DYNAMIC TIME WARPING APPLIED TO DETECTION OF CONFUSABLE WORD PAIRS IN AUTOMATIC SPEECH RECOGNITION

Jan Anguita Ortega, Francisco Javier Hernando Pericás

{jan,javier}@talp.upc.es (TALP Research Center)
Universitat Politècnica de Catalunya, Barcelona, Spain

ABSTRACT

In this paper we present a method to predict if two words are likely to be confused by an Automatic Speech Recognition (ASR) system. This method is based on the classical Dynamic Time Warping (DTW) technique. This technique, which is usually used in ASR to measure the distance between two speech signals, is used here to calculate the distance between two words. With this distance the words are classified as confusable or not confusable using a threshold. We have tested the method in a classical false acceptance/false rejection framework and the Equal Error Rate (EER) was measured to be less than 3%.

1. INTRODUCTION

Using speech to communicate with the machines is a great improvement since it has a lot of advantages: speech is the natural way of communication for humans, speaking is faster than typing, while speaking hands and eyes are free for other tasks, some channels (Phone) are made for speech, etc. Unfortunately, although Automatic Speech Recognition (ASR) technology is already mature enough for some consumer products, in order to obtain acceptable performances the vocabulary and the structure of the sentences that the system is able to recognize must be limited. Even so, the systems make errors. These errors are sometimes caused by the words of the vocabulary that are phonetically similar. Therefore, the error rate can be reduced by designing the vocabulary with words as less similar as possible. The aim of this study is to design a method to predict if two words are likely to be confused by an ASR system. A tool like this can help to design the vocabulary of a speech recognition system since it can warn the designer if two words are too similar, and sometimes one of them can be changed for another one with the same meaning but less similar.

For example, suppose that we want a speech recognition system for a mail application, where the user will be able to control all the options by his voice. First of all, we have to define the vocabulary that the system will be able to recognize. For example, suppose that we chose the following words:

- Supprimer: to delete a message.
- Imprimer: to print a message.
- Envoyer: to send a message.
- Lire: to read a message.
- Ecrire: to write a message.
- Suivant: to go to the next message.
- Précédent: to go to the previous message.

We have chosen French words because in this project we have worked in French. Once the vocabulary is chosen we have to define the syntax, i.e. the structure of the sentences the system will recognize. In this case isolated words is enough. This means that the user can only say one word each time, preceded and followed by a silence. He cannot say supprimer suivant for example. This application may seem very simple but, as we have already said, we must do these simplifications because, nowadays, the ASR technology is not good enough to let the user say what he wants and how he wants. Even if this system is very simple it will make errors. Sometimes the user will say one word and the system will recognize another one. This is very dangerous because, imagine that the

user says lire and the system understands supprimer. The message is lost forever. Therefore, these errors must be reduced as much as possible. Imagine that, when the system is already in use, we realize that it often confuses two words, for example supprimer and imprimer. If we would have known this when we were designing the vocabulary of the system, we could have changed one word by a synonym, for example supprimer by effacer. In this case the application would have been exactly the same and we would have avoided the confusions between supprimer and imprimer.

In this project we have developed a method to predict if two words are likely to be confused by an ASR system if they are both in its vocabulary. We will use the terms confusable and not confusable to refer to the pairs of words that are often confused by an ASR system and the ones that are not confused respectively. The developed method is based on a technique called Dynamic Time Warping (DTW) [1], which is usually used in ASR to measure the distance between two speech signals. Here, we use it to calculate a measure of distance between the phonetic transcriptions of the words to compare [2,3] and, after, we classify the pair of words as confusable or not confusable using a threshold. This method can help to design the vocabulary of an ASR system, because it will warn the designer if he chose confusable words so, he can change them if it is possible. The principle of this measure is to do an alignment between the phonetic transcriptions of the two words and calculate the distance as the sum of the distance between the phones that are in correspondence according to the alignment. Although the developed method can be used in any language, the used language in this work is French.

