Recovery from Model Inconsistency in Multilingual Speech Recognition

Report from JHU workshop 2007

Hynek Hermansky [4], Lukáš Burget [1], Petr Schwarz [1], Pavel Matějka [1],
Mirko Hannemann [2], Ariya Rastrow [3], Christopher White [3],
Sanjeev Khudanpur [3], and Jan "Honza" Černocký [1]

(1) Speech@FIT, Brno University of Technology, Czech Republic,
{burget,schwarzp,matejkap,cernocky}@fit.vutbr.cz
(2) Magdeburg University, Germany, mirko.hannemann@student.uni-magdeburg.de
(3) Johns Hopkins University, USA, {ariya,cmileswhite,khudanpur}@jhu.edu
(4) IDIAP Research Institute, Switzerland, hynek@idiap.ch

Summary of planned work

Current ASR has difficulties in handling unexpected words that are typically replaced by acoustically acceptable high prior probability words. Identifying parts of the message where such a replacement could have happened may allow for corrective strategies.

We aim to develop data-guided techniques that would yield **unconstrained estimates of posterior probabilities** of sub-word classes employed in the stochastic model solely from the acoustic evidence, i.e. without use of higher level language constraints.

These posterior probabilities then could be compared with **the constrained estimates of posterior probabilities** derived with the constraints implied by the underlying stochastic model. Parts of the message where any significant mismatch between these two probability distributions is found should be re-examined and corrective strategies applied.

This may allow for development of systems that are able to indicate when they "do not know" and eventually may be able to "learn-as-you-go" in applications encountering new situations and new languages.

During the 2007 Summer Workshop we intend to focus on detection and description of out-of-vocabulary and mispronounced words in the 6 language Call-home database. Additionally, in order to describe the suspect parts of the message, we will work on language-independent recognizer of speech sounds that could be applied for phonetic transcription of identified suspect parts of the recognized message.

Team members

Team Leader

Hynek Hermansky	IDIAP	hynek@idiap.ch

Senior Researchers

Lukas Burget	Brno University of Technology	burget@fit.vutbr.cz
Sanjeev Khudanpur	Johns Hopkins University	khudanphur@jhu.edu
Chin-Hui Lee	Georgia Technical Institute	chl@ece.gatech.edu
Haizhou Li	Institute for Infocomm Research	hli@i2r.a-star.edu.sg
Jon Nedel	Department of Defense	jnedel@gmail.com
Geoffrey Zweig	Microsoft	gzweig@microsoft.com

Graduate Students

Pavel Matejka	Brno University of Technology	matejkap@fit.vutbr.cz
Ariya Rastrow	Johns Hopkins University	ariya@jhu.edu
Petr Schwartz	Brno University of Technology	schwarzp@fit.vutbr.cz
Rong Tong	Nanyang Technological University	tongrong@i2r.a-star.edu.sg
Chris White	Johns Hopkins University	cmileswhite@jhu.edu

Undergraduate Students

Mirko Hannemann	Magdeburg University, Germany	mirko.hannemann@idiap.ch
Sally Isaacoff	University of Michigan	sisaacof@umich.edu
Puneet Sahani	NSIT; Delhi University	sahani.puneet@gmail.com

Acknowledgements

This research was conducted under the auspices of the 2007 Johns Hopkins University Summer Workshop, and partially supported by the US Department of Defense (Contract No H-98230-07-C-0365),

National Science Foundation (Grant No IIS-0121285) and Defense Advance Research Projects Agency (GALE Program).

Speech@FIT researchers were partly supported by European project AMIDA (IST-033812), by Grant Agency of Czech Republic under project No. 102/05/0278 and by Czech Ministry of Education under projects No. MSM0021630528 and LC06008. Lukáš Burget was supported by Grant Agency of Czech Republic under project No. GP102/06/383. The hardware used in this work was partially provided by CESNET under projects Nos. 162/2005 and 201/2006.

Thanks to Martin Karafiát and Franta Grézl who provided valuable support to the JHU team at Brno base-station. Thanks also to AMIDA LVCSR team, especially to Thomas Hain from University of Sheffield.

We are grateful to Milind Mahajan from MSR for the MaxEnt code.

Contents

Chapter 1

Introduction

A message is sent by an addresser to an addressee. For this to occur, the addresser and addressee must use a common code, a physical channel, or contact, **and the same frame of reference, or context**.
Roman Jakobson

Insperata accidunt magis saepe quam quae speres.
(Things you do not expect happen more often than things you do expect)
Plautus (cca 200 B.C.)

As put out so eloquently by Roman Jakobson, without some common prior knowledge on both sides of the conversation, the human speech communication would not be possible. Thus, prior knowledge of a speaker and of a listener that are engaged in human speech communication plays a major role in getting the message through.

However, any meaningful communication must also include at least some element of surprise. As pointed out by Plautus more than two millennia ago, unexpected happens. Without unexpectedness, there is no information conveyed.

The two quotations above characterize well the dilemma in use of prior information in speech communication. Skillful balancing of the predictable and the unpredictable elements in the message makes a master in communication.

The same can be said about balancing these two elements in automatic recognition of speech (ASR). One of the most significant contributions of stochastic approaches to ASR is its systematic use of a language model, i.e. of a prior knowledge (what R. Jacobson may call the context). However, too much reliance on priors can be damaging. Incidents such as the performance of the otherwise successful system in the early ARPA-SUR program where a cough of one of reviewers caused the system to recognize the right command for a move by a machine in a chess-play (since that was the only reasonable move given the position in a game) [1] are still possible even with current state-of-the-art ASR systems. Just as in human speech communication, reaching the right balance between the reliance on priors and the reliance on the actual sensory is needed.

The problem is the most obvious when dealing with words that are unknown to the recognition machine. These so called **out-of-vocabulary words (OOVs)** are source of serious errors in current large vocabulary continuous speech recognition systems (LVCSR). They are *unavoidable*, as human speech contains proper names, out-of-language and invented words, and also *damaging*, as it is known, that one OOV in input speech typically generates two or more word recognition errors.

Still, OOVs usually do not have large impact on the word error rate (WER) of LVCSR, as they are rare. As such, in the current ASR culture where the WER is of the prime interest, they sometimes

do not get the the full attention.

And the attention they rightly deserve! Rare and unexpected words tend to be information rich. Their reliable detection can significantly increase the practical utility of ASR technology in may important applications since it can lead to a possible additional (machine or human) processing, to automatic update of recognizer's vocabulary, or to description of the OOV region by phonemes.

1.1 Current ASR

Current ASR finds the unknown utterance w by finding such a model $M(w_i)$ that with the highest probability $P(M(w_i))$ best accounts for the data x, i.e.

$$w = \arg\max_i P(M(w_i)|x)$$

This is being solved trough the use of the Bayes rule:

$$w \propto \arg\max_i p(x|M(w_i))P(M(w_i)) \tag{1.1}$$

where likelihood $p(x|(M(w_i))$ represents the conditional probability of the observed data and the prior knowledge is represented by the language model $P(M(w_i))$.

In the large vocabulary conversational speech (LVCSR) the model of the utterance $M(w_i)$ typically represents a cascade of models of (conditionally independent) individual words, the likelihood of the whole utterance w is given by a product of likelihoods of the individual words $M(w^j)$:

$$p(x|M(w_i)) = \prod_j p(x|M(w_i^j)).$$

As clear, for the particular w_i to be chosen as the recognized one, both the likelihood $p(x|(M(w_i))$ and the language model $P(M(wi))$ need to be reasonably high.

As long as the goal is to correctly recognize as many words as possible, i.e. to optimize the average word error rate (WER), the use of Bayes rule is optimal. However, in communication of information by speech, not words are created equal. Especially the words that are highly predictable from the context of a discourse are typically less important than the words that are unexpected.[1]

So when the information extraction is a target, the crucial issue may be to correctly recognize high information value words, and these might be precisely these words that are not favored by the language model $P(M(w))$, since low likelihood of any of the words in the utterance makes the overall likelihood of the utterance low.

Thus, low prior probability words are less likely to be recognized than the high probability ones. Even worse, when a particular word is not in the lexicon of the machine – so called out-of-vocabulary word (OOV). Its prior probability is by definition:

$$P(M(w^{OOV})) = 0$$

and such a word is **never** recognized. In this case, the unexpected OOV is usually substituted in the utterance by at least one but more typically several acoustically similar shorter higher prior probability words. Thus, each OOV can cause more than word error. The observed rule-of-thumb is that each OOV causes 2-3 word errors. Still, the OOVs are typically rare, their impact on the overall word error rate (WER) is still quite limited.

However, the situation changes when the goal is information extraction from speech. As discussed above, unexpected words typically carry more information than the entirely predictable words. Therefore, substituting the OOV by another words might have disastrous consequences.

[1] Remember, after asking your boss for a pay-rise a number of time and always heard categorical "no", she suddenly says "yes". Sure you do want to understand this "yes"!

We argue that the way this prior knowledge is applied appears to be inconsistent with data on human language processing. We suggest an alternative architecture of ASR that may help in alleviating one of the most obvious shortcomings of the current technology, the dealing with unexpected OOVs.

1.2 Processing of unexpected events in nature

There is enough evidence that unexpected events in nature receive different (typically more) attention than the expected ones. After all, their occurrence may represent a new opportunity or a new danger.

