Speaker Identification and Assessment on the YOHO database

J.W. Koolwaaij

Department of Language and Speech, University of Nijmegen
E-mail: koolwaaij@let.kun.nl

November 14, 1996

Abstract

Within the CAVE project (Caller Verification in Banking and Telecommunication) we have done extensive research on speaker verification. However, speaker verification does not give sufficient information about inter-speaker relations, and that is one of the topics we want to investigate. So we decided to use speaker identification experiments to address the following two questions. Is there a reliable measure to classify speakers as resistant, dependable, unreliable or vulnerable? And can we construct a distance measure between speakers based on log likelihoods? And naturally, what is the relation between these two questions? First we will describe the speaker database and the HMM topologies.

1 Database description

The database is the YOHO Speaker Verification Corpus. The particular vocabulary employed in this collection consists of two-digit numbers ("thirty-four", "sixty-one", etc), spoken continuously in sets of three (e.g. "36-45-89"): the so called combination-lock phrases. There are 138 speakers (106 male, 32 female); for each speaker, there are 4 enrollment sessions of 24 utterances each, and 10 verification sessions of 4 utterances each, for a total of 136 utterances in 14 sessions per speaker. The speech data was collected over a three month period from an office environment with a high quality microphone.

2 Method definition

From the experiments done within CAVE we choose the best one with left-to-right HMM's (01326, SI EER = 0.109%), the best one with ergodic HMM's (03142, SI EER = 0.826%) and also the most basal method (01220, SI EER = 0.666%) in order to be able to study the impact of the acoustic resolution on the identification results. The definitions of these methods are described in table 1. For each method we specify the HMM topology, the text dependence, the kind of covariance matrix and the enrollment set. Subword is conform Higgin's definition [1] of partial words: {one,...,seven, nine, ty, Twen,...,Nine}. Further on we will use the subset *tens* {Twen,...,Nine} and the subset *units* {one,...,seven, nine}. In this paper we use the following definitions for subword, word and utterance.

Subword	All Higgin's partial words
Word	Combination of {ten,ty,unit} spoken continuously
Utterance	Combination of three words spoken continuously

The speech files of YOHO were in linear format. We applied a preemphasis with factor 0.97 and used a Hamming window (length 25.6 ms, step 10.0 ms) to calculate the mel-frequency zero-mean cepstral coefficients $c_0, \ldots, c_{12}$ and their first and second time derivative as well. So we use 39-dimensional feature vectors as input of our identification experiments.

Table 1: Method definition

Method number	01220	01326	03142
HMM description	Left to right	Left to right	Ergodic
Text dependence	Subword	Subword	Independent
States / subword	1	1	8
Mixtures / state	1	5	8
Covariance kind	Diagonal	Diagonal	Diagonal
Enrollment set	96 utt/speaker	96 utt/speaker	24 utt/speaker

3 The enrollment and identification sets

For the enrollment of methods 01220 and 01326 we used all available enrollment speech of 118 speakers (96 male, 22 female), because the other 20 speakers in YOHO were used to train a world model. So enrollment was done with on the average $4 \times 24 \times 3/8 = 36$ occurrences of each ten and 36 occurrences of each unit and $96 \times 3 = 288$ ty's per speaker. For the enrollment of method 03142 we used only one enrollment session. We did not make a real choice of enrollment sets, because this choice was decided by the choice to use the methods performing best in verification.

The choice for the identification set was more open. YOHO has 10 verification sessions per speaker and each session contains 4 utterances. Our identification set was defined as all the speech of 2 of those sessions per speaker randomly chosen out of the 10 available sessions. Thus with 8 utterances, 24 words and 48 subwords per speaker. It may be important to know that in our configuration speaker $1, \ldots, 22$ are females and speaker $23, \ldots, 118$ are males.

4 Scoring procedure

We use the likelihood test statistics to score the results of our experiments. For the methods with text dependence on subword level each speaker has models for all 17 subwords, but the identification is done on three different levels and the log likelihoods are calculated as follows

Subword	Log likelihoods computed per subword
Word	Sum of the log likelihoods of {ten,ty,unit}
Utterance	Sum of the log likelihoods of the three words in this utterance

And for the text independent methods for all utterances the labeling is the same: *sil, speaker, sil, speaker, sil, speaker, sil.* So each speaker has only one model. So the identification can be done on only two different levels:

Word	Log likelihoods computed per word
Utterance	Sum of the log likelihoods of the three words in this utterance

So for text independent methods there are no subword results available. Also note that the silences between the words are discarded in the scoring procedure.

