Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für
Technische Informatik und
Kommunikationsnetze

Automatic pronounciation verification

Claudia Meder

Semester Thesis SA-2011-28

Computer Engineering and Networks Laboratory

Supervisor:
Dr.B.Pfister und S.Hoffmann

Professor:
Prof. Dr. L. Thiele

23 Dezember 2011

Contents

Summary **2**

1 Introduction **3**

2 Speech recognition with pattern matching **4**

2.1 Principle of pattern matching 4

2.2 Application of pattern matching 4

3 Approaches for improvement **7**

3.1 Speech material 7

3.2 Initial methods 7

3.3 Error causes and improvement 8

3.3.1 DTW Algorithm 8

3.3.2 Blow noise 10

3.3.3 Quiet signal parts 10

3.3.4 Error threshold and minimum error region 11

3.3.5 Sound segmentation 12

4 Evaluation **14**

4.1 Evaluation of approaches of improvement 14

4.2 Evaluation of final approach 14

4.3 Errors of neural network 16

4.4 Pronounciation Verification Application 16

5 Conclusion and outlook **18**

References **19**

A Task **20**

Summary

In this project the use of an automatic pronounciation verification in the context of a computer based-language course was analysed. The work was based on a previous thesis about the use of speech recognition technologies in such a setting. It had implemented methods to compare a recorded student signal with a given teacher signal in order to verify and correct the student's pronounciation. Errors in prosody and sound omission, insertion or substitution would then be localized and reported back to the student in a suitable way. To match the student signal to the teacher signal the approach of pattern matching with a neural network was used. In this project only the pattern matching is of importance, since the neural network developed in the previous thesis was not changed or investigated on. The causes for wrong error decisions were analyzed and approaches to improve performance were tested. The evaluation of the methods showed some improvement of the overall performance of the system, but it clearly showed the limitations of sound differentiation due to the neural network. The network was initially trained for a speech recognition task, and therefore is not suitable for a pronounciation verification as given here. Further investigation with respect to feature extraction and training of the neural network for a pronounciation verification task are necessary to further improve the performance.

1 Introduction

For the purpose of this project, the detection and localisation of errors in a student utterance in the context of a computer-based language course was considered. For a student using a computer-based language course who has to repeat a word or phrase after hearing it said by a teacher voice, having a feedback about whether and where he made a mistake in pronounciation could give him the possibility to improve his pronounciation of certain sounds or words more specifically. The learning effect of such a language course for pronounciation could be greatly increased.
The automatic pronounciation verification in a computer based-language course should give the student a feedback about whether and where he made a mistake in his pronounciation of a word that was just played to him from a teacher voice. It would be in the students interest to keep the false positive rate, meaning an error is detected when the student utterance was correct, to a minimum, because else it can easily get frustrating for the student. Having a higher false negative rate would not hurt him so much, since that only means that some errors he makes are not detected by the system. Furthermore the error detection needs to run stably, meaning the same utterance said in a similar way should always produce the same feedback, because having different errors for every repetition of the utterance would only confuse the student instead of helping him. The methods implemented should also provide a suitable way of giving the student feedback about his utterance, so that he can work specifically on his pronounciation of certain sounds or words.
This project is based on a previous diploma thesis [1], which dealt with the topic of speech processing technologies for computer-based language courses. For the purpose of that thesis methods were implemented to compare a student utterance with the corresponding teacher signal and detect errors in pronounciation. The suitability of these methods for the given task were not fully analysed. In this project the methods based on the previous thesis were tested and improved.
In chapter 2 the concept of pattern matching as well as its specific application to this task will be depicted in detail. In chapter 3 the results of the analysis of the previously implemented methods and approaches for improvement will be discussed. The evaluation of these approaches will then be described in chapter 4. The conclusions from the evaluation and possibilities for further projects will be presented in chapter 6.

2 Speech recognition with pattern matching

2.1 Principle of pattern matching

In order to compare two speech signals it is necessary to determine a measure of how much they differ, independent of prosody and signal intensity. This measure will in the following simply be called distance. In order to do so the relevant features have to be extracted from segments of the signal, so that it can then be represented by a sequence of feature vectors, usually refered to as speech pattern. These features of course need to be independent of the speakers voice characteristics and the signal intensity and only give a measure of similarity of the spoken sound. The signals not only differ in intensity but ususally als differ in length, even if they are the same utterance of the same speaker. Therefore they have to be time matched in a way that the corresponding features in each signal coincide. This cannot simply be done linearly since the length of each sound differs from speaker to speaker. Especially the length of a vowel in a signal can differ greatly, whereas plosives are so short that they usually don't differ much. This calls for a way to stretch and compress signal parts so that the final signal speech pattern compares the correct features with each other. This is done using the Dynamic Time Warping Algorithm which will be described in general in the follow sections.A more precise description can be found in [2].After this time matching of the signals the distance can then be computed from the distance of the corresponding signal features in each analysis window.

2.2 Application of pattern matching

The task of matching two speech patterns in time is illustrated using the teacher pattern $X = x_1, ...x_n$ and the student pattern $Y = y_1, ...y_m$ which were extracted from a student and a teacher signal. The idea of a dynamic time matching is to locally change the time axis of the patterns X and Y so that the minimum distance of the corresponding patterns can be obtained. Plotting the sequence of feature vectors of X on the x axis and the sequence Y on the y axis results in a matrix of all possible matching of a feature vector from X to a feature vector from Y. The sequence of such assignments can be described using a $warping\ curve\ W(.)$.

