ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

Context-dependent ASR

PhD Study Report

Jan Hejtmánek

Technical Report No.
May, 2009

Distribution: Public

Abstract

Computer speech recognition gains more and more attention these days with its implementation in nearly everyday life. But the ultimate goal is still out of reach. The automatic recognition (ASR) systems can very precisely work on small domain. However the bigger the domain is the worse is the performance of the ASR system.

The aim of many researchers is to diminish this problem on various levels of the ASR.

This work describes components of an ASR system, how they are working together and delves into prosody and how it is used in ASR. From the usage of prosody, the main part of work describes how the ASR can be improved better modeling of the speech variance. We discuss usage of triphones, syllables and other models as well as algorithms and techniques for clustering.

Copies of this report are available on
http://www.kiv.zcu.cz/publications/
or by surface mail on request sent to the following address:

University of West Bohemia in Pilsen
Department of Computer Science and Engineering
Univerzitn¶³ 8
30614 Pilsen
Czech Republic

Contents

1. Automatic Speech recognition

1.1. Brief History of ASR

The first ASR systems were developed in the 1950s and were based on template matching algorithms. This approach is very limited and is usable only for recognition of only few words. The best know recognizer was built in Bell laboratories in 1952 for isolated digit recognition.

Researches were than searching for a better way how to make the ASR system more robust and usable for more general tasks. Limited success brought utilizing only expert knowledge on linguistics, phonetics etc. Next success was brought by several Japanese laboratories that demonstrated speaker independent recognition of speech segments (vowels, digits,…) and most notably used segmentation and phonemes for speech recognition. This marked the path to the continuous speech recognition system.

In 1960s an alternative to speech segmenter gain its credit. The concept of adopting a *non-uniform time scale* for aligning speech patterns deals with non-uniformity in repeated speech events. This approach better model the alignment between two segments (utterances).

Real breakthrough came in 1970s with the dynamic programming techniques and success in applying of the *Linear Predictive Coding* (LPC). Based on the earlier success Tom Martin developed the first real commercial ASR system. Though the system was used only in a few simple cases its real impact was in DARPA[1].

Thanks to DARPA, numerous ASR systems were built. Worse noting is the *Harpy* system. It was able to recognize over thousand words with reasonable accuracy and had the structure of a modern ASR system.

Also in the 1970s, IBM and AT&T Bell Laboratories gain their attention. IBM started to develop the VAT (Voice-Activated Typewriter). The aim was to create the recognition system with reasonable accuracy and with as huge vocabulary as possible as the system was supposed to be used in

[1] Advanced Research Project Agency of U.S. Departement of Defense

everyday correspondence (i.e. task of *transcription*). This system was speaker-dependent and the first to use the n-gram language model.

The AT&T laboratories aimed at new services for public as voice dialing, voice commands etc. for which the system ought to be speaker-independent. The ultimate acoustic variety in pronunciation led to clustering algorithm for acoustic models representation (first for pattern matching but lately for statistic models). The research continued in dealing with keyword spotting and acoustic modeling.

The shift from pattern matching to the more robust statistical modeling took its place in 1980s and 1990s. Although the *hidden Markov model* was previously used it gains more attention and is widely used. As a parallel, the idea of modeling the speech with artificial neuron networks was introduced. Both methods started the wave of steady improvement.

These improvements revived the DARPA's attention. With the DARPA's support the standard evaluation techniques were pursued in 1990s. In these years, great progress was made although in development of software tools for ASR (such as the HTK[2]). This made the research available to wide public.

Late 1990s saw the first real wide public applications. It was mainly thanks to AT&T (automated handling of operator-assisted calls and customer service line). The constant advancement in technology in the turn of the millennium brought in the ASR into every life. Voice-dialing (mobile phone contact list) and voice-commanding (turn-by-turn navigation) is nowadays used in ordinary situations. However the ultimate goal is still out of reach. The research turns to the local groups to adopt the ASR systems to various languages and much bigger vocabularies. The goal of the IBM group from 1970s is trying to meet numerous research groups all across the world, however still with only limited success and only on carefully selected domains.

The milestones of speech recognition are shown on the figure 1.

[2] Hidden Markov Model Tool Kit, University of Cambridge

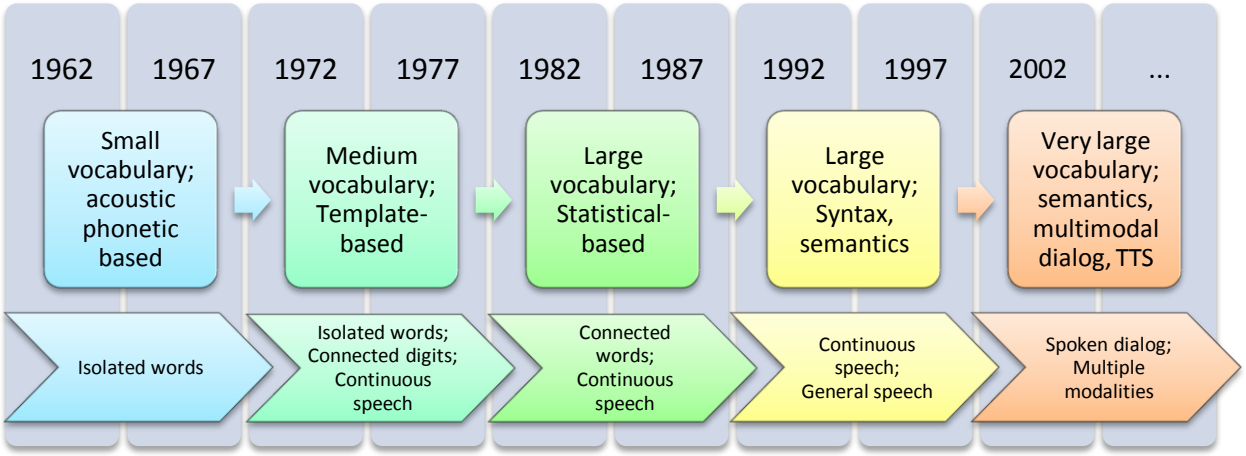

Figure 1: Milestones of automatic speech recognition

1.2. Components of ASR system

The ASR as a complex task was divided into several subtopics. In this work we only focus on a general-purpose strict speech-text transcription.

