# Bayes Risk Approximations Using Time Overlap with an Application to System Combination

*Björn Hoffmeister, Ralf Schlüter, Hermann Ney*

Human Language Technology and Pattern Recognition
Computer Science Department, RWTH Aachen University, Germany,
hoffmeister@cs.rwth-aachen.de

## Abstract

The computation of the Minimum Bayes Risk (MBR) decoding rule for word lattices needs approximations. We investigate a class of approximations where the Levenshtein alignment is approximated under the condition that competing lattice arcs overlap in time. The approximations have their origins in MBR decoding and in discriminative training. We develop modified versions and propose a new, conceptually extremely simple confusion network algorithm. The MBR decoding rule is extended to scope with several lattices, which enables us to apply all the investigated approximations to system combination. All approximations are tested on a Mandarin and on an English LVCSR task for a single system and for system combination. The new methods are competitive in error rate and show some advantages over the standard approaches to MBR decoding.

**Index Terms**: speech recognition, minimum bayes risk, confusion network, system combination, discriminative training

## 1. Introduction

Minimum Bayes Risk (MBR) decoding in large vocabulary continuous speech recognition (LVCSR) aims at finding the word sequence which minimizes the expected Levenshtein distance given a sequence of acoustic features $x_1^T$. In the presence of several systems the MBR decoder can be simply extended to a system combination approach by averaging the posterior probabilities of a word sequence. Equation (1) shows the resulting MBR decoding rule for the combination of $I$ systems.

$$\mathcal{MBR}(x_1^T) := \operatorname*{argmin}_{w_1^N \in \mathcal{H}} \sum_{v_1^M \in \mathcal{S}} \left[ \sum_{i=1}^{I} \lambda_i p_i(v_1^M | x_1^T) \right] \mathcal{L}(w_1^N, v_1^M) \tag{1}$$

The system weights $\lambda_i$ sum up to one and the hypothesis space $\mathcal{H}$ and summation space $\mathcal{S}$ contain ideally all possible word sequences. In practice, MBR decoding is usually applied in a rescoring step on an $N$-best list or word lattice. The summation space $\mathcal{S}$ becomes then the union of the system-dependent lattices $L_1^I$ and the hypothesis space is usually either equal to or a superset of $\mathcal{S}$, e.g. in confusion network (CN) decoding.

Still with the restricted summation and hypothesis space a direct computation of the MBR hypothesis is prohibitive and additional approximations are required. The approximation can happen by further reducing hypothesis and summation space, e.g. by lattice pinching [1] or using rather short $N$-best lists, or by approximating the cost function, i.e. the Levenshtein distance. In this work we investigate a class of Levenshtein distance approximations working on lattices and relying on arc-wise time overlap.

The most popular approximation of this kind is the CN algorithm introduced in [2]. The time overlap is used to avoid the alignment of two arcs lying on the same path. The CN algorithm can be extended to system combination in two ways: either by directly constructing the CN from the lattice union or by building a CN from each lattice and subsequently aligning the CNs [3]. An alternative approximation based on the time-frame error (min.fWER) and its extension to system combination are introduced in [4, 5].

In the next section we discuss the bias of MBR decoders, especially of the approximations applied in CN and min.fWER decoding. A deeper investigation of the min.fWER approach and a comparison with the approximations used in lattice-based discriminative model training yield several new, improved approximative costs. Though having some nice properties, the new approximations require precise word boundaries and are computationally expensive. An algorithm that overcomes these problems is presented in Section 3: a new, conceptually very simple CN algorithm based on the same statistics as used for min.fWER decoding.

Section 4 describes the experimental setup and presents and discusses the results. The last section draws conclusions.

## 2. Approximations for Lattice-based MBR Decoding using Arc-wise Time Overlap

The lattice produced by the $i$th system is denoted by $L_i$ and the union of the lattices by $L$. A path $\pi$ through the lattice consists of a set of lattice arcs and each arc $a$ has a begin time $b(a)$, an end time $e(a)$, and a label $l(a)$. All labels not representing a word, like silence and noise, are mapped to the empty word $\epsilon$. We abbreviate the notation of the arc duration by $d(a)$ and the time overlap of two arcs by $o(a, a')$.

