of distinguishing error pronunciations from canonical ones rely on the domain knowledge of phonetics, as measured by acoustic likelihood scores calculated by automatic speech recognizer. We may still need to consider other phonological knowledge to further improve the performance for each individual phone. In this section, we continue to explore the procedure of feature selections that may provide more insights on identifying pronunciation errors for individual phonemes. Concisely, the various types of prior phonological knowledge is denoted as linguistic features (`ling`) in the following sections.

5.3.1 Exhaustive Search for *1-best* Feature Subset

Considering linguistic features, the whole feature set is expanded as illustrated in Table 4.1. We conduct feature selection experiments for each phone based on best configuration (LinearSVC and Auto-Weighting) as suggested in Section 5.2.2. Ablation operation is applied on the whole feature set in order to select best subset for each phone. A 10-fold cross validation is performed on the training set. Figure 5.10 illustrates the performances of the best feature subset and baseline `ac+wi`. As expected, the best feature subset achieved a better `f1_score` for individual phones than the baseline. Specifically, for those phones /h/(HH), /k/(K), /b/(B), /f/(F), /g/(G), /ɔɪ/(OY), /ʊ/(UH), the performances are improved by a large margin. These selected best feature subset for each individual phones is shown in Table 5.3.

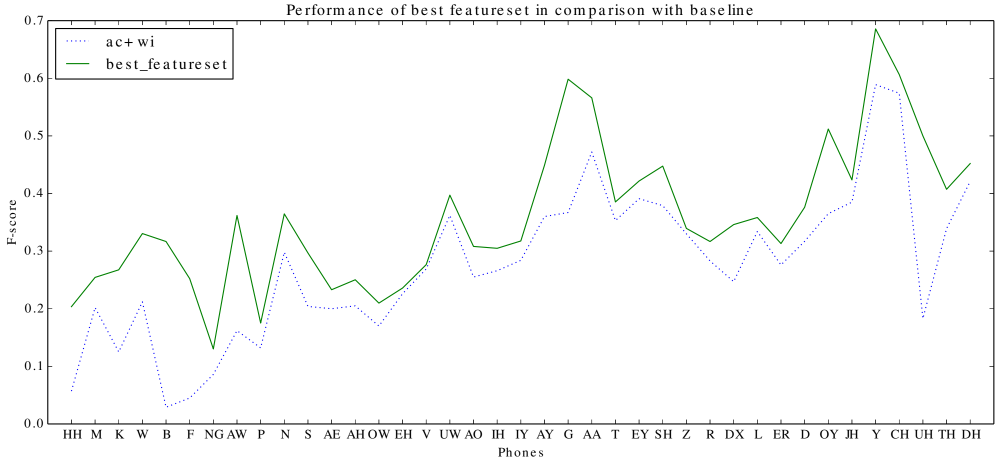


Figure 5.10: Comparison between the best feature subset and `ac+wi` baseline on the training set using LinearSVC and the Auto-Weighting approach, where the `f1_score` is averaged over all 10 folds. The sequence of phones is sorted by the percentage of pronunciation errors.

Table 5.3: The *1-best* feature subset for individual phones.

