USING EM-TRAINED STRING-EDIT DISTANCES FOR APPROXIMATE MATCHING OF ACOUSTIC MORPHEMES

Michael Levit, Elmar Nöth

University Erlangen-Nuremberg, Germany

{levit,noeth}@informatik.uni-erlangen.de

Allen Gorin

AT&T Laboratories-Research

algor@research.att.com

ABSTRACT

Our research concerns spoken language understanding within the domain of automated telecommunication services. In the recent papers we presented a new methodology for training of statistical language models for recognition and understanding of utterances from large corpora of phone sequences obtained as the output of a task-independent ASR-system. The advantage of this strategy compared to the traditional word-based strategy is that we don't have to manually transcribe large amounts of data in order to extract acoustic morphemes to train the classifier. Since the baseline strategy suffered high False Rejection Rates caused by finding no acoustic morphemes in the test data, we describe in this paper how approximate matching can be incorporated in the Bayes-classifier to reduce FRR. The experiments are evaluated for *"How May I Help You?"*-task.

1. INTRODUCTION

The subject of our research is machine understanding of spoken natural language. The popular methodology in this field is a word-based training which requires training corpora annotated at the word level. Since annotation of large amounts of speech data is time consuming and expensive, we suggested in [4] an understanding system that acquires lexicon, syntax and semantics from untranscribed speech. In particular, our strategy makes use of clusters of semantically meaningful (*salient*) phone sequences, which we call *acoustic morphemes*, for classification of utterances. The representations of the utterances at the phone level are obtained as the output of a task-independent phone recognizer [7].

We evaluate our algorithms for the *"How May I Help You?"* (HMIHY) task [3], where an automated dialogue system is designed to infer appropriate machine actions upon the service requests made over the phone by non-expert users. Elicited by an open-end prompt *"How May I Help You?"*, these requests are made in form of natural language utterances and are to be categorized into one of 15 known call-types including an open-class denoted "OTHER".

The classification of utterances is made based upon semantic associations of acoustic morphemes encountered in it. In [5] we described our approach for extraction of salient phone phrases from a training corpus. Acoustic morphemes are then obtained as FSM-representations of clusters of acoustically and semantically similar salient phone phrases. In this paper we describe the application of Bayes-classifier for the call-classification task, whereby our attention is focused on the issues of approximate matching and EM-estimation of string-edit distances used hereby.

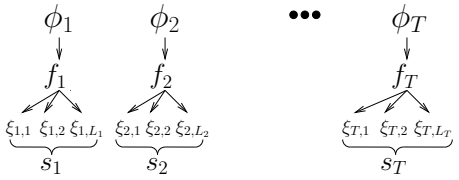

Fig. 1. Approximate fragment instantiation as three-step production mechanism.

2. UTTERANCE GENERATION AS PRODUCTION MECHANISM AND STOCHASTIC PROCESS

In this section we derive a statistical formulation for the task of searching for acoustic morphemes in the test utterances. Our goal is to find approximate instances of acoustic morphemes in the output of phone recognizer. Let us first assume that each utterance is a linear sequence (*string*) of phones $S = \xi_1 \dots \xi_L$. Then, if we introduce the term *"fragment"* subsuming acoustic morphemes and phones, the process of utterance generation can be considered as a three-step production mechanism (Figure 1). The source emits a sequence Φ of fragments $\phi_t,\ t = 1 \dots T$, each of the latter gets instantiated as s_t, a subsequence of S, which is hypothesized to arise due to distortion of some path f_t through ϕ_t (at the moment, f_t are just phone phrases that make up this fragment). The observed string S can thus be represented as a sequence of T fragment instances $s_1 \dots s_T$, originating from the sequence F of fragment paths f_t. The inferring of the underlying sequence F from S is an *approximate matching*-problem; it will be addressed in the next sections.

The generation process can also be formulated as a triple stochastic process $(\boldsymbol{\phi}, \boldsymbol{f}, \boldsymbol{s})$ with the only observed variable $\boldsymbol{s}$. If we assume *statistical independence* of generated fragments, the joint likelihood of Φ, F and S can be decomposed:

$$P(\Phi, F, S) = \prod_t P(\phi_t, f_t, s_t) = \prod_t P(\phi_t)P(f_t|\phi_t)P(s_t|f_t, \phi_t) = \prod_t P(\phi_t)P(f_t|\phi_t)P(s_t|f_t) \quad (1)$$

where the last equality holds since possible distortions of paths f_t are independent of the fragment this path is taken from.