The organization of this paper is as follows. In section 2 we explain the DTW distance. The first step of this technique is to align the phonetic transcriptions to compare. Therefore, first of all we explain the notation used to describe an alignment. After, the formulation of DTW and its algorithm are presented. In section 3 we present the distance measure between phones that we have used to calculate the distance between phonetic transcriptions. In section 4 we describe the experiments performed to test the method and the obtained results. Finally, in section 5 we present the conclusions of this work.

2. DYNAMIC TIME WARPING

2.1. Alignment between Phonetic Transcriptions

Let $W_1=\{p_{1i}\}$ and $W_2=\{p_{2j}\}$, with $i=1,\ldots,I$ and $j=1,\ldots,J$, be the phonetic transcriptions of the two words to compare. The values I and J are the lengths of the phonetic transcriptions and p_{1i} and p_{2j} are their phones. Let us consider an i-j grid, shown in Fig. 1, where W_1 and W_2 are developed along the i-axis and the j-axis respectively. A path through the grid is written as $F=\{c(1),c(2)...c(K)\}$, and it represents an alignment between the two transcriptions. The generalised element of the path is $c(k)$ and it consists of a pair of coordinates in the i and j directions. The i and j coordinates of the kth path element are $i(k)$ and $j(k)$ respectively.

$$c(k)=(i(k),j(k)) \quad (1)$$

The path F fulfils the following conditions [1]:

1) Monotonic conditions:

$$i(k-1) \le i(k) \text{ and } j(k-1) \le j(k) \quad (2)$$

2) Continuity conditions:

$$i(k)-i(k-1)\le 1 \text{ and } j(k)-j(k-1)\le 1 \quad (3)$$

3) Boundary conditions:

$$i(K)=I \text{ and } j(K)=J \quad (4)$$

The alignment is defined by the path F as follows:

- if $i(k)=i(k-1)+1$ and $j(k)=j(k-1)+1$ then $p_{1i(k)}$ and $p_{2j(k)}$ are aligned.

- if $i(k)=i(k-1)+1$ and $j(k)=j(k-1)$ then $p_{1i(k)}$ is aligned with the null character (symbol of an insertion or an omission)

- if $i(k)=i(k-1)$ and $j(k)=j(k-1)+1$ then $p_{2j(k)}$ is aligned with the null character.

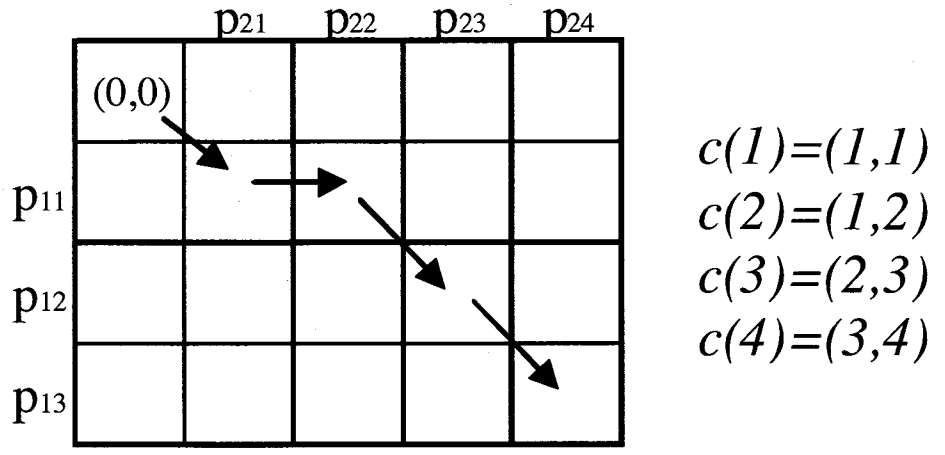


Fig. 1. Example of a path F *in the grid, and the steps* c(k).