We replace the cost by an accuracy function which simplifies the following formulas. The accuracy functions we investigate share two properties: the accuracy calculation is local, i.e. the computation is independent of the accuracies of other arcs, and it depends only on arcs that overlap in time. The following equation shows the general form of the approximated MBR decoding rule using a local accuracy.

$$\mathcal{MBR}_{\text{approx.}}(L_1^I) := \operatorname*{argmax}_{\pi \in \mathcal{H}} \sum_{i=1}^{I} \lambda_i \sum_{\tilde{\pi} \in L_i} p_i(\tilde{\pi} | x_1^T) \sum_{a \in \pi} acc(a, \tilde{\pi}) \tag{2}$$

Figure 1 lists the approximative accuracies we discuss in the remainder of the Section. We define the set of arcs in path $\tilde{\pi}$ that compete with arc $a$ as $O_\beta(a, \tilde{\pi}) := \{\tilde{a} \in \tilde{\pi} : o(a, \tilde{a})/d(\tilde{a}) > \beta\}$; $\beta$=0 means any overlap in time. For the sake of simplicity, we omit details like the handling of $O_\beta(a, \tilde{\pi}) = \emptyset$ and $\epsilon$-arcs.

The requirement of the time overlap anticipates that arcs lying on the same path compete with each other. Contradictory, in practice the Levenshtein distance aligns words and arcs without overlap in time, especially short words like the English *a* or *I*. The reasons include the fuzzy word boundaries in continuous speech, the discretisation of the audio signal, and variations in the time stamps across systems. This gives the approximated MBR decoding a general bias towards more deletions.

$$acc_{\text{CN}}(a,\tilde{\pi}) \quad := \quad \max_{\tilde{a}\in O_0(a,\tilde{\pi})} \left\{ \delta\left(\text{slot}(a), \text{slot}(\tilde{a})\right) \delta\left(l(a), l(\tilde{a})\right) \right\} \tag{3}$$

$$acc_{\text{frame}}(a,\tilde{\pi}) \quad := \quad \sum_{\tilde{a}\in O_0(a,\tilde{\pi})} \left[ \gamma \frac{o(a,\tilde{a})}{1+\alpha(d(a)-1)} + (1-\gamma)\frac{o(a,\tilde{a})}{1+\alpha(d(\tilde{a})-1)} \right] \delta\left(l(a), l(\tilde{a})\right) \tag{4}$$

$$acc_{\text{mDT}}(a,\tilde{\pi}) \quad := \quad \max_{\tilde{a}\in O_\beta(a,\tilde{\pi})} \left\{ \phi\left(-1+\frac{o(a,\tilde{a})}{d(\tilde{a})}\right) + \chi\left(-1+\frac{o(a,\tilde{a})}{d(a)}\right) + \frac{o(a,\tilde{a})}{d(\tilde{a})}\delta\left(l(a), l(\tilde{a})\right) \right\} \tag{5}$$

$$acc_{\text{disc.}}(a,\tilde{\pi}) \quad := \quad \max_{\tilde{a}\in O_\beta(a,\tilde{\pi})} \left\{ \delta\left(l(a), l(\tilde{a})\right) \right\} \tag{6}$$

Figure 1: Local accuracies based on time overlap for hypothesized arc $a$ and competing path $\tilde{\pi}$; $\delta(\cdot,\cdot)$ is the Kronecker-$\delta$.

The first approximation, Equation (3), is the accuracy for CN decoding. In contrast to the following accuracies, the computation itself does not depend on time overlap, but on a global alignment. In a preceding step all arcs are clustered into slots, where lattice-based CN algorithms use the time overlap to ensure that only competing arcs are clustered together. The slots build the CN and define the alignment between hypotheses and competitors. Figure 3 illustrates the derivation of a CN from a lattice. The hypothesis space $\mathcal{H}$ for CN decoding includes all paths through the CN, which is a superset of the union $L$. However, the single global alignment and the arc clustering step cause a further deletion bias, for a detailed discussion see [2].