	am_raw	am_dur	cs_raw	cs_dur	logcs_raw	logcs_dur	onset	full	str	wrdIn	word	score
HH		x	x		x				x		x	0.203571
M	x	x			x			x		x	x	0.254393
K	x	x		x		x		x			x	0.267527
W			x		x				x		x	0.330476
B		x	x		x			x	x	x		0.316667
F		x	x	x		x	x		x	x	x	0.252222
NG	x		x		x	x				x	x	0.130207
AW	x	x		x						x		0.361905
P	x	x	x	x	x	x		x	x	x	x	0.175151
N		x		x		x		x	x	x	x	0.364656
S		x	x	x			x	x				0.296779
AE			x		x	x			x		x	0.232980
AH	x	x			x	x			x	x	x	0.250278
OW						x						0.209761
EH		x			x	x			x		x	0.235892
V	x	x				x				x	x	0.276525
UW				x	x					x		0.397157
AO	x					x						0.308145
IH	x	x	x	x	x				x		x	0.304873
IY		x	x		x				x	x	x	0.317590
AY	x	x			x					x		0.448871
G		x		x		x			x		x	0.598413
AA	x	x			x				x		x	0.566037
T		x	x	x	x	x	x		x	x	x	0.385360
EY	x	x			x				x		x	0.421666
SH	x		x	x		x	x	x			x	0.447554
SH	x		x	x		x	x		x		x	0.447554
SH	x		x	x		x	x			x	x	0.447554
Z	x		x	x	x	x		x	x	x	x	0.339440
R				x	x		x					0.316642
DX			x			x		x	x			0.346059
L	x					x					x	0.358340
ER	x		x	x	x	x		x		x		0.313118
ER	x		x	x	x	x			x	x		0.313118
D	x		x						x		x	0.376115
OY		x		x						x		0.511905
JH				x	x			x	x		x	0.423563
Y				x	x	x		x	x	x		0.685754
CH		x				x	x	x			x	0.606472
CH		x				x	x		x		x	0.606472
UH	x	x										0.500000
TH	x	x	x	x		x						0.407279
DH	x	x			x	x	x		x			0.452019

5.3.2 Evaluation with *1-best* Feature Subset

Based on the *1-best* feature subset for each phone which achieved the best `f1_score` on the training set, we trained a separate LinearSVC classifier for each phone using the Auto-Weighting approach. Thereafter, experiments on the held-out test set are performed to analyze the generalization power. As shown in Figure 5.11, most phones obtained better score than the baseline `ac+wi` except /m/(M), /f/(F), /ŋ/(NG), /s/(S) and /ɾ/(DX). Table 5.4 illustrates the mean values of the `f1_score` over all phones for the *1-best* feature in comparison to the `ac+wi` model. The performance of the *1-best* feature model dropped 1.07% from 0.2623 on the `ac+wi` model, however, it remains 57.85% better than the baseline (imbalanced data).

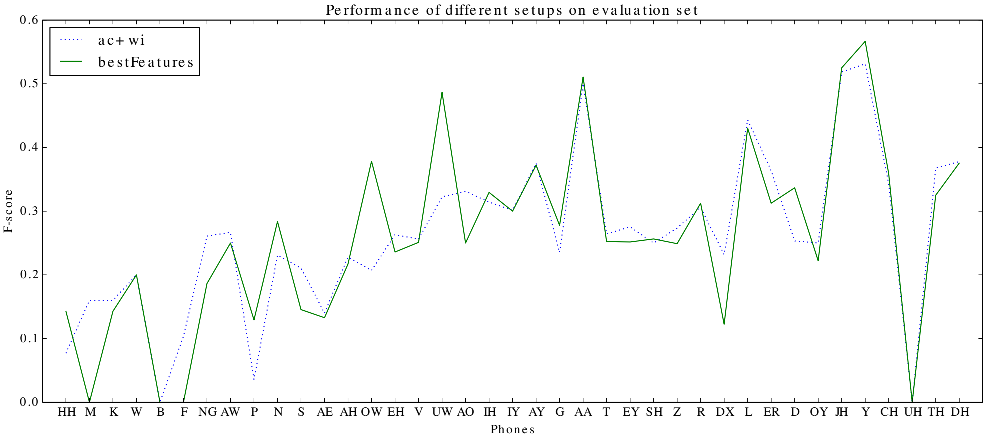


Figure 5.11: Comparison of performances on evaluation set based on *1-best* feature subset and `ac+wi` baseline using LinearSVC as classifier, where the sequence of phones is sorted by the percentage of pronunciation errors.

Table 5.4: Mean values of `f1_score` for different setups.

	`ac+wi`			*1-best* feature
	Imbalance	SMOTE	Auto-Weighting	Auto-Weighting
LinearSVC	0.1644	0.2529	**0.2623**	0.2595

5.4 Word-Level Pronunciation Error Identification

Previous sections in this chapter studied the task of identifying pronunciation errors at phone level. We are also interested in detecting those pronunciation errors at the word level using models similar to those selected in phone level experiments. In this section, we continue to conduct word level experiments on different feature models by leveraging two balancing approaches, and make evaluation on the held-out test set. LinearSVC is the only classifier to be applied. The word level pronunciation error is triggered if any one of phones in this word pronunciations is mispronouned.

