FLaVoR: a Flexible Architecture for LVCSR

Kris Demuynck, Tom Laureys, Dirk Van Compernolle and Hugo Van hamme

K.U.Leuven/ESAT/PSI
Kasteelpark Arenberg 10
3001 Leuven (Belgium)

{kris.demuynck,tom.laureys,dirk.vancompernolle,hugo.vanhamme}@esat.kuleuven.ac.be

Abstract

This paper describes a new architecture for large vocabulary continuous speech recognition (LVCSR), which will be developed within the project FLaVoR (Flexible Large Vocabulary Recognition). The proposed architecture abandons the standard all-in-one search strategy with integrated acoustic, lexical and language model information. Instead, a modular framework is proposed which allows for the integration of more complex linguistic components. The search process consists of two layers. First, a pure acoustic-phonemic search generates a dense phoneme network enriched with meta-data. Then, the output of the first layer is used by sophisticated language technology components for word decoding in the second layer. Preliminary experiments prove the feasibility of the approach.

1. Motivation and aims

Current speech recognition research programmes aim at the recognition of unconstrained speech input, higher accuracy, less domain dependency and richer transcription output [1]. Yet, the introduction of powerful techniques necessary to realize these goals is often hampered by the standard, monolithic recognition framework: as all knowledge sources—lexicon, acoustic model and language model—are combined into a single search space, they must be kept extremely simple. This has particularly inhibited progress at the linguistic level. Consequently, almost all recognizers still employ non-optimal linguistic knowledge components such as static lexica (lexicalization of morphological processes) and N-gram language models.

In this paper we deviate from the standard framework and present a novel, flexible speech recognition architecture. We believe that more sophisticated linguistic models are indispensable in order to meet the current challenges in speech recognition. Therefore we opt for a framework which allows for the direct integration of such complex knowledge sources. The development of the architecture and its different components will take place within the project FLaVoR (2002 – 2006).

The key aspect of the proposed framework consists of splitting up the search engine into two separate layers. The first layer performs phoneme recognition and outputs a dense phoneme network, which acts as an interface to the second layer. In this second layer, the actual word decoding is accomplished by means of sophisticated probabilistic morpho-phonological and morpho-syntactic models. These models can be made more complex because the decoupling of the acoustic-phonemic decoding from the word decoding eliminates most of the traditional constraints on them.

The concept of multi-layered recognition as such is not new to this project: it is applied by so-called progressive (multi-pass) search techniques, which target faster recognition. One progressive framework described in the literature uses fast and simple models in a first pass to list word candidates at each time index [2]. On the basis of these lists the search space is confined in subsequent passes. An alternative approach again applies simplified models in a first pass, but records the most probable parts in the search space in word graphs [3]. The following recognition passes are then limited to the strings encoded in the word graphs. It is important to note that the framework presented in this paper is very different from the mentioned progressive techniques. Progressive recognition systems start with simple versions of all knowledge sources and later switch to more complex models. As a result, models with different complexity are needed at each step and crucial information is sometimes lost early in the process. In the proposed architecture, on the other hand, every layer will employ the best models available in order to minimize information loss. Moreover, the acoustic model will be isolated completely from the other knowledge sources. Based on the resulting modularity, we want to achieve more accurate and less domain dependent recognition, probably at the expense of recognition speed.

The incorporation of more accurate linguistic knowledge is not new to speech recognition research either. Especially syntax-based language models have recently been explored [4, 5]. Yet, the sophisticated information has always been applied to the rescoring of word graphs generated by simpler language models. Obviously, the incorrectness and incompleteness of these word graphs cannot be solved by providing complex knowledge sources. In the proposed framework the direct integration of powerful models should again avoid such problems.

This paper is organized as follows. Section 2 compares the traditional recognition framework with the one proposed in this paper. In section 3, we look at the proposed architecture in more detail. First results in phoneme recognition are discussed in section 4. We end with conclusions.

2. Standard vs. FLaVoR architecture

As depicted in the left column of figure 1, standard speech recognition architecture brings in all available knowledge sources very early in the process. More precisely, an all-in-one search strategy is adopted which completely integrates the acoustic model with the linguistic models, optionally followed by rescoring the produced word graphs on the basis of more complex language models. The main advantage of such an approach is the efficiency of the search. Due to the high degree of acoustic confusability in speech, early inclusion of higher level information, as offered by the lexicon and the language model, has proven to be fruitful for reliably pruning away the most unlikely options from the search space.