A relatively well known is phenomenon of mismatch negativity [2] that refers to a negative peak in brain event-related potential observed in human EEG signal about 150-200 ms after the onset of out-of-order stimulus in the train of repetitive auditory stimuli.

The P300 positive peak in the EEG activity occurs about 300 ms after the presentation of unexpected stimulus in most sensory modalities — more unexpected the stimulus, larger is the peak.

The N400 negative peak occurring about 400 ms after the presentation of unexpected stimulus that is hard to integrate semantically. It is most obvious in presentation of incongruous words in sentences and it is not limited only to the auditory modality (e.g. the picture of incongruous animal in the acoustically presented sentence may also trigger the N400).

1.3 Unexpected words

Two interesting works, one in physiology and one in psycho-physics of human speech communication, both related to human use of prior information in speech communication, have been discussed by J.B. Allen in his recent book [3] and are reviewed below.

1.3.1 Physiological evidence for the local application of the prior knowledge (context) in human recognition of speech

It appears that there is a plethora of physiological evidence for special processing of unexpected sensory stimuli. The N400 appears to be most closely related to high-level processes involved in processing of information in spoken language. The work of van Petten et al. [4] deals with the issue of timing of the triggering of the N400 event by incongruous words that differ from the expected congruous word (e.g. "dollars" in the sentence "Pay with ?") either at their beginning (e.g. "scholars") or or at their ends (e.g."dolphins"). By a careful experimental design where the instant of recognition of the individual words has been first established, they are able to demonstrate that the rhyming words ("scholars") trigger the N400 **earlier** than do the incongruous words with the correct first syllable ("dolphins") . This observation supports the notion of instantaneous integration of both the top-down prior context information and the bottom-up acoustic information.

However, van Petten et al. data are inconsistent with the way the prior information is integrated in the current ASR. As seen from Eq. 1.1, the prior $P(M(w))$ is invoked **globally** during the search for the best match of the whole utterance while the van Petten et al. data indicate that in human speech recognition, the prior knowledge is applied **locally** at the instant of recognition of every word.

1.3.2 Psycho-physical evidence for the parallel prior knowledge (context) channel in human recognition of speech

Information about a word to be recognized is in bottom-up sensory data (acoustics) and on the top-down prior knowledge (context). Following [5], the error e of recognition in context is given by

$$e = e_a e_c,$$

where $e_a = (1 - p_a)$ represents the error of the acoustic channel and $e_c = (1 - p_c)$ represents the error of the hypothetical "context" channel that contributes to the correct word recognition with the probability p_c.

Further, since the context words and the target word are being perceived under the same degradation (noise, etc.), the e_a and e are related. Let's assume that the relation is $e = e_a^k$, where $k > 1$ to account for the fact that the error in context e is less that the error e_a from the acoustics alone.

This is supported by results of Miller et al. [6] who ran an experiment where they presented words from the closed set, first in the isolation and then forming the grammatically correct sentences, and evaluated accuracy of their recognition by human subjects in the increasing levels of noise. Accuracy of the recognition obviously decreased with the decreasing signal-to-noise ratio. Further, as expected, the accuracy decreased slower for the words in the sentences than for the words presented in isolation.

As shown in [7], data for errors in recognition of the out-of-context isolated words and of the in-context words in sentences, when plotted on the log-log scale, indeed lay on two lines with the common origin of close to 100% error for the very noisy data but with different slopes. That means the in-context e is related to the out-of-context e_a through $e = e_a^k$, where

$$k = \frac{\log e_a}{\log e} > 1.$$

1.4 A short divertissimo: The multiplication of error

The multiplication of error rule has important implications for the implied architecture of the human speech processing system. It has been experimentally observed in early experiments in band-limited speech by Fletcher and his colleagues (see [3] for a review) and the fact it has been also observed for the acoustic and the context channel is suggesting its universal applicability in processing of sensory information. As discussed below (following loosely the speculations of R. Galt on p. 148-151 of his notes (# 38138) [refer to CD] about the statistically independent frequency channels), it suggests the existence of parallel statistically independent processing channels. This means that a reliable recognition (i.e. a small error) in any of these channels yields a small final recognition error. This, in its turn, is the property that is highly desirable and that makes a lot of sense in information processing in biological systems.

The "multiplication of errors" rule can be derived as follows: The compound probability of the correct recognition is given by a sum of recognition probabilities under three mutually exclusive conditions:

1. Both the acoustics and the context are correct, the context channel supporting the acoustic evidence. Under the assumption of a conditional independence of these two information channels, the probability of correct recognition is given by a product of probabilities $p_1 = p_a p_c$.

2. The acoustic is correct but the context is in error (e.g. unexpected word). Then the probability of correct recognition is $p_2 = p_a(1 - p_c)$.

3. The acoustic is in error (e.g. noise) but the context is correct i.e. the probability of correct recognition is $p_3 = (1 - p_a)p_c$.

This speculation tacitly assumes that the system does not hallucinate correct recognition when both channels are in error.

In other words, the words will be correctly recognized in three situations:

1. $p_1 = p_a p_c$ words that are supported by both the acoustics and the context. These words will be correctly recognized.

2. The context will not support $p_2 = p_a(1 - p_c)$ words that are suggested by the acoustics. These words will be correctly recognized because of the acoustic evidence.

3. Out of the p_c words that makes sense due to the context, $p_3 = p_c(1 - p_a)$ will not be supported by the acoustics. These words will be correctly recognized because of their context.

We do not have to worry about $p_4 = (1 - p_a)(1 - p_c)$ words which are supported neither by the acoustics nor by the context since they will never be correctly recognized.

Since these three events are mutually exclusive, the final probability of getting the word right is given by summation of the individual event probabilities

$$p = p_a p_c + p_a(1 - p_c) + (1 - p_a)p_c$$

and the error of recognition is

$$e = 1 - p = 1 - p_a p_c - p_a(1 - p_c) - (1 - p_a)p_c = (1 - p_a)(1 - p_c) = e_a e_c.$$

Thus the data of Miller et al. and Boothroyd et al. support the existence of two statistically independent acoustic and context channels.

So there appear to be two independent channels, one bottom-up acoustic channel and one top-down context channel. When an expected word comes and the acoustics is reliable, both channels indicate the same word. When the word is unexpected (e.g. out-of-vocabulary) or the acoustics is unreliable, the channels may disagree, each indicating different word. Still, the words will be recognized as long as the evidence in either the acoustic or the context channel is strong enough.

The existence of parallel processing channels in decoding linguistic information in speech is not a minor detail. The parallel processing is the universal way of increasing reliability of any system. It allows for compensation of partial failures of some of the parallel channels by providing supporting evidence from the other, more reliable, channels. The fact that the parallel context-specific and acoustic-specific channels are supported by results of the perceptual experiments described above suggest necessary major modifications of existing machine recognition architectures.

1.5 Here is the Idea

Suppose that, in addition to recognizing the word from the information in the two parallel information channels as described above, we also find out which word would be suggested from the acoustics alone:

- In the first case when the acoustic supports the context, the recognized word is identical to the word suggested by the acoustics.
$$1 - e = p \approx p_a.$$

- In the second and the third case, the recognized word may be different from the word suggested by the acoustics.
$$p \neq p_a$$
  - The second case would arise when the context is overriding the acoustics (but the acoustics is reliable).
  - The third case would arise when the acoustics is unreliable and context is supplying the correct word.

- To differentiate between the second and the third cases, we need to know the reliability of the acoustic channel.

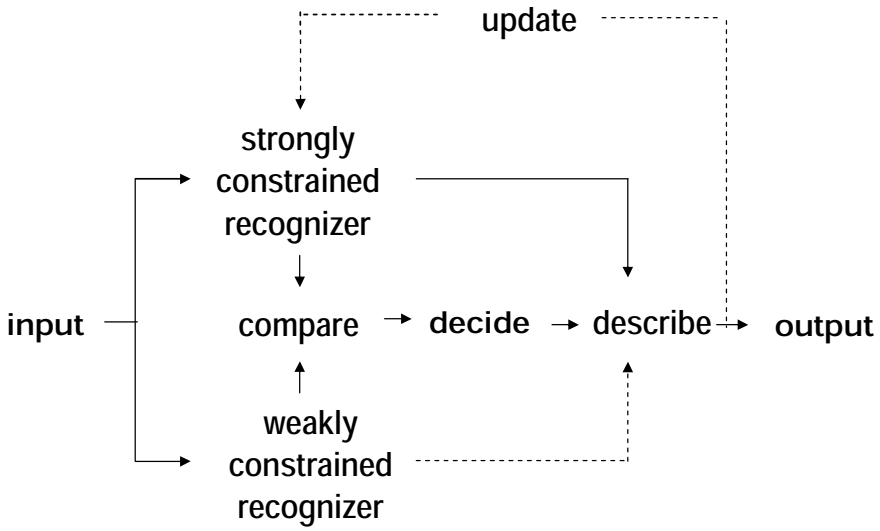

Figure 1.1: The block diagram of a system that could be used for detection and description of unexpected out-of-vocabulary words

Thus, to build the artificial system that would be capable of indicating which of the three situations occurred requires:

1. recognizing word from both the acoustics and the context,
2. recognizing the word only from the acoustics,
3. comparing the results from both recognition streams,
4. estimating reliability of recognition results in both parallel streams,
5. means for interpreting the indicated OOV it in terms of its parts (phonemes) that would allow for its description as well as for updating the lexicon.