5.4.1 Cross Validation on Training Set

The 10-fold cross validations are performed using different combinations of feature set and balancing approaches. Table 5.5 lists the mean values of `f1_score` over all phones. At word level, word identity feature that represents the difficulty level of words provides a reasonable baseline since we expect our models to identify mispronunciations rather than learn difficulty words. For each feature model (see the headers of Table 5.5), the scores increases by a large margin when applying balancing approaches, and Auto-Weighting method achieves the best performance all the time. Adding linguistic features (`ac+wi+ling`) does not make improvement while leads to drop by 1% in comparison to model `ac+wi`. As expected, *1-best* model achieves the best performance on the training set.

Table 5.5: Comparison of 10-fold cross validation among different feature sets in terms of `f1_score`. This table only shows the results of the LinearSVC classifier.

	`wi`	`ac`	`ac+wi`	`ac+ling`	`ac+wi+ling`	*1-best*
Imbalance	0.3173	0.1520	0.3392	0.1953	0.3300	NaN
SMOTE	0.3695	0.3730	0.4179	0.3750	0.4176	NaN
Auto-Weighting	**0.3746**	**0.3749**	**0.4276**	**0.3834**	**0.4233**	**0.4457**

5.4.2 Evaluation

In this evaluation task on held-out test set, we take into account two models: `ac+wi` and *1-best* in comparison along with two balancing approaches. Table 5.6 illustrates that these two models hold competitive scores when applying Auto-Weighting approach.

Table 5.6: Comparison on held-out test set in terms of `f1_score`. This table only shows the results of the LinearSVC classifier.

	`ac+wi`	*1-best*
Imbalance	0.3468	NaN
SMOTE	0.4190	NaN
Auto-Weighting	0.4229	**0.4282**

CHAPTER 6

DISCUSSION AND CONCLUSION

6.1 Discussion

In this thesis, we compared the performance of different machine learning algorithms and approaches to handling imbalanced data sets in the context of the task of pronunciation error detection in a large-scale language assessment. We found that the best performance could be achieved by combining information about word identity and the acoustic properties of the word, although the performance of the models varied across phones. We note that a simple model based solely on the word identity achieved relatively high performance, especially on the balanced data set (for /ɑ/(AA), `f1_score` = 0.55). Despite this high performance, such a model has little use in assessment or training: it only distinguishes difficult and easy words without regard to the correctness of a particular pronunciation. Nonetheless, this potential effect of word identity has been often ignored in previous studies. We recommend that a model based on word identity should be used as one of the baselines in all future studies on pronunciation error detection to ensure that the model performance is not limited to the identification of difficult words. Besides word identity and acoustic features, we also observed that linguistic factors contributed to the improvement of the performance for each individual phones, which suggested a promising future work on further exploring linguistic features.

Both cost-sensitive (Auto-Weighting) and sampling (SMOTE) approaches substantially improved the performance of the model in comparison to the baseline trained on the original imbalanced data. For some classifiers, the Auto-Weighting approach outperformed SMOTE. Since the Auto-Weighting approach only considers the cost associated with misclassifying samples, while the SMOTE approach synthesizes artificial data, the lower performance of the SMOTE method may be due to the fact that the synthetic data was generated from only the minority class and may have led to an increased overlap between classes.

Finally, we found that support vector classifiers and logistic regression obtained better classification performance than decision trees, random forest and gradient boosting classifier. We also observed that classification performance differed by phone.

6.2 Conclusion

This thesis investigates the task of phone-level pronunciation error detection as a binary classification problem, of which the performance is highly affected by the imbalanced distribution of classes (error vs. correct) in the data set. We explored the use of a word identity feature as a baseline, and acoustic features combined with it improved the overall classification performance on both imbalanced and balanced data. Meanwhile, two imbalanced learning approaches (Auto-Weighting and SMOTE) were applied and both achieved a better average `f1_score` by a large margin. In the end, empirical experiments were also performed for different machine learning classifiers, among which support vector machines with a linear kernel and logistic regression were the most effective classifiers on the general task of identifying pronunciation errors.

