WORD JUNCTURE MODELLING BASED ON THE *TIMIT* DATABASE

Xue Wang and Louis C.W. Pols
Institute of Phonetic Sciences/IFOTT, University of Amsterdam
Herengracht 338, 1016 CG Amsterdam, the Netherlands
Tel. +31 20 5252183, Fax: +31 20 5252197
E-mail: {wang, pols}@fon.let.uva.nl

ABSTRACT

In this study, we develop data-based word juncture models, which account for the pronunciation variations at word boundaries, as an optional form of phonological rules. We used the American English TIMIT database. Issues in generating the models and using them in a continuous recognition task are discussed. A comparison is given between the coverage of the pronunciation variations by the models and by a set of phonological rules. There is a fairly good agreement between the models and the rules in predicting the pronunciation variations, whereas the models cover a larger set of variation phenomena. Furthermore, use of the models improved recognition performance.

1. INTRODUCTION

It is well known that the pronunciation of sequences of words in fluent speech can deviate substantially from the normative lexical forms, especially at the junctures between words. Such deviations include for instance deletion of vowels and consonants, and substitutions of one vowel by another vowel. On the other hand, in many automatic speech recognisers, for technical simplicity, the pronunciation of each word is only based on the norm sequence of phones according to the lexicon. Probabilistic models in the recogniser may model minor variations of a phone in its acoustic realisation, but they can hardly model the serious deletion of phones and substitutions by other phones. This then introduces mismatches between the norm phone sequence for a word to be recognised and the acoustic signal which simply represents the sequence of the actually pronounced phones. This might in turn reduce recognition accuracy. One solution to this problem is to use "phonological rules" ([2], for both French and English, and [3]) which account for various pronunciation variations in the language. However, it is hard to obtain reliable sets of such rules. Because of the compactness of such rules, many variation phenomena cannot be covered.

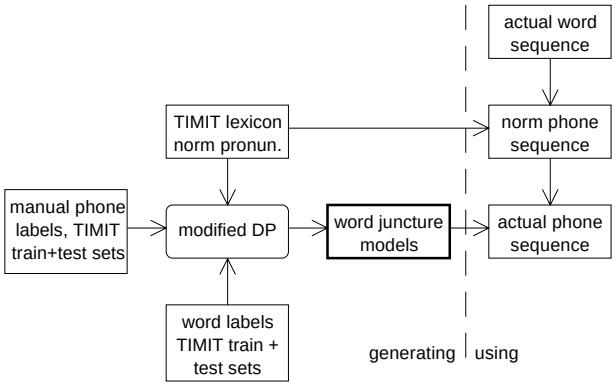

Figure 1. Generating and using the word-juncture models.

In this study, we propose a different approach. We only concentrate on the pronunciation variations at word junctures, although generally within-word deviations may also occur. We directly generate word-juncture models based on the statistics of the pronunciation variations from a training data set. Then we use these models to predict those actual phone sequences deviating from the normative sequences. The actual phone sequences are used in word recognition. The whole process of generating and using the models is illustrated in Figure 1.

In this paper, we mainly address the procedures of generating and using the juncture models. The juncture models are stored as a list of items each describing how a normative phone sub-sequence should be converted into a predicted sub-sequence, for each sort of word junctures.

2. GENERATING THE JUNCTURE MODELS

For the purpose of generating the juncture models, we have chosen the American English TIMIT database [7] because information about both normative and actual pronunciations is available, and because comparison with data in literature can then more easily be made. It is hand-labelled at the phone and at the word level, and is provided with one unique lexicon pronunciation form for each word. Therefore correspondences can be found between the normative and the actual pronunciations.

2.1. Dynamic programming for symbol sequences

In this study, the actually pronounced phone sequences are based on the TIMIT manual labelling. For instance the word pair "what time" is actually pronounced as /w aa cl t ay m/ rather than /w aa cl t.cl t ay m/ ("cl" is a

closure phone and "." indicates the word boundary), as would have been predicted from concatenating the norm phone sequences of the two words. In other examples, more complicated situations may occur between the norm and the actual phone sequences.

We used a dynamic programming (DP) procedure (e.g. [1]) to match the two phone sequences, in order to find the correspondence between the two sequences. DP was performed at the level of a whole (sentence) utterance. Insertion and deletion penalties are adjusted in order to find an optimal match. Given the fact that most vowels are not deleted in TIMIT (although they can be substituted by other vowel phones), vowels were used as good anchor points for isolating phone sub-sequences of each word-pair in the utterance. For this purpose a heavy penalty for substitution between different classes of phones (especially between vowels and non-vowels) was used.