The input of the system is a spoken language in the form of waveform and the output is a string of transcribed discourse. The recorded discourse goes through stages as shown on the figure 2. Each state can be a component of a ASR system.

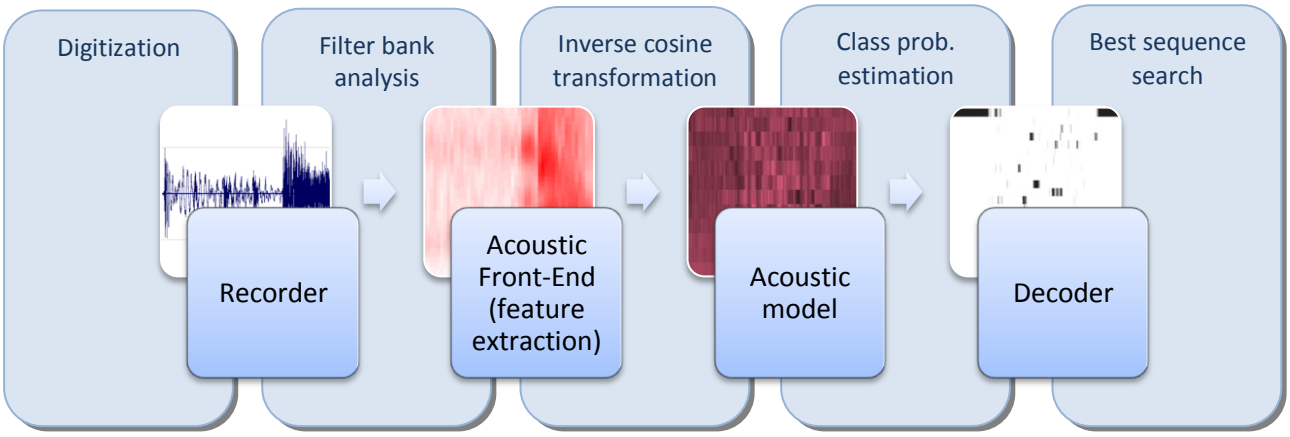

Figure 2: Components of a ASR system

Recorder

This part of system is usually responsible for digitization of speech. The key to a good recognition system is to have all the speech. This means to bring in silence detection, noise canceling and other algorithms that ensure the discourse to be complete. The most common is 16kHz/16bps sampling rate. The extracted recording is then passed to the next stage (component).

Audio Front-End

The signal from recorder is split into small segments (frames). From these segments the features are extracted. Features are usually 12 *mel-frequency cepstral coefficients* (MFCCs), energy and corresponding delta parameters. However, much research is given into this topic. The extracted features (or feature vectors) is bases for the recognition process.

The second task for the audio front-end is to reduce the data caught by the recorder.

Acoustic model

This component usually classifies (or assigns probabilities or scores) every frame into phonetic class. This task can be done by probability component of a HMM or an ANN.

Decoder

The decoder searches for the best possible path through the sequence of scored phonetic classes given by the acoustic model. Usually some restrictions are needed in order to obtain the path fast. The most common is the *Viterbi algorithm* for this task.

1.2.1. Feature Extraction

The raw digitized speech is segmented into very short frames, usually 10-50 milliseconds. The segments overlap to prevent aliasing effects at the segment boundaries.

Today, the most common is to extract frames based on frequency distribution in the speech frame. For example a bank of band pass filters output a feature vector of short time energy values for each specified band. The most successful filterbank has the filters equally spaced up to 1 kHz and logarithmically spaced bands above 1 kHz.

By modeling the vocal tract, another method was created. *Linear Predictive Coding* (LPC) extracts the features by applying an all-pole filter with its features computed by a fast autocorrelation method. This method is faster than the filterbank analysis but the results are worse.

For the next component of the ASR system the features need to be uncorrelated [PAV09]. Unfortunately coefficients acquired from both LPC and mel-scale filterbank are highly correlated. Applying the inverse cosine transform to the mel-scale filterbank coefficients leads to the new set of coefficients (cepstrum) that are close to be uncorrelated. These features are therefore used as a standard set of features and called mel-scale filterbank cepstral coefficients (MFCCs).

Some novel approaches have been developed but the contribution to the overall ASR system performance is too small. Therefore the main focus of researchers is directed to other components.

1.2.2. Phonetic units

Phonetic units are classes from which the output of the system is composed. In general purpose ASR systems it would be unfeasible to use whole sentences or even whole words. However for very small domains with very small variety in recognizable units (consider for example "yes"/"no") supersizing leads to better performance.

For general-purpose ASR the phonetic units usually start with units called *monophones*. These roughly correspond to phonemes as described in phonology of the used language [RLR79]. This is the simplest possible model with only few classes (for the Czech language usually 36, the number can vary as no strict rule can be given).

The pronunciation of the monophones differs from utterance to utterance. To model this extending the context was proved [OSM02] to be the solution.

Diphones

To model the pronunciation variety the first to come to mind is to look to the right or left and depending on the neighbor model the phoneme. However, as is described in [OSM02] the usage of these phonetic units is highly problematic. When used on the same set of data, the result is different for the left and right diphones so it is unclear which to use. The result of the ASR is also inferior to the one which uses triphones

Triphones

Today's state-of-the-art ASR systems use so-called triphones. Like with diphones, the model takes advantage of the context. This model takes from both sides and thus can cover the whole pronounceable (and of course unpronounceable) space of phonetic units.

Pentaphones

Like triphones, this approach takes advantage of the two neighboring phones to model the phonetic unit. The number of such units when used on mid-sized vocabulary is enormous.

Syllables

Syllables are the smallest pronounceable part of the speech. Thus, when used properly, should lead to better results in ASR. The length of a syllable

varies but the boundaries are easier to find. Moreover, the pronunciation of syllable is usually very consistent.