6.3 Future Works

6.3.1 Context Feature of Each Phone

Current experiments considered acoustic features, word identity and a few phonological characteristics of phones. We expect to explore more features that could contribute to enhance the error identification power.

For the task of identifying pronunciation errors for each individual phone by training a robust classifier, the variation within the same phone is also useful information to help classifiers discriminate error from canonical pronunciations. Take Allophones for example. Allophones are defined as one of two or more variants of the same phoneme. For instance, two pronunciations /l/ and /əl/ for the letter "l" in the word "little", or /p/(P) in "pin" and "spin", each pair has different pronunciation. These pronunciation variants of the same phone at different positions within a word may lead to different error types. In other words, such errors may result from the context of each phone. Motivated by this observation, triphone identity may have good potential for further exploration.

6.3.2 Acoustic Landmark Theory

Recent application of landmark-based distinctive features in ASR motivated researchers to further explore the features' utility in pronunciation error detection problems. Quantal nonlinearities in articulatory-acoustic relations provide a theoretical basis for selecting distinctive features, complementary to the empirical foundations of most L2 research [49]. Stevens [50, 51] proposed four different candidate landmark locations for English, including the vowel peak landmark, oral closure landmark, glide valley landmark in glide-like consonants, and oral release landmark. These four landmark categories were proposed by Stevens to be language-universal.

In the field of identifying English pronunciation errors, acoustic landmark based distinctive features have gained great success for detecting six phonemes for which Korean (L1) speakers made frequent pronunciation errors in previous work [15, 16]. In the field of Mandarin Chinese mispronunciation detection, we explored two approaches [52] to select Mandarin Chinese salient phonetic landmarks for the Top-16 frequently mispronounced phonemes by Japanese (L1) learners, and extracted features at those landmarks including mel-frequency cepstral coefficients (MFCC) and formants. One approach is to directly map well-founded English landmark theory into Chinese language since there exists correspondences of articulatory-manner and articulatory-place between English and Mandarin Chinese after applying Stevens theory. Second, we defined distinctive Chinese landmarks for the Top-16 frequent pronunciation errors by conducting human speech perception experiments in collaboration with linguists. Experiments showed that acoustic cues of MFCC and formants at both Chinese landmarks and English landmarks led significantly better performance over the strong GOP baseline.

However, determining the acoustic landmark positions that best represent categorical phonological distinctions remains a difficult problem, since the acquisition of this knowledge requires large-scale experiments of human speech perception [53]. The lack of this knowledge hinders the progress of the application on identifying pronunciation errors. The discovery of acoustic landmark theory and its success on identifying Chinese and English pronunciation errors shed light on our future research of detecting pronunciation errors in the context of large-scale assessment of English proficiency.

APPENDIX A

STATISTICS OF CORPUS

A.1 Error Distributions on Training Set and Evaluation Set

Table A.1: Error distributions on the training set and evaluation set.

(a) Training Set

	Correct	Error	Rate (%)
AA	514	47	8.378
AE	1625	83	4.859
AH	5874	309	4.998
AO	887	73	7.604
AW	165	7	4.070
AY	1042	90	7.951
B	376	9	2.338
CH	172	36	17.308
D	1906	228	10.684
DH	1385	462	25.014
DX	430	44	9.283
EH	955	52	5.164
ER	1175	136	10.374
EY	969	90	8.499
F	857	22	2.503
G	228	20	8.065
HH	728	9	1.221
IH	1618	137	7.806
IY	1513	129	7.856
JH	252	41	13.993
K	1782	28	1.547
L	1908	214	10.085
M	1499	23	1.511
N	3259	153	4.484
NG	636	22	3.343
OW	744	40	5.102
OY	100	15	13.043
P	1470	68	4.421
R	2066	208	9.147
S	2123	108	4.841
SH	237	23	8.846
T	3336	305	8.377
TH	217	54	19.926
UH	31	7	18.421
UW	635	51	7.434
V	1241	91	6.832
W	707	16	2.213
Y	335	57	14.541
Z	1599	158	8.993
Total	46596	3665	7.292