The block diagram of the system is illustrated in Fig. 1.1.

1.6 WHAZWRONG Group at the 2007 JHU Workshop – a summary

We have assembled a group of researchers that all agreed on the necessity of a progress towards dealing with unexpected acoustic inputs in ASR to initiate the work on the system proposed above. The chosen researchers formed a group called Recovery from Model Inconsistency in Multilingual Speech Recognition (informally WHAZWRONG?)[2]. At the 2007 JHU summer workshop, the group concentrated on the detection of OOV segments in the output of large vocabulary continuous speech recognition (LVCSR) system on Wall Street Journal (WSJ).

All aspects of the system have been studied, with most effort devoted to

1. recognizing speech with a minimal use of the context,
2. comparing estimates from the "strongly-constrained" (i.e. both the acoustics ad the context constrained) and the "weakly-constrained" (mostly acoustics) recognition streams

[2] `http://www.clsp.jhu.edu/ws2007/groups/rmimsr/`

3. the confidence measures,

4. phoneme recognition without use of any context.

The task we have chosen was the detection of errors in ASR induced by the existence of OOVs in the data. The data material consisted of WSJ data, down-sampled to 8 kHz to make our research more applicable to recognition of telephone-quality speech, with about 20% of least frequent words left out from the lexicon, thus emulating the targeted OOVs. In addition to the test data-set, a development set (used for a training of some of data-guided techniques) has been also created.

A state-of-the-art LVCSR recognizer, adopted from the LVCSR system developed in the AMI(DA) EC project [3] has been used as the strongly-constrained recognizer. As the main weakly-constrained recognizer, we have used the same LVCSR system modified for recognition of phonemes (rather than words). Both recognizers were trained on the independent telephone-quality data and not on the targeted WSJ data. Additionally, the strong constrains were also induced on the recognized phoneme string by a transducer-based system from Microsoft Research.

The compared parameters, representing results of the strongly-constrained and the weakly-constrained recognizers were posterior probabilities of phonemes. A number of comparison techniques have been investigated in addition to many state-of-the-art confidence measures derived from both the strongly-constrained and the weakly-constrained recognition streams. In the final system, results from most of the investigated techniques have been fused using a state-of-the-art classifier from Microsoft Research.

A significant research efforts were also dedicated to advancements in phoneme recognition. The Universal Phoneme Model (UPM) recognizer from DoD has been investigated in addition to a Mandarin phoneme recognizer from a Singapore group, resulting in important improvements from adaptation of these recognizers to the targeted data.

The progress has been evaluated by comparing the developed error-detection techniques to the state-of-the-art C_{max} technique [8]. Compared to these baseline techniques, we have reached significant improvement in both detection of OOVs and detection of plain recognition errors (see Figures 3.7 and 3.8).

As discussed in more detail later in this report, these improvements are due to many individual advances in several aspect of the complete system. Namely:

- Burget devoted most of his efforts towards appropriate confidence measures,
- Schwarz worked on weakly-constrained recognizer,
- Matejka, Rastrow and Khudanpur studied comparison techniques,
- White and Zweig were developing alternative system of strongly and weakly constrained recognizers based on phoneme recognition and a transducer,
- Nedel, Lee, Le, Isacoff and Marco worked on improvements of phoneme recognition,
- Hannemann and Sahani worked on system and database aspects,
- ... and Hermansky acted as moderately dogmatic cheerleader and coordinated the efforts.

The individual efforts are described in more detail in the following chapters.

[3] the main modifications consisted of the exclusive use of the NN-based TANDEM features and of some simplifications of the AMI(DA) multi-stage processing

Chapter 2

Task and data

2.1 The task

Although the primary task is the detection of OOV, we actually wanted to build a system, which solves the following two tasks:

1. Word Error Detection task: detect words in LVCSR output which are wrong.
2. OOV detection task: detect words in LVCSR output, which are wrong recognized and do overlap with OOV words.

We performed a recognition using a lexicon (see the following section), that was limited to the most frequent 4968 words from the language model training texts, and thus it was defined what will be treated as OOV.

2.1.1 Word-based labeling of OOVs

For each word in the output of the recognizer, we had to determine whether it was mis-recognized. Therefore we performed a time-mediated alignment (see for example the SCTK/sclite toolkit [1]) of reference and recognition output words - this should result in more precise alignments than using the standard Levenshtein algorithm - and each substituted or inserted word was marked as incorrect. These labels correspond to the Word Error Detection task (upper panel of Figure 2.1). Silences between words were also detected and labeled differently.

In a second step, we determined for each mis-recognized word, whether it shares time-frames with an OOV word in the reference. If this was the case, we labeled it as a target for our OOV detection task (lower panel of Figure 2.1).

2.1.2 Frame-based labeling of OOVs

For several tasks, we also needed sub-word labels, on a frame by frame level. Depending on the task, the definition of an "incorrect"/"OOV" frame is different. The first possibility is declaring each frame belonging to a word labeled as overlapping with OOV as being an "OOV frame", in the same way as in Figure 2.1. These labels could serve as targets for word level classificators using frame level input (see also lower panel of Figure 3.4 – red means mismatch/ OOV frame, gray means correct). Silence was in all levels often labeled with a separate tag.

Even in an OOV, most of the phones (70%) are correctly recognized since the OOV is replaced by the in-vocabulary word, which is phonetically closest to the OOV. Keeping that in mind, we could label only those frames as "OOV", which are inside a word which is overlapping with an OOV and which

[1] `ftp://jaguar.ncsl.nist.gov/current_docs/sctk/doc/sclite.htm`

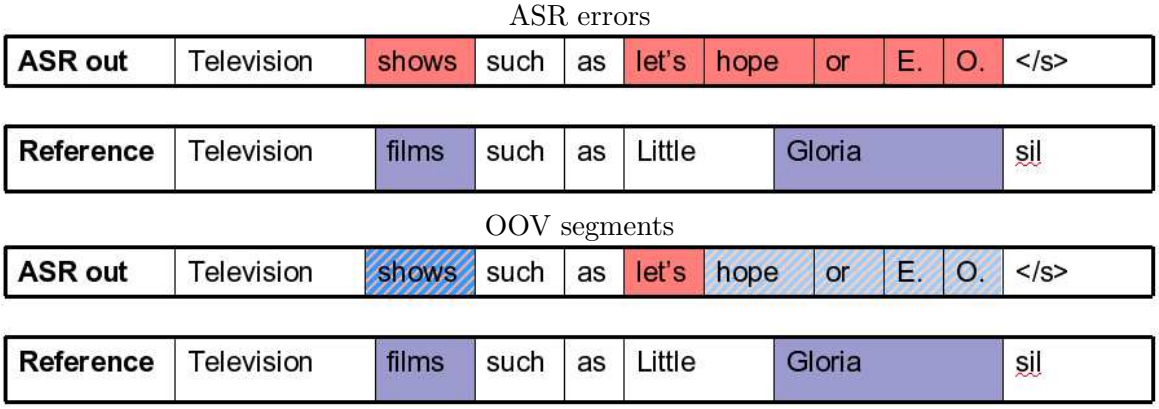

Figure 2.1: Recognizer's errors and OOV segments: LVCSR vocabulary does not contain the words "films" and "Gloria". While all the words with colored background in the upper pannel are considered recognition errors, only "hope or E. O." are overlapping with an OOV and should be tagged as OOV segment.

belong to a phone, which does not match the reference, using the boundaries from the recognition. These labels could serve for examining the properties of phone patterns, which are not correctly recognized/ caused by OOVs. This is depicted in the depicted in the middle panel of Figure 3.4.

Also within matching phones, not all frames match, due to changing phoneme boundaries. So we could label according to a frame-by-frame comparison, and could observe properties of generally mismatching frames (upper panel of panel of Figure 3.4).

A known weakness of most of the labling schemes described here is that there is no way of marking deletions. Since the LVCSR output obviously doesn't have them, we did not assign them to output words - so we will neither evaluate for deletion errors, nor train detectors on deletions.

Also, since in general, we discard silence segments, we will not detect/ label cases where something was substituted by silence. Fortunately at least at the word level in sentences containing OOV, deletion errors are the least frequent ones.

2.1.3 Evaluation measures

To evaluate a new confidence measure / detector, we obtain a score for each LVCSR output word in a test set, and then set a threshold to determine, whether it should be marked as detected or not. A false alarm is when a word is detected, which was labeled as "correct" and a miss is a word which was not detected, but was labeled as mis-recognized/ overlapping with OOV.

We trade off false alarm and miss probabilities by using multiple thresholds and we show them in standard detection error trade-off (DET) curves, which are used to compare several confidence measures. We also compute some one-number metrics such as Equal-Error-Rate (EER) or Confidence-Error-Rate (CER), but since they are dependent on the ratio of correct targets to overall number of tokens, we leave the choice of the operating point open by reporting the whole DET curve.

2.2 Data

2.2.1 Choice of Wall Street Journal

Based on the good results on a small vocabulary task [10] we wanted to extend our approach to a system for large vocabulary continuous speech recognition (LVCSR).