5 Closed Set Identification Results

As described in the EAGLES document [2], the relative number of times the system fails in identifying correctly an applicant speaker is the most natural measure for the performance of a SI system. These so called misclassification rates γ were computed and averaged over the speakers, first gender dependent

($\bar{\gamma}_F, \bar{\gamma}_M$) and also gender independent ($\bar{\gamma}_{MF}$) and the same for the mistrust rates ($\hat{\gamma}$), the relative number of times the system falsely assigns an attempt to a registered speaker. Because the misclassification rate on utterance level is less than 2.5%, we are also interested in the misclassification rates on word and subword level. Table 2 shows the results of identification on those three levels. The misclassification rates are low for utterance and word level, but much higher for subword level. We see that the γ on

Table 2: Identification results per method

		$\bar{\gamma}_F$	$\bar{\gamma}_M$	$\bar{\gamma}_{MF}$	$\bar{\bar{\gamma}}_F$	$\bar{\bar{\gamma}}_M$	$\bar{\bar{\gamma}}_{MF}$	γ
Method 03142	Word	37.311	37.283	37.297	31.368	34.195	32.782	37.288
	Utterance	3.977	1.693	2.835	1.010	1.833	1.422	2.119
Method 01326	Subword	13.731	16.254	14.993	10.690	16.343	13.517	15.784
	Word	3.598	2.517	3.058	0.972	2.922	1.947	2.719
	Utterance	1.136	0.130	0.633	0.000	0.347	0.174	0.318
Method 01220	Subword	39.489	33.268	36.378	27.917	34.041	30.979	34.428
	Word	10.795	6.944	8.870	6.044	7.488	6.766	7.662
	Utterance	0.000	0.521	0.260	0.000	0.440	0.220	0.424

utterance level roughly correspond the the SI EER's from the verification experiments. (03142: 0.826%, 01326: 0.109%, 01220: 0.666%).

Because these results are an average over all subwords, the misclassification rates per subword are listed in table 3, which reveals that the Eigh is the bad guy in the midst of all subwords. A possible explanation is that Eigh needs to be modeled with more than one state, which is the case now, because Eigh is a one-phoneme subword and both methods use 1 state per phoneme.

Another point is that it seems that males perform better than females with low complexity HMM's, but with high complexity HMM's the balance is turned. This is exactly what we experienced in our verification experiments as well.

Table 3: Misclassification rates per subword ($\times 100$)

Units	01220	01326
one	0.1792	0.0961
two	0.6066	0.1922
three	0.4067	0.1226
four	0.3448	0.1466
five	0.2676	0.1514
six	0.5462	0.1204
seven	0.1080	0.1278
nine	0.1951	0.0732

Tens	01220	01326
Twen	0.4375	0.1278
Thir	0.2521	0.1360
Four	0.2869	0.1449
Fif	0.3821	0.1672
Six	0.3096	0.1452
Seven	0.1080	0.1939
Eigh	0.7247	0.4242
Nine	0.3771	0.1564

6 Goats, sheep and other animals

In general we state that it is difficult to extract speaker specific characteristics from the results of one method, because it is not clear then if you are really measuring the speaker specific characteristics or just the impact of this particular method on those characteristics. And because we now address our first question "Is there a reliable measure to classify speakers as resistant, dependable, unreliable or vulnerable?" we will do the classification based on different method and compare the results afterwards to see if they are method independent.

The more popular names for the mentioned classes of speakers are

Goat	Unreliable applicant speaker (with high misclassification rate)
Sheep	Dependable applicant speaker (with low misclassification rate)
Lamb	Vulnerable registered speaker (with high mistrust rate)
Ram	Resistant registered speaker (with low mistrust rate)

In speaker verification some other terms are used, which can be compared with the popular names above. For example: wolf (SV) can be compared with goat (SI) since someone having much misclassifications in SI implies that this goat is probably able to enter a SV system while making a claim to be the speaker as whom he is misclassified most in the SI system.