(b) Evaluation Set

	Correct	Error	Rate (%)
AA	251	18	6.691
AE	738	28	3.655
AH	2889	156	5.123
AO	425	42	8.994
AW	74	5	6.329
AY	520	44	7.801
B	196	4	2.000
CH	91	8	8.081
D	923	112	10.821
DH	683	203	22.912
DX	221	21	8.678
EH	460	36	7.258
ER	594	69	10.407
EY	493	30	5.736
F	398	7	1.728
G	121	6	4.724
HH	367	3	0.811
IH	765	82	9.681
IY	715	54	7.022
JH	115	26	18.440
K	850	9	1.048
L	891	136	13.242
M	740	8	1.070
N	1564	70	4.284
NG	319	18	5.341
OW	372	23	5.823
OY	53	4	7.018
P	687	49	6.658
R	990	101	9.258
S	1003	48	4.567
SH	128	5	3.759
T	1630	128	7.281
TH	108	30	21.739
UH	14	0	0.000
UW	310	19	5.775
V	620	45	6.767
W	332	4	1.190
Y	170	24	12.371
Z	771	80	9.401
Total	22591	1755	7.209

REFERENCES

[1] M. Levy, *Computer-Assisted Language Learning: Context and Conceptualization.* Oxford University Press, 1997.

[2] D. Luo, X. Yang, and L. Wang, "Improvement of segmental mispronunciation detection with prior knowledge extracted from large L2 speech corpus," in *12th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2011, pp. 1593–1596.

[3] Y. Tsubota, T. Kawahara, and M. Dantsuji, "Practical use of English pronunciation system for Japanese students in the CALL classroom," in *8th International Conference on Spoken Language Processing (ICSLP)*, 2004, pp. 1689–1692.

[4] A. M. Harrison, W. Y. Lau, H. M. Meng, and L. Wang, "Improving mispronunciation detection and diagnosis of learners' speech with context-sensitive phonological rules based on language transfer," in *9th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2008, pp. 2787–2790.

[5] L. Wang, X. Feng, and H. M. Meng, "Automatic generation and pruning of phonetic mispronunciations to support computer-aided pronunciation training," in *9th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2008, pp. 1729–1732.

[6] X. Qian, H. M. Meng, and F. K. Soong, "On mispronunciation lexicon generation using joint-sequence multigrams in computer-aided pronunciation training (CAPT)," in *12th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2011, pp. 865–868.

[7] W. K. Lo, S. Zhang, and H. M. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," in *11th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2010, pp. 765–768.

[8] L. Neumeyer, H. Franco, V. Digalakis, and M. Weintraub, "Automatic scoring of pronunciation quality," *Speech Communication*, vol. 30, no. 2, pp. 83–93, 2000.

[9] J. Zheng, C. Huang, M. Chu, F. K. Soong, and W. Ye, "Generalized segment posterior probability for automatic Mandarin pronunciation evaluation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. IV, 2007, pp. 201–204.

[10] S. Kanters, C. Cucchiarini, and H. Strik, "The goodness of pronunciation algorithm: A detailed performance study," *ISCA International Workshop on Speech and Language Technology in Education (SLaTE)*, pp. 49–52, 2009.

[11] S. M. Witt, "Use of speech recognition in computer-assisted language learning," Ph.D. dissertation, University of Cambridge, 1999.

[12] S. M. Witt and S. J. Young, "Phone-level pronunciation scoring and assessment for interactive language learning," *Speech Communication*, vol. 30, no. 2, pp. 95–108, 2000.

[13] L. F. Weigelt, S. J. Sadoff, and J. D. Miller, "Plosive/fricative distinction: The voiceless case," *The Journal of the Acoustical Society of America*, vol. 87, no. 6, pp. 2729–2737, 1990.