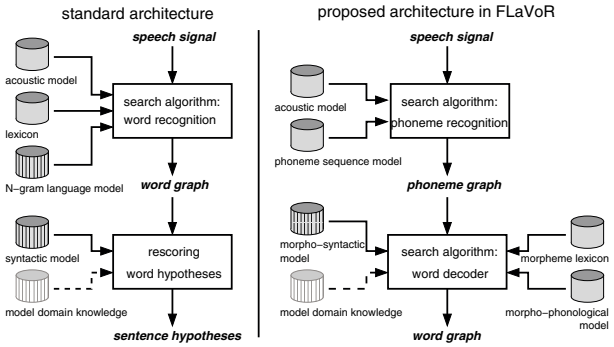

Figure 1: *Standard vs. FLaVoR architecture.*

However, the monolithic search strategy has some important drawbacks as well. First, the simultaneous application of all knowledge components obviously results in a huge search problem which requires very fast evaluation of the models. Consequently, only extremely simple models can be used (e.g. N-grams, lexicalization) which involve crude approximations. Second, the left-to-right operation of the acoustic model requires the other knowledge sources to work from left to right as well. This is often inefficient for decisions which (partially) depend on a right context, for example in the language model. Third, the standard framework complicates the inclusion of generic information. For example, the required simplicity of the models does not allow for a generic approach to morphology in the lexicon. As a result, building applications for a new knowledge domain comes down to starting almost from scratch. Fourth, when targeting unconstrained speech input, i.e. when working with large (possibly unlimited) lexica and flexible language models for handling spontaneous speech phenomena, the impact of lexicon and language model on the acoustic search diminishes. So, judging by the current trends in recognition, it seems that the disadvantages of a monolithic recognition strategy outweigh its advantages.

We claim that the mentioned research challenges can be handled more efficiently by layered recognition. A simplified sketch of the recognition architecture proposed in this paper can be seen in the right column of figure 1. The acoustic-phonemic decoding is clearly separated from the word decoding. As a result, fewer constraints are imposed upon the linguistic models in the word decoder. A detailed discussion of the proposed architecture and its expected benefits follows in the next section.

3. The FLaVoR architecture in detail

3.1. First layer: acoustic-phonemic decoding

A detailed description of the FLaVoR architecture can be found in figure 2. In the first layer, a search algorithm determines the network of most probable phoneme strings F given the acoustic features X of the incoming signal. The knowledge sources employed are an acoustic model $\mathrm{p}(X|F)$ and a phoneme transition model $\mathrm{p}(F)$. The resulting phoneme network is enriched with meta-data (prosody, speaker identity, etc.) in order to restrict the loss of information as much as possible. In section 4 we will demonstrate that accurate pure acoustic-phonemic decoding is possible.

Important to note is the isolation of the low-level acoustic-phonemic search, with a dense phoneme network to the higher layers. This decoupling is made possible by the high quality of

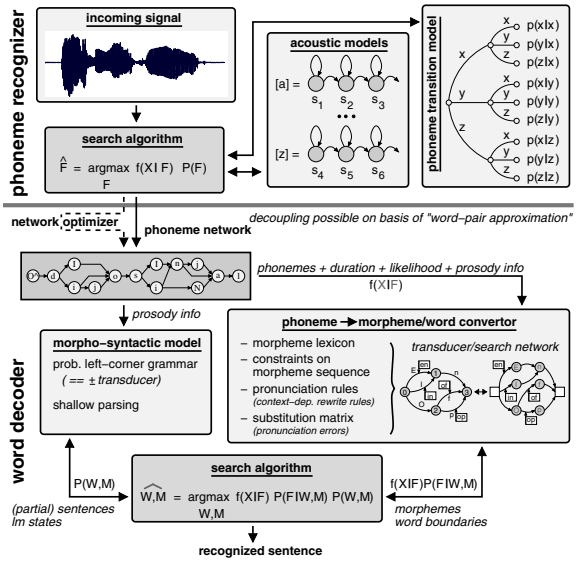

Figure 2: *FLaVoR architecture in detail.*

current acoustic modelling and by an extension of the so-called word-pair approximation [6]. For a description of this extension we refer to section 4.