Let us concentrate on the methods with left-to-right HMM's for the moment. The confusion matrix on utterance level does not contain enough information because there are only 3 misclassifications. And for this classification problem we need a reasonable amount of misclassifications. (You see that we run out of terminology.) In figure 1 the misclassification rates and the mistrust rates are depicted for word and subword level and for the two methods 01220 and 01326. For the construction of the confusion matrix,

Table 4: Dividing clients into categories (1-best)

	Method 01220		Method 01326	
	Subword	Word	Subword	Word
Goat	71 19 14	71 97 19	71 35 97 11	71 11
Sheep	42 100	-	60	-
Ram	21 36 4	-	72 3	-
Lamb	56 89	11 40	40	66

only the first best client (= the one with maximum log likelihood) is used. But we have a lot of questions which can not be answered by simply looking at the confusion matrix. (From now on we will use client for registered speaker and just speaker for applicant speaker.) For example: How many clients were almost as likely to be the speaker as the client who is eventually chosen by the system? What is the distance of the first best and the second best client? Which speakers can easily imitate a particular client? (In this case 1-best do not really suffice, because, for example, a client always being second best for each speaker is more a lamb than a client being first best for few speakers.) Which clients are very typical? Which male sounds most like a female and vice versa? All these questions can not be answered with the aid of the confusion matrix. What we need is a method which takes into account the 1-best till the T-best identity, where t varies depending on the application. (We will choose T equal to the number of clients.) So we tried to find a measure for the overall performance of speakers and clients.

Therefore we need to define a ranking function R

$$R(s,c,t) = \text{The number of times that client } c \text{ reaches the } t^{th} \text{ rank in the attempts of speaker } s \quad (1)$$