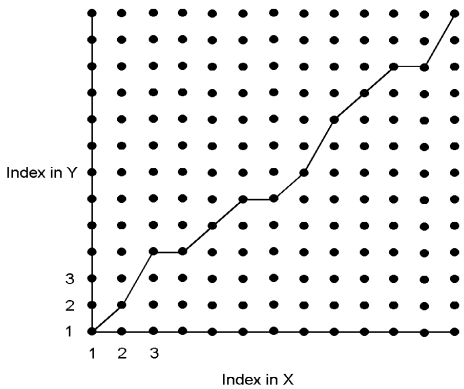

Figure 1: *Illustration of pattern matching between a student and teacher signal using the DTW Algorithm*

$$W(k) = (i(k), j(k)), \quad 1 \leq k \leq T \tag{1}$$

Where $i(.)$ denotes the index in X and $j(.)$ the index in Y, and T the length of the warping curve.An example of such a Warping curve is depicted in Figure 1. The corresponding distance between the two feature vectors is denoted as $d(x_i, y_j)$, which leads to the total distance of the warping curve $W(.)$ being:

$$D_W(X, Y) = \frac{\sum_{k=1}^{T} d(W(k))w(k)}{\sum_{k=1}^{T} w(k)} \tag{2}$$

In this application the weights are not used therefore $w(k) = 1$.
The Dynamic Time Warping Algorithm then finds a warping curve which minimizes the total distance so that

$$D(X, Y) = \min_{W} D_W(X, Y) \tag{3}$$

$W(.)$ has to fullfill certain boundary conditions, because else it would be possible that the Warping curve would flip one of the signals or skip great lengths of it. That's why the path extensions are introduced. They determine the boundary conditions and the steps the warping curve can take in any point (X, Y). In order to compensate for an incorrect utterance detection in the student signal, the warping curve can go vertically or horizontally for a certain amount of steps in the beginning and the end. That way the beginning or end of one of the signals can be omitted. Other than that, only certain path extensions as depicted in Figure 2 are allowed.

1. This path extension corresponds to a stretching of the student speech signal for the case that the student utterance is faster than the teacher. In order to limit the allowed speed up of the student signal this path extension can not be applied twice in a row.

2. This path extension does not change the student or the teacher signal in the area of interest.

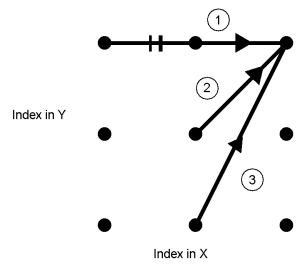

Figure 2: *Illustration of allowed path extensions for the DTW Algorithm*

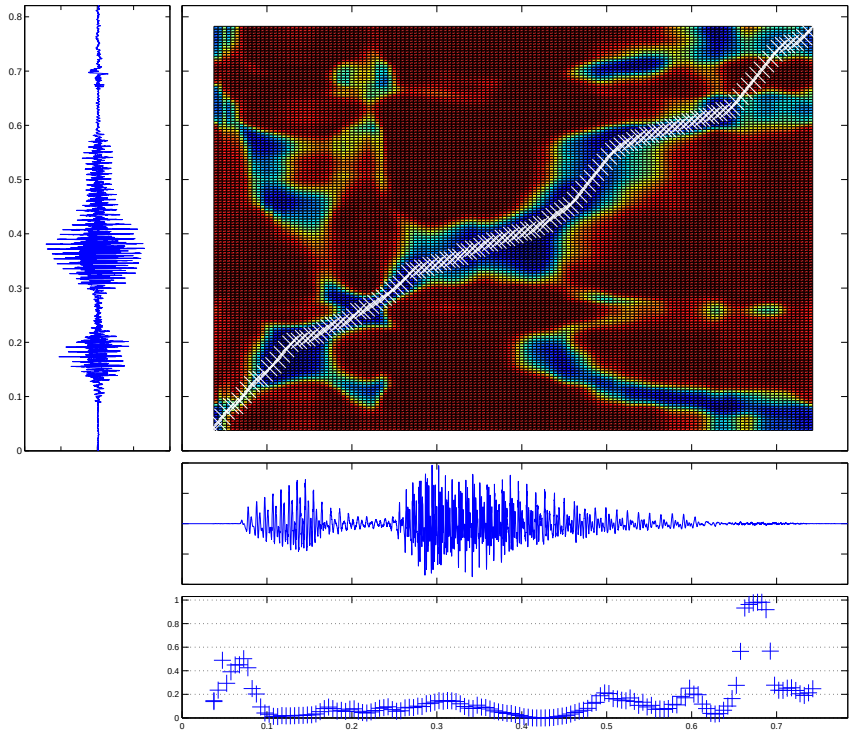

Figure 3: *Example of the distance matrix for the word 'evade'*

3. The third path extension allows to skip one feature vector of the student signal in order to compensate for a slow student utterance. It can also be used to skip short noisy parts of the student utterance. This is usually not necessary for the teacher signal, since it is assumed to be recorded properly and without interfering noise.

In this application not only the warping curve but also the corresponding distances are of interest, therefore as so called distance matrix is used. It contains the local distances in all points (x, y) whereas red corresponds to high distances and blue to low distances. Furthermore the student and teacher signals are plotted in below and to the left of the distance matrix, which makes it easier to identify the signal parts that are assigned to each other by the warping. An example of such a distance matrix is given in Figure 3.

3 Approaches for improvement

3.1 Speech material

For the analysis of the existing methods 50 words from six speakers repeated once as correctly as possible and once with a delibarate error were used. Speaker two, three and four were native Swiss German speking men, speaker five a Swiss German speaking woman. Speaker six was a natve German speaking woman, speaker seven a native Persian speaking man. This resulted in 600 speech signals to be used for the analysis, whereas in the initial phase of the project only speaker two through four were used. These signals were already recorded during the previous thesis, and therefore speakers five through seven commited the same deliberate errors in their recordings, although these do not represent very common pronounciation errors a student in a language course would make. For the analysis of the different approaches all signals were used. The three initial speakers were also used for the training of the neural network, which further motivated recording more signals from speakers outside of the set of training voices. It can easily be observed that these speakers performed much better in the error analysis.
The neural network used was trained in the previous thesis. In the following the parameters for the training and feature extraction are summarized.

1. The training data consisted of 200 words recorded by six male speakers of which two were native English speakers.
2. 12 MFCCs and 14 filters were used.
3. The analysis window was 75ms and the window shift 10ms.