The foregoing was based on the assumption that the phone string S is a perfect reflection of acoustic observations. However, the string itself is just an interpretation hypothesis at the phone level for the actually observed acoustic signal $O = \omega_1 \dots \omega_N$, and it can be provided with an acoustic score $P(O|S)$, which is composed of fitness scores $P(o_l|\xi_l)$ of individual phones $\xi_l \in S$

with respect to o_l, corresponding subsequences of O. In fact, a typical phone recognizer would output a lattice (weighted acyclic graph), representing multiple concurrent phone-hypotheses that can account for O. The arcs of such a graph σ (one arc represents one phone in the hypothesis) are weighted according to the fitness score. It is crucial for understanding that all information about conditional probabilities $P(O|S)$ is contained in the graph σ, so that we can use simpler notations: $P_\sigma(S)$ and (for substrings of S) $P_\sigma(s_t)$.

In this case we have to go one step further and extend our production model by an additional level of acoustic observations. Then stochastic process becomes $(\boldsymbol{\phi}, \boldsymbol{f}, \boldsymbol{s}, \boldsymbol{o})$, and joint likelihood (1) turns into:

$$P_\sigma(\Phi, F, S) = \prod\nolimits_t P(\phi_t)P(f_t|\phi_t)P(s_t|f_t)P_\sigma(s_t). \qquad (2)$$

Conditional distribution $P(f|\phi)$ is an intrinsic characteristic of each fragment ϕ, which can reflect occurrence statistics of the linear phone phrases making up this fragment or/and their salience. In the HMIHY-framework acoustic morphemes are created not only to compensate for possible mistakes of the phone recognizer, but can also represent unions of *distinct* phrases having similar acoustic and semantic characteristics. This is why we decided to abandon the stochastic condition for these probabilities: $\sum_{f\in \mathrm{FP}(\phi)} P(f|\phi) \equiv 1$, where $\mathrm{FP}(\phi)$ is the set of paths through ϕ. The new measure $\tilde{P}(f|\phi)$ which we suggest to choose from the interval $[0.5, 1]$ and interpret as a *representativeness* of the phrase in the fragment, will be also referred to as *score* of the corresponding path through the fragment. Now, we can reduce the degree of freedom of our production system in such a way that if it is possible for string s_t to be instantiation of ϕ_t at all, then it is definitely the instantiation of that path $f = f_{\phi_t}^{s_t}$ through ϕ_t, for which the product $\tilde{P}(f|\phi_t)P(s_t|f)$ is maximal. Since the permissible sequence F_Φ^S of taken paths through fragments is now uniquely determined by the sequences Φ and S, we finally obtain stochastic process $(\boldsymbol{\phi}, \boldsymbol{s}, \boldsymbol{o})$, such that we can rewrite joint likelihood (2) as:

$$P_\sigma(\Phi, S) = \prod\nolimits_t P(\phi_t)\tilde{P}(f_{\phi_t}^{s_t}|\phi_t)P(s_t|f_{\phi_t}^{s_t})P_\sigma(s_t).$$

The classification task implies that for each output lattice σ from the phone recognizer, likelihoods of all found instances of acoustic morphemes must be estimated. The easiest way to do this is to consider the *best parse*, i.e. the pair of sequences Φ, S with the highest likelihood, given graph of acoustical observations σ:

$$(\Phi^*, S^*) = \mathrm{argmax}_{(\Phi,S)}\, P_\sigma(\Phi, S),$$

and to accept all instances of fragments occurred in it for further processing in the classifier. For example, if for some t ϕ_t is fragment ϕ from the lexicon, then we proclaim to have found instance of fragment ϕ in s_t with likelihood:

$$P_\sigma(\phi, s_t) = P(\phi)P(f_\phi^{s_t}|\phi)P(s_t|f_\phi^{s_t})P_\sigma(s_t).$$