Word-dependent units

These units are used in very small ASR systems where the advantage of very clear boundaries can be used. In all other cases and general-purpose system these units are losing since the number of units grows with vocabulary. This leads to the problem when the ASR system cannot be trained because of lack of training data.

However, even in the general-purpose ASR these units can have a place – when used with problematic words.

Other units

Much research was done in the field of search for the best phonetic units. In the Sphinx ASR only the states of HMM (called Senones) were used. This led to superior performance than when using monophones or triphones. Polygraphs are next units, described in [SCh02].

Depending on the domain, purpose, the amount of training data and computational resources the right compromise has to be made when building the ASR system.

1.2.3. Acoustic model

Two approaches are usually used to do acoustic modeling – Hidden Markov models and artificial neuron networks (especially MLP[3]).

The standard approach is to use Gaussian mixtures (GM) with HMMs [SCh02]. This method was proven to deliver accuracy and generally good results. With the need of even better results some new techniques were created. Today's most successful solution is referred to as a hybrid approach where the artificial neural network estimates the emission probabilities of hidden Markov model [PAV09].

The multi-layer perceptron is capable of learning any arbitrary function as well as output class posterior probabilities. If each network output is associated with a HMM state S_j than the activations of output layer neurons can be interpreted as $P(S_j|o)$. From the Bayes' rule we can get than:

$$P(o|S_j) = \frac{P(S_j|o) \cdot P(o)}{P(S_j)}$$

[3] Multi-level perceptron

Using ANN instead of GM bring some potential advantages. First, the ANN doesn't need any assumptions before the training of the model. The usage of the ANN allows more general context sensitivity of the model. Third, the ANN/HMM hybrids require less trainable parameters and allow for a more efficient pruning during the decoding phase. However, the ANN also has some disadvantages (like longer training times when used with longer vocabularies, etc.) and hybrid systems are still in the research.

1.2.4. Training

Training of the statistical model is the key process of the ASR system.

HMM

Before the parameters of an HMM can be trained some initial estimates of the have to be chosen. Usually this is done by flat-start procedure. It means to create one prototype of a HMM that represents an unknown phonetic unit with global means and variances computed on all the training data. All the models of the phonetic units are then assigned the parameters of this proto-unit. Models are then trained with training data and the procedure expects that:

1. Each training training utterance is uniformly segmented
2. Enough of the models will align with their actual realizations (so that in next iterations of training the models will improve as intended)

ANN

When using the ANN as a model each frame in the training set must be labeled with the class it belongs to. For this purpose an existing recognizer can be used (regardless whether HM or ANN/HMM hybrid). This technique is called forced Viterbi alignment and it automatically assigns labels to the frames. Repeating the labeling process several times can further improve the recognition accuracy of the hybrid ANN/HMM. This technique is also called recursive labeling.

After the labels were assigned the actual training (e.g. rescoring of the weight of the ANN) can start. The most common weight update strategy is the incremental version of the backprop algorithm.

1.2.5. Decoding

The fundamental problem of the speech recognition is to choose the one most likely word sequence of words $W^*=w_1,w_2,...w_n$ from the sequence of observation vectors $O=o_1,o_2,...o_n$ that

$$W^* = \max_{\forall W} P(W|O) = \operatorname{argmax}_{\forall W} \frac{P(O|W)P(W)}{P(O)}$$

The *P(O|W)* is the acoustic model likelihood of the word sequence and the *P(W)* is a prior probability of the sequence computed by the *language model*. We can exclude the *P(O)* as it is independent of the word sequence and thus remains constant.

Using standard computation power in general-usage ASR it is not possible to compute the *P(W|O)* for every possible word sequence in a finite time. Thus, a compromise has to be made and some sequences have to be ruled out.

Most large vocabulary ASR systems include pronunciation model. Then, the search is approximated as:

$$W^* \approx \underset{\forall W}{\operatorname{argmax}} \left[\max_{\phi}(x|\phi) P(\phi|W) P(W) \right]$$

Language model

The language model computes the prior probability of the sequence *P(W)*. The simplest deterministic model is the one that assigns only ones or zeroes (the sequence is either possible or impossible). A example would be a dictionary that holds all the possible pairs of words that can follow. A more powerful deterministic language model can be a finite state automation (FSA) with words associated with the transitions. If a word sequence is accepted its probability is set to one, else it is set to zero. This FSA can be generated from a non-recursive context-free grammar (like Extended Backus Naur Form).

The stochastic language models are the next step in increasing of the complexity. These models try to actually estimate the probability of the *P(W)* and usually have trainable parameters. The most commonly used is the *N-gramm language models*, where the N is referred to as the *order of the model*. The probability of the word sequence is computed like

$$P(w_1, w_2, \dots w_n) = \prod_{i=1}^{N} P(w_i | w_1, w_2, \dots w_{i-1})$$

The unigram language model is the simplest possible choice where N=1. With the N growing so is the computation complexity. Moreover, the higher the order of the model is the more the model suffers from the sparse data problem [OSM02]. Therefore, various strategies to compensate the the ill-trained N-gramms exist.

The most commonly used is the bigram (N=2) language model.

Pronunciation model

The pronunciation model is the last contribution to the model. As that a wide variety of approaches has been created to design the most correct

$p(\phi|W)$.[OSR02] The majority assumes that the intermediate unit ϕ_i corresponds to a phone.

Time Synchronous Decoder

An efficient decoder is a Viterbi algorithm. To use it in ASR, the lexical and language models must exist. The language model (as described earlier) defines the all allowed sequences of words (or their probabilities). The lexical model consists of all words together with their representation in the phonetic units.

The Vitebi algorithm does not maximize the probability of the word sequence but instead it maximizes the probability of a single most likely path along the HMM states in HMM (or hybrid ANN/HMM) model. This was proven [PAV09] to be a better way since this procedure allows a direct retrieval of the best word sequence.

The Viterbi algorithm is only usable with bigrams or word-pair grammars where the probability of the next word is only dependent on the current word. This satisfies the first Markov assumption and is computable in a finite time.