For example, the alignment associated to the path of Fig. 1 is the following one:

p21	p22	p23	p24
p11	–	p12	p13

2.2. DISTANCE BETWEEN TWO PHONETIC TRANSCRIPTIONS

The proposed application of this work is to predict if two words are likely to be confused by an ASR system, i.e, if they are confusable or not. In order to do this, a distance is calculated between the two words and, if the distance is lower than a threshold, the word pair is considered confusable:

$$\begin{cases} \text{if } D_{DTW}(W_1,W_2) \le Threshold \Rightarrow Confusable \\ \text{if } D_{DTW}(W_1,W_2) > Threshold \Rightarrow Not\ Confusable \end{cases}$$

where $D_{DTW}(W_1,W_2)$ is a distance between the phonetic transcriptions of the words W_1 and W_2. Dynamic Time Warping (DTW) [1] is a technique that was used in speech recognition to calculate a distance measure between two speech signals. In this work we apply this technique to calculate the distance between the phonetic transcriptions of two words:

(5)

$$D_{DTW}(W_1,W_2) = \min_F \left[\frac{\sum_{k=1}^{K} d(c(k))w(k)}{\sum_{k=1}^{K} w(k)} \right]$$

where w(k) is a weighting function introduced to normalise by the path length and d(c(k)) is a distance measure between the elements that are aligned according to c(k). For example d(c(1)) in Fig. 1 is the distance between the phones and p11. How to obtain the distance d(c(k)) is explained in the following section. In this work we have used the following weighting function [1]:

$$w(k)=i(k)-i(k-1)+j(k)-j(k-1) \qquad (6)$$

This implies that:

$$\sum_{k=1}^{K} w(k) = I + J$$

Then, the denominator of (5) is constant and, therefore, independent of the path F. The DTW distance is the minimum weighted summation of the distances between the aligned phones, for all the possible alignments between the phonetic transcriptions of the words. Since the denominator N(w)=I+J is a constant, in order to solve (5) we only have to minimize the numerator and after, divide by I+J. Recall that the points that can lead to the point (i(k),j(k)) are (i(k)-1,j(k)), (i(k)-1,j(k)-1) and (i(k),j(k)-1) (monotonic and continuity conditions). Therefore, the weights associated to each step are (using (6)):

(i(k)-1,j(k)) → (i(k),j(k)): w(k) = i(k)-i(k-1)+j(k)-j(k-1) = i(k)-(i(k)-1)+j(k)-j(k) = 1

(i(k)-1,j(k)-1) → (i(k),j(k)): w(k) = i(k)-i(k-1)+j(k)-j(k-1) = i(k)-(i(k)-1)+j(k)-(j(k)-1) = 2

(i(k)-1,j(k)) → (i(k),j(k)): w(k) = i(k)-i(k-1)+j(k)-j(k-1) = i(k)-i(k)+j(k)-(j(k)-1) = 1

The solution, i.e DDTW(W1,W2), can be found using the variable s(i,j) defined as follows:

(8)

$$s(i,j) = \min \begin{cases} s(i-1,j) + d(i,j) \\ s(i-1,j-1) + 2d(i,j) \\ s(i,j-1) + d(i,j) \end{cases}$$

where s(i,j) is the accumulated distance of the optimal path that goes from the point (0,0) to the point (i,j). Therefore,

(9)

$$D_{DTW}(W_1,W_2) = \frac{s(I,J)}{I+J}$$

When all the values s(i,j) have been calculated over all i,j DDTW(W1,W2) can be calculated using (9). Below we present the algorithm to find the solution [4].

```
s(0,0) = 0
for(j=1;j J;j++) s(0,j)=
for(i=1;i I;i++){
 s(i,0)=
 for(i=1;i I;i++){

s(i,j) = min { s(i-1,j) + d(i,j)
             { s(i-1,j-1) + 2d(i,j)
             { s(i,j-1) + d(i,j)

                                    }
}
```

$$D_{DTW}(W_1,W_2) = \frac{s(I,J)}{I+J}$$

3.DISTANCE BETWEEN PHONES

In the previous section we have explained the DTW technique, which is used to calculate a distance measure between two phonetic transcriptions. Since this technique depends on a distance between the phones of the phonetic transcriptions, in this section we explain how to obtain this distance. In modern ASR systems the acoustic units are usually modeled by Hidden Markov Models (HMM) [5]. Therefore, it is possible to obtain a distance measure between two phones by calculating the distance between their HMMs. In this paper we propose the following distance between two HMMs:

(10)

$$d_{HMM}(p_1,p_2) = \begin{cases} \dfrac{\sum_{Q} P(Q)\left[\dfrac{1}{L}\sum_{i=1}^{L} D_N\left(N_{q_{1i}}, N_{q_{2i}}\right)\right]}{\sum_{Q} P(Q)} & \text{if } p_1 \neq p_2 \\ 0 & \text{if } p_1 = p_2 \end{cases}$$

where Q is an alignment between the states of the HMMs of the phones p_1 and p_2, $P(Q)$ is the probability of Q, L is the length of the alignment, q_{1i} and q_{2i} are states of the models that are aligned according to Q, $N_{q_{1i}}$ and $N_{q_{2i}}$ are the Gaussian distributions associated to the states q_{1i} and q_{2i}, and $D_N(\cdot)$ is a measure of distance between the two Gaussian distributions. The numerator is a weighted sum of the average distance between the Gaussians of the aligned states for each alignment Q. In [6], this average distance between Gaussians is calculated for each Q and the minimum one is chosen. On the other hand, we sum all these average Gaussian distances weighted by the probability of the alignment. Since only a subset of the possible alignments is used, the denominator is introduced in order to normalise by the probability of the subset of alignments. In this work, we used the alignments associated to the possible paths in a grid of dimension $M_1 \times M_2$, where M_1 and M_2 are the number of states of the models. Fig. 2 shows an example with $M_1 = M_2 = 3$. This subset avoids the alignments where there are loops in states of the two models at the same time.

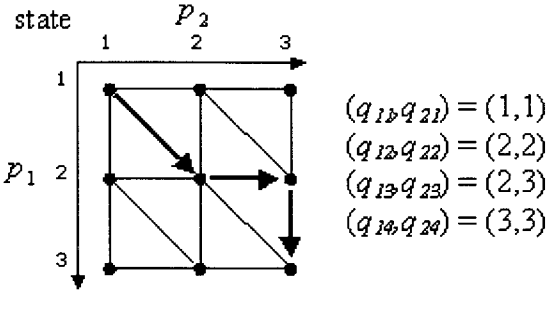


Fig. 2: Subset of alignments used to calculate the inter-HMM distance. The bold line shows one of these alignments. The values of q1i and q2i are the aligned states according to the path in bold.

The models used to obtain a distance value between the phones with the proposed measure have one Gaussian per state. This does not imply that the real ASR systems must have one Gaussian per state. We considered several monomodal Gaussian distances such as Euclidean, Mahalanobis, Kullback-Leibler, Bhattacharyya and Jeffreys-Matusita [7,8]:

Euclidean distance:

$$D_{EUC}(N_1,N_2) = (\mu_2 - \mu_1)^T (\mu_2 - \mu_1) \qquad (11)$$

Bhattacharyya distance:

(12)

$$D_{BHA}(N_1,N_2) = (1/8)\ (\mu_2 - \mu_1)^T ((\Sigma_1+\Sigma_2)/2)^{-1}$$
$$(\mu_2 - \mu_1) + 1/2 \log\ (|(\Sigma_1+\Sigma_2)/2|)/(|\Sigma_1||\Sigma_2|)^{(1/2)}$$

Jeffreys-Matusita distance:

(13)

$$D_{JM}(N_1,N_2) = \sqrt{2}(1 - \exp(D_{BHA}(N_1,N_2)))^{1/2}$$

Kullback-Leibler distance:

(14)

$$D_{KL}(N_1,N_2) = 1/2\ (\mu_2 - \mu_1)^T (1/\Sigma_1+1/\Sigma_2)\ (\mu_2 - \mu_1)$$
$$+ 1/2\ \mathrm{tr}((1/\Sigma_1)\Sigma_2 + (1/\Sigma_2)\Sigma_1 - 2I)$$

Mahalanobis distance:

(15)

$$D_{MAH}(N_1,N_2) = (\mu_2 - \mu_1)^T (\Sigma_1\Sigma_2)^{-1}(\mu_2 - \mu_1)$$

where μ_i and Σ_i are the mean vector and the covariance matrix of the Gaussian Ni respectively.