2.2. Juncture area

For clarity, we define a "juncture area" which is the sub-sequence of phones around the boundary of a pair of words undergoing pronunciation variation. We first concatenate the norm phone sequences of the two words. The juncture area starts to extend from the word boundary into both words. If the first phone is a vowel, then this vowel is included and the extension stops. If the first phone is not a vowel, the extension will continue until a vowel is encountered (then that vowel is not included). Such a definition of a juncture area includes the majority of pronunciation variations at word boundaries for American English. The number of phones included in such a juncture area is thus not fixed (in [3], the juncture area always has two phones to the left and one phone to the right, of the boundary).

2.3. Two types of models

According to the way of storing the items, we distinguish between two types of models. The type-1 model searches through the database to find all different word-pairs, and then identifies the most frequently occurring realisation of the phone sub-sequence for the juncture area of each unique word-pair. Only those sub-sequences different from the norm forms are stored in the model. Examples of such items are:

liked to ⇒ cl t 1 1
object to ⇒ cl t 1 1
respect to ⇒ cl t 1 1
subject to ⇒ cl t 6 7
invoked technology ⇒ cl t pau t 1 1

where /cl/ is a closure phone, and ⇒ separates the word pair on its left from the actual phone sub-sequence on its right. The norm phone sub-sequence in the juncture areas for all these pairs or words is /cl k cl t.cl t/ where "." indicates the word boundary. For all these cases the actual phone sequence according to the model is /cl t/.

In the above example, two integer numbers are given at the end of each item. The second integer is the number of all different instances of phone sub-sequences for the given word pair (7 such instances for the word pair "subject to"). The first integer indicates the number of the instances of the phone sub-sequence which occurs the most frequently (the winner) for the given word pair, as given on the right (6 instances of this word pair have their juncture areas realised as /cl t/). There are 8,815 items of type-1 model as extracted from the TIMIT training and test sets.

The type-2 model takes a further step to cluster all word pairs for which the norm phone sub-sequences in the juncture areas are the same. The five items of the type-1 model in the above example can then be summarised into one item of type-2 model:

cl k cl t.cl t ⇒ cl t 9 11

The two integers at the end indicate similar statistics as in the type-1 model. Note that the first integer is 9 rather than the sum (10) of the first integers for all the items in the type-1 example, since the last item with a different actual phone sub-sequence /cl t pau t/ from the winner /cl t/ is excluded. The statistics of such occurrences are thus collected on larger samples, whereas the number of items of the model is reduced to 1,654. This type of model is also better to use since the relative prediction of realisations in an independent data set by the "training set" of the models will be larger for the phone-cluster based model than for the word-pair based model.

2.4. Coverage of the model on the data set

The coverage of the type-2 model is checked here with the training data (the training and test sets of TIMIT). Three example segments of the list of pronunciations are given in the next page. Each segment gives rise to one item in the type-2 model. The number after each pronunciation is the count of instances, whereas the two numbers after the model are the same as above. In the whole training data, there are 36,117 instances of pronunciations. 16,353 items are non-normative (19,764 normative), of which 9,052 instances are correctly predicted by the type-2 model (53.4%). This percentage is not high, because the number of winner instances (8 in segment 2) can still be much smaller than that of the total instances (33). Norm pronunciations may also be present as non-winners (the first two segments). There are 2,002 such instances out of the total of 19,764 +2,002=21,766 normative instances (9.2%). These 9.2% of normative instances will be forced to take a non-normative pronunciation by the model in the recognition process, which is the error introduced by the model. Of course, in using the models with a different data set from the training data, the coverage may be different.

segment 1:	cl t s.cl k ⇒ cl s.cl k 16 winner cl t s.cl k ⇒ cl t s.cl k 6 norm cl t s.cl k ⇒ q s.cl k 1
model item:	cl t s.cl k ⇒ cl s.cl k 16 23

segment 2:	ax.ay ⇒ .ay 3 ax.ay ⇒ ah.q aa 1 ax.ay ⇒ ah.q ay 4 ax.ay ⇒ ax.ay 1 norm ax.ay ⇒ ax.q aa 1 ax.ay ⇒ ax.q ay 5 ax.ay ⇒ er.ay 1 ax.ay ⇒ ih.eh 1 ax.ay ⇒ ih.q ay 1 ax.ay ⇒ ix.aa 1 ax.ay ⇒ ix.ay 1 ax.ay ⇒ ix.er 1 ax.ay ⇒ ix.q ay 1 ax.ay ⇒ iy.aa 3 ax.ay ⇒ iy.ay 8 winner
model item:	ax.ay ⇒ iy.ay 8 33