Pruning

To make the Viterbi search work faster some algorithm were developed. Prunig tries to omit computation of path that with very low probability. The most common technique is called *beam search*.

The beam search defines the beam (i.e. the threshold or number of states) for every frame. When looking for the next path only the states from the beam are taken into account. Thus the beam search crops the space.

Best-First Decoder

The best-first search is time-asynchronous algorithm and is used as a alternative to the Viterbi algorithm in many large vocabulary systems [SCh02]. It is based on A* algorithm and uses a composite function to evaluate the score $f_h(t)$ of each hypothesis h at time t:

$$f_h(t) = a_h(t) + g_h^*(t)$$

The $a_h\ (t)$ is the score of the partial hypothesis based on the information collected to the time t. The $g_h^*(t)$ is a heuristic expectation of the remainder of the score up to the end of the utterance. If the reminder is an upper bound on the actual score of the hypothesis the search guaranties to find the best path.

The time-asynchronous search offers some potential advantages:

1. The computational time can be saved as the most likely path can be found before all the hypotheses are computed.
2. A higher order language model can be used during the search

The sensitivity to the chosen heuristic $g_h^*(t)$ is the main disadvantage [OSM02].

2. Improving ASR with phonetic units

2.1. Prosody and acoustic variation

The easiest way how to add prosody to the ASR is to add a feature in the observation vector. Such a action however dig deep into the recognizer and the feature have to be used later in the recognition process.

Another approach how to model the prosody is to incorporate a hidden variable that can be leveraged in pronunciation modeling and/or acoustic model clustering. Also, word-level features from the expert-knowledge can be used.

All these approaches assume that the number of phonetic units (or their parameters) is bigger than can be properly trained (be it GM or context-dependent units).

2.2. Extending the Context

Large part of degradation of speaker-independent ASR system is related with acoustic variation, speaking style or loosely structured language. The work on pronunciation modeling in terms of phoneme-level substitutions, insertions and deletions yielded only small improvement.

Instead of this, the pronunciation modeling pronunciation on the state level brought much better results. This allows the Gaussian mixture model to be shared across all states.

Experiments with adding the phonetic context to the model brought another wave of success [SYo08]. Experiments proved that the context-dependent units can greatly increase the recognition accuracy.

2.2.1. Triphones

The triphones are today's most commonly used context-dependent units to model the acoustic variance in the words.

Let's assume the ASR system is built upon the five-state HMM (three omitting states). The triphones are then built around a center phoneme unit. This can be done from monophones. Every word is than transcribed into the new units as shown on the figure 3.

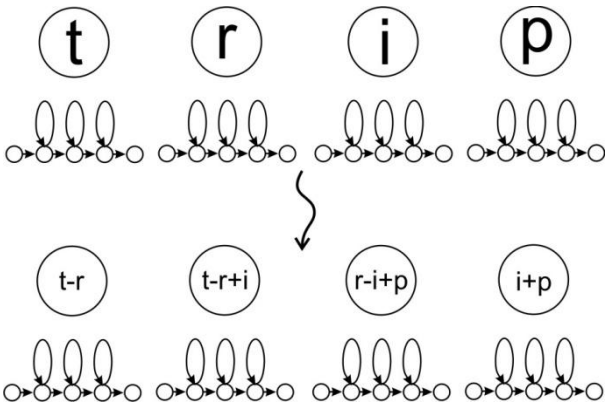

Figure 3: Monophones to triphones transition

Obviously, creating triphones looses on the boarders of the words. There are two solutions. So called cross-word triphones can be created or the system can work with mixed acoustic models (monophones, bihpones and triphones).

Creating cross-word phonetic units have some advantages and disadvantages. To use pure triphone model the cross-word system works and triphones are created as shown on the figure 4. However, a problem with word transitions occurs. As phonetic studies show, a pause between words can be small or large with no particular link to the phonetic units (or phonemes). So, lots of phonetic units are created with very small amount of training data.

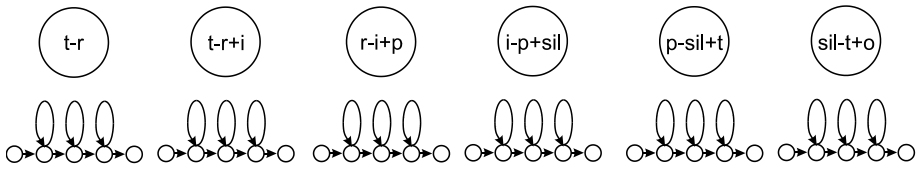

Figure 4: Cross-word triphones

Again, the most common approach for ASR systems with small or medium sized corpora is to build a mixed acoustic model.

2.2.2. Syllables

Syllables are the smallest possible pronounceable part of speech. Studying the English language the psychoacoustics argued for the syllable to be a unit of perception. The main differences between syllables and triphones are in the length of the units (with syllables it is variable) and in the construction.

Whereas the triphones (or pentaphones) are artificially created phonetic units to cover the variations in pronunciation, the syllables are built naturally.

In series of studies, the systematic variation with respect to the syllable structure. The comparison test [OSR02] showed some interesting results:

1. The onset of the syllable maintains its canonical identity at most times (regardless of the speaking style)
2. The coda is less often realized in canonical form in conversational speech than in read speech

Thus, the syllables appear to be a good phonetic unit for the ASR. There are many ways how to build up an acoustic model from syllables. Several researchers succeeded in outperforming the triphone-based ASR system [OSR02],[JON97],[GAN01],[SET03],[MES04],[HAN05].

There are several choices when mapping the syllables onto the underling HMM. Again, as the research is in progress, the best way is not clear yet and most common problem is the lack of training data [OSR02],[HAN05].

Summary of mapping and construction of syllable models is shown in [TOD98].

One-to-One mapping

This approach replaces the phonetic units with syllables. There is the very same model for all syllables with no respect to the length of a syllable.

The problem of this approach is when trying to cluster the parameters using decision-tree.

Variable length

To overcome the problem with clustering, variable length of a model can be taken. Such model is usually a compound of monophones. Then it is clear which state represent which phoneme.