The decoupling of acoustic and word decoding not only eliminates the left-to-right operation constraint on linguistic models, but it also significantly lowers the event rate. First, there is a pure reduction in data rate, from the 100 feature vectors per second at the input to an average of 12 phones (plus alternatives) per second at the output. In addition, the number of parallel options each input stands for is reduced. While a monolithic search engine has to match all incoming feature vectors with all possible combinations of phonemes and end positions at that point in the search, the phoneme network will only contain the set of best matching phonemes with their optimal start and end times. This opens up the possibility of using more complex modelling strategies in the word decoder (layer 2).

Another important aspect is the generic nature of the first layer for a full natural language. That is, the phoneme recognizer can function in any knowledge domain for a specific language. In addition, the phoneme information itself could be used in certain applications (e.g. language learning) or for handling specific problems (e.g. recognition of proper names).

3.2. Second layer: word decoding

The phoneme network and associated meta-data serve as input to the second layer which performs the actual word decoding. The search algorithm has two knowledge sources at its disposal: a morpho-phonological and a morpho-syntactic component.

The morpho-phonological component converts the phoneme network into sequences of morphemes and hypothesizes word boundaries. The morpho-phonological knowledge consists of a morpheme lexicon, constraints on morpheme sequences, pronunciation rules and a substitution matrix. All knowledge sources will be combined into one finite-state transducer or a search network. As illustrated in the work by AT&T [7] and ourselves [8], such transducers are a very compact and efficient solution for decoding.

The morpheme lexicon consists of a set of both bound (affixes) and unbound (roots) morphemes, with a phonemic tran-

scription for each lemma. The constraints on morpheme sequences determine which morphological positional slot can be filled by which positional morpheme. These constraints will still be quite general. For example, a prefix should attach at the front of a root, etc. The pronunciation rules qualitatively and quantitatively describe the contextual influence on the pronunciation of a sequence of phonemes. More precisely, probabilistic context-dependent rewrite rules will formalize the processes of assimilation, insertion, etc. on the intra- as well as the inter-word level. As such these rules provide the indispensable link between the isolated standard phonemic transcription of morphemes (in the morpheme lexicon) and their realization in 'real-life speech'. Finally, the substitution matrix copes with non-regular pronunciation variants such as dialectical influence, swallowed sounds in fast pronunciation and slips of the tongue.

The morpho-syntactic language model will provide a probability measure for each hypothesized word based on morphological and syntactic information of the word and its context. A fine-grained, hierarchical, probabilistic morphological analysis (covering flexion, derivation and compounding) is provided for each input word. This morphological information is then integrated into a syntax-based language model. Two different syntax-based approaches will be developed: one based on full-sentence left-corner parsing [5], the other based on shallow parsing techniques [9].

3.3. Advantages of the proposed architecture

The advantages of the proposed layered architecture are many. We will highlight the use of a dynamic lexicon, the integration of better and more generic linguistic models, the higher degree of modularity and the improved richness of output.

The probabilistic description of regular morphological processes leads to a dynamic lexicon. Standard large-vocabulary recognizers typically use a static lexicon of 40K up to 65K word forms. Inclusion of more words is often technically impossible, brings about a significant increase in processing time or introduces many words for which the language model probability cannot be reliably estimated. As people are 'creative wordsmiths' this limitation leads to a relatively high number of Out-Of-Vocabulary (OOV) words. Moreover, OOV words have a particularly negative impact on the recognition accuracy, as they often cause their neighbouring words to be misrecognized as well [10]. This problem is all the more important when unconstrained speech input is targeted. By modelling the regular morphological processes rather than listing lexicalized word forms, lexicons can be kept concise and at the same time a large part of OOV cases can be dealt with. Obviously, especially morphologically rich languages (German, Dutch, Turkish) will benefit from the adoption of a dynamic lexicon.

The linguistic knowledge sources incorporated into the proposed architecture are complex, accurate and highly generic. Experiments with syntax-based language models have already proven their superiority to N-gram modelling techniques [4, 5]. By introducing morphological knowledge we believe that their accuracy can even be increased. In addition, the morphological information allows for better generalization. Whereas lexicalization makes the standard language model blind to semantic dependencies (e.g. between 'high', higher' and 'highest'), the morphological analyses do reveal this information and can therefore provide better statistical estimates for unseen events. Finally, constraints on morphology and syntax apply to the language as a whole. Our morpho-syntactic language model therefore holds promise as a generic, reusable knowledge source in speech recognition architectures.