3.2 Initial methods

The initial methods for error detection were based on performing dynamic time warping to match the student and the teacher signals, and then finding error regions from the distance metric along the warping curve. An error region was detected if at least six consecutive entries of the distance curve were higher than 0.6. These regions were then reported back. Since it was decided that a sensible feedback would only be one error per signal, for this analysis of the initial methods already only the most significant error was used. This error region was determined by the maximum of the sum over the local distances of the individual error regions.

	false negative	false positive	wrong error	total
speaker 2	13	19	8	40
speaker 3	14	7	7	28
speaker 4	9	4	7	20
speaker 5	6	31	14	51
speaker 6	7	17	20	44
speaker 7	8	23	20	51
total	57	101	76	234 (39

Table 1: *Error analysis of initial methods*

For the error analysis of the speech material up to two errors were marked in each signal and then the pronounciation verification was applied to the signal. If there was no error marked and no error detected, or if the marked error was located in the maximum error region the signal was counted as correctly analysed. The errenously detected signals will in the following be classified as false negative (no error detected although an error was marked), false positive (no error was marked but an error was detected) and or wrong error (another error than the one that was marked was detected).
As shown in Table 1 the error rate is almost 40 % with a very high false negative rate, which is very undesirable for the given application.

3.3 Error causes and improvement

In the following the main causes for misdetermination of pronounciation errors that were found in the existing methods and approaches for improvement thereof will be discussed.

3.3.1 DTW Algorithm

The boundary conditions of the Dynamic Time Warping Algorithm give the possibility for the warping curve to go horizontally or vertically in the beginning and end for a certain amount of steps. This is done to compensate a possible inaccurate utterance detection of the student utterance. Initially this condition was set to 5 steps and led to some errors at the beginning and end of the warping curve. The boundary conditions of the Dynamic Time Warping Algorithm were loosened to allow the warping curve to go horizontally of vertically for an incresed number of steps, to compensate for an incorrect utterance detection (see boundary condition 10,15,20 in Table 3).
This loosening of the boundary condition will cause high distance entries in the horizontal and vertical parts of the warping curve because the first or last part of one of the signals is compared to the first or last analysis window of the other signal, which is not of any interest for error detection. Therefore another measure to suppress errors originating from this horizontal or vertical part of the warping curve had to be taken. In the first case the corresponding distance entries were all set to zero (see start end suppression 0 in Table 3) in the other case this was done with a linear increase over the last 5 entries of the horizontal or vertical part(see start end suppression window) to take into account the long window size, which causes the distance entries in one point to be influenced significantly by the adjacent sound. This had to be done after the actual warping because else a warping curve with a long horizontal or vertical start and end would in most cases be preferrable due to the zero distances in these regions over the actual warping curve. That means the warping curve was obtained using the loosened boundary condition and afterwards the distance curve was weighted in the beginning and end as previously described.
An example of this problem with the corresponding approach is given in Figures 4 and 5.

Other than that, the warping worked correctly even if sounds were replaced, omitted or new ones included in a word. If the utterance detection is done very poorly and the student and teacher signals differ by more than a factor of two, the warping doesn't work anymore because of the limits of the slope of the warping curve. In this case no error detection is done and a suitable feedback to the student would make sense.

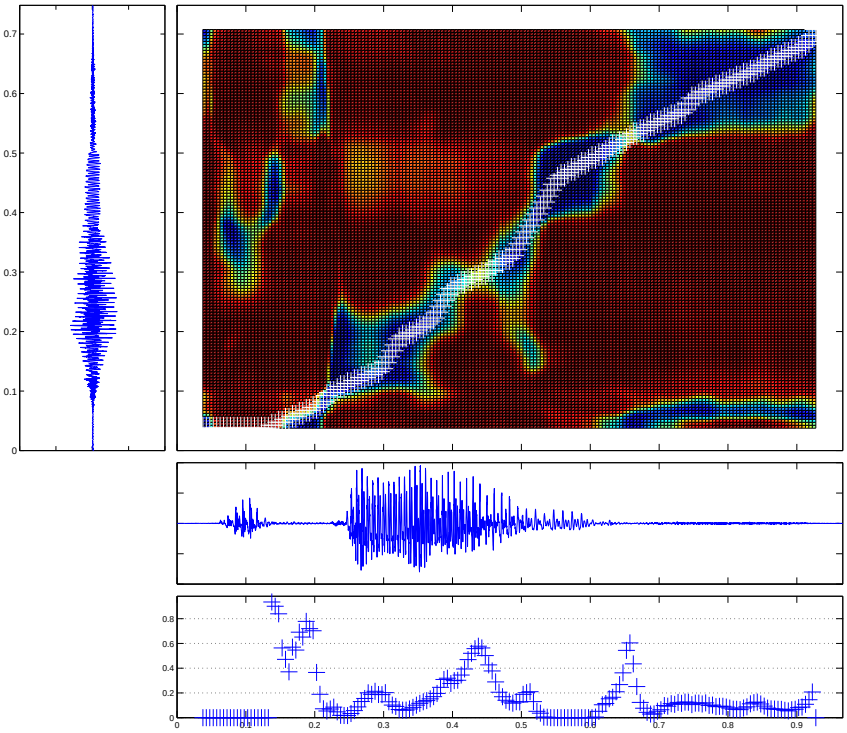

Figure 4: *Warping for a signal with bad utterane detection for the word 'appearance'*