We have chosen to work with the Wall Street Journal Corpus (WSJ), because it is well known, we had parts of the setups available and we wanted to have high quality speech to be able to concentrate on the effects and mismatches produced by out-of-vocabulary words (OOV) rather than additionally dealing with effects arising from spontaneos, noisy speech.

We wanted to have a controlled experiment, where we "make a healthy animal sick" by introducing OOV by limiting the recognition dictionary to the most frequent words, and thus having "naturally" occuring OOVs.

2.2.2 Evaluation set

To evaluate the performance of our system, we defined an evaluation set, which is composed of two parts:

- the already mentioned November 92 Hub2 test set: 330 utterances from WSJ0, `si_et_05`, which have no OOVs (closed vocabulary)
- and the 5k development test set: 913 utterances from WSJ1, `si_dt_05` which is defined as open vocabulary.

The November 92 Hub2 was internally called "test1" and the 5k development test set was split into two portions, one containing only sentences covered by the vocabulary (internally "test2": 537 utterances) and the other containing all sentences with OOVs ("test3": 376 utterances).

These three subsets were so far always used together, that means all results refer to "test1+test2+test3". They are only used for organization purposes and to get insights by computing separate statistics.

The second part of the evaluation set (test2+test3) has some properties which are interesting for analysis: these texts were collected one year after the language model training, which as we later see results in a little worse recognition results (without OOVs) and a high number of OOVs.

Many of the utterances (400, 360 of them in test3) are utterances, which have been repeated by 10 different speakers and have the same words. These can be analyzed to compare the different recognition results caused by the same OOV across the speakers.

Using the sub-partitions, we can analyze mis-recognitions not caused by OOV separately from those caused by OOV, and we can test our system with several different OOV token rates.

Table 2.1 shows all kinds of statistics about the evaluation set. Token counts were obtained from time mediated alignment not counting pause tags. Substitution/Insertion/Deletion errors were counted using standard Levenshtein algorithm.

The whole evaluation set consists of 1243 utterances (2.49 hours), after decoding with the limited recognition dictionary, it has an OOV token rate of 4.95% in the reference, and in the LVCSR output we had 13.83% tokens marked as mis-recognized, and within 8.51% OOV tokens (recognized words overlapping with OOV words in the reference).

We see, that the word error rate (WER) drops dramatically with the introduction of OOVs in test3: (42.13%), which is mainly caused by substitution errors, but also but a lot of insertion errors, and we also see that the rule of thumb - one OOV causes two word errors - seems to be true.

We have 1665 tokens marked as being overlapping with an OOV, which are caused by 952 occurences in the reference, but which itself only contain 104 different OOVs - that means in average every OOV is happening ten times - this is due to the properties of the development set described above.

2.2.3 Development set

To compare the outputs of the strongly and weakly constrained recognizers, or in general to compute a confidence measure, we can directly compute a measure, or we can train a classifier/ score estimator, using the Maximum Entropy framework or the neural nets for frame-by-frame scores.

	Evaluation Set			
source corpus	**WSJ0** si_et_05	**WSJ1** si_dt_05		**Test1+ Test2+**
name	**Test1**	**Test2**	**Test3**	**Test3**
number of utterances	330	537	376	1243
number of audio seconds	2411	4112	2445	8969
different words in reference	1270	1857	454	2543
different words in ref., in dictionary	1270	1857	350	2439
different words in ref., OOV	0	0	104	104
number of tokens in reference	5353	8805	5061	19219
number of tokens in reference, OOV	**0**	**0**	**952**	**952**
average reference token per utterance	16.22	16.40	13.46	15.46
number of tokens from LVCSR	5344	8704	5508	19556
number of tokens LVCSR, overlap OOV	0	0	1665	1665
number of tokens LVCSR, misrecogn.	207	459	2039	2705
OOV-rate word lists	0%	0%	22.91%	4.09%
OOV-rate reference token	**0%**	**0%**	**18.81%**	**4.95%**
OOV-rate LVCSR token	**0%**	**0%**	**30.23%**	**8.51%**
mis-recognized-rate LVCSR token	**3.87%**	**5.27%**	**37.02%**	**13.83%**
LVCSR, Substitution Error	183(3.42)	403(4.58)	1473(29.10)	2059(10.71)
LVCSR, Deletion Error	32(0.60)	150(1.70)	106(2.09)	288(1.50)
LVCSR, Insertion Error	23(0.43)	49(0.56)	553(10.93)	625(3.25)
LVCSR, Word Error Rate	**4.45**	**6.84**	**42.13**	**15.46**

Table 2.1: Properties of the evaluation set.

For that purpose we defined a development set, which provides recognition examples to train classifiers on Word Error and OOV detection task.

We used all utterances from WSJ0 `si_tr_s`, which belong to the "c" set (fourth letter of name, because of non-verbalized punctuation), and which did not contain any disfluencies or noises. The term "training" in this setting does not refer to acoustic training since our acoustic models were already trained on Switchboard data, and were not trained on WSJ at all. Overall, the development set covers 4088 sentences and 7.73 hours of speech, which were also labeled according to sections 2.1.1 and 2.1.2.

Similar to the evaluation set, internally the development set was also split into three portions: 963 utterances containing no OOVs ("devtrain1"), 1963 utterances containing medium level OOV ("devtrain2") and 1162 utterances containing "a lot of" OOVs ("devtrain3"). The split between "devtrain2" and "devtrain3" was performed randomly, but giving a higher prior for "devtrain3" the more OOVs the utterance contained.

Again, these three subsets were so far always used together, that means all classifier training was done on "devtrain 1+2+3", but for organization purposes and to be able to select subsets with varying OOV token rates, they are reported separately here.

Table 2.2 shows all kinds of statistics about the development set. First we can observe again the extreme increase in WER with rising OOV content, and again the thumb rule one OOV, two word errors holds, and the errors are substitutions and in second place also insertions of short words to form acoustics close to the OOV. On the whole development set, about $\frac{4}{5}$ of the mis-recognized tokens really overlap with OOV words.

Since there are no repeated sentences, we can also observe Zipf's Law (-¿ "devtrain3"): if we have

	Development Set			
source corpus	**WSJ0** si_tr_s/???c????			**DevTrain1+ DevTrain2+**
name	**DevTrain1**	**DevTrain2**	**DevTrain3**	**DevTrain3**
number of utterances	963	1963	1162	4088
number of audio seconds	5725	13112	8985	27821
different words in reference	1765	4899	4728	6685
different words in ref., in dictionary	1765	2766	2381	3410
different words in ref., OOV	0	2133	2347	3275
number of tokens in reference	14482	32712	21969	69163
number of tokens in reference, OOV	**0**	**3652**	**4350**	**8002**
average reference token per utterance	15.04	16.66	18.91	16.92
number of tokens from LVCSR	14388	34450	24080	72918
number of tokens LVCSR, overlap OOV	0	6834	8153	14987
number of tokens LVCSR, misrecogn.	626	8370	8874	17870
OOV-rate word lists	0%	43.54%	49.64%	48.99%
OOV-rate reference token	**0%**	**11.16%**	**19.80%**	**11.57%**
OOV-rate LVCSR token	**0%**	**19.84%**	**33.86%**	**20.55%**
mis-recognized-rate LVCSR token	**4.35%**	**24.30%**	**36.85%**	**24.51%**
LVCSR, Substitution Error	559(3.86)	6044(18.48)	6314(28.74)	12917(18.68)
LVCSR, Deletion Error	153(1.06)	529(1.62)	395(1.80)	1077(1.56)
LVCSR, Insertion Error	61(0.42)	2273(6.95)	2509(11.42)	4843(7.00)
LVCSR, Word Error Rate	**5.34**	**27.04**	**41.96**	**27.24**

Table 2.2: Properties of the development set.

19.8% OOV tokens in the reference, we still have only covered half of the total words (49.64% OOV) - which means the OOV words are really rare.

Interesting is also the direct correlation of the sentence length with the number of OOV tokens in the reference - an indication of more complex sentences.

2.2.4 The good, the bad and the silence

Table 2.3 shows the distribution of frames (speech segments of 10 ms) for the different sets. The frames were labeled according to middle panel of Figure 3.4: phoneme level comparison frame-by-frame, taking phoneme boundaries from the LVCSR output.

We can compute, that 4.18% of the speech (non-silence) frames in the evaluation set and 7.70% of the speech frames in the development set were labeled as mismatching, while from table 5 and 6 we see, that 13.85% of the tokens from evaluation and 24.51% of the tokens from development were labeled as mis-recognized on the word level. We see that only around 30% of the phones in an OOV (mis-recognized word) are actually misrecognized.

Dividing the number of non-silence frames by the number of tokens, we obtain an average word length of 335 ms stable for all subsets.