As the proposed framework is highly modular, the different knowledge sources no longer put heavy constraints on each other (e.g. left-to-right operation, complexity). Consequently, new models could be plugged in more easily, irrespective of their structure and complexity.

Finally, richer transcription output comes almost for free with the layered architecture. The acoustic decoding generates the network of phonemes with their durations. The morpho-syntactic component gives information on the structure within and among words. Lexical stress, phrase accent and prosodic boundaries will be predicted by the prosodic patterns. All this information can be applied by subsequent modules in, for example, language learning applications, post-synchronization tools or dialogue systems.

4. Phoneme recognition results

The key prerequisite for the proposed framework is the generation of high quality phoneme lattices in the acoustic-phonemic layer. The quality of phoneme lattices is defined by both a low phoneme error rate and a low event rate or density. In this section, we investigate whether this prerequisite can be met with current state-of-the-art HMM technology.

We chose to perform the experiments on the Wall Street Journal (WSJ) test suite. Since we have worked with this test suite extensively, we have excellent acoustic models for the task. In addition, there are plenty of WSJ reference results—from our own and other recognizers—available for comparison. However, in the context of evaluating phoneme lattices the WSJ database poses one major problem: it does not include reference phoneme transcriptions.

We created reference phoneme transcriptions by converting the orthograpic transcription of each sentence into a phoneme network by means of a pronouncing dictionary and assimilation rules, and used the Viterbi algorithm to decide on the best path through the network given the acoustic signal. As shown in [11], for dictated speech this approach delivers phonemic transcriptions of a quality close to that of manually labeled data when an accurate orthography is available. In this setup, we limited the assimiliation rules to the most important ones: degemination and (de)voicing assimilation.

Phoneme decoding was based on the architecture described in [8]. The two main topics to be addressed when using this architecture as a pure acoustic-phonemic decoder, are the extension of the word-pair approximation to the phoneme level and the overall quality of the output.

Viterbi decoding only optimizes the scores (and indirectly the phoneme boundaries) of the best hypothesis (phoneme sequence): for efficiency reasons, Viterbi decoding blocks (recombines) search paths from the very moment the path is known to be sub-optimal given the set of first order Markov knowledge sources (HMM, lexicon network, N-gram). The principle behind word or phoneme lattices is to record all 'interesting' paths that were blocked. A possible issue is the fact that the word (or phoneme) boundaries may not yet be optimal. However, in practice it was observed that if search paths are prevented from being recombined for at least one word, e.g. when using a bigram language model, the vast majority of the boundaries between two words A and B are already optimal when the search engine reaches the end of word B [6]. For phoneme recognition this horizon of 'at least one word' has to be mapped to a certain number of phonemes. Hence, to obtain high quality phoneme lattices, a phoneme transition model (an N-gram be-

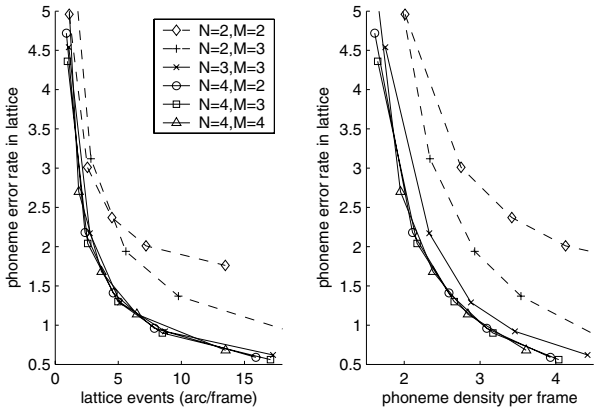

Figure 3: *Summary of phoneme recognition results.*

tween phonemes) of a sufficiently high order N has to be used, or the LM-context has to be artificially expanded to $(M-1)$ phonemes. Note that values of $M \leq N$ are useful as well since typical N-grams are not full, i.e. certain events are only modelled with lower order n-grams. The role of M in this context, is to artificially prevent the Viterbi search from recombining search paths based on this lower order n-grams when $n < M$.

The acoustic models used are context-dependent (CD) tied state phoneme models (2966 states) with mixtures of tied gaussians (64080 gaussians) as observation density functions. These models result in a 7.7% WER on the Nov92 20K nvp open vocabulary test set using the standard trigram. The phoneme N-grams were estimated on the acoustic training material in the WSJ database (automatic phonemic transcription, cf. supra). Figure 3 shows the results of phoneme recognition experiments with different values of N and M and with different lattice densities. The values on the ordinate are the phoneme error rates (ins. + del. + sub.) of the phoneme lattice, i.e. the error rate of that path which matches best with the reference transcription. The values on the abscissa of the left plot are the average number of events (an arc representing a CD-phoneme) per frame in the lattice. The values on the abscissa of the right plot are the average number of different phonemes (ignoring the context) in parallel per frame in the lattice.