This distance has to be extended to cover pairs consisting of a phone and the null character, which corresponds to the operation of insertion or omission. The extended inter-phone distance, which is the one used to calculate the DTW distance measure is:

(16)

$$d(c(k)) = \begin{cases} d\_ & \text{if } \begin{pmatrix} i(k) = i(k-1) \text{or} \\ j(k) = j(k-1) \end{pmatrix} \\ d_{HMM}(p_{1i(k)}, p_{2j(k)}) & \text{otherwise} \end{cases}$$

where d_ is the distance between a phone and the null character This value was set at the arithmetic mean of the distances between all the phones:

(17)

$$d\_ = \frac{1}{P^2}\sum_{i=1}^{P}\sum_{j=1}^{P} d_{HMM}(p_i, p_j)$$

where P is the total number of phones.

4.EXPERIMENTS AND RESULTS

4.1.Experimental Setup

In order to test our method we need to determine which pairs of words are usually confused by ASR systems to compare them with the prediction of our method. We constructed two kinds of ASR systems: one to detect the confusable word pairs, and the other to detect the not confusable word pairs.

-NCD Systems (No Confusability Detection): 223 systems, each one with only one word in its vocabulary and a garbage model to reject out-of-vocabulary data. Each system was tested with the 223 words.

-CD System (Confusability Detection): One system with 841 words and a garbage model, tested with the 841 words.

If one of the NCD systems, with only the word A in its vocabulary, is tested with another word B and they are never confused, it means that they are very different and, therefore, they are not confusable. On the other hand, if they are sometimes confused, it only means that B is more similar to A than to the garbage model, not necessarily that A and B are similar. Therefore, with this kind of systems we can only determine if two words are not confusable in general.

If we test the CD system with several pronunciations of a word A, and a word B is never recognized, we cannot say that A and B are not confusable, we can only say that A is more similar to some of the other words of the vocabulary than to B. On the other hand, if they are sometimes confused, we can assure that they are quite confusable. Therefore, with this system we can detect confusable word pairs.

The vocabulary of the CD and NCD systems consisted of French isolated words such as numbers, cities, commands, etc. Each word was pronounced by 700 speakers in average. The speech signal was sampled at 8 kHz and parameterized using MFCCs. The feature vectors consisted of 27 coefficients: the frame energy, 8 MFCCs, and the first and second time derivatives. The models of the words were constructed by concatenating context dependent HMMs of the phones with one Gaussian per state. By testing these systems the following three groups of word pairs are obtained:

"Low Probability of Confusion (LPC): 21506 word pairs which were never confused when the NCD systems were tested.

"Medium Probability of Confusion (MPC): 150 word pairs which had a confusion rate lower than 5% and higher than 0% when the CD system was tested.

"High Probability of Confusion (HPC): 189 word pairs which had a confusion rate higher than 5% when the CD system was tested.

We consider a False Rejection to classify as confusable a LPC word pair, and a False Acceptance to classify as not confusable a HPC word pair. The MPC word pairs were not taken into account in the evaluation because we considered that is not a severe error neither to classify them as confusable nor as not confusable.

The HMMs used to calculate the inter-phone distances are not the models used in recognition. In the first case we used models without context with 3 states and 1 Gaussian per state.

4.2. RESULTS

In order to test our method we measure the False Rejection Rate (FRR) and the False Acceptance Rate (FAR). The FRR is the error rate when classifying the pairs of words that are not confusable, i.e., the percentage of pairs words belonging to the group LPC classified as confusable. The FAR is the error rate when classifying the pairs of words that are confusable. That is to say, the percentage of word pairs belonging to the group HPC classified as not confusable. Our objective is to minimize both FRR and FAR, and both values depend on the chosen threshold. If we decrease the threshold the FRR decreases and the FAR increases and vice versa. In order to compare the different methods with only one value we use the Equal Error Rate (EER). The EER is the False Acceptance Rate and the False Rejection Rate obtained with the threshold that makes them equal as shown in Fig. 3.