3. APPROXIMATE STRING MATCHING

Let us now turn to the question, how to calculate the probabilities $P(s_t|f_{\phi^k}^{s_t})$? Both s_t and $f_{\phi^k}^{s_t}$ are linear phone strings, so that we have to solve the problem of estimating likelihood of observed string of tokens $s = \xi_1 \dots \xi_K$ if the source actually emitted string $r = \zeta_1 \dots \zeta_M$. We can approximate $P(s|r)$ as the probability of token-wise (without context consideration) transformation r into s. To illustrate this, let us build a model of a noisy channel which takes sequences of tokens from the input stream and sends distorted sequences to the output stream. Thereby, the transmission operations (*mappings*) that can take place within the channel shall be restricted to the following four types:

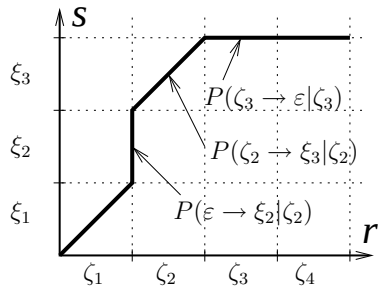

Fig. 2. Computing probability of string transformation with DP-algorithm.

1. *identity mapping*: the input token is correctly forwarded to the output (noisefree transmission);
2. *substitution*: token x is absorbed from the input stream and token $y \neq x$ is sent to the output stream;
3. *insertion*: given the next token x coming from the input stream, the channel keeps it there and emits token y;
4. *deletion*: the next token from the input stream x is absorbed without resulting in any output token.

Here x is from the alphabet L_r of input strings and y is from the alphabet L_s of output strings; furthermore, not absorbing or not writing any tokens can be interpreted as absorbing or writing *empty token* ε respectively, so that now we can express all mappings in the form $x \to y|x$ and $\varepsilon \to y|x$, with next input token $x \in L_r$ and next output token $y \in L_s \cup \{\varepsilon\}$.

The mistakes represented by the last three types of mappings are typical for a phone recognizer, whereas other applications may have additional types of noise (e.g. swaps in typing). All mappings are provided with probabilities which are sufficient to describe the channel properties. Given these probabilities, we can avail ourselves of the Viterbi-approximation to obtain probability $P(s|r)$. Ignoring context dependency we write:

$$P(s|r) = \prod\nolimits_{m_i} P(m_i)$$

with mappings m_i from the cheapest sequence of mappings transforming r into s. This sequence can be determined with DP-algorithm (see Figure 2). In the illustrated example the probability of the chosen transformation can be calculated using the formula: $P(s|r) = P(\zeta_1 \to \xi_1|\zeta_1)P(\varepsilon \to \xi_2|\zeta_2)P(\zeta_2 \to \xi_3|\zeta_2)P(\zeta_3 \to \varepsilon|\zeta_3)P(\zeta_4 \to \varepsilon|\zeta_3)$. We explain now how to obtain probabilities of mappings $x \to y|x$ and $\varepsilon \to y|x$ with $x \in L,\ y \in L \cup \{\varepsilon\}$, that is, probabilities of observing token y at the sink of the noisy channel given that the next input token is x. The probabilities will be inferred from two corpora: undisturbed linear input sequences (phone transcriptions) and the corresponding sequences observed at the sink of the channel (output of phone recognizer).

The solution is based on the algorithm presented in [8]. This algorithm makes use of EM-framework [2] to estimate probabilities of the *elementary edit operations* on tokens. It furthermore acts on the assumption that at any moment each edit operation

is possible and the probability of seeing a particular edit operation doesn't change over time. This allows for a description in the form of a simple memoryless "flower"-transducer accounting for all possible edit operations $z \in Z : L \cup \{\varepsilon\} \to L \cup \{\varepsilon\}$ (identities, deletions, insertions and substitutions)[1]. The probabilities $\delta(z)$ of the edit operations are then iteratively refined. For details and discussion see [1, 8].