From Equation (4) we get the min.fWER decoding rule by setting $\gamma$=1. The basic idea is to count time frame instead of word accuracies, which however favors long words. In order to get a more word-wise accuracy the factor $\alpha$ in the denominator of Equation (4) allows a smooth normalization of the time frame accuracy. But because the normalization happens on the hypothesis side, the decoding penalizes substitutions and insertions, but deletions are ignored. On the other hand, normalizing on the reference side ignores insertions. Setting $\gamma$ to a value between zero and one interpolates between both normalizations and allows to balance between deletion and insertion bias. By default we use the time conditioned form of the union $L$ as hypothesis space for the min.fWER decoder [5].

In min.fWER decoding we sum up the fractional mismatches between $a$ and each overlapping arc. Alternatively, we can align $a$ to the competing path. This approach has its origin in lattice-based discriminative acoustic model training. Equation (5) with $\phi$=1 and $\chi$=0 (and $\beta$=0) is the approximated accuracy used in [6] for minimum word or phone error training. In its original form the approximation has a tendency to penalize insertions higher than deletions as pointed out in [7]. In the same paper an alternative accuracy function is proposed. However, this function is expensive and it requires an HMM alignment which we do not necessarily possess when doing cross-site system combination. Instead, by setting $\chi$>0 we add an additional deletion penalty: if a long hypothesis word $a$ competes with a much shorter word $\tilde{a}$, then presumably a deletion takes place and is penalized by the $\chi$-term.

The use of fractional values is a tribute to the locality of the accuracy approximation, because two hypothesis words $a$ and $a'$ can be assigned to the same competing word $\tilde{a}$. We can avoid the flaw in the alignment by requiring that a $a$ (or $a'$) can only be aligned with $\tilde{a}$ if the fractional overlap exceeds one half. Setting $\beta$=0.5 and $\phi$=$\chi$=1 in Equation (5) implements the approach. Interestingly, this changes the interpretation of the $\phi$- and the $\chi$-term: the $\phi$-term accounts for the deleted fraction of competitor $\tilde{a}$ and the $\chi$-term for the fractional insertion by hypothesis $a$. Instead of the fractional values we can use discrete accuracies, i.e. 1 for a correctly aligned word, which yields Equation (6).

## 3. CN algorithm based on Frame-Wise Word-Posterior Probabilities

Besides the CN algorithm all the accuracy approximations discussed in the last section rely on precise word boundaries. Also, we discussed in the last section why word boundaries are usually not reliable, so much the worse for cross-site system combination. Lattice-based CN decoders like [2] are more robust to fuzzy word boundaries, because the time information is only used (among other information) in the arc clustering step. For cross-site system-combination the CN combination technique described in [3] can be used, which does not rely at all on word boundaries. From a theoretical point of view a further advantage of a CN decoder is the larger hypothesis space.

The crucial step in lattice-based CN decoding is the arc clustering which yields the CN; decoding the CN is then straightforward. We propose a new algorithm for arc clustering which has the following properties

1. all overlapping arcs with the same label are clustered together (if a non-ambiguous solution exists)
2. $\epsilon$-arcs, e.g. noise arcs, do not affect the clustering result
3. no distance function is required

and aims at finding a compact CN, i.e. a CN with few slots; the number of slots is directly related to the deletion bias. The first property is desired, because in most cases it means that these arcs are competitors. Properties two and three go hand-in-hand. Usually, CN algorithms use distance functions which compute the similarity between arcs and arc clusters. This has the undesirable consequence that the boundaries between succeeding $\epsilon$-arcs influence the outcome of the clustering. Simply not using distance functions avoids the problem and makes the proposed algorithm conceptually extremely simple.

The pseudo code of the proposed clustering algorithm is given in Figure 2. For each iteration the algorithm updates the frame-wise word-posterior probabilities $p_t(w|x_1^T)$ and use them to find a time frame $t_S$ that represents the next slot. The basic concept is illustrated in Figure 3.

The main iteration starts in line 3 and consists of three steps. The first step updates the frame-wise word posteriors; $l_t(\pi)$ is the label of that arc in path $\pi$ that overlaps with time frame $t$.