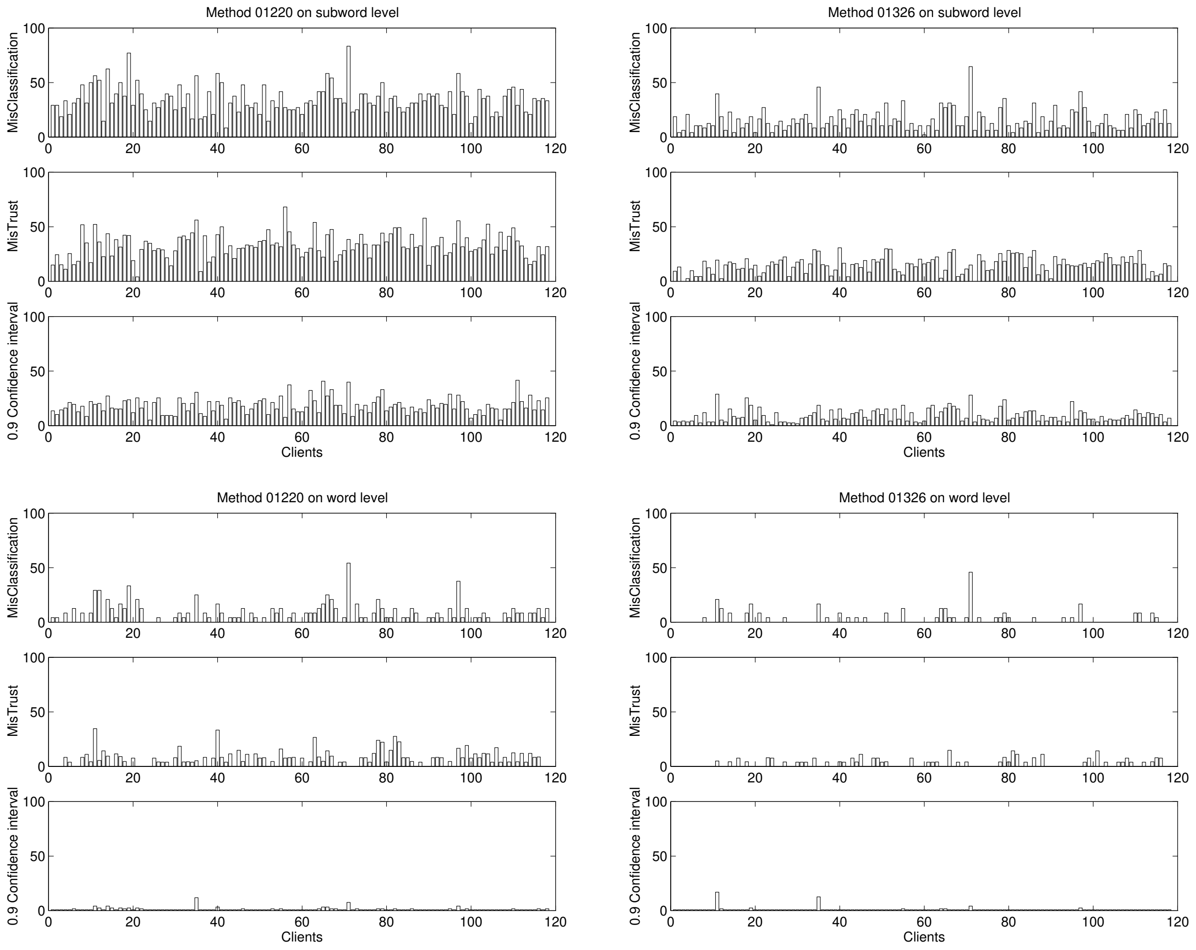

Figure 1: Misclassification and mistrust rates for methods 01220 and 01326

The goat-sheep-curve GS is then defined as

$$GS(s,t) = \sum_{i=1}^{t} \frac{R(s,s,i)}{n_u(s)}, t = 1, \ldots, n_c (= T) \tag{2}$$

This definition is very similar to the definition of confidence intervals as in EAGLES, except that this score is normalized such that $GS(n_s, n_c) = 1$. And the ram-lamb-curve RL is defined as

$$RL(s,t) = \frac{1}{(n_c - 1)} \sum_{i=1}^{t} \sum_{c \neq s} \frac{R(s,c,i)}{n_u(s)}, t = 1, \ldots, n_c (= T) \tag{3}$$

where n_c is the number of clients, n_s is the number of speakers and $n_u(s)$ is the number of attempt units of speaker s. To clarify these definitions suppose that speaker 1 makes 8 attempts and all 8 times she is recognized correctly (she is dependable). Moreover, when all other 117 clients in the database make 1 attempt this speaker 1 is always in the lowest rank: 118 (she is resistant too). Then $R(1,1,1) = n_u(1) = 8$, $R(1,c,118) = n_u(c) = 1$ and so $GS(1,t) = 1, t = 1, \ldots, n_c$ and $RL(1,t) = \delta(t, 118)$. To make an ordering of the lambs we could have chosen the speakers who firstly cross the $RL = \alpha$-line, like is done with the confidence intervals. However, because we want to give an overall performance measure, we choose for a kind of integration over the ranking t. So the goat-sheep-score and the ram-lamb-score can then be defined as

$$S_{GS}(s) = \frac{1}{n_s}\sum_{t=1}^{n_s} GS(s,t) \text{ and } S_{RL}(s) = \frac{1}{n_s}\sum_{t=1}^{n_s} RL(s,t) \tag{4}$$

These scores are normalized between 0 and 1. We can now review the definition of goat, sheep, ram and lamb (although we are aware of the fact that is sounds more legitimate to base the definition of goat and sheep on the first best results only, but we also want to know if the results of those two selection methods correspond and which one is the most consistent.)

Goat	Unreliable speaker (with low goat-sheep-score)
Sheep	Dependable speaker (with high goat-sheep-score)
Ram	Resistant speaker (with low ram-lamb-score)
Lamb	Vulnerable speaker (with high ram-lamb-score)

In table 5 you see the classifications for two methods. Speakers that are not mentioned are modal by definition. We plotted the RL curves and the GS curves for these two methods as well in figures 2 - 4. Taking into account the confidence intervals of these results, which are not depicted in this paper, we state that speaker 71 is a goat. And it is nice to see that in the text independent case (method 03142) he is by far the most unreliable speaker. There are no real sheep, because the system performs very well overall, and there is no real ram but there is quite a set of resistant speakers. The lamb is client 66, although it is not really convincing, but in the text independent case it is much more clear that he is lamb.

Table 5: Dividing clients into categories (N-best)

	Method 01220		Method 01326	
	Subword	Word	Subword	Word
Goats	71 6 65	71	71 35 6 79	35 97 71 11
Sheep	28 24 107 42	-	24	-
Ram	72 52	52 72 56 89 64	72 64 89 28	89 64 72 28
Lamb	80 63	80 66	66 43 40	66 43 81

The advantage of this method compared to the confusion matrix method is that, because it uses the overall performance of speakers and clients and not only their champion performance, it also gives a more consistent classification which is less method dependent.

Finally it is important to know how this knowledge can be used in real life applications. Suppose a new client is enrolled for such a real life application. Then you, as the maintainer of that application, want to know the classification of this client to get a kind of a preview of his/her future performance. So you calculate the GS and the RL score of this new client and compare them with the scores of all registered speakers, and when he/she appears to be a sheep or a ram everything is fine and you do not expect severe problems with this speaker, but when he/she is a lamb you have to be careful with attempts assigned to that particular client or when he/she is a goat maybe you can even decide to give this new client entrance to your system via a human operator and not via the automatic SV/SI system.