These probabilities obey the following statistic condition[2]: $\sum_z \delta(z) \equiv 1$. Employing our noisy channel analogy, we can now take advantage of the fact that at each time point there is exactly one pending token in the input stream, and go over to conditional probabilities. Suppose that we have determined the probabilities $\delta(z)\ \forall(z = x \to y) \in Z$, then conditional probability of edit operation $w \to y$ given that the next input token is $a \neq \varepsilon$ (we call such operations "mappings") can be computed according to the following formulae:

$$P(w \to y|a) = \begin{cases} \frac{\delta(a \to y)}{\sum_x \delta(a \to x)}(1 - \sum_x \delta(\varepsilon \to x)), & \text{if } w = a; \\ \delta(\varepsilon \to y), & \text{if } w = \varepsilon; \\ 0, & \text{otherwise.} \end{cases}$$

The derivation becomes easily comprehensible as soon as one thinks of the process of transformation of one string into another as a dual stochastic process in which we first make a binary decision whether the channel will absorb the pending token from the input stream, and the second decision to make is concerned with the token generated at the sink of the channel.

4. BAYES-CLASSIFIER FOR THE TASK OF CALL-CLASSIFICATION

In this section we show how a simple Bayes-classifier can be constructed for the call-classification task, when the statistical concepts described in the foregoing sections are used. Let $\{M^k, k = 1 \dots K\}$ be the set of semantic categories in the task. To classify phone sequence S we consider posterior probabilities of all categories given this sentence and choose the one with maximal value

$$P(M^k|S) \simeq P(S|M^k)P(M^k). \tag{3}$$

Using Viterbi approximation, we replace $P(S|M^i)$ in (3) by conditional joint probability of the best parse $(\Phi^{(S,k)}, S)$ given M^i:

$$\begin{aligned} & P(S|M^k)P(M^k) \approx \max_{\Phi} P(S, \Phi|M^k)P(M^k) \\ := \quad & P(S, \Phi^{(S,k)}|M^k)P(M^k) \\ = \quad & P(S|\Phi^{(S,k)}, M^k)P(\Phi^{(S,k)}|M^k)P(M^k) \\ = \quad & P(S|\Phi^{(S,k)})P(M^k|\Phi^{(S,k)})P(\Phi^{(S,k)}), \end{aligned}$$

where the last equality is due to Bayes-rule for $P(\Phi^{(S,k)}|M^k)$ and to the fact that the first probability term describes characteristics of the noisy chanel modeling intrinsic characteristics of the employed phone recognizer (Section 3) and is thus independent from task semantics. Assuming statistical independence of fragments which constitute sequence $\Phi^{(S,k)} = \phi_1^{(S,k)} \dots \phi_T^{(S,k)}$ we can decompose this formula with the chain rule into:

$$\prod_t P(s_t|\phi_t^{(S,k)})P(M^k|\phi_t^{(S,k)})P(\phi_t^{(S,k)}). \tag{4}$$

Tieing up to discussion in Section 2, we rewrite it in terms of the most likely paths $f_t^{(S,k)}$ through fragments $\phi_t^{(S,k)}$, given the parts of S they account for. Besides, if phone recognizer produces genuine phone lattices σ and not linear phone strings, we extend the formula by acoustic score of recognized phone strings $S \in \sigma$. Then (4) becomes:

$$\prod_t P_\sigma(s_t)P(s_t|f_t^{(S,k)})P(f_t^{(S,k)}|\phi_t^{(S,k)})P(M^k|\phi_t^{(S,k)})P(\phi_t^{(S,k)}).$$

Probabilities $P(\phi)$ and $P(M^k|\phi)$ can be Maximum-Likelihood-estimated from the training corpus. Instead of probabilities we can also compare costs: $C(\cdot) = -\log P(\cdot)$. Then the utterance represented by graph σ will be mapped by the classifier into the category with the smallest cost:

$$\begin{aligned} M^* \quad = \quad & \operatorname*{argmin}_{M^k} \sum_t \Big(C_\sigma(s_t) + C(s_t|f_t^{(S,k)}) \\ + \quad & C(f_t^{(S,k)}|\phi_t^{(S,k)}) + C(M^k|\phi_t^{(S,k)}) + C(\phi_t^{(S,k)}) \Big) \end{aligned}$$