Frame sets	silence	correct	incorrect	total	time
test1	58549(24.28%)	180585(74.89%)	1991(0.83%)	241125	0.67 h
test2	118946(28.92%)	287940(70.02%)	4349(1.06%)	411235	1.14 h
test3	75350(30.82%)	148567(60.77%)	20574(8.42%)	244491	0.68 h
$\sum$ *evaluation*	*252845(28.19%)*	*617092(68.81%)*	*26914(3.00%)*	*896851*	*2.49 h*
devtrain1	98687(17.24%)	468213(81.79%)	5572(0.97%)	572472	1.59 h
devtrain2	221536(16.90%)	1005039(76.65%)	84605(6.45%)	1311180	3.64 h
devtrain3	147345(16.40%)	663118(73.81%)	87993(9.79%)	898456	2.50 h
$\sum$ *development*	*467568(16.81%)*	*2136370(76.79%)*	*178170(6.40%)*	*2782108*	*7.73 h*

Table 2.3: Frame distributions for evaluation and development sets.

2.3 Data sources

Wall Street Journal corpus is supplied by the Linguistic Data Consortium (LDC) [2] under the catalog numbers: WSJ0: LDC93S6A and WSJ1: LDC94S13A.

2.3.1 Acoustic data and transcriptions

WSJ0, `si_tr_s` can be found on disks 01-03, disk 04 holds the transcriptions, the November 92 evaluation test set (`si_et_05`) can be found on disk 14. The WSJ1 development test set can be found on WSJ1 disk 16. All audio files were recorded using the Sennheiser microphone (indicated by the ending .wv1) and had be converted because they are saved in "shorten" compression.

2.3.2 Language model data

We used a standard bigram backoff (Katz) language model, using open vocabulary and non-verbalized punctuation - it is provided with the WSJ0 distribution, on disk 13 (file `bcb05onp.z`), and it was computed on text material from Wall Street Journal which was collected in the years 1987-89.

2.3.3 Graphemes to phonemes

The phonetic transcriptions were partly taken from the dictionary of the AMI NIST RT meeting recognizer [15], which itself uses UNISYN [3] dictionary, which was augmented by automatically generated pronunciations by the Festival system and in some parts corrected by hand.

To cover also all OOVs for reference alignments we also created a reference dictionary, where we had to hand-craft some of the phonetic transcriptions. We decided on a phone set and mapping which has 45 TIMIT-style phonemes including silence.

We build our reference transcriptions by discarding all files with noises and disfluencies, then all punctuations were removed, all words were converted to upper case, we had mappings for common spelling errors, and numbers and acronyms were spelled out in a normalized way.

Since our original dictionary was in British English, we also had to perform a mapping between British and American English and vice versa, which can be done in the following steps:

- First we mapped multiple spellings which were all valid in both dialects to a unique form, then we performed British-American mappings using a dictionary and keeping a special dictionary for ambiguos mappings, which have different words with different meanings in one dialect, but only one word in the other.

[2] `http://www.ldc.upenn.edu`

[3] The Centre for Speech Technology Research, University of Edinburgh, `http://www.cstr.ed.ac.uk/projects/unisyn/`

Chapter 3

Combination of strongly and weakly constrained recognizers for reliable detection of OOV

3.1 Basis

3.1.1 Confidence measures

Confidence measures (CM) [8] are being routinely used for detecting incorrectly recognized words. We start with them also for the detection of OOVs. Under the assumption that OOVs should be easier errors to detect, CM should start detecting those before ordinary mis-recognitions. We are actually comparing our results to the C_{max} measure computed from word lattices, that was evaluated as the best performing in [8]. In this work, the use of frame-based word- and phone- posteriors is investigated. Frame-based posteriors have been used as CM too, for example in [9], they served to estimate confidence of words from a hybrid system.

3.1.2 Strongly and weakly constrained systems

By comparing posteriors from *two* systems: *strongly constrained* (word-based, with language model) and *weakly constrained* (only phones) (Fig. 1.1), we however aim at not only detecting where the recognizer is not sure (which is the task for confidence estimation) but also to detect places where the recognizer is sure about wrong thing.

We are motivated by the visualization in Fig. 3.1. The mismatch in LVCSR-posteriors and posteriors generated by a weakly constrained system has a chance to reveal the OOV, although the LVCSR itself is quite sure of its output. Preliminary work in this direction was done by Ketabdar and Hermansky [10], the results were however obtained only on a small connected-digit recognition task.

3.2 Posteriors

All posteriors used in our work are **frame-based** and are denoted $p(u|t)$, where u is the respective unit (word, phone) and t is time in frames.

3.2.1 Posteriors from strongly constrained system

LVCSR output is represented as recognition lattice with arcs representing hypothesized words w_i^j, where w_i is the word identity and j is the occurrence of word w_i in the lattice. Each w_i^j spans certain time interval and has associated acoustic and LM scores. Note that occurrences of several w_i^j for the

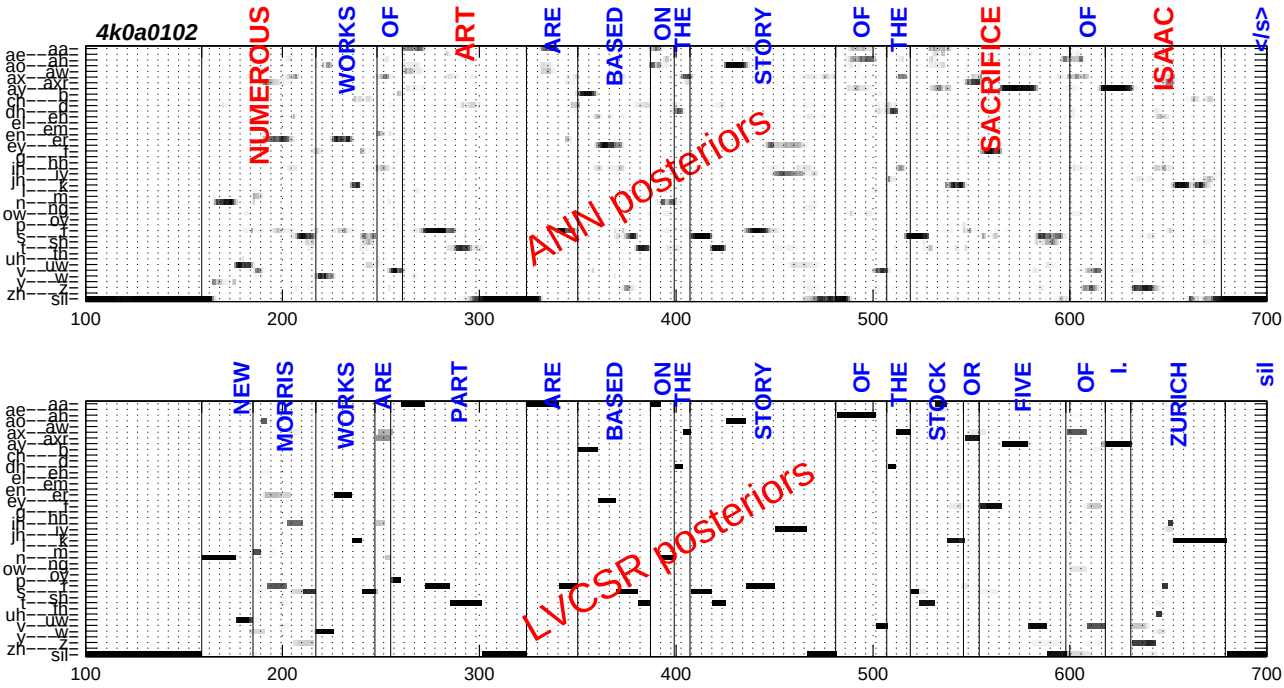

Figure 3.1: Comparison of ANN posteriors and LVCSR posteriors. The reference transcription is: NUMEROUS WORKS OF ART ARE BASED ON THE STORY OF THE SACRIFICE OF ISAAC, words "sacrifice" and "Isaac" are OOVs. Note the mismatch in the posteriors in places of OOVs.

same word w_i can overlap in time. Lattice arc posteriors $p(w_i^j)$ are estimated from the lattice by standard forward-backward algorithm.

Frame-based word-posterior $p(w_i|t)$ (left panel of Figure 3.2) is given by summing all $p(w_i^j)$ active at the given time t. *Word entropy* (right panel of Figure 3.2) for time t is estimated as:

$$H(t) = -\sum_i p(w_i|t) \log_2 p(w_i|t),$$

and, in the case of C_{max} confidence measure, the confidence of hypothesized word w_i spanning time (t_s, t_e) is

$$C_{max}(w_i, t_s, t_e) = \max_{t \in (t_s, t_e)} p(w_i|t).$$

The second set of posteriors from strongly constrained system are *LVCSR-phone posteriors*. In our decoder, phones are parts of recognition lattices [11]. It is straightforward to run the forward-backward algorithm on the level of phones and obtain $p(g_i^j)$, where g_i^j denotes jth occurrence of ith phone from the alphabet. Note that there is still a possibility to have concurrent hypothesis of the same phone at the same time. Similarly to words, fame-based phone-posterior $p(g_i|t)$ is given by summing all $p(g_i^j)$ active at the given time t.

3.2.2 Posteriors form weakly constrained system

First set of "weak" posteriors was obtained from a system having the same front-end and acoustic models as LVCSR, but with phones populating the vocabulary and a simple bigram phonotactic model. The resulting phone lattices were processed in the same way as above. We will call these *Phone recognizer posteriors*.