The two approaches are described on figure 5.

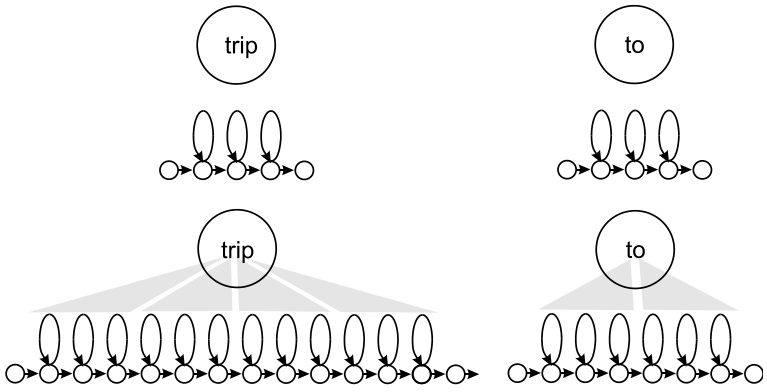

Figure 5: Syllable modeling

We distinguish four basic types of syllables.

A. Heavy syllables

Has a branching rhyme. All syllables with a branching nucleus (long vowels) are considered heavy. Some languages treat syllables with a short vowel (nucleus followed by a consonant (coda) as heavy.

B. Light syllables

Has a non-branching rhyme (short vowel). Some languages treat syllables with a short vowel (nucleus) followed by a consonant (coda) as light.

C. Closed syllables

Syllables end with a consonant coda.

D. Open

Has no final consonant.

These classes can be later used to better phonetic unit clustering.

2.2.3. Bigger units

Even though the most commonly used are the triphones, much research is done in the field of acoustic modeling. To model the bigger context, pentaphones half-syllables and other units have been successfully tested. The results are at the best the same as using triphones.

In some works the problematic words are modeled as a one model. Empirical tests prove that this can be a solution in small or mid-sized vocabularies tightly bind to a domain.

2.3. Tightening the number of units

Extending the context and thus increasing the number of phonetic units lead inevitably to less training data for the units. The two solutions are at hand – to get more training data and to tighten the number of units.

Getting more training data is the best choice but is not always the right option. Depending on the number of phonetic units the training corpus would have to be extremely large. Acquiring bigger corpus for specific domain is usually either very expensive or takes very long time.

Therefore most common [RLR79], [NEE05], [EST02] practice is to tighten the number of units or their parts. This can be done manually or using sophisticated algorithms. The simplest way how to reduce the number of units is to extract the units from the corpus. Then, two contradiction actions take place. We need to tighten the number of phonetic units but do not lose the ability of model to recognize. The most common ways is to tie the parameters of the underlying HMM states. These states can be tied on the base of their distance or their classification into groups.

An overwhelming number of methods and algorithms was created with various effect.

2.3.1. Data-driven clustering

Data-driven clustering is based on previous knowledge of the phonetic units' statistics. Most of the algorithms work on the basis of computing the distance between the Gaussian distributions and the using some criteria classifying the units (or states of the HMM).

Euclidian, Hamming or Bhattacharyya distance can be used. The classification is then a standard task. Again, standard algorithm can be used.

Very interesting attempt to improve the data-driven method was to use "speech trajectory clustering" [HAN05],[HAN06]. It was built to model multi-path topologies, and the algorithm was successfully applied to longer-length acoustic models (linguistics-based Head–Body–Tail models [CHOU94]) for connected digits recognition. In this approach, speech observations are regarded as continuous trajectories along time in acoustic feature space, and clustered based on mixtures of regressions of these trajectories. Each trajectory cluster is modeled as a prototype polynomial function with some variability around it. The variability within the clusters is described in terms of a mixture of Gaussians. The EM algorithm is employed to train the cluster model in a maximum likelihood manner. Using the results of trajectory clustering, multipath models can be trained based on the training tokens in different trajectory clusters.

2.3.2. Decision-Tree clustering

Most ASR systems use decision-tree clustering to reduce the number of model parameters. For large scale vocabulary ASR systems it is necessary to map the large number of distributions (i.e. phonetic unit's parameters) to a smaller set that can be robustly estimated [OSM02],[HAN05],[TOD98].

This technique is particularly attractive for parameter tying as it allows mapping of any sub-word unit that was not seen in the training data. Typically, a separate tree is used for each state of a HMM.

During the training phase, all the context-specific observations data are pooled at the root of the decision-tree. A set of predefined questions is at each node of the tree. These questions are typically about phonetic context but can be also about the word-position, prosodic features, etc.

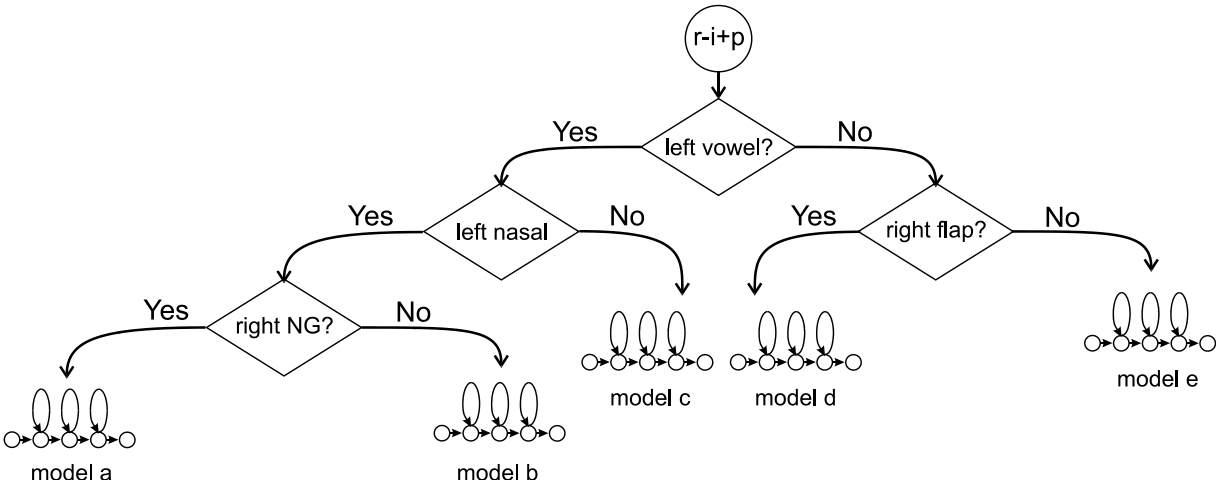

Figure 6: Example of a decision tree for acoustic modeling.