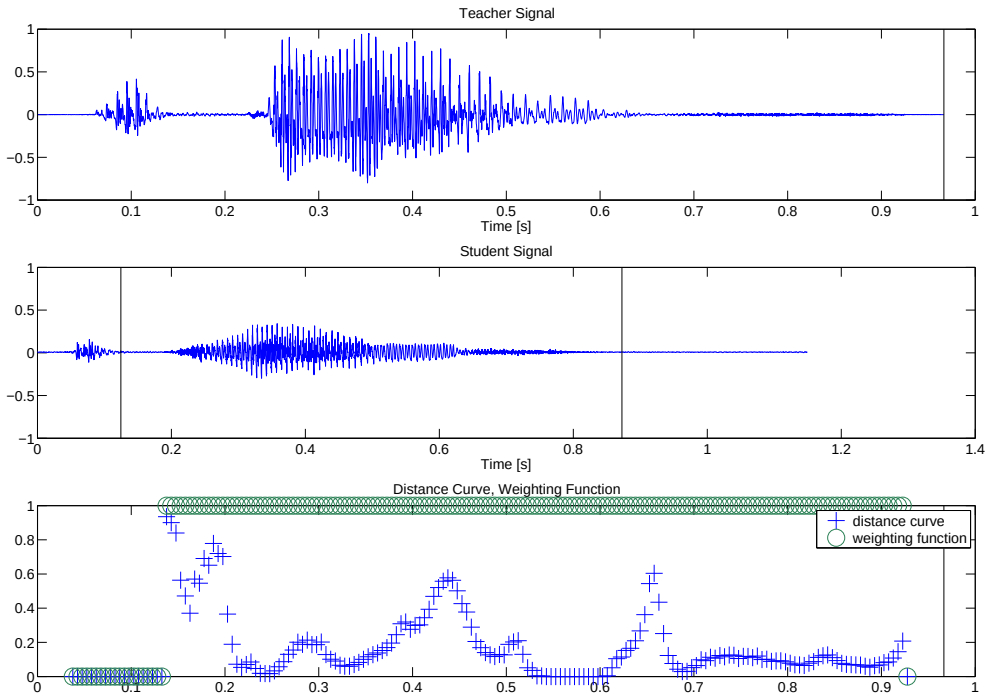

Figure 5: *Suppression of distance at beginning of warping for the word 'appearance'*

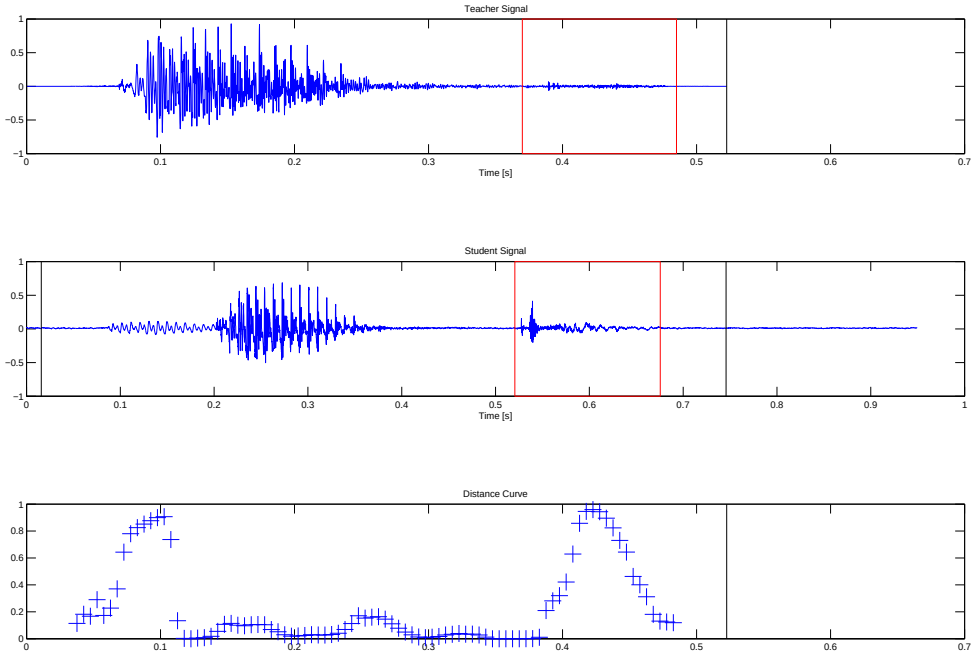

Figure 6: *False positive error due to blow noise in student signal for the word 'duck'*

3.3.2 Blow noise

If the student has the microphone positioned in front of his mouth clearly audible blow noise will be recorded, which leads to significant errors in the pattern matching. This can be seen in Figure 6 Some of the blow noise can be filtered out right after recording, additionally a blow noise detection was introduced to minimize the errors caused by blow noise. If blow noise is detected in an error region this error is discarded from the error analysis. It is assumed that in this region there was no error made but that only the blow noise caused this error. A feedback to the student to adjust the position of his microphone accordingly makes sense in this case.

3.3.3 Quiet signal parts

For quiet signal parts like a preplosive pause or a pause in between two words, the features for the distance metric can only be extracted from the background noise. Since the spectral composition of two different signals usually strongly differs, the features extracted from the signals may differ greatly in these parts, which leads to a large distance. Since in general these signal parts are of no interest for the pronounciation verification, the distances have to be filtered out in the quiet parts of the signal, so that these parts cannot cause errors.
An additional weighting scaled the distance in quiet signal parts according to the RMS signal intensity of the student and the teacher signals. If either the teacher or the student RMS signal intensity lies below a certain threshold the corresponding entry in the distance curve is scaled linearly with the RMS intensity value of that signal. The results from this approach are found