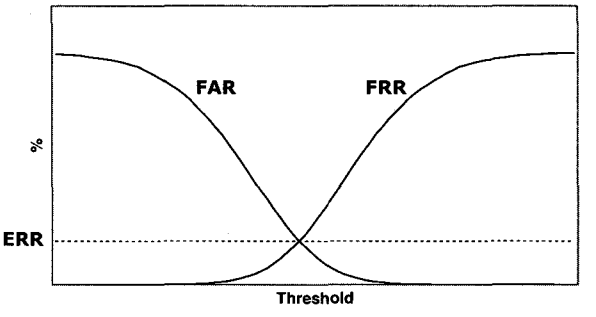


Fig. 3: FRR and FAR in terms of the threshold. The EER is the point where the two lines intersect.

Table 1 shows the EER for each Gaussian distance in (10). We can see that the better results are obtained when using the Mahalanobis Gaussian distance to calculate the distances between the phones. With the Euclidean and the Kullback-Leibler distances also low error rates are obtained. On the other hand, the Battacharyya and the Jeffreys-Matusita distances give high EERs and therefore are not useful to our purpose. We can conclude that the developed method can be used useful to predict if two words are likely to be confused by an ASR system because we have obtained an EER of 2.6%.

5.CONCLUSIONS

In this paper we have presented a method to predict if two words will be confused by an ASR system. This method is based on the classical DTW technique, which is used to calculate a distance

between two phonetic transcriptions. We also have described how to obtain the data to test. We have tested the method in a classical false acceptance/ false rejection framework and the EER was measured to be less than 3%.

	DTW
EUC	3,1%
KL	3,2%
MAH	2,6%
JM	7,5%
BHA	8,9%

Table 1: EER obtained with each Gaussian distance in (10)

6.ACKNOWLEDGMENTS

This paper reports the work performed during an internship at Telisma R&D, Lannion, France. The authors would like to thank all the members of Telisma specially Stephane Peillon and Alexandre Bramoulle.

7.REFERENCES

[1] Hiroaki Sakoe and Seibi Chiba, "Dynamic Programming algorithm optimization for spoken word recognition". In IEEE Trans. on ASSP, vol. ASSP-26, N°1, 1978.

[2] Beng T. Tan, Yong Gu, Trevor Thomas, "Word Confusability Measures for Vocabulary Selection in Speech Recognition", Proceedings of the ASRU, Keystone, December 1999.

[3] Sandrine Pouységur, "Etude du taux de confusion de mots pour la reconnaissance de mots isolés". 4e Rencontres jeunes chercheurs en parole, 2001.

[4] Calliope, La Parole et son Traitement Automatique, Masson, Paris, 1989.

[5] Rabiner L. "A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition". Proceedings of the IEEE, 77(2):257-286, Feb. 1989.

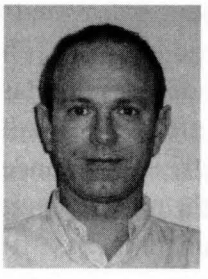
[6] Bahlmann C. and Burkhardt H. "Measuring HMM similarity with the Bayes probability of error and its application to online handwriting recognition", Proceedings of the ICDAR, pp 406-411, 2001.

[7] Basseville M. "Distance Measures for Signal Processing and Patter Recognition", Signal Processing, Vol. 18(4), pp. 349-369, 1989.

[8] Sooful J., Botha E. "An Acoustic Distance Measure for Automatic Cross-language Phoneme Mapping". Proceedings of the PRASA, 2001.

8.AUTHORS

Jan Anguita Ortega received his degree in telecommunication engineering and the European master in speech and language from the Universitat Politècnica de Catalunya (UPC) in 2003. He is currently a PhD student in the department of signal theory and communications at UPC. His interests are speech processing, robust speech and speaker recognition and speech perception.



Francisco Javier Hernando Pericás received his degree and PhD degree in telecommunication engineering from the Universitat Politècnica de Catalunya (UPC) in 1988 and 1993 respectively. He is presently an associate professor in the department of signal theory and communications at UPC. His main interests are speech processing and robust speech and speaker recognition. He has published papers in international journals and conferences and he is member of both national and international associations. He works in projects supported by Spanish and European institutions.