In the second step (lines 9 to 14) we choose $t_S$. First, we go over all arcs and consider for each arc only those time frames for which the overlap with competing arcs having the same label is maximized, i.e. where property 1 is fulfilled. Then, we choose among the pre-selected time frames the one $t_S$ that minimizes the probability of the empty word, which turned out to be a reasonable heuristic for getting a compact CN.

In step three (lines 17 to 22) we select the competing arcs for the slot and set their labels to $\epsilon$ for the coming updates of the frame-wise word-posteriors. Then we proceed with step one.

```
# build slots from non-$\epsilon$ arcs
$CN \leftarrow []$;  $A \leftarrow \{a \in Arcs(L) : l(a) \neq \epsilon\}$
while $A \neq \emptyset$ do
  # update frame-wise word-posteriors
  foreach $(t, w) \in [1, T] \times \mathcal{V} \cup \{\epsilon\}$ do
    $p_t(w|x_1^T) \leftarrow \sum_{\pi \in L: l_t(\pi)=w} p(\pi|x_1^T)$
  # find slot-building time frame
  $t_S \leftarrow \infty$
  foreach $a \in A$ do
    p_max $\leftarrow \max_{b(a) \leq t \leq e(a)} p_t(l(a)|x_1^T)$
    foreach $b(a) \leq t \leq e(a)$ do
      if $p_t(l(a)|x_1^T) =$ p_max then
        if $p_t(\epsilon|x_1^T) < p_{t_S}(\epsilon|x_1^T)$ then $t_S \leftarrow t$
  # build slot
  $S \leftarrow []$
  foreach $a \in A$ with $b(a) \leq t_S \leq e(a)$ do
    p_max $\leftarrow \max_{b(a) \leq t \leq e(a)} p_t(l(a)|x_1^T)$
    if $p_{t_S}(l(a)|x_1^T) =$ p_max then
      insert($S$, $a$)
      $A \leftarrow A \setminus \{a\}$;  $l(a) \leftarrow \epsilon$
  insert($CN$, $S$)
finalize($CN$)
```

Figure 2: Confusion network algorithm based on frame-wise word-posterior probabilities.

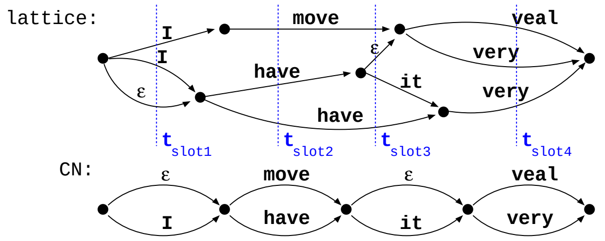

Figure 3: Illustration of the proposed CN algorithm.

## 4. Experimental Results

We test the accuracy approximations discussed in Section 2 and the CN algorithm proposed in Section 3 on two tasks. The first is a Mandarin BN/BC task. For our experiments we use three systems that are ML-trained on 230 hours [8]. All systems share the same pronunciation lexicon and language model, but have different acoustic front-ends: MFCCs, PLPs, and gammatone filter bank based features. Lattice-based MBR decoding is performed on character level. The development and test set are taken from the GALE 2007 Evaluation and consist of 2.5 and 1.6 hours, respectively.

For the second task we use the lattices that were shared across sites within the TC-Star/EPPS 2007 English Evaluation. The corpus and the lattice sets are described in [9]; development and test set consist of 3.2 and 2.9 hours, respectively.

The CN decoders and the min.fWER decoder use the extended hypothesis spaces as described in the previous sections. Our standard CN decoder is based on an arc clustering algorithm similar to the approach described in [2], but incorporates the idea of using a pivot-path as described in [10] to speed up the clustering. On the EPPS English task the decoder proofed to be competitive to the CN decoders from other sites.

The other accuracy approximations use only the lattice union as hypothesis space. This has no crucial impact on the Mandarin task, where the acoustic segments are short. Unlike the English EPPS task, where a lattice spans over approximately 30 minutes. Using the lattice union as hypothesis space does not allow to switch between hypotheses from different systems within the 30 minutes, which causes a degradation in WER.