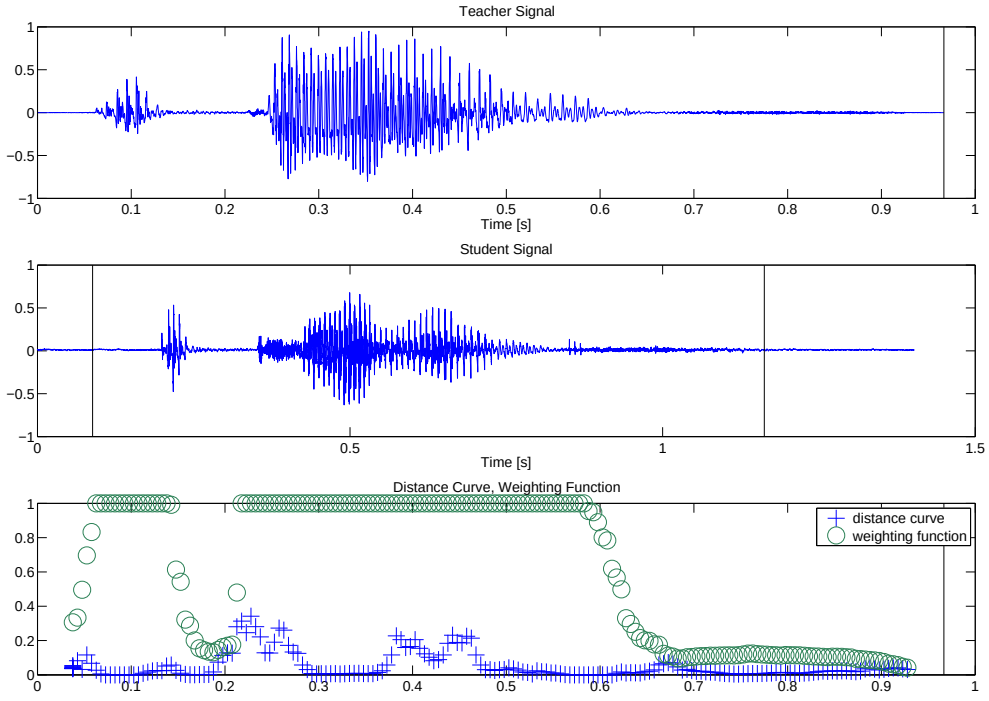

Figure 7: *Suppression of quiet signal parts according to RMS teacher and student signal intensity for the word 'appearance'*

under low suppression in Table 3.
An example of such a weighting of the quiet signal parts is shown in Figure 7.

3.3.4 Error threshold and minimum error region

Because the error threshold was fixed to a distance of 0.6 over at least 6 entries in the distance curve, errors for short sounds like plosives are hard to detect. The window length for the feature extraction is rather long, and so short sounds or errors only affect a few distance entries, since they are still influenced by great parts by the adjacent sounds. Changing the error threshold and minimum error length to another fixed value is not very promising, because this would only cause longer sounds to yield more errors, therefore causing undesired fault negative errors. The condition of 6 consecutive distance entries being higher than 0.6 can be loosened to allow one entry in the middle to be a bit lower. Adjusting the error threshold according to the average distance can be used to increase performance of the error localisation for rather badly spoken student utterances. Very poor utterances can be reported back to the student as insufficient according to the overall distance in the signal. These measures will not be discussed further since they proved to be very ineffective.

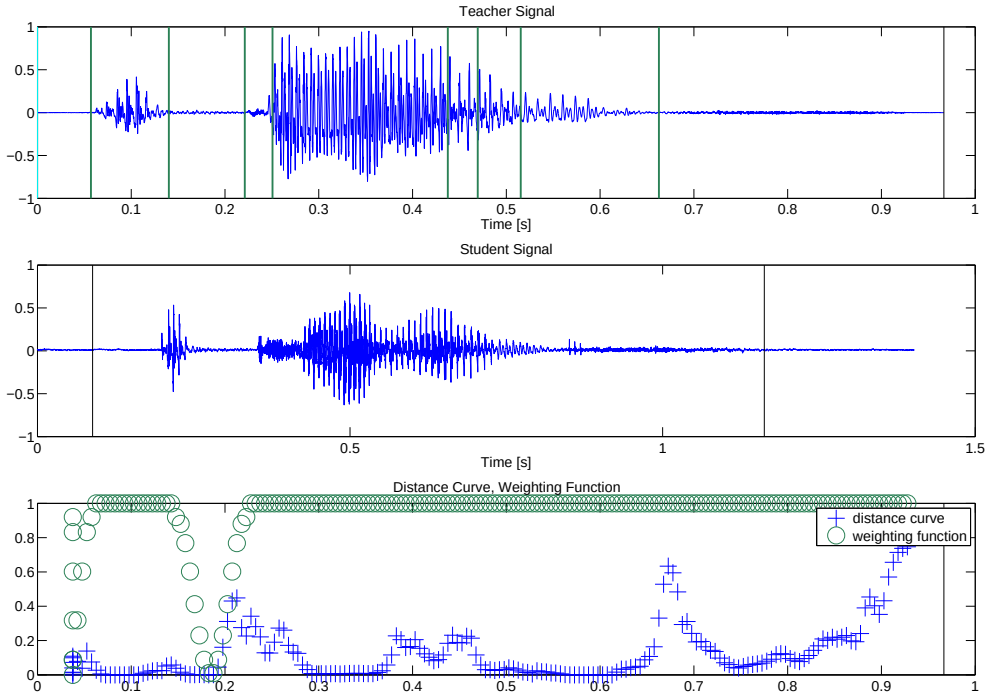

Figure 8: *Suppression of preplosive pauses and quiet signal parts accoring to sound segmentation for the word 'appearance'*

3.3.5 Sound segmentation

A sound segmentation was introduced for an error analysis based on distinction of different sound classes, which is shown in Table 2. This showed clearly that plosives were hard to detect correctly, as was mentioned before because of their short duration. Also because of the long analysis window they are greatly influenced by neighboring sounds. Furthermore the voiced and unvoiced f, meaning the distinction between /f/ and /v/ was very poor. Also a substitution of /w/ for /v/ was hardly ever detected. The discrimation of the different pronounciations of a was also insufficient, as was a substitution of `i:` for `3:`. The sounds /n/,/m/,/ŋ/,/s/,/z/ and /l/ yield a very high false positive rate, and the sound /a͡ʊ/ was very badly differentiated. These were the most significant classes where errenous decision were made.

In order to improve error performance the sound segmentation was used to determine the error threshold and minimum error region according to the current sound.The sounds [/n/,/m/,/ŋ/,/s/,/z/] yield a very high false positive rate, and therefore a higher threshold for these sounds was introduced. These sounds were also considered as not very critical in respect to pronounciation, so that not detecting errors for theses sounds would not degrade the oveall performance of the system significantly. Of course different choices for sound specific threshold or minimum error region could be taken, this one was only chosen as one possibility in order to evaluate the approach for this setting. The altering of the error threshold was both implemented with a rectangular window and a hamming window per sound. The results are labeled error threshold sound and error threshold sound window in Table 3.