[14] H. Strik, K. Truong, F. De Wet, and C. Cucchiarini, "Comparing different approaches for automatic pronunciation error detection," *Speech Communication*, vol. 51, no. 10, pp. 845–852, 2009.

[15] S.-Y. Yoon, M. Hasegawa-Johnson, and R. Sproat, "Landmark-based automated pronunciation error detection," in *11th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2010, pp. 614–617.

[16] S.-Y. Yoon, M. Hasegawa-Johnson, and R. Sproat, "Automated pronunciation scoring using confidence scoring and landmark-based SVM," in *10th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2009, pp. 1903–1906.

[17] X. Yang, A. Loukina, and K. Evanini, "Machine learning approaches to improving pronunciation error detection on an imbalanced corpus," in *IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 300–305.

[18] W. Yong, W. Jingli, and C. Zhou, "The use of 'I think' by Chinese EFL learners: A study revisited," *Chinese Journal of Applied Linguistics*, vol. 33, no. 1, pp. 3–23, 2010.

[19] Y. Ohno, M. Mashimo, A. Lee, H. Kawanami, H. Saruwatari, and K. Shikano, “A study on pronunciation evaluation for English learner using English acoustic models,” in *Proceedings of the Autumn Meeting of Acoustic Society of Japan*, 2002, pp. 1–6.

[20] A. Ito, Y.-L. Lim, M. Suzuki, and S. Makino, “Pronunciation error detection for computer-assisted language learning system based on error rule clustering using a decision tree,” *Acoustical Science and Technology*, vol. 28, no. 2, pp. 131–133, 2007.

[21] S. Witt and S. Young, “Phone-level pronunciation scoring and assessment for interactive language learning,” *Speech Communication*, vol. 30, no. 2, pp. 95–108, 2000.

[22] B. Mak, M. Siu, M. Ng, Y.-C. Tam, Y.-C. Chan, K.-W. Chan, K.-Y. Leung, S. Ho, J. Wong, and J. Lo, “PLASER: Pronunciation learning via automatic speech recognition,” in *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing.* Association for Computational Linguistics, 2003, pp. 23–29.

[23] A. Neri, C. Cucchiarini, and H. Strik, “ASR-based corrective feedback on pronunciation: Does it really work?” in *9th International Conference on Spoken Language Processing (ICSLP)*, 2006, pp. 1982–1985.

[24] J. van Doremalen, C. Cucchiarini, and H. Strik, “Using non-native error patterns to improve pronunciation verification,” in *11th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2010, pp. 590–593.

[25] H. Strik, K. Truong, F. de Wet, and C. Cucchiarini, “Comparing different approaches for automatic pronunciation error detection,” *Speech Communication*, vol. 51, no. 10, pp. 845–852, 2009.

[26] C. Molina, N. B. Yoma, J. Wuth, and H. Vivanco, “ASR based pronunciation evaluation with automatically generated competing vocabulary and classifier fusion,” *Speech Communication*, vol. 51, no. 6, pp. 485–498, 2009.

[27] J. van Doremalen, C. Cucchiarini, H. Strik, and J. V. Doremalen, “Automatic pronunciation error detection in non-native speech: The case of vowel errors in Dutch,” *The Journal of the Acoustical Society of America*, vol. 134, no. 2, pp. 1336–1347, 2013.

[28] S. Wei, G. Hu, Y. Hu, and R.-H. Wang, “A new method for mispronunciation detection using Support Vector Machine based on pronunciation space models,” *Speech Communication*, vol. 51, no. 10, pp. 896–905, 2009.

[29] H. Franco, H. Bratt, R. Rossier, V. Rao Gadde, E. Shriberg, V. Abrash, and K. Precoda, "EduSpeak®: A speech recognition and pronunciation scoring toolkit for computer-aided language learning applications," *Language Testing*, vol. 27, no. 3, pp. 401–418, 2010.

[30] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, pp. 161–168, 2006.

[31] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[32] D. Mease, A. J. Wyner, and A. Buja, "Boosted classification trees and class probability/quantile estimation," *The Journal of Machine Learning Research*, vol. 8, pp. 409–439, 2007.