In our experiments we use *Final State Machines* (FSM) to implement the classifier. Let FSM $\mathcal{S}$ represent phone lattice σ. With FSM-operations: $\cup$ (*union*), $\circ$ (*composition*), $\cdot$ (*concatenation*), $(\cdot)^*$ (*concatenative closure*) and *bestpath*, the decision rule is carried out by comparison of costs of FSMs $\mathcal{B}^k = \textit{bestpath}(\mathcal{F}^k \circ \mathcal{S})$ for each category M^k, where

$$\mathcal{F}^k = (\mathcal{W}^k \cup \bigcup\nolimits_j (\mathcal{I}^{\varphi_j} \cdot (\varphi_j^k \circ \mathcal{W}) \cdot \mathcal{O}^{\varphi_j}))^*. \tag{5}$$

Here following notations are used (see Figure 3): $\mathcal{W}$ is the *"flower"*-transducer reflecting channel characteristics, $\mathcal{W}^k$ accounts additionally for priors of the phones and probabilities of M^k given these phones, φ_j^k is an FSM representing acoustic morpheme ϕ_j along with its prior and probability $P(M^k|\phi_j)$, and $\mathcal{I}^{\varphi_j}$ and $\mathcal{O}^{\varphi_j}$ are special "parentheses" FSMs, which will indicate begin and end of acoustic morphemes in the sequence of labels along the best path. In general case, φ_j^k has arbitrary arc and final costs, whereby it is convenient to think of the arc cost as reflection of occurrence statistics of linear salient phone phrases which constitute ϕ_j, and of final costs as containing priors and semantic probabilities. $\mathcal{S}$ can also be an arbitrary acyclic weighted graph. We also recommend to use only those acoustic morphemes to compute the optimal cost of the utterance for given category, which are significantly salient for this category.

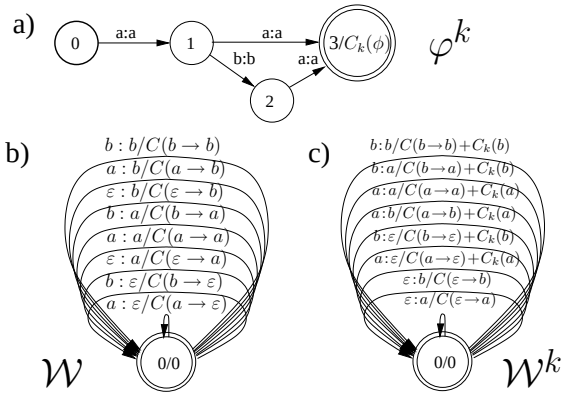

Fig. 3. FSM-transducers used in formula (5): a) FSM for acoustic morpheme ϕ with final cost $C_k(\phi) = C(\phi) + C(M^k|\phi)$ b) flower FSM representing noisy channel c) flower FSM with additional cost for each phone $C_k(x) = C(x) + C(M^k|x)$.

[1] See Section 5.

[2] Additionally we postulate: $\delta(\varepsilon \to \varepsilon) \equiv 0$

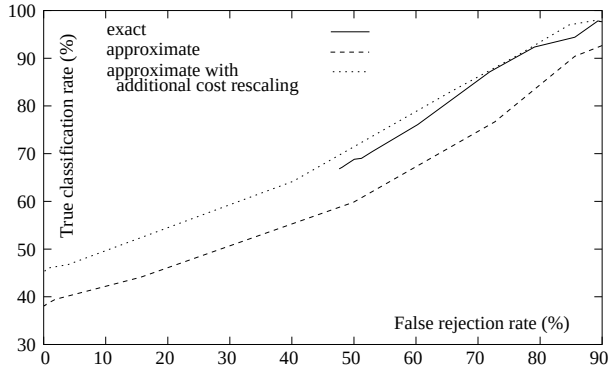

Fig. 4. Effect of approximate matching by call-classification

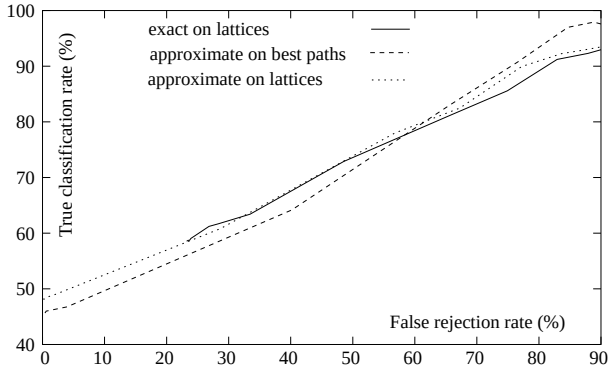

Fig. 5. Classification with lattices and approximate matching

5. EXPERIMENT

We conducted our classification experiments on the HMIHY-database, a collection of recordings of callers responding to the prompt *"AT&T. How may I help you?"* [3]. There are 7642 and 1000 sentences in our training and test sets respectively. For the classification experiment FSM-toolkit [6] was used.