Assuming that all the data in the root share common Gaussian, the question that maximizes the likelihood of the data in a node is chosen as a candidate for the next split. The best partitions of the new clusters resulting from the split are added to the list and the tree is grown until a stoping criterion.

That can be either a final phonetic unit (triphone, pentaphone, syllable,…), the amount of members in a class or the deep of the tree.

Usually, evaluating the split involves computing a generalized log likelihood ratio:

$$\log\left[\frac{\left(\left(\max_{\mu_L,\Sigma_L} P(\chi_L|\mu_L,\Sigma_L)\right)\left(\max_{\mu_R,\Sigma_R} P(\chi_R|\mu_R,\Sigma_R)\right)\right)}{\max_{\mu_P,\Sigma_P} P(\chi_P|\mu_P,\Sigma_P)}\right]$$

Where L, R and P indicate Left, Right and Parent node, μ stands for means and $\sum$ for covariances. χ indicates a data subset. Union of L and P parts of χ gives all previous data subset on the level (P).

The complexity of the algorithm is typically *O(QN)*, where *Q* is the number of question candidates and *N* is the number of observed contexts.

When building the word model for decoding, a particular context-specific phoneme is dropped into the tree and answering the questions is led to the proper class.

In most ASR systems an expert-knowledge is used to create the questions. These are usually hand-written. However, some other features are being experimentally used. These can be:

- Position in word.
- Stres on the (part of) phonetic unit (if previously labeled)

- Length of the unit (if it is syllable or other such unit).
- Special features (if previously created).

Clustering using this extra information is also referred to as *tagged clustering*.

2.3.3. Tagged clustering

The work [OSR02] shows that using the word and syllable level features increases the likelihood and the questions that are related to these features are asked very early.

However, the use of word position (initial, medial, final) alone has not so far proved to be useful. Usage of other features also lead to only very little gain.

Some researchers propose that using more of such features can eventually lead to better results.

2.3.4. Multi-stage clustering

When trying to cluster more complex acoustic models a few limitations in the decision tree procedure appear. With the growing number of features and units the memory needed increases dramatically. Also the sparse-data problem reoccurs. There is a big amount of units that are very rare and therefore poorly trained and clustered.

One possible approach is to divide the clustering process to multiple stages. The decision tree can be viewed as a function that maps a feature vector to an index (a particular state of an acoustic model).

The group of contextual information is divided into several feature vectors, for this example we assume two of them. In the first stage for the first tree function $T_1: f_1 \rightarrow b$ where b is a leaf of B. In the second stage, the training data is annotated with f_2 along with the value of b. The b is obtained by dropping its context f_1 into the tree T_1. Using the new data (that came from the T_1) new set of features f_2 can be used to build the second tree T_2.

Whereas the questions of f_1 and f_2 are ususally all hand-written, the value of b is not. Moreover, the b is large and not all the leaves will be used. Therefore the set of binary questions is build from the tree T_1. These questions ensure that only the valid leaves of the T_2 will be grown.

After both trees are grown they can be merged into one depending on the proprieties.

The multistage clustering helps to diminish the problem of sparse data since only part of the features are used in the stage.

The storage and computational cost of the multistage clustering depends on various factors:

1. The number of components (features) in *f.*
2. The uniformity of the training data.
3. The size of the tree in every stage (the stopping criteria).

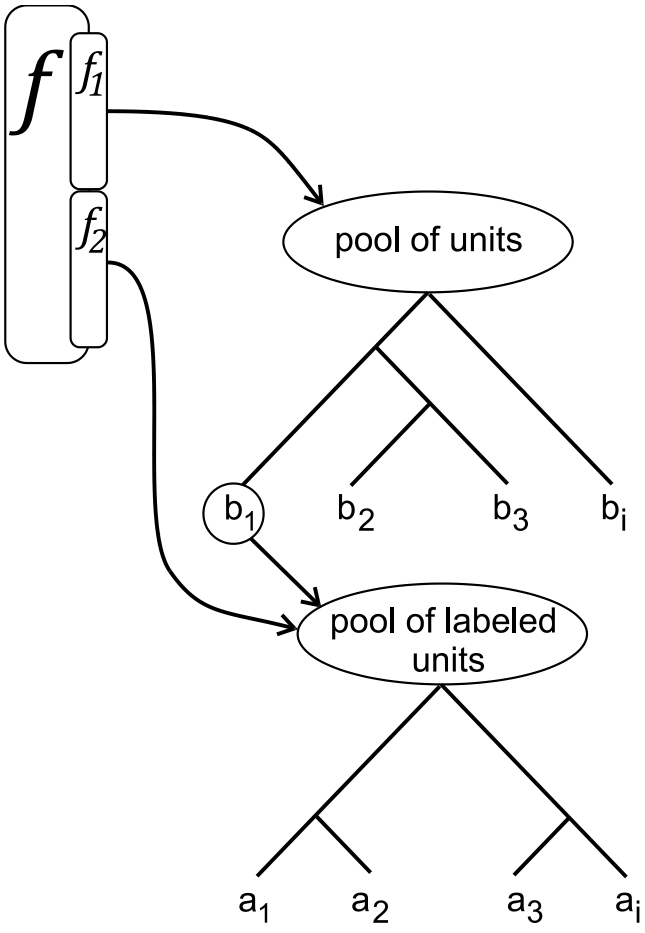

Figure 7: Example of a multilevel decision tree clustering.

2.4. Other types of clustering

As the works show the problem of clustering is not solved. The research shows that it is highly correlated with the training data, domain of the ASR and speech corpora.