Since at a certain point lower lattice error rates no longer improve the accuracy of the total system, assessing whether a lattice phoneme error is acceptable would already require a complete system. Yet, when regarding our standard word-based recognizer as a refined phoneme recognizer with lexicon, word N-gram and assimilation rules to predict the next phoneme, we obtain a phoneme error rate of 2.7%. Hence, when building the second layer in our new approach, we should at least start from a lattice error rate of 2.7% or lower to rival with the performance of the standard recognizer. As figure 3 shows, this point is already reached with an event rate less than 2.

Further, the difference between the N=2,M=2 and N=2,M=3 curves clearly demonstates that a one phoneme horizon is insufficient. We also see that a 3-gram transition model is to be preferred to a 2-gram model as it leads to substantially lower event rates. The gain obtained from going to a 4-gram is too small considering the specificity of high order N-grams (modelling task-specific words).

5. Conclusions

We presented a flexible, layered architecture for LVCSR which allows for an easier integration of knowledge sources. The architecture largely hinges on a highly accurate pure acoustic-phonemic decoding, the feasibility of which was proved in experiments. In the future, the described linguistic components will be developed within the project FLaVoR. The target language will shift to Dutch as that language is morphologically considerably richer than English.

6. Acknowledgements

The research reported in this paper was funded by IWT in the GBOU programme, project FLaVoR (Project number 020192). http://www.esat.kuleuven.ac.be/∼spch/projects/FLaVoR.

7. References

[1] C.L. Wayne, "BAA #02-06: Effective, Affordable, Reusable Speech-to-text (EARS - DARPA/ITO)," 2002.

[2] L. Nguyen and R. Schwartz, "The BBN single-phonetic-tree fast-match algorithm," in *Proc. ICSLP*, Sydney, Australia, Nov. 1998, vol. V, pp. 1827–1830.

[3] H. Murveit, J. Butzberger, V. Digalakis, and M. Weintraub, "Large-vocabulary dictation using SRI's DECIPHER$^{\text{tm}}$ speech recognition system," in *Proc. ICASSP*, Minneapolis, U.S.A., Apr. 1993, vol. II, pp. 319–322.

[4] C. Chelba and F. Jelinek, "Structured language modeling," *Comp. Speech and Lang.*, vol. 14, no. 4, pp. 283–332, 2000.

[5] D.H. Van Uytsel, D. Van Compernolle, and P. Wambacq, "Maximum-likelihood training of the PLCG-based language model," in *Proc. ASRU 2001*, Madonna di Campiglio, Italy, Dec. 2001, 4 pages.

[6] H. Ney and X. Aubert, "Dynamic programming search strategies: From digit strings to large vocabulary word graphs," in *Automatic Speech and Speaker Recognition*, C. Lee, F.K. Soong, and K.K. Paliwal, Eds., pp. 385–411. Kluwer Academic Publishers, 1996.

[7] M. Mohri, "Finite-state transducers in language and speech processing," *Comp. Ling.*, vol. 23, no. 2, pp. 269–311, 1997.

[8] K. Demuynck, J. Duchateau, and D. Van Compernolle, "A static lexicon network representation for cross-word context dependent phones," in *Proc. EUROSPEECH*, Rhodos, Greece, Sept. 1997, vol. I, pp. 143–146.

[9] W. Daelemans, S. Buchholz, and J. Veenstra, "Memory-based shallow parsing," in *Proc. CoNLL*, Bergen, Norway, June 1999.

[10] M. Adda-Decker and L. Lamel, "The use of lexica in automatic speech recognition," in *Lexicon Development for Speech and Language Processing*, Frank Van Eynde and Dafydd Gibbon, Eds., Text, Speech and Language Technology, pp. 235–266. Kluwer Academic Publishers, 2000.

[11] K. Demuynck, T. Laureys, and S. Gillis, "Automatic generation of phonetic transcriptions for large speech corpora," in *Proc. ICSLP*, Denver, U.S.A., Sept. 2002, vol. I, pp. 333–336.