Table 1: Results for the Mandarin BN/BC task.

| | CER[%] (del/ins) error | |
|---|---|---|
| | dev07 | eval07 |
| single system | | |
| Viterbi | (2.7/1.7) 14.8 | (4.5/0.9) 15.0 |
| CN (standard) | (2.9/1.5) 14.5 | (4.6/0.8) 14.7 |
| CN (proposed) | (2.9/1.5) 14.5 | (4.6/0.8) 14.7 |
| min.fWER | (3.1/1.4) 14.5 | (4.8/0.8) 14.8 |
| $acc_{\text{frame}}, \gamma{=}1$ | (3.0/1.5) 14.6 | (4.8/0.8) 14.8 |
| $acc_{\text{frame}}, \alpha{=}1$ | (2.7/1.6) 14.5 | (4.5/0.9) 14.7 |
| $acc_{\text{mDT}}, \beta{=}0, \phi{=}1, \chi{=}0$ | (2.9/1.5) 14.6 | (4.7/0.8) 14.8 |
| $acc_{\text{mDT}}, \beta{=}0$ | **(2.3/1.9)** 14.5 | **(4.2/1.1)** 14.8 |
| $acc_{\text{mDT}}, \phi{=}\chi{=}1$ | (2.8/1.5) 14.5 | (4.6/0.8) 14.7 |
| $acc_{\text{disc}}$ | (2.7/1.6) 14.5 | (4.5/0.9) 14.8 |
| three systems | | |
| ROVER | (2.7/1.3) 13.2 | (4.5/0.7) 13.9 |
| CNC (standard) | (2.8/1.3) 13.2 | (4.7/0.7) 13.7 |
| CNC (proposed) | (2.8/1.3) 13.2 | (4.6/0.7) 13.6 |
| CN (standard) | (2.9/1.2) 13.1 | (4.8/0.7) 13.7 |
| CN (proposed) | (2.8/1.3) 13.1 | (4.6/0.7) 13.7 |
| min.fWER | (3.1/1.2) 13.2 | (4.7/0.7) 13.7 |
| $acc_{\text{frame}}, \gamma{=}1$ | (2.7/1.4) 13.2 | (4.5/0.8) 13.7 |
| $acc_{\text{frame}}, \alpha{=}1$ | (2.8/1.3) 13.1 | (4.6/0.7) 13.6 |
| $acc_{\text{mDT}}, \beta{=}0, \phi{=}1, \chi{=}0$ | (3.0/1.2) 13.3 | (4.7/0.7) 13.8 |
| $acc_{\text{mDT}}, \beta{=}0$ | **(2.4/1.5)** 13.1 | **(4.3/0.8)** 13.7 |
| $acc_{\text{mDT}}, \phi{=}\chi{=}1$ | (2.7/1.3) 13.2 | (4.5/0.7) 13.7 |
| $acc_{\text{disc}}$ | (2.6/1.4) 13.2 | (4.5/0.8) 13.9 |

For each setup we tune the following parameters on the development set using the Nelder-Mead downhill simplex algorithm: acoustic and language model scale, a system weight, and all free parameters shown in Figure 1(including $\beta$).

Table 1 shows the results for the Mandarin task, the upper part for a single system and the lower part for the combination of three systems. For the CN-based system combination we tested the CN combination proposed in [3] (CNC) and the derivation of the CN directly from the lattice union (CN).

From the results we see that the accuracy approximations from Section 2 work well and are competitive to the standard approximations for lattice-based MBR decoding. In a direct comparison the extended min.fWER approach ($acc_{\text{frame}}, \alpha{=}1, \gamma$ optimized) performs a little better than the standard min.fWER approach ($acc_{\text{frame}}, \alpha$ optimized, $\gamma{=}1$). As expected, the original version of the accuracy approximation used in lattice-based minimum word or phone error training ($acc_{\text{mDT}}, \beta{=}0, \phi{=}1, \chi{=}0$) shows a tendency to produce many deletions. The modified versions can compensate for the deletion bias and show slightly better error rates. Especially, $acc_{\text{mDT}}$ with $\beta{=}0$ is one of the best performing accuracies and has the lowest del/ins ratio among all approximations.