7 Distance measures based on log likelihoods

Suppose a speaker s produces $n_u(s)$ genuine test utterances, the set of which will be denoted as

$$\mathbf{x}_s = \{x_s^u\}_{1\le u\le n_u(s)}. \tag{5}$$

For each utterance we compute the log likelihoods for each client model available. This gives a set of sorted log likelihoods **LL** with corresponding identities **I**.

$$\mathbf{LL}_s = \{LL^u_{s,t}\}_{1\leq u\leq n_u(s),1\leq t\leq n_c} \text{ with } LL^u_{s,1} \geq \ldots \geq LL^u_{s,n_c} \text{ and } \mathbf{I}_s = \{I^u_{s,t}\}_{1\leq u\leq n_u(s),1\leq t\leq n_c}. \tag{6}$$

We use the following distance measure based on utterance u of speaker s.

$$d(s, I^u_{s,t}) = (LL^u_{s,1} - LL^u_{s,t})/\mathrm{std}(LL^u_{s,.}), t = 1, \ldots, n_c \tag{7}$$

Please note that this is an asymmetric distance and also it is not necessarily so that $d(s,s) = 0, \forall s$, but

$$d(s,s) = 0, \forall s \Rightarrow \gamma = 0. \tag{8}$$

This is exactly where the relation with the classification appears again: the goats are the speakers with the highest distance to themselves, and the sheep are the speakers with $d(s,s) = 0$. The distance measure may be a more logical way to find the goats and sheep, since it takes into account the log likelihoods. And when the maximum log likelihood moves more into the tail of the distribution the ranking function stays the same, but the distance will increase as is correctly modeled by the introduced distance measure. In table 6 you see the goats and sheep according to this distance measure. It is not so easy to classify clients as ram or lamb based on this distance measure. But this is one point we are working on.

Table 6: Dividing clients into categories (using distance measure)

	Method 01220		Method 01326	
	Subword	Word	Subword	Word
Goats	71 66 65	71 19 97 11	71 110 65	11 71 35 97
Sheep	24 107 42	-	24	-

8 Conclusion

From this paper it slowly becomes clear how difficult it is to make a method independent classification of speakers. We have done a new proposal for the goat, sheep, ram, lamb classification, which is more consistent over the text dependence levels than the existing definition. But it seems almost impossible to construct a classificator which is totally method independent, although some things are clear with all classificators and almost all methods: speaker 71 is a unreliable speaker and speaker 66 is a vulnerable speaker. So we recommend that for each different method a new classification has to be done, because the results show that within each method the different classificators have a much higher degree of agreement. Further we showed that especially for the ram-lamb-classification 1-best is not enough. Therefore we propose to use a n-best classificator. At least we have the point that there is a real need for general classificators, since the proposals as made in EAGLES do not suffice.

References

[1] Higgins A., Bahler L. and Porter J., *Speaker Verification Using Randomized Phrase Prompting*, Digital Signal Processing, **1**, pp. 89-106, 1991.

[2] European Advisory Groups on Language Engineering Standards, Spoken Language Working Group, 1995.