Furthermore the sound segmentation was used to filter the quiet signal parts by weighting the

distance curve according to the previously identified preplosive pauses and quiet parts at the beginning and end of the signal. This would compared to the previous low suppression not affect sounds with low signal intensity. The weighting curve was in one case set to zero for all entries where a preplosive pause or silence occured (low suppression sound), and in the other case these signal parts were filtered out with an hamming window (low supression sound window). An example of such a suppression of preplosive pauses and other quiet signal parts can is shown in Figure 8.

sounds	false negative	false positive
plosives	51	134
[A:,q,@]	45	35
[v,f,w]	21	18
[n,m,N]	2	17
[l]	1	9
[3:]	18	7
[s,z]	0	18
[a_U]	11	7

Table 2: *Error analysis per sound*

4 Evaluation

4.1 Evaluation of approaches of improvement

Each of the approaches of improvement was analyzed for all speakers. Although there were great differences between the speakers, only the sum of errorenous decisions for all speakers was counted and minimized. Only the overall improvement for the given approach was of interest, since it should in the end work optimally for a great variety of speakers. The effect of the different measurements taken can be observed in Table 3.
Adjusting the boundary condition of the DTW Algorithm only slightly increased the performance, due to the mentioned increased error region in the vertical and horizontal parts of the warping curve. With the combination of a 20 step boundary condition and suppression of the horizontal and vertical parts of the warping curve a significant improvement of the performance was achieved. The blow noise detection did not improve much, since the blow noise is very hard to differentiate from some aspirated sounds. With further investment in this method, better results might be achieved. The suppression of quiet signal parts based on the teacher and student RMS signal intensities showed great improvement over the initial method, although very onesided towards a high false negative and a low false positive rate. Although desired this case might be a bit too extreme, since a pronounciation verification that does not detect errors anymore is not desirable. The error analysis based on the sound segmentation did not prove to be as efficient as hoped, which is most likely due to the very large window size used. An error threshold or weighting of the signal based on the sound marked in the segmentation is in principle very accurate and effective, with a large analysis window this effect diminishes, because the analysis window overlaps to one or even more other sounds and therefore affects the error detections for the other sounds as well. Filtering out preplosive pauses even with a smooth window as the hamming window used here, will most likely filter out most of the plosive too, and further increases the problem of pronounciation verification for plosives. The same is true for adjusting the error threshold for certain sounds. Since this not only alters the error threshold for the particular sound but as in the case of [/n/,/m/,/ŋ/,/s/,/z/] used here also the neighbouring sounds were affected, which in consequence did not improve performance for this approach.

4.2 Evaluation of final approach

The error analysis clarly showed that the sound segmentation was not useful in combination with the neural network used in this project. Therefore the final approach was limited to the adjustment of boundary condition with corresponding error suppression for the start and end point, the blow noise detection and a suppression of quiet signal parts based on the RMS signal intensity. The results from the combination of all these improvements are shown in Table 4. The final approach relatively decreased the incorrect decisions by approximately 10 % over the inital approach. The false positive rate clearly decreased, but on the contrary the false negative rate is now with 40% very high. This means an error is only detected in a little more than half of the cases. This of course is not desirable for a pronounciation verification task.

	false negative	false positive	wrong error	total
initial methods	57	101	76	234
boundary warping (10)	62	99	69	230
boundary warping (15)	63	99	68	230
boundary warping(20)	64	97	67	228
blow noise detection	57	98	77	232
start end suppression (boundary 20)	69	84	63	216
start end suppression window (boundary 20)	78	80	64	222
low suppression	113	44	59	216
low suppression sound	90	86	71	247
low suppression sound window	86	79	74	239
error threshold sound	57	99	78	234
error threshold sound window	57	98	75	230
final approach	122	51	39	212

Table 3: *Error analysis of different approaches*

	false negative	false positive	wrong error	total
speaker 2	22	6	8	36
speaker 3	20	3	7	30
speaker 4	17	1	6	24
speaker 5	19	18	11	48
speaker 6	25	4	5	34
speaker 7	19	7	14	40
total	122	51	39	212 (35%)