segment 3:	ax.cl p ⇒ .cl p 3 ax.cl p ⇒ .cl p ao 1 ax.cl p ⇒ ah.cl p 3 ax.cl p ⇒ ax.cl p 72 winner (norm) ax.cl p ⇒ ax.cl p ao 1 ax.cl p ⇒ ax.cl p eh 1 ax.cl p ⇒ ax.cl p ow 1 ax.cl p ⇒ ax.th cl p 1 ax.cl p ⇒ ey.cl p 4 ax.cl p ⇒ ih.cl p 2 ax.cl p ⇒ ix.cl p 41 ax.cl p ⇒ ix.cl p eh 1 ax.cl p ⇒ ix.cl p y 1
model item:	no model

3. COMPARISON OF THE JUNCTURE MODELS WITH THE RULES

One way to compare the quality of the cluster-based models (type-2) with a set of rules is to count the percentage of the instances that our clusters coincide with each of the rules (We take the 11 rules used by [3]), using the data set. This comparison is given in Table 1. Because the juncture areas are defined differently for the rules than for our models, we actually merged all those different items together which have the same norm phone sub-sequences at the word boundaries as defined in the juncture areas of the rules. For example, the norm phone sub-sequences:

p t.k r
p t.p
p t.t r
k t.t

will all be compared with rule (2) /stop stop.stop/ in Table 1, while the last /r/ is irrelevant

Table 1. A set of word-juncture rules according to [3], and statistics of the matching of our type-2 cluster-based juncture models with these rules. "C" and "V" refer to consonants and vowels, respectively. "St" (stop) here refers to the set of phones /p,t,k,b,d,g/. "sil" is matched against /q,pau,cl,vcl/. "[f|n]" means /f/ or /n/. "Total w." is the total number of winner instances of the type-2 models for one rule. "Cor." is the number of winners that are also correctly matched with the predicted sequences of the rule. The percentages is "correct/total winner". "Total" is the total number of instances corresponding to the same kind of norm instances as the rules, but may have different predicted form from the winner.

Rule	norm	predicted	cor.	tot w.	%cor.	total
1	C.sameC	⇒ C	316	347	91.1	414
2	st st.st	⇒ st	1141	1145	99.65	1357
3	t.y	⇒ ch	7	38	18	87
4	d.y	⇒ jh	27	30	90	59
5	V t.V	⇒ V dx V	197	221	89.1	635
6	[f|n] st.st	⇒ [f|n] st	206	217	94.9	319
7	[s|z].sh	⇒ sh	103	103	100.0	137
8	t.[d|dh]	⇒ sil [d|dh]	162	167	97.0	232
9	V t.dh	⇒ V dh	148	148	100.0	165
10	n d.dh	⇒ n dh	41	41	100	87
11	dh ax.V	⇒ dh ih V				
1-10			2348	2457	95.56	3492
all				9052		16353

Furthermore, some of the rules in [3] will have overlapping phone sub-sequences if the juncture areas are extended with more phones in order to be compared with our models. For example the phone in rule 8 before phone /t/ may be /f/ or /n/, thus such a sub-sequence overlaps with that of rule 6. For the purpose of counting a fair coverage of the rules by our list of model items, we removed all the overlaps from the rules (for the above example, "[f|n]t.d" is excluded for counting with rule 8). Rule 11 cannot be compared since its juncture area is outside our defined area.

Most rules give rise to phone deletion in their actual sub-sequences. After such a deletion, the remaining phones may belong to either word of the pair. For the output of the rules such a placement makes a difference. However, in generating the actual phone sequence for the process of an *N*-best re-scoring process (see the next section), the word-boundary positions are irrelevant since we only need the phone sequence for the whole *utterance*, instead of the sequences for all the individual *words*.

Of the total of 9,052 instances of pronunciation variations in our data set, which can be explained by our model, only 2,457 instances can be explained by 10 out of the 11 rules. This means that our model covers a larger amount of different types of variations than the rules do. Of these 2,457 instances that can be explained by the rules, 2,348 instances get the same predicted variations form from our model as from the 10 rules. This means that our model has a good agreement (95.6%) with the rules, but is actually more powerful in modelling the phenomena of pronunciation variations.