Comparing our proposed CN algorithm with the standard implementation we observe almost equal results, though the proposed method is conceptually much simpler. For CN-based system combination we see neither an advantage for the CN combination nor for building the CN from the lattice union.

The results for the English EPPS task are summarized in Table 2. We give system combination results only for two systems, because for more systems the parameter optimization was infeasible when using one of the accuracies based on a local alignment($acc_{\text{mDT}}$ and $acc_{\text{disc}}$). In general, the results on the English task support our observations on the Mandarin task. The rather bad results for the new accuracy approximations for the system combination experiments come presumably from the restricted hypothesis space. Looking at the min.fWER

Table 2: Results for the English EPPS task.

| | WER[%] (del/ins) error | |
|---|---|---|
| | dev07 | eval07 |
| single system | | |
| Viterbi | (1.5/1.3) 8.5 | (1.9/1.3) 9.7 |
| CN (standard) | (1.5/1.1) 8.2 | (2.1/1.1) 9.5 |
| CN (proposed) | (1.5/1.1) 8.2 | (2.0/1.1) 9.5 |
| min.fWER | (1.8/1.0) 8.2 | (2.4/1.0) 9.5 |
| $acc_{\text{frame}}, \gamma{=}1$ | (1.8/1.0) 8.3 | (2.4/1.0) 9.6 |
| $acc_{\text{frame}}, \alpha{=}1$ | (1.7/1.0) 8.2 | (2.3/1.0) 9.5 |
| $acc_{\text{mDT}}, \beta{=}0, \phi{=}1, \chi{=}0$ | (1.6/1.1) 8.3 | (2.1/1.1) 9.6 |
| $acc_{\text{mDT}}, \beta{=}0$ | (1.6/1.1) 8.3 | (2.1/1.1) 9.5 |
| $acc_{\text{mDT}}, \phi{=}\chi{=}1$ | (1.5/1.2) 8.3 | (2.0/1.2) 9.6 |
| $acc_{\text{disc}}$ | (1.6/1.3) 8.5 | (2.0/1.3) 9.7 |
| two systems | | |
| ROVER | (1.7/0.9) 6.7 | (2.2/0.8) 7.8 |
| CNC (standard) | (1.4/0.8) 6.4 | (1.9/0.8) 7.5 |
| CNC (proposed) | (1.4/0.8) 6.4 | (1.9/0.8) 7.6 |
| CN (standard) | (1.6/0.8) 6.4 | (2.2/0.7) 7.6 |
| CN (proposed) | (1.5/0.7) 6.4 | (2.0/0.7) 7.5 |
| min.fWER | (1.6/0.9) 6.6 | (2.0/0.8) 7.7 |
| $acc_{\text{frame}}, \gamma{=}1$ | (1.8/0.9) 7.2 | (2.1/0.9) 8.3 |
| $acc_{\text{frame}}, \alpha{=}1$ | (1.5/1.2) 6.9 | (1.9/1.0) 7.8 |
| $acc_{\text{mDT}}, \beta{=}0, \phi{=}1, \chi{=}0$ | (1.7/0.8) 6.9 | (2.3/0.8) 8.0 |
| $acc_{\text{mDT}}, \beta{=}0$ | (1.5/0.9) 6.8 | (2.1/0.8) 8.0 |
| $acc_{\text{mDT}}, \phi{=}\chi{=}1$ | (1.5/0.9) 6.8 | (2.0/0.8) 8.0 |
| $acc_{\text{disc}}$ | (1.5/0.9) 7.0 | (2.0/0.9) 8.1 |

experiments using the extended hypothesis space (min.fWER) and using the hypothesis space restricted to the lattice union ($acc_{\text{frame}}, \gamma{=}1$) we observe a clear degradation in error rate.

Again, CNC is not better than building the CN from the lattice union. Noteworthy, for combining three or four systems from different sites we observe a small advantage for CNC.