Our experiments evaluate the utility of approximate matching (AM) on the stage of classification. To do this, we decided to consider a simple Bayes-classifier (as opposed to experiments described in our previous publications, where a more sophisticated classifier was used). Figure 4 shows ROC-curves (rank 1) for classification on best paths through lattices produced by the ASR from test utterances, where a) AM was not allowed at all, b) AM was used with mapping costs optimized as described in Section 3 and c) EM-estimated mapping costs were additionally rescaled: costs of identity mappings were all set to zero, and the rest was multiplied by factor $\lambda \in [2; 4]$ (in the shown plot $\lambda = 3$ is used).

We see that direct use of AM impairs true classification rates (TCR) of the recognizer while making classification with low false rejection rates (FRR) possible. The latter effect is due to the fact that it is possible to instantiate anything anywhere when finite costs for all phone mappings are given. As for the decreasing TCR, we explain it by noting that the estimation of mapping costs has been done without consideration of task semantics. With the costs of mapping altered as shown in plot c) however, it was possible to optimize performance of the AM-classifier so that it produced TCR surpassing baseline measures by 3 percent points by same FRR values. The maximal achievable percentage of correctly classified and correctly rejected utterances grew from 42.8% to 46.9% and was reached with no false rejections.

In [5] we had already reported on classification experiments on lattices instead of best paths. Figure 5 compares ROC-curves achieved with the simple Bayes-classifier described above for exact instantiation on 100-best lattices and for AM on the best paths.

As expected, lattices still reveal superior performance even though they were not able to operate in the area of low FRRs. However, if we combine both approaches and look for approximate instantiations of acoustic morphemes in lattices, the result appears to combine advantages of both strategies. The maximal achievable percentage of correctly classified or rejected utterances rises to 49.2% with no false rejections.

6. CONCLUSION

Our experiments showed that the straightforward application of approximate matching for call-type classification can be used if one would like to keep the false rejection rate as small as possible. To achieve improvement of true classification rates, additional rescaling of costs of phone mappings is needed. The classification on phone lattices remains superior compared to classification on best paths using approximate matching of acoustic morphemes, however the combination of both strategies allows for operating in the area of low FRRs and improves TCR at the same time.

7. REFERENCES

[1] Bahl R. L. and Jelinek F.: *Decoding with channels with insertions, deletions and substitutions with applications to speech recognition*. IEEE Trans. Information Theory, IT-21:404-411 (1975).

[2] Dempster A.P, Laird N.M and Rubin D.B.: *Maximum Likelihood from Incomplete Data via the EM Algorithm*. Journal of the Royal Statistical Society, Series B (Methodological), 39(1):1–38, 1977.

[3] Gorin A. L., Riccardi G. and Wright J. H.: *How may I help you?*. Speech Communication 23, pp. 113–127, 1997.

[4] Gorin A. L., Petrovska-Delacrétaz D., Riccardi G. and Wright J. H.: *Learning Spoken Language without Transcriptions*. IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU'99, Colorado, USA, Dec. 1999.

[5] Levit M., Gorin A. L. and Wright J. H.: *Multipass algorithm for acquisition of salient acoustic morphemes*. Proc. Eurospeech, Aalborg, Denmark, Sept. 2001

[6] Mohri M., Pereira F. and Riley M.: *FSM Library — general-purpose finite-state machine software tools.* http://www.research.att.com/sw/tools/fsm/

[7] Riccardi G.: *On-line Learning of Acoustic and Lexical Units for Domain-Independent ASR*. 6th International Conference on Spoken Language Processing, ICSLP'2000, Beijing, China, Oct. 2000.

[8] Ristad E. S. and Yianilos P. N.: *Learning string edit distance*. In Machine Learning: Proceedings of the 14th International Conference (San Francisco, July 8–11 1997), D. Fisher, Ed., Morgan Kaufmann, pp. 287–295.