Even though the decision tree algorithm and its derivates is highly the one most commonly used, other clustering techniques are under research. Usually it a hybrid type clustering that combines some ideas from prosody, decision-tree techniques and data-driven clustering.

2.5. Evaluation

Although an official evaluation methodology for evaluation of acoustic models was created (started 1982, last update in 2008), it is very extensive and not suitable for fast research. This methodology comes from DARPA program, more specifically from the Acoustic Model Evaluation Committee (AMEC). However part of the methodology can be used.

The standard methods of evaluation of the acoustic models are mostly based on the overall performance of the whole ASR system.

%WER

Word Error Rate is the most common criteria. ASR system should work with less than 10% of WER.

%Corr and %Acc

From the statistics of the ASR system the accuracy (%ACC) and correctness (%Corr) is computed.

During the testing phase of the ASR correct hits (*H*), deletions (*D*), insertions (*I*) and substitutions (*S*) are counted. From these values the overall scores are computed as follow:

These metrics however do not evaluate only the acoustic models but the whole system.

Measuring validity of clustering

To determine if the clustering algorithm succeeded, several methods can be used.

1. Davies-Bouldin index (*DB*)

   The DB index is the average similarity between each cluster and its most similar one. It is desirable for the clusters to have the least possible similarity to each other, so the smaller the DB index, the more clusters tend to be compact and not overlap, thus better expected separation. The number of clusters which minimizes the DB index is the optimal one.

2. F-Measure

   The F-measure is an index which describes how well a clustering configuration fits a classification. It also gives means to compare

different clustering's and determine which is most likely to correspond to the classification.

Purity of a clustering describes the average purity of the clusters obtained. In other words, it is a measure of how good the clustering is, if one seeks to have clusters which represent only one class.

3. Purity

   In a similar way to the F-measure, one can define the purity of a clustering ρ_i of each cluster K_i as the highest precision p_{ij} reached over the different classes:

$$\rho_i = \max_{l \leq j \leq C} p_{ij}$$

   The weighted average of ρ_i over all clusters yields a measure of quality of the whole clustering:

$$\rho_i = \sum_{l \leq j \leq C} \frac{card(K_i)}{N} \rho_i$$

   The closer purity is to 1, the more clustering tends to break down classes into clusters one by one. In other words, we try to perform a covering of each class using clusters, in order for clusters to be able to define classes in return.

4. PCA

   Principal component analysis, or PCA, helps to discover or to reduce the dimensionality of the data set and to identify new meaningful underlying variables. It performs a linear transformation on an input feature set, to produce a different feature set of lower dimensionality in a way that maximizes the proportion of the total variance that is accounted for.

   The PCA analysis can help to understand which cluster

3. Conclusions and Future Work

3.1. Work already done

I have successfully built and tested several triphone-based and syllable-based ASR systems. Thanks to context-dependency the baseline results of a syllable-based ASRs were much higher than a monophone ASR and slightly worse than fine-tuned triphone ASR.

On the syllable-based ASR I have also successfully tested data-driven clustering, which led to visible improvement. The preliminary results show that better clustering is the key to get better performance.

I have also successfully tested all the main method of the clustering on the triphone-based ASR. From variety of tests the clear winner is the decision-tree clustering algorithm for which I have manually created all the necessary context questions for the Czech language.

For the statistics and training purposes I have also adopted and improved a FSA syllabification algorithm for Czech and English.

All the built ASR systems were tested on the corpora accessible in LIKS[4] ZČU.

3.2. Conclusions

Although the studied researches and materials are mainly for the English language waste majority of approaches can be adopted to a Czech ASR system. The acoustic models and problems that are to be dealt with are very similar.

Moreover, all the approaches developed for the Czech language in the topic of acoustic modeling can be easily adapted to other languages.

[4] Laboratory of Intelligent Communication Systems, Dept. of Computer Science and Engineering, University of West Bohemia in Pilsen, Czech Republic

3.3. Aims of doctoral thesis

1. Propose, implement, test and evaluate the best possible clustering technique for English and Czech syllables.

2. Build syllable-based ASR that would perform better than our current best recognizer.

3. Test and confirm the proposed methods on bigger corpus

Bibliography

[JLR04] B.H. Juang, Lawrence R. Rabiner. Automatic Speech Recognition – A Brief History of the Technology. [Online] 2004. http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/354_LALI-ASRHistory-final-10-8.pdf.

[RLR79] L. R. Rabiner, S. E. Levinson, A. E. Rosenberg and J. G. Wilpon. s.l., Speaker Independent Recognition of Isolated Words Using Clustering Techniques. : IEEE Trans. Acoustics, Speech and Signal Proc., Aug. 1979, Vols. Vol. Assp-27, pp. 336-349.

[SYo08] S. Young, et. al. the HTKBook. [Online] 2008. http://htk.eng.cam.ac.uk/.

[MRa96] Brian Mak, Etienne Barnard. Phone Clustering Using the Bhattacharyya Distance. Portland : Center for Spoken Language Understanding Oregon Institute of science and Technology, 1996.

[LZe05] J. Lánský, M. Žemlička. Text Compression: Syllables. Proceedings of the Dateso 2005 Annual International Workshop on DAtabases, TExts, Specifications and Objects. CEUR-WS. 2005, Vol. 129, pp. 32-45.

[JHe07] Hejtmánek, J. Use of context-dependent units in speech recognition (Master Thesis). Pilsen : University of West Bohemia in Pilsen, Faculty of Applied Sciences, 2007.

[HPA07] J. Hejtmánek, T. Pavelka., Use of context-dependent units in Czech speech. Balatonfüred, Hungary : Proc. of PhD Workshop 2007, 2007.

[YMO05] Yu, K. and J. Mason, J. Oglesby. s.l., Speaker recognition models. : Proceedings of Eurospeech 95, 1995. pp. 629-632.

[MEd96] M. Edgington et al., Prosody and speech generation.1996. BT Technology Journal. Vol. 14, pp. 84-99.

[SIL08] International, SIL. Glosary of linguistic Terms. [Online] 2008. www.sil.org.