[33] N. Chawla and K. Bowyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[34] J. Yuan and M. Liberman, "Speaker identification on the SCOTUS corpus," *Journal of the Acoustical Society of America*, vol. 123, no. 5, p. 3878, 2008.

[35] A. Neri, C. Cucchiarini, and H. Strik, "Selecting segmental errors in non-native Dutch for optimal pronunciation training," *International Review of Applied Linguistics (IRAL) in Language Teaching*, vol. 44, no. 4, pp. 357–404, 2006.

[36] R. Weide, "The Carnegie Mellon pronouncing dictionary [cmudict. 0.7b]," *Carnegie Mellon University*, vol. 9, 2015. [Online]. Available: http://www.speech.cs.cmu.edu/cgi-bin/cmudict.Accessed

[37] P. Bonaventura, P. Howarth, and W. Menzel, "Phonetic annotation of a non-native speech corpus," in *Proceedings International Workshop on Integrating Speech Technology in the (Language) Learning and Assistive Interface, InStil.* Citeseer, 2000, pp. 10–17.

[38] T. Cincarek, R. Gruhn, C. Hacker, E. Nöth, and S. Nakamura, "Automatic pronunciation scoring of words and sentences independent from the non-native's first language," *Computer Speech & Language*, vol. 23, no. 1, pp. 65–88, 2009.

[39] D. Higgins, X. Xi, K. Zechner, and D. Williamson, "A three-stage approach to the automated scoring of spontaneous spoken responses," *Computer Speech & Language*, vol. 25, no. 2, pp. 282–306, 2011.

[40] K. Evanini and X. Wang, “Automated speech scoring for non-native middle school students with multiple task types,” in *14th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 2435–2439.

[41] S. J. Young, J. J. Odell, and P. C. Woodland, “Tree-based state tying for high accuracy acoustic modelling,” in *Proceedings of the Workshop on Human Language Technology.* Association for Computational Linguistics, 1994, pp. 307–312.

[42] M. Benzeghiba, R. D. Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, R. Rose, V. Tyagi, and C. Wellekens, “Automatic speech recognition and speech variability: A review,” *Speech Communication*, vol. 49, no. 1011, pp. 763–786, 2007.

[43] B. J. Winer, D. R. Brown, and K. M. Michels, *Statistical Principles in Experimental Design.* McGraw-Hill New York, 1971, vol. 2.

[44] T. J. Hastie, R. J. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2009.

[45] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning.* Springer, 2013.

[46] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[47] D. Blanchard, N. Madnani, M. Heilman, N. M. Llerena, D. M. Napolitano, A. Cahill, K. Evanini, and C. W. Leong, “Scikit-learn laboratory (SKLL) 1.0.0,” 2014. [Online]. Available: http://dx.doi.org/10.5281/zenodo.12825

[48] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., “Scikit-learn: Machine learning in Python,” *The Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[49] K. N. Stevens, *Acoustic Phonetics.* MIT Press, 2000, vol. 30.

[50] K. N. Stevens, “Evidence for the role of acoustic boundaries in the perception of speech sounds,” in *Phonetic Linguistics: Essays in Honor of Peter Ladefoged*, V. A. Fromkin, Ed. Orlando, Florida: Academic Press, 1985, pp. 243–255.

[51] K. N. Stevens, S. Y. Manuel, S. Shattuck-Hufnagel, and S. Liu, "Implementation of a model for lexical access based on features," in *Second International Conference on Spoken Language Processing (ICSLP)*, vol. 1, Banff, Alberta, 1992, pp. 499–502.

[52] X. Yang, X. Kong, M. Hasegawa-Johnson, and Y. Xie, "Landmark-based pronunciation error identification on Chinese learning," *submitted in Speech Prosody*, 2016.

[53] M. Wang and Z. Meng, "Classification of Chinese word-finals based on distinctive feature detection," *Third International Symposium on ElectroAcoustic Technologies (ISEAT)*, vol. 35, no. 9, pp. 38–41, 2011.