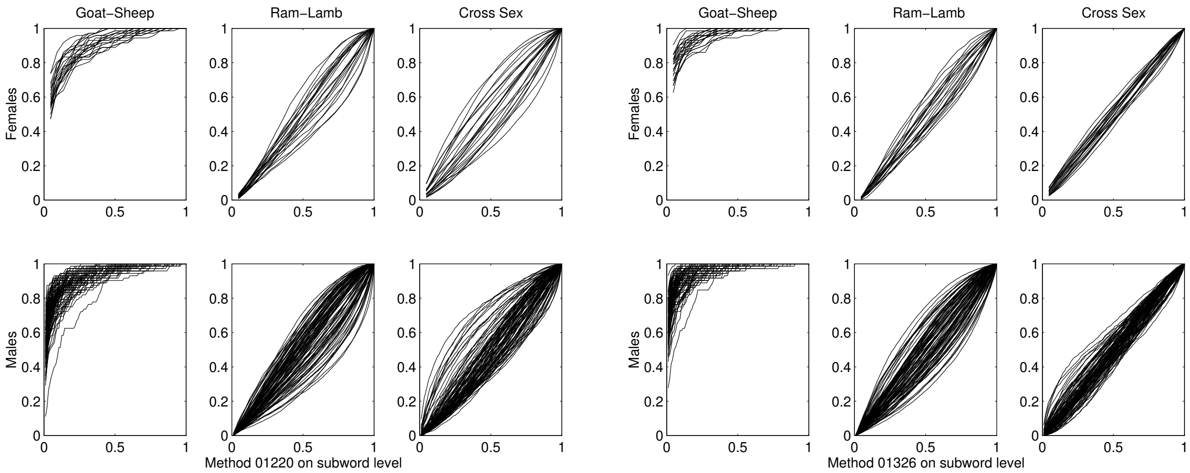

Figure 2: Curves on subword level for methods 01220 and 01326

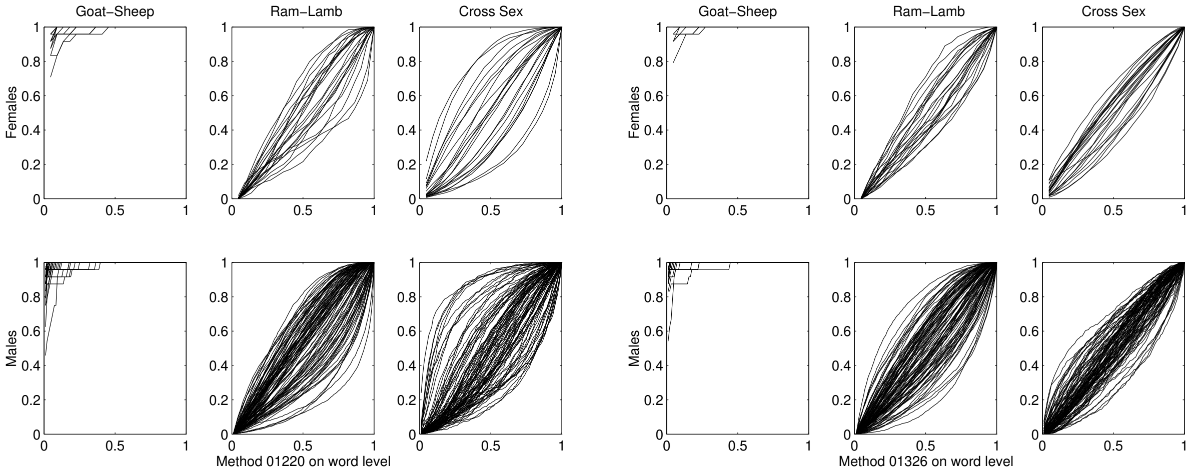

Figure 3: Curves on word level for methods 01220 and 01326

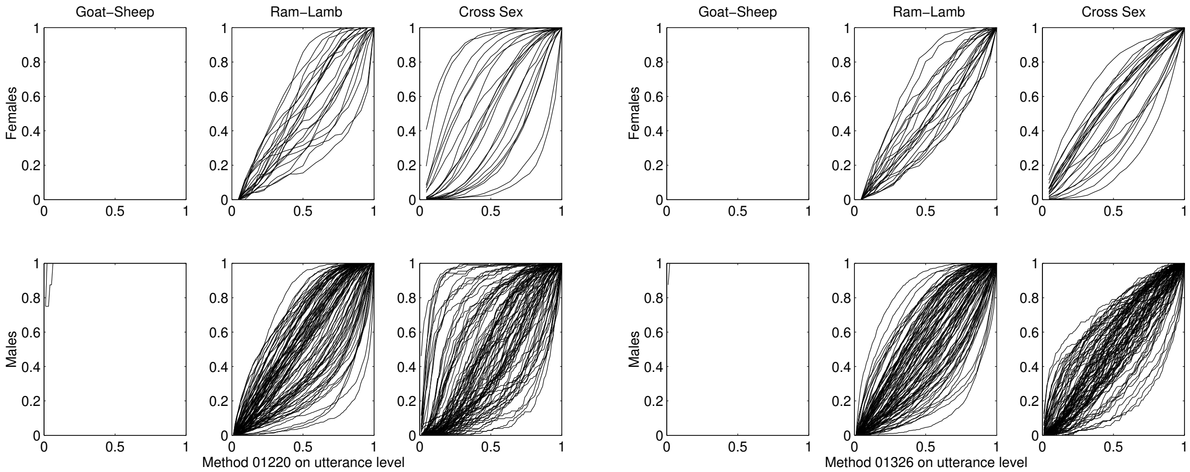

Figure 4: Curves on utterance level for methods 01220 and 01326