In the remainder we discuss shortly the runtime and computational problems of the different approximations. Measurements are done on a Core2Duo with 2.4GHz and 4GB memory. Our Viterbi decoder (inclusive confidence score computation), our standard CN decoder, and our min.fWER decoder have approximately the same runtime and need around 15 seconds for each development set. For system combination the runtime increases almost linear with the number of systems: for two systems it doubles, for three it triples.

The runtime of the other approximations depend much more on the acoustic segmentation and on the current lattice. The proposed CN algorithm needs between 30 seconds on Mandarin and 120 seconds on the long English lattices; for the combined lattices it needs 90 and 130 seconds, respectively. In the computation of the extended min.fWER accuracy ($acc_{\text{frame}}, \alpha{=}1, \gamma{<}1$) we have to iterate for each arc over all competing arcs. The resulting computational cost depends highly on the lattice structure and density: for the English lattices the runtime is equal to the fastest algorithm, but for the more dense Mandarin lattices it goes up to 30 seconds. In system combination the runtime for English is still equal to the standard approaches, but 150 seconds for the Mandarin task.

The remaining approximations are based on an alignment of the hypothesis arc $a$ with the partial path consisting of all consecutive lattice arcs that overlap with $a$. The explicit iteration over all these partial paths can be very expensive if the lattice has a malicious structure, e.g. a long word competing with a large, highly connected cloud of short words, so happened for the combination of three or four systems for the English task. The runtime for these algorithms is around 100 seconds for Mandarin and 50 seconds for English; for the lattice unions it is 750 and 100 seconds, respectively.

## 5. Conclusions

In this work we developed and compared several approximations for lattice-based minimum Bayes risk (MBR) decoding which rely on the time overlap of lattice arcs. The approximations include a new confusion network (CN) algorithm and variations of the accuracy approximations used in lattice-based discriminative acoustic model training.

We tested the approximations on a Mandarin and a English task for single lattice decoding and for system combination. The approximations turned out to be competitive to standard MBR approximations like the CN combination and decoding, though some of them are computationally much more expensive. The results indicate that the approximations used in discriminative training are reasonable. However, this work introduces some modifications which improve the approximations.

The proposed CN algorithm is conceptually extremely simple but competitive in error rate with our standard CN decoder.

## 6. Acknowledgements

This material is partly based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA).

This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

## 7. References

[1] V. Goel, S. Kumar, and W. Byrne, "Segmental minimum bayes-risk decoding for automatic speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 3, pp. 234–249, 2004.

[2] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, 2000.

[3] G. Evermann and P. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *NIST Speech Transcription Workshop*, College Park, MD, 2000.

[4] F. Wessel, R. Schlüter, and H. Ney, "Explicit word error minimization using word hypothesis posterior probabilities," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, May 2001.

[5] B. Hoffmeister, T. Klein, R. Schlüter, and H. Ney, "Frame based system combination and a comparison with weighted ROVER and CNC," in *Proc. Int. Conf. on Spoken Language Processing*, Pittsburgh, PA, USA, Sep. 2006.

[6] D. Povey and P. C. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 1, Orlando, FL, USA, May 2002, pp. 105–108.

[7] J. Zheng and A. Stolcke, "Improved discriminative training using phone lattices," in *Proc. European Conf. on Speech Communication and Technology*, Lisboa, Portugal, Dec. 2005, pp. 2125–2128.

[8] C. Plahl, B. Hoffmeister, M.-Y. Hwang, D. Lu, G. Heigold, J. Lööf, R. Schlüter, and H. Ney, "Recent improvements of the RWTH GALE mandarin LVCSR system," in *Proc. Int. Conf. on Spoken Language Processing*, Brisbane, Australia, Sep. 2008, pp. 2426–2429.

[9] B. Hoffmeister, R. Schlüter, and H. Ney, "iCNC and iROVER: The limits of improving system combination with classification?" in *Proc. Int. Conf. on Spoken Language Processing*, Brisbane, Australia, Sep. 2008, pp. 232–235.

[10] D. Hakkani and G. Riccardi, "A general algorithm for word graph matrix decomposition," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Hong Kong, Apr. 2003.