[HPa08] Hejtmánek, J., Pavelka, T. Automatic Speech Recognition Using Context-dependent Syllables. Izola, Slovenia : s.n., 2008. Proceedings of 9th International PhD Workshop on Systems and Control (YGV2008). ISBN 978-961-264-003-3.

[NOH00] NOUZA J., Holada M., A Voice-Operated Multi-Domain Telephone Information System. Istambul : s.n., 2000. Proc. of 25th Int. Conference on Acoustics, Speech and Signal Processing. pp. 3755-3758. ISBN 0 7803-6296-9.

[LLi06] S. Laurinčiukaitė, A. Lipeika., Syllable-Phoneme based Continuous Speech Recognition. Vilnius : Institute of Mathematics and Informatics, 2006. ISSN 1392-1215.

[CSh02] Chang, Shuangyu. Berkeley , A Syllable, Articulatory-Feature and Stress-Accent Model of Speech Recognition. INTERNATIONAL COMPUTER SCIENCE INSTITUTE, 2002.

[NAS07] Narayanan, Sankaranarayanan Ananthakrishnan and Shrikanth, IMPROVED SPEECH RECOGNITION USING ACOUSTIC AND LEXICAL CORRELATES OF PITCH ACCENT IN A N-BEST RESCORING FRAMEWORK.. Los Angeles : Speech Analysis and Interpretation Laboratory Department of Electrical Engineering Viterbi School of Engineering University of Southern California, 2007.

[CHC04] K. Chen, M. Hasegawa-Johnson, and A. Cohen., An automatic prosody labeling system using ANN-based syntactic-prosodic model and GMM-based acoustic-prosodic model. 2004. International Conference on Acoustics, Speech and Signal Processing. Vol. 1, pp. 509-512.

[ISh06] Shafran, Izhak. Baltimore, Acoustic and Language Modeling for Czech ASR in MALACH. : The Center for Language and Speech Processing (CLSP), The Johns Hopkins University (JHU), 2006.

[STN03] Abhinav Sethy, Bhuvana Ramabhadran, Shrikanth Narayanan, IMPROVEMENTS IN ENGLISH ASR FOR THE MALACH PROJECT USING SYLLABLE-CENTRIC MODELS.. s.l. : IEEE, 2003.

[BYL06] Boves, Yan Han and Lou. Nijmagen, EM Algorithm with Split and Merge in Trajectory Clustering for Automatic Speech Recognition. : Department of Language and Speech, Radboud University Nijmegen, 2006.

[OSM02] Ostendorf, Izhak Shafran & Mari. Washington , Acoustic Model Clustering Based on Syllable Structure.: Department of Electrical Engineering, 2002.

[HAN06] Han, Y., Hamalainen, A., Boves, L., Trajectory Clustering of Syllable-length acoustic models for continuous Speech Recognition, Centre for Language and Speech Technology (CLST),
Radboud University Nijmegen, The Netherlands, 2006

[PAV09] Pavelka, T., Hybrid Methods of Automatic Speech Recognition: University of West Bohemia in Pilsen, Faculty of Applied Sciences, 2009.

[OSR02] Ostendrof, M., Schafran I., Bates, R., Prosody models for conversational speech recognition: EE Dept., University of Washington, 2002.

[JON97] Jones, R.J., Downey, S., and Mason J.S., “Continuous speech recognition using syllables,” in Proc. Eurospeech-97, vol. 3, pp. 1171-1174, 1997.

[GAN01] Ganapathiraju, A., Hamaker, J., Ordowski, M., Doddington, G., and Picone J., “Syllable-based large vocabulary continuous speech recognition,” IEEE Transactions on Speech and Audio Processing, vol. 9(4), pp. 358-366, 2001.

[SET03] Sethy, A., and Narayanan, S., “Split-lexicon based hierarchical recognition of speech using syllable and word level acoustic units”, in Proc. ICASSP-2003, vol. 1, pp. 772-776, 2003.

[SETH03] Sethy, A., Ramabhadran, B., and Narayanan, S., “Improvements in ASR for the MALACH project using syllable-centric models,” in Proc. IEEE ASRU-2003, St. Thomas, US Virgin Islands, 2003.

[MES04] Messina, R., and Jouvet D., “Context-dependent long units for speech recognition,” in Proc. ICSLP-2004, pp. 645-648, 2004.

[HAM05] Hamalainen, A., de Veth, J., and Boves, L., “Longer-length acoustic units for continuous speech recognition,” in Proc. EUSIPCO-2005, Antalya, Turkey, 2005.

[HAMA05] Hamalainen, A., Boves, L., and de Veth, J., “Syllable-length acoustic units in large-vocabulary continuous speech recognition,” in SPECOM-2005, pp. 499-502, 2005.

[HAN05] Han, Y., de Veth, J., and Boves, L., “Trajectory Clustering for Automatic Speech Recognition,” in Proc. EUSIPCO-2005, Antalya, Turkey, 2005.

[HANA05] Han, Y., de Veth, J., and Boves, L., “Speech Trajectory Clustering for Improved Speech Recognition,” in Proc. Interspeech-2005, Lisbon, Portugal, 2005.

[CHOU94] Chou, W., Lee, C.-H., and Juang, B.-H., “Minimum error rate training of inter-word context-dependent acoustic model units in speech recognition,” in Proc. ICSLP-94, pp. 439-442, 1994.

[TOD98] Stephenson, T.A., "Speech Recognition using Phonetically Featured Syllables", Centre for Cognitive Science, University of Edinburgh, Master Thesis, 1998

[NEE05] Neel, J., "Cluster analysis methods for speech recognition", Royal Institute of Technology, Master Thesis, Stockholm, 2005

[EST02] Estivill-Castro, V., Why so many clustering algorithms: a position paper. SIGKDD Explorations, 4(1):65{75, 2002.

[HAL02] Halkidi, M., Batistakis, Y, Vazirgiannis, M., "Clustering algorithms and validity measures". IEEE Transactions on pattern analysis and machine intelligence, 24(12), 2002.