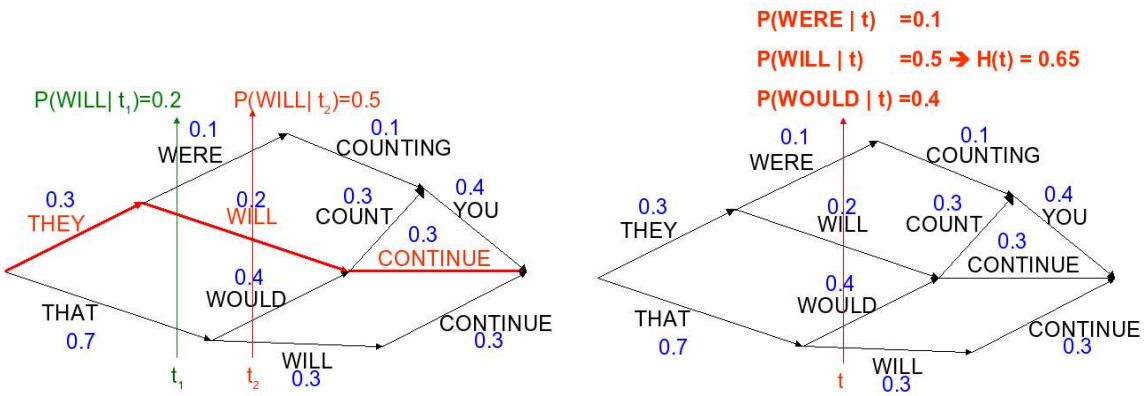

Figure 3.2: Estimation of word posteriors and word entropy from recognition lattice. The numbers in the lattice represent word posteriors $p(w_i^j)$ obtain by the forward-backward algorithm.

The second set of "weak" posteriors is generated by phone posterior estimator based on a neural net (NN). The NN contains the soft-max non-linearity in the output layer, so that its outputs can be directly considered as frame-based posteriors. These will be denoted *NN phone posteriors*.

Weak posteriors of any kind will be further denoted $p(f_i|t)$. Note that phone recognizer posteriors will be more "focused" because of use of 3-state HMMs and phonotactic LM.

3.3 Comparison of posteriors

To compare posteriors from strong and weak systems, we have investigated the following three approaches:

1. **Posteriors of the weak system** were found for the phones hypothesized by the strong system.

2. **Kullback-Leibler (KL) divergence** between the posteriors from the strong and weak system was evaluated. The classical formula:

$$KL(t) = \sum_i p(g_i|t) \log \frac{p(g_i|t)}{p(f_i|t)}$$

   was not sufficient and some engineering was needed. First, some posteriors (especially from LVCSR) tend to have zero values, so that thresholding is necessary. Second, there is a temporal alignment problem between the phones generated by the strong and weak systems. We solved it by a soft-alignment: first, for time t, the strongest phone posterior from LVCSR was detected: $s^\star = \arg\max_i p(g_i|t)$. A context of $2N+1$ frames ($t_1 = t-N, t_2 = t+N$) from the weak system was taken and a weighting corresponding to the posterior of $s^\star$ in its output was applied:

$$KL_{avg}(t) = \frac{\sum_{t' \in (t_1,t_2)} p(f_{s^\star}|t') \sum_i p(g_i|t) \log \frac{p(g_i|t)}{p(f_i|t')}}{\sum_{t' \in (t_1,t_2)} p(f_{s^\star}|t')}$$

3. The third and most successful approach relied directly on the estimated posteriors. A **neural network** was trained to combine posterior vectors from strong and weak systems and come up with the final decision (e.g. in OOV word or not).

3.3.1 Post-processing of frame-based values into scores

To convert the described frame-based CM to word-scores, several techniques were investigated:

- maximum over hypothesized word boundary,
- variance over hypothesized word boundary,
- average over hypothesized word boundary,
- and averaging over hypothesized phonemes normalized by the number of phonemes.

We also experimented with arithmetic vs. geometric averages.

Averaging over hypothesized phonemes normalized by the number of phonemes worked well for most of the measures described above. But for example, for KL divergences, variance over hypothesized word boundary worked the best. For the following combination, we have selected few well performing post-processing methods for each frame-based CM.

3.3.2 Combination of word scores

The combinations of word-scores generated by the individual techniques were post-processed by conditional models trained using the maximum entropy (MaxEnt) criterion [12]. Conditional maximum entropy models were chosen based on their history of good performance for speech and language related tasks including language modeling, [13], parsing, [14], etc.

Our model is of the form $p(y|\mathbf{x})$. Here, y is a discrete random variable representing the class 'correct' or 'incorrect', and $\mathbf{x}$ is a vector of discrete or continuous random variables – the word confidence scores. In the conditional MaxEnt framework, the model interacts with the random variables $\mathbf{x}$ and y through a vector of feature functions $f_i(\mathbf{x}, y)$ and parameters λ_i.

$$p_\Lambda(y|\mathbf{x}) = \frac{\exp\left(\sum_{i=1}^{F} \lambda_i f_i(\mathbf{x}, y)\right)}{\sum_{y'} \exp\left(\sum_{i=1}^{F} \lambda_i f_i(\mathbf{x}, y')\right)}$$

This conditional MaxEnt model is regularized by using a Gaussian prior on the parameters λ_i.

Besides MaxEnt classifier, we have experimented also with NN- and SVM-fusing, with similar results, so that we stick with MaxEnt.

3.4 Experimental setup

3.4.1 Data

The data used in our eperiments are thoroughly described in the respective chapter (section 2.2).

3.4.2 LVCSR and NN-phone posterior estimator

The **LVCSR** was a CTS system derived from AMI[DA] LVCSR [15]. It was trained on 250 hours of Switchboard data. The decoding was done in three passes, always with a simple bigram language model:

- In the *first pass*, PLP+Δ+ΔΔ+ΔΔΔ features were used, they were processed by Heteroscedastic Linear Discriminant Analysis (HLDA), and the models were Minimum-Phone Error (MPE) trained.
- In the *second pass*, vocal-tract length normalization (VTLN) was applied on the same PLP+Δ+ΔΔ+ΔΔΔ features, HLDA and MPE were used, and in addition, constrained maximum likelihood linear regression (CMLLR) and speaker adaptive training (SAT) were used for speaker adaptation.

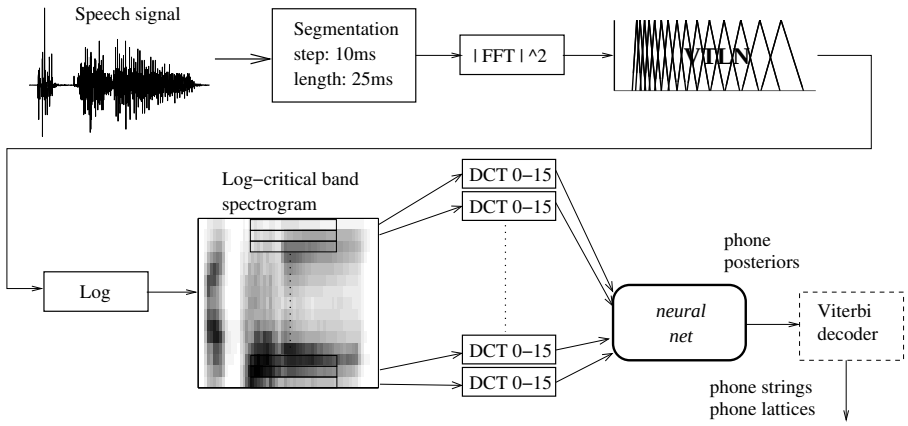

Figure 3.3: Phone posterior estimator, with optional Viterbi decoder producing phone lattices (dotted). Thanks to Franta Grzl (Speech@FIT, BUT) for providing elements to draw this figure.

- Finally, the *third pass* was the same as pass 2, but posterior-features with their deltas replaced the PLP-based features (these features were generated by the NN phone posterior estimator).

On WSJ0, Hub2 test from November 92, this system reached word error rate (WER) of 2.9% on the closed-set 5k word task.

The **NN phone-posterior estimator** (Figure 3.3) was based on NN processing long (300 ms) temporal trajectories of Mel-filter bank energies. On contrary to [16], we used a simple system with only one 3-layer NN with 500 neurons in the hidden layer. The output layer of NN represents phone-state posteriors, but these were summed for each phone to form phone-posteriors. In [16], we have shown that phone-states in the final layer of the NN always greatly improve the accuracy so that we kept this scheme also in this work.

3.4.3 Score estimators

The only parameter to set in the conditional MaxEnt model is the Gaussian prior on the parameters λ_i. The performance on the development data was insensitive to the variance of this prior, which is not surprising given the size of our training data set. As a result, it was fixed at a value of 100 for all of our experiments.

When NN was used for direct estimation of frame-based scores, the network was directly fed by posteriors from strong and weak systems. The NN was a 3-layer perceptron with 100 neurons in the hidden layer and the final layer having 3 outputs: OOV, non-OOV and silence. Different schemes of frame-labeling for NN training were devised, the best was to label all frames of an ASR word overlapping with an OOV as "OOV".

Three labelling schemes for NN training (Fig. 3.4) were devised: the first two were based on *phone* alignment between the reference and LVCSR output:

1. In *frame-based* training, only frames where the LVCSR phone label did not match the reference phone were tagged "OOV".