4. USING THE JUNCTURE MODELS IN SPEECH RECOGNITION

The development of our word juncture models is part of a research project in our institute for duration modelling [4] in the context of continuous speech recognition [5, 6]. One of the approaches to incorporate durational knowledge into the recogniser is to use an *N*-best algorithm to generate hypotheses of transcriptions at *word* level. However, in the re-scoring procedure the additional knowledge about duration is brought into the recogniser in terms of duration scores of the *phones*. In the *N*-best recognition the pronunciation dictionary used has a single norm form for each lexical word. In order to be more faithful to the actual pronunciation, the phone sub-sequence in the juncture area in each pair of words of all the word-level *N*-best transcriptions are converted into the predicted phone sub-sequence using the type-2 juncture model. If an item of the model does not exist for the norm phone sub-sequence of the word pair, this juncture area remains the norm sub-sequence.

The above is a two-step procedure of recognition using the *N*-best algorithm. Such a procedure actually has solved a controversy problem in continuous speech recognition. One hopes to faithfully model the pronunciation variations of all the lexical words. Using the two-step procedure, in the first step a simple pronunciation-dictionary with one norm pronunciation per word can be used to generate the *N*-best hypotheses. In the separated second step, the word-juncture model can be used to generate the predicted phone sequence. This model is relatively simple since only one particular word sequence per hypothesis is dealt with, rather than a comprehensive pronunciation-dictionary which ought to include all the variations associated with all possible between-word transitions. It may nevertheless be possible to use the word-juncture models in other procedures than the *N*-best.

Our word recognition was performed with the whole TIMIT test set [5]. The recogniser used was based on monophone HMMs with linear topology and 8 Gaussian densities per state. The front-end processing consisted of 12 MFCC plus normalised frame energy, the first and the second time derivatives, together making a 39-dimensional vector. The frame shift was 8 ms. The language model used was a word-pair grammar derived from the whole training and test sets of TIMIT.

In the recognition tests presented by Giachin et al. [3], some of the 11 rules were used optionally. In our relatively preliminary tests of the word-juncture models, the optionality of applying the models was either with or without the whole set of the model items. Under these two test conditions, the word correct scores are 79.90% (with) versus 79.36% (without), respectively. These both are improved by using duration modelling from a baseline score (79.07% word correct) which uses no duration modelling. Although the difference in scores is not large, this shows a positive contribution of the word-juncture modelling, although 9.2% of normative pronunciations are wrongly predicted as non-normative.

5. CONCLUSION

In this paper we addressed the problem of the serious pronunciation variations at word-junctures in fluent speech, that cannot be modelled properly by the often used single norm pronunciation per word in automatic speech recognition. Our results in using the word juncture models, as an preliminary investigation of the usefulness of the models, showed already that the recognition performance can be improved.

Our word-juncture models are data-based, and show good agreement with the rule-like models in [3] for those variation phenomena also covered by the rules, but actually cover a much larger amount of variation phenomena than these rules do. For any recognition task, as long as a training corpus is available with manual labelling and a norm lexicon, similar word-juncture models can be extracted and used in the recognition.

For future work, a "universal" juncture model can be made using a large and representative training corpus, which then should cover most of the phenomena of pronunciation variations at word junctures, for a given language. Such a juncture model can be compared with each test set of a recognition task, in order to investigate the coverage of such a model.

6. REFERENCES

[1] S. Furui, “Digital speech processing, synthesis, and recognition”, M. Dekker, Inc. New York. 1989.

[2] J.L. Gauvain, L.F. Lamel, G. Adda and M. Adda-Decker, “Speaker-independent continuous speech dictation”, *Speech Comm.* Vol. 15, pp. 21-37, 1994.

[3] E.P. Giachin, A.E. Rosenberg and C.-H. Lee, “Word juncture modeling using phonological rules for HMM-based continuous speech recognition”, *Computer Speech and Language* Vol. 5, pp. 155-168, 1991.

[4] L.C.W. Pols, X. Wang and L.F.M. ten Bosch, “Modelling of phone duration (using the TIMIT database) and its potential benefit for ASR”, *Speech Comm*. Vol. 19, pp. 161-176, 1996.

[5] X. Wang, “Incorporating knowledge on segmental duration in HMM-based continuous speech recognition”, Ph.D. thesis, no. 29 in the series “Studies in Language and Language Use”, University of Amsterdam, 1997.

[6] X. Wang, L.F.M. ten Bosch and L.C.W. Pols, “Integration of context-dependent durational knowledge into HMM-based speech recognition”, *Proceedings ICSLP'96*, Philadelphia, PA, pp. 1073-1076, 1996.

[7] V. Zue, S. Seneff, and J. Glass, “Speech database development at MIT: TIMIT and beyond”, *Speech Comm.* Vol. 9, pp. 351-356, 1990.