Table 4: *Error analysis of final approach*

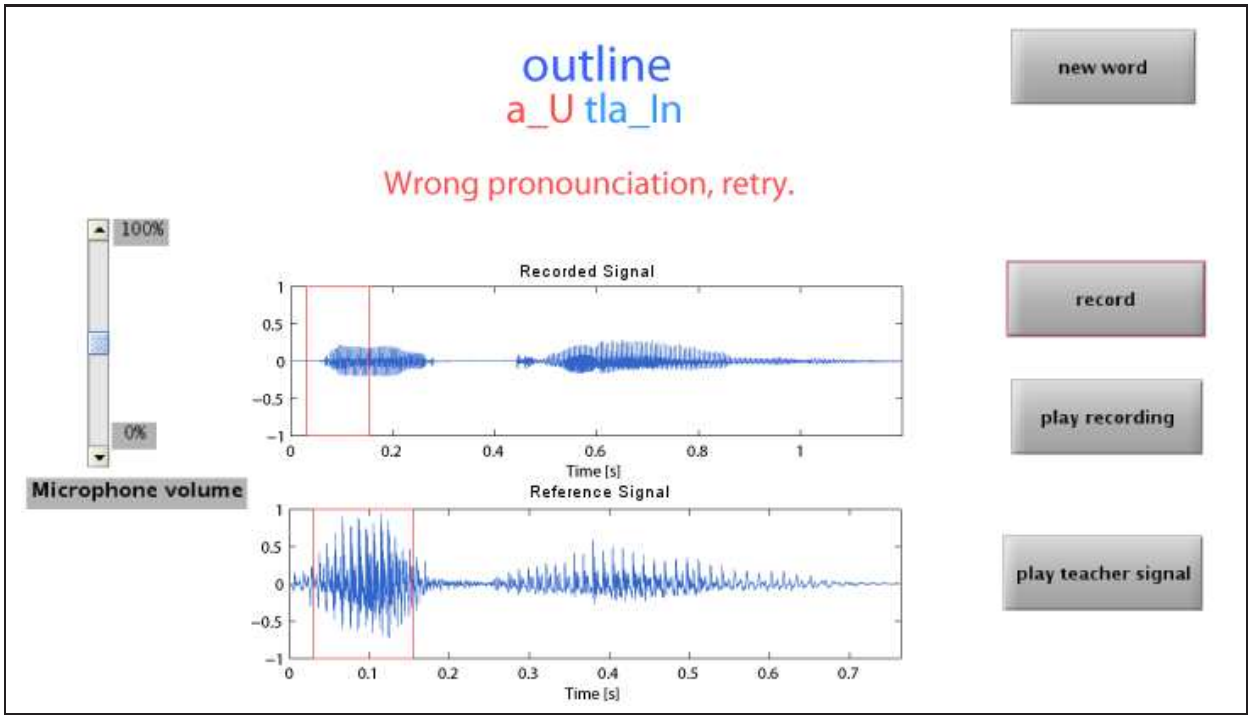

Figure 9: *Pronounciation Verification Application*

4.3 Errors of neural network

The neural network used for this project was developed and trained in a previous thesis, and therefore not changed or investigated upon. A great number of errors were caused by an insufficient sound discrimination of the neural network. Also the long analysis window used was suboptimal for the task. In the previous thesis, a network was designed and optimized for a speech recognition task of finding out wheter the student utterance matched a given teacher signal most of all possible answers. A similarly trained network was used for this pronounciation verification task, for which the network clearly is not suitable or optimal. The analysis of errors per sound or group of sounds clearly shows that some sounds cannot be differentiated by the network. Using a different network from the set of available networks from the previous thesis containing also a female voice in the training data did not improve the performance on female speakers. Furthermore an additional solution for the detection of voiced and unvoiced f needs to be found in order to make pronounciation verifiation of these sounds reliable, because this was not the case in this application.

4.4 Pronounciation Verification Application

A simple example application was developed based on the improved pronounciation verification. The student repeats a word he sees displayed as a word and also in phonetic script. The pronounciation verification gives a feedback about whether the signal recorded was too long (due to bad utterance detection), too badly pronounced or blow noise was detected and he needs to adjust the microphone. In these cases the utterance has to be repeated. Otherwise the student sees the signal he just recorded above to the teacher signal and the errenous region is marked

in his signal. Furthermore the errenous phones are marked in red, in order to give a specific feedback to the student. The Application is shown in Figure 9.
Using the application proved that only grave errors were detected, which was expected from the previous error analysis, but a correctly spoken signal was at least after two or three tries accepted as spoken correctly. This means the application could be used for beginners in an English course to detect very strong errors and give an adequate feedback, but for advanced students it would certainly not be very helpful.

5 Conclusion and outlook

The approaches for improved pronounciation verification based on the neural network given from the previous thesis led to a relative reduction of errenous verification decisions by approximately 10%, which is not sufficient to make the pronounciation verification reliable for the environment it was designed for, the application only detects obvious errors and therefore will only give relevant feedback to people commiting grave pronounciation errors. For advanced students it would most likely not be suitable.
The sound differentiation was strongly limited by the neural network, which is the main cause for wrong decision and needs to further be investigated upon. First of all, the training data needs to have precise pronounciation, most likely meaning professional speakers, in order not to train the data to the typical pronounciation errors any non native speaker would commit, such as the wrong pronounciation of /æ/,/ɑ/ and /ə/. Second of all the analysis window needs to be much shorter in order to make use of the sound segmentation in order to filter quiet signal parts and vary the error threshold based on the current sound. Furthermore since the error analysis for the speakers contained in the training data proved to be much better the speaker independence of the network needs to be questioned, and the number of speakers used for training thus should possibly be increased. Since the sound segmentation for this project only consisted of the segmentation of approximately 400 signals it was done by hand, for a further investigation on pronounciation verification using a sound segmentation and automatic sound segmentation for the teacher voices needs to be developed.

References

[1] S. Müller. Sprachverarbeitungstechnologien für die comptergestützte Sprechschulung. Master's thesis, ETH Zürich, 2008. (DA-2008-05).

[2] B.Pfister und T.Kaufmann. *Sprachverarbeitung: Grundlagen und Methoden der Sprachsynthese und Sprachverarbeitung*. Springer Verlag, 2008.

A Task

(SA-2011-28)

SEMESTERARBEIT

für

Frau Claudia Meder

Betreuer: Dr. B. Pfister, ETZ D97.6
S. Hoffmann, ETZ D97.5

Ausgabe: 19. September 2010
Abgabe: 23. Dezember 2010

Automatische Überprüfung der Aussprache

Einleitung

Für das Lernen von Fremdsprachen werden in zunehmendem Masse computerbasierte Übungen eingesetzt, mit denen auch die mündliche Kommunikation trainiert werden kann. So gibt es bereits verschiedene Sprachkurse, die Dank des Einsatzes einer Spracherkennung mit den Lernenden einfache mündliche Dialoge führen können.

Wünschbar wäre, wenn der Computer auch überprüfen könnte, ob ein Schüler die Laute, Wörter und Sätze korrekt ausspricht und allenfalls mitteilt, was nicht korrekt gesprochen worden ist. Dies scheint jedoch schwieriger zu sein. Jedenfalls vermögen Sprachkurse, die mit einer automatischen Überprüfung der Aussprache ausgestattet sind, in dieser Hinsicht noch nicht zu überzeugen.

Grundsätzliches zur Aussprache

Bei der gesprochenen Sprache unterscheidet man zwischen der segmentalen und der suprasegmentalen Ebene, oder anders ausgedrückt zwischen den Lauten und der Prosodie (siehe z.B. [1], Kapitel 1.2). Wenn nun zu beurteilen ist, ob eine Äusserung korrekt artikuliert worden ist, dann müssen beide Ebenen in Betracht gezogen werden.