2. In *phone-based* approach, this label was extended the whole incorrect phone.

3. The third approach was *word-based* – whole mismatched words were tagged as OOVs.

The third technique performed the best.

A lot of improvement was obtained when temporal context was used in the NN input (see the following section).

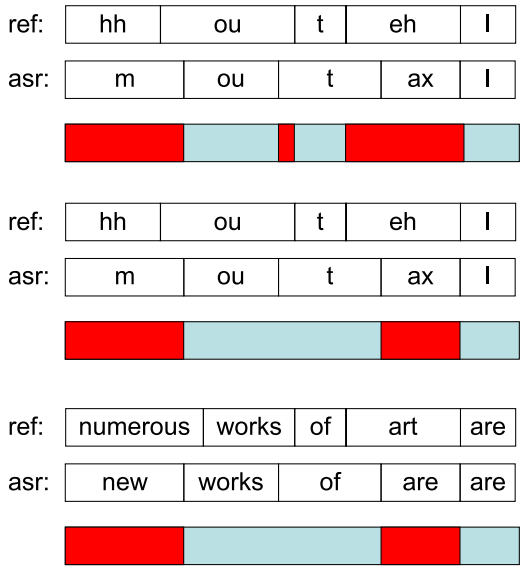

Figure 3.4: Three schemes of NN training: frame-based, phone-based and word based. Dark regions denote segments where the NN was presented with the "OOV" label in the training.

3.5 Results

The first set of DET curves in Fig. 3.5 shows the results without the use of NN for OOV detection. Average word entropy significantly outperformed standard C_{max} confidence measure and was found to be the best single score for this task (not considering NN based scores). Two remaining curves show performance obtained with MaxEnt combination of groups of features:

3.5.1 LVCSR-based features

These features include

- C_{max},
- average word posterior,
- average word entropy,
- word posterior and entropy from confusion networks [17]
- measures related to acoustic stability [8]
- lattice link entropy
- number of different active words,
- word lattice width
- acoustic, LM-score and duration measures from 1-best word string.
- average phone posterior (MPCM) [9] based on LVCSR posteriors
- and phone entropy based on lattice from LVCSR.

3.5.2 Weak features

The group of *weak features* consisted of phone entropy based on lattice from phone recognizer, phone entropy based on NN output (both weak recognizers only) and a group of features comparing LVCSR and weak:

- KL-divergence between LVCSR and NN posteriors,

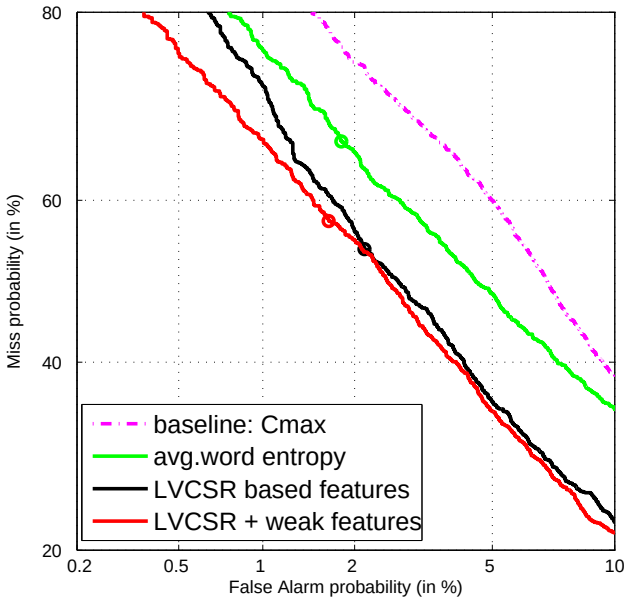

Figure 3.5: OOV detection using strong system only and combination of strong and weak systems.

- KL-divergence between LVCSR and phone recognizer posteriors,
- average phone posterior (MPCM) based on NN posteriors,
- average phone posterior (MPCM) based on phone recognizer posteriors,
- and several variations of the KL-divergence.

The weak features themselves had poor results, but they provide a nice improvement when combined with LVCSR-based features.

3.5.3 Results of the combination

The second set of results in Fig. 3.6 shows the results for the NN detecting OOVs from the combination of strong (LVCSR-phone) and weak (NN-phone) posteriors. Note that even the simplest NN-based method taking into account only 1 frame of **phone** posteriors without any context has performance comparable to above mentioned techniques based on **word** posteriors. As mentioned before, we have experimented with the different NN training schemes (Fig. 3.4), the word-based training was the best, as expected.

3.5.4 Context for the combining NN

Several experiments were done regarding the context for NN. We found that it was optimal to take the strong and weak posteriors from the current frame t, 1 frame in past: $t-6$ and 1 frame in future: $t+6$. This corresponds to sampling neighboring phonemes. The last DET curve in Fig. 3.6 shows that this is the best single technique for OOV detection.

3.5.5 Combination

Finally, MaxEnt classifier was used to fuse the results from LVCSR+weak features and NN – see Fig. 3.7. In Fig. 3.8, we present the performance of the same systems in the detection of **all** recognition errors. We see that in both tasks, the NN combined with LVCSR+weak features performs excellently.

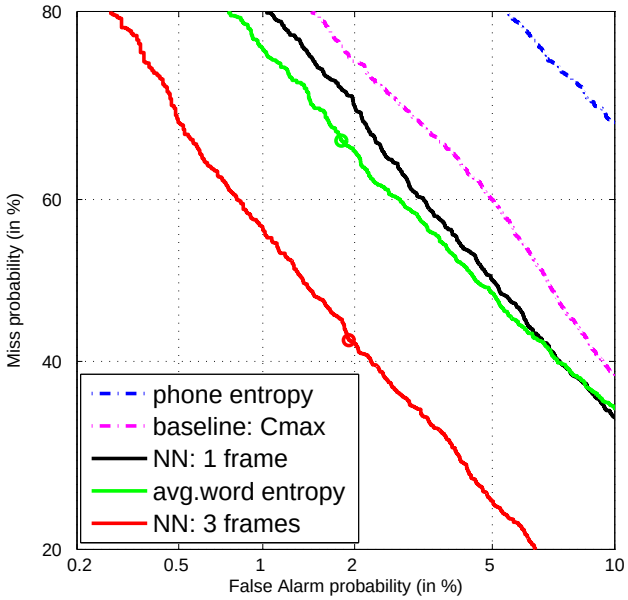

Figure 3.6: OOV detection using NN with 1-frame and 3-frame input $(t, t-6, t+6)$.

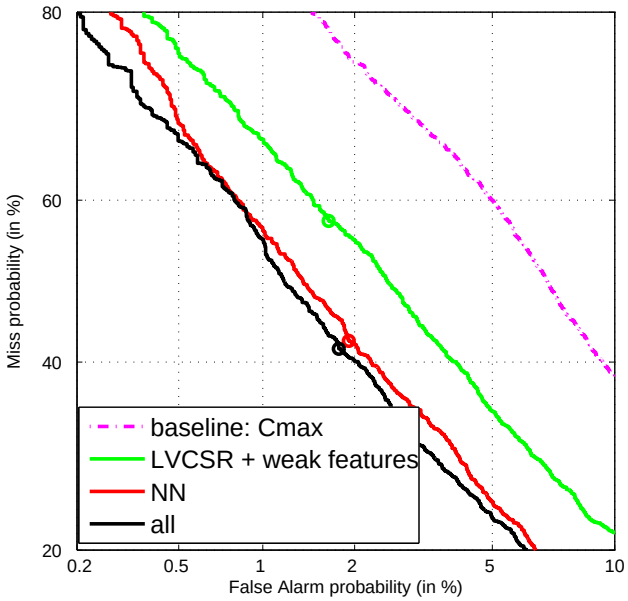

Figure 3.7: OOV detection using combination of LVCSR+weak features and NN.

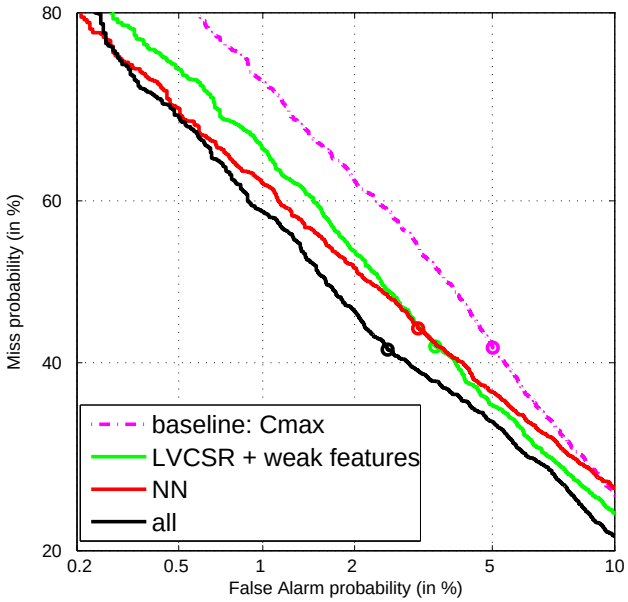

Figure 3.8: Recognition error detection using combination of LVCSR+weak features and NN.

Chapter 4

Conclusions

We have shown that combination of parallel strong and weak posterior streams is efficient for detection of OOVs and also for the detection of recognition errors.