Hinsichtlich der Laute ist die Beurteilung grundsätzlich einfach: die aus einer Äusserung

wahrgenommene Abfolge von Lauten muss der korrekten Aussprache der betreffenden Wörter entsprechen und die einzelnen Laute müssen identifizierbar sein. Im konkreten Fall sind Laute jedoch schwierig zu beurteilen, weil sie einerseits von der Sprache abhängig sind, anderseits von der Stimmcharakteristik der Person geprägt werden und zudem durch benachbarte Laute beeinflusst werden können. Instanzen eines Lautes können somit sehr verschieden sein und trotzdem völlig klar und eindeutig als Beispiele dieses Lautes wahrgenommen werden.

Hinsichtlich der Prosodie ist die Beurteilung eher noch schwieriger. Die Prosodie umfasst Aspekte wie Sprechrhythmus, Sprechmelodie, Betonung, Gruppierung usw. Im Sprachsignal wird die Prosodie mit den physikalischen (im Sprachsignal messbaren) Grössen Grundfrequenz, Lautdauer und Intensität ausgedrückt. Dabei hängt die Prosodie stark von der Intention der sprechenden Person und vom Kontext ab, in dem eine bestimmte Äusserung gemacht wird. Zudem haben beispielsweise Geschlecht und Alter einen grossen Einfluss auf die Grundfrequenz. Es ist somit oft eine Ermessensfrage, ob in einem bestimmten Kontext die Prosodie angemessen ist oder nicht.

Problemstellung

Das Beurteilen sowohl der lautlichen als auch der prosodischen Korrektheit sprachlicher Äusserungen ist folglich keine einfach zu beschreibende Aufgabe. Es gibt keine relativ einfachen Kriterien, anhand derer sich diese Frage entscheiden liesse.

In dieser Semesterarbeit soll deshalb ein Ansatz zur automatischen Überprüfung der Aussprache untersucht werden, der auf die spezielle Begebenheit eines Sprachkurses zugeschnitten ist. Es kann nämlich davon ausgegangen werden, dass für eine zu beurteilende Äusserung einer Wortfolge mehrere als korrekt bekannte Äusserungen dieser Wortfolge vorliegen. Diese als korrekt bekannten Äusserungen sind von den Lehrern gesprochen worden, deren Stimmen im Sprachkurs zu hören sind.

In diesem Fall kann somit zur Beurteilung der Korrektheit der Aussprache ein Verfahren eingesetzt werden, das auf einem Mustervergleich basiert (vergl. [1], Kapitel 11). Die Idee ist, die Korrektheit einer Äusserung anhand geeigneter Merkmale zu beurteilen, z.B. aufgrund bestimmter Eigenheiten der Warping-Kurve (siehe [2], Seiten 45 und 46), der Grösse der lokalen Distanz, des zeitlichen Verlaufs der Grundfrequenz usw.

Vorgehen

Für diese Semesterarbeit wird das folgende Vorgehen empfohlen:

1. Arbeiten Sie sich zuerst in die Thematik ein. Machen Sie sich mit der Merkmalsextraktion und mit dem DTW-Algorithmus (*dynamic time warping*) vertraut. Dies sind die wichtigsten Grundlagen für den Vergleich von Sprachmustern. Studieren Sie insbesondere die Kapitel 10.7 und 11.1 in [1] und lösen Sie die Übung 12 aus der Reihe der Sprachverarbeitungsübungen.

2. Studieren Sie Kapitel 5.3 “Überprüfung der Aussprache” der Diplomarbeit [2]. Schauen Sie die zugehörigen Programme und Daten an und führen Sie damit ei-

nige Experimente durch um herauszufinden, welche Arten von Aussprachemängeln mit dem gewählten Ansatz gut bzw. schlecht erkannt werden.

3. Erarbeiten Sie aufgrund der Ergebnisse von Punkt 2 Verbesserungsmöglichkeiten und besprechen Sie diese mit Ihren Betreuern. Legen Sie zusammen mit Ihren Betreuern fest, welche Verbesserungsmöglichkeiten im Rahmen Ihrer Arbeit weiterverfolgt werden sollen. Überlegen Sie auch, welche Programme und Daten für die entsprechenden Untersuchungen nötig sind.

4. Verwirklichen Sie diese Programme und akquirieren Sie die nötigen Daten. Führen Sie geeignete Tests durch, um die Stärken und die Schwachstellen der neuen Ansätze zu ermitteln.

Die ausgeführten Arbeiten und die erhaltenen Resultate sind in einem Bericht zu dokumentieren (siehe dazu [3]), der in gedruckter und in elektronischer Form (als PDF-Datei) abzugeben ist. Zusätzlich sind im Rahmen eines Kolloquiums zwei Präsentationen vorgesehen: etwa zwei Wochen nach Beginn soll der Arbeitsplan und am Ende der Arbeit die Resultate vorgestellt werden. Die Termine werden später bekannt gegeben.

Literaturverzeichnis

[1] B. Pfister und T. Kaufmann. *Sprachverarbeitung: Grundlagen und Methoden der Sprachsynthese und Spracherkennung.* Springer Verlag (ISBN: 978-3-540-75909-6), 2008.

[2] S. Müller. Sprachverarbeitungstechnologien für die computergestützte Sprechschulung, 2008. Diplomarbeit am Institut für Technische Informatik und Kommunikationsnetze, ETH Zürich (DA-2008-05).

[3] B. Pfister. *Richtlinien für das Verfassen des Berichtes zu einer Semester- oder Diplomarbeit.* Institut TIK, ETH Zürich, Februar 2009.
(http://www.tik.ee.ethz.ch/spr/SADA/richtlinien_bericht.pdf).

[4] B. Pfister. *Hinweise für die Präsentation der Semester- oder Diplomarbeit.* Institut TIK, ETH Zürich, März 2004.
(http://www.tik.ee.ethz.ch/spr/SADA/hinweise_praesentation.pdf).

Zürich, den 19. Dezember 2011