Different scores perform differently for the two tasks; NN seems especially suitable for the OOV detection. We are however aware of the simplicity of the defined task, and in future we plan to test the outlined approaches on more representative spontaneous speech data.

Appendix A

Out Of Vocabulary Visualization Toolkit (OOVTk)

A.1 Introduction

The general purpose of the toolkit is the visualization and analysis of all recognizer components used in the Summer Workshop - it helps to get a better idea about the working of recognizers and to understand the OOV problem, and the involved patterns and distributions of posteriors.

It mainly uses phone posterior probability files obtained from different speech recognizers to create so called posteriograms overlayed with several cues like word boundaries, words and phones in reference and recognition, and many measures like framewise phone entropy, word lattice word entropy, Kullback-Leibler divergence between different sets of posteriors.

Thus, one can visually see the variation of these cues over OOV/mis-recognized and non-OOV/correctly recognized regions of speech and get a good idea of typical patterns which happen in these regions, and of the performance of several measures to detect OOVs in continuous speech.

Since the toolkit also allows to hear the audio and to zoom into any portion of the file, it can for example also be used to hear and verify if a word is mispronounced, if it is not an OOV but still the strong and weak recognizer outputs differ.

The toolkit is also capable for statistical analysis of the distribution of any measure (at the moment word entropy) in regions of speech segregated according to different labeling schemes, for example the agreement of strong and weak recognizer outputs or silence regions.

A.2 Usage example

```
--> change to the directory of the OOVTk
Load necessary variables and data:
--> oovtk;
Load an example file:
-->load_file('4k0a0102','test3');
Display posteriograms:
--> oov_display;
--> oov_colorplay;
Load and play audio:
--> load_wav;
--> play_audio;
```

Figure A.1 shows an example output from the OOV display. Gray scaled posteriors from the weak recognizer are plotted in the upper half and from the strong recognizer in the lower half of the figure.

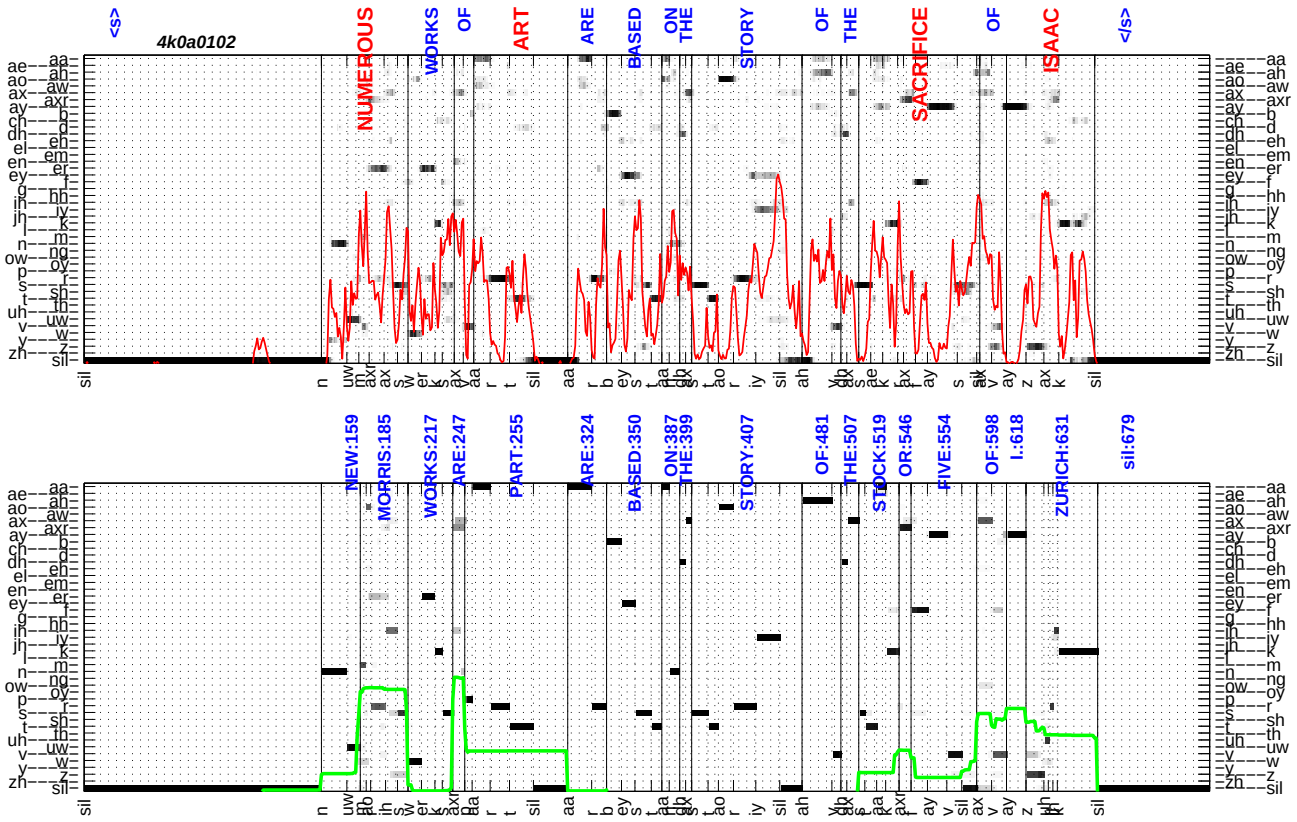

Figure A.1: Example output of `oov_display`

The darkness of the color indicates the value of the phone posterior: black equals one and white equals zero. Vertical black lines indicate hypothesized word boundaries: in the upper part from forced aligned reference transcripts, in the lower part from the output of the strong recognizer. Red and bold words indicate OOVs. The x axis is the time in frames, and the phoneme outputs are displayed. In the upper plot, scaled posteriors frame entropy is shown in red, while in the lower plot, the word lattice word entropies is plotted in green.

A.3 How to obtain the toolkit

The toolkit makes use of most of the files generated in the workshop and relies at the moment on the directory structure present at the workshop. It can be obtained by contacting the author:
`mirko.hannemann@student.uni-magdeburg.de`

Bibliography

[1] D. H. Klatt: Review of the ARPA speech understanding project, *J. Acoust. Soc. Am.* 62 (Dec. 1977), pp. 1345–1366.

[2] I. Winkler, G. Karmos, and R. Näätänen: Adaptive modeling of the unattended acoustic environment reflected in the mismatch negativity event-related potential. *Brain Research*, 742, 1996, pp. 239–252.

[3] J.B. Allen: *Articulation and Intelligibility*, Morgan & Claypool 2005.

[4] C. Van Petten, S. Coulson, S. Rubin and E. Plante: Parks, Marjorie, Time course of word identification and semantic integration in spoken language, *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 25(2), Mar 1999, pp. 394–417.

[5] A. Boothroyd: Speech perception and sensorineural hearing loss, in *Auditory Management of Hearing-Impaired Children*, M. Ross and G. Giolas, Eds., University Park, Baltimore, MD, 1978.

[6] G. A. Miller, G. A. Heise, W. Lichten: The intelligibility of speech as a function of the context of the test material, *J. Exp. Psychol.*, 41, pp. 329–335, 1951.

[7] A. Boothroyd, and S. Nittrouer: Mathematical treatment of context effects in phoneme and word recognition, *J. Acoust. Soc. Am.*, 84(1):101-114], 1988.

[8] F. Wessel, R. Schlüter, K. Macherey and H. Ney: "Confidence measures for large vocabulary continuous speech recognition", *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.

[9] G. Bernardis and H. Bourlard: "Improving Posterior Based Confidence Measures in Hybrid HMM/ANN Speech Recognition Systems", in *Proc. of International Conference on Spoken Language Processing (ICSLP'98)*, Sydney, Australia, 1998.

[10] H. Ketabdar, M. Hannemann and H. Hermansky: "Detection of Out-of-Vocabulary Words in Posterior Based ASR", in *Proc. Interspeech 2007*, Antwerp, 2007.

[11] A. Ljolje, F. Pereira, and M. Riley: "Efficient General lattice Generation and Rescoring". In *Proc. Eurospeech '99*, Budapest, Hungary, 1999.

[12] C. White, J. Droppo, A. Acero and Julian Odell: "Maximum entropy confidence estimation for speech recognition", in *Proc. ICASSP 2007*, Hawaii, 2007.

[13] R. Rosenfeld: *Adaptive Statistical Language Modeling: A Maximum Entropy Approach*, Ph.D. thesis, Carnegie Mellon University, 1994.

[14] A. Ratnaparkhi: *Maximum Entropy Models for Natural Language Ambiguity Resolution*, Ph.D. thesis, University of Pennsylvania, 1998.

[15] T. Hain, et al: "The AMI System for the Transcription of Speech in Meetings", In *Proc. ICASSP 2007*, Hawaii, 2007, pp. 357-360

[16] P. Schwarz, P. Matějka, and J. Černocký: "Hierarchical structures of neural networks for phoneme recognition", in *Proc. ICASSP 2006*, Toulouse, 2006, pp. 325-328

[17] L. Mangu, E. Brill and A. Stolcke: "Finding Consensus Among Words: Lattice-Based Word Error Minimization". In *Proc. Eurospeech'99*, Budapest, 1999, pp. 495-498.