INFERRING LINGUISTIC STRUCTURE IN SPOKEN LANGUAGE

M. Woszczyna *A. Waibel*

Carnegie Mellon University — USA
University of Karlsruhe — Germany

ABSTRACT

We demonstrate the applications of Markov Chains and HMMs to modeling of the underlying structure in spontaneous spoken language. Experiments with supervised training cover the detection of the current dialog state and identification of the speech act as used by the speech translation component in our JANUS Speech-to-Speech Translation System. HMM training with hidden states is used to uncover other levels of structure in the task. The possible use of the model for perplexity reduction in a continuous speech recognition system is also demonstrated. To achieve improvement over a state independent bigram language model, great care must be taken to keep the number of model parameters small in the face of limited amounts of training data from transcribed spontaneous speech.

1. INTRODUCTION

In spoken language understanding productive interpretation of an utterance has to incorporate the underlying linguistic structure in a dialog. This structure comprises the current topic, discourse state, speech act, and common phrases. It has been shown [4] that if topic or speech act is known, the perplexity of the task can be reduced significantly. Furthermore a good estimate of the most likely speech acts for a given sentence helps to reduce ambiguities in language understanding and speech translation.

A variety of approaches use knowledge based techniques like discourse trees, plan-recognition, finite state grammars and other linguistic structures, by conditioning statistical language models on the detected linguistic state. While such methods do lead to significant perplexity reduction, they involve the definition and analysis of useful linguistic structure and the development of grammars to detect and parse relevant constituents. Both tasks are labor intensive and there is no guarantee of optimality.

As an alternative, we propose in this paper an approach in which we treat linguistic structure as states in a Markov Chain. Section 2 gives an overview of the task used for the experiments. Section 3 shows how supervised training can be used to model transitions between known constituents like dialog states or speech acts. Section 4 shows how hidden linguistic states and their meaning can be learned and optimized by the familiar HMM forward-backward learning procedure. These hidden states are then used to build state dependent language models for perplexity reduction.

2. THE SPONTANEOUS SCHEDULING TASK

All experiments have been performed on transcribed text taken from the English section of the Spontaneous Scheduling Task (ESST). This task consists of human to human dialogs recorded in an office environment. A push-to-talk button was used to avoid crosstalk. Each dialog contains 10-15 utterances of two speakers trying to schedule a two hour meeting within a given two week scenario.

190 transcribed ESST dialogs were available for training and testing. The training Data (155 Dialogs) contained 50,000 tokens with a vocabulary size of 1200 words.

3. SUPERVISED TRAINING

3.1. Modeling Dialog States with Markov Chains

To automatically build a dialog structure graph, we used a Markov Model to learn the transition probabilities between dialog states.

The six dialog states choosen for this task were:

state	example
opening:	well I think we need to meet for another two hours to discuss this matter
suggest:	well I'm free on Monday; how does the beginning of June look for you
constraint:	Wednesday through Friday I'm like in a in seminars all day
accept:	wonderful looks like we have a date
reject:	no I'm sorry but that is a bad week for me
closing:	see you later then

The transition probabilities a_{ij} between these states were computed on 15 manually labeled dialogs.

$$\begin{aligned} a_{ij} &\doteq P(q_t = S_i | q_{t-1} = S_j) \\ &= \frac{\text{number of transitions from } S_i \text{ to } S_j}{\text{number of transitions from } S_i} \end{aligned}$$

The discourse model in figure 1 was then derived automatically by starting with a fully interconected model and removing all transitions that fall below a threshold.

This procedure not only eliminates the need to draw dialog charts, it also assigns a probability to each transition that can be used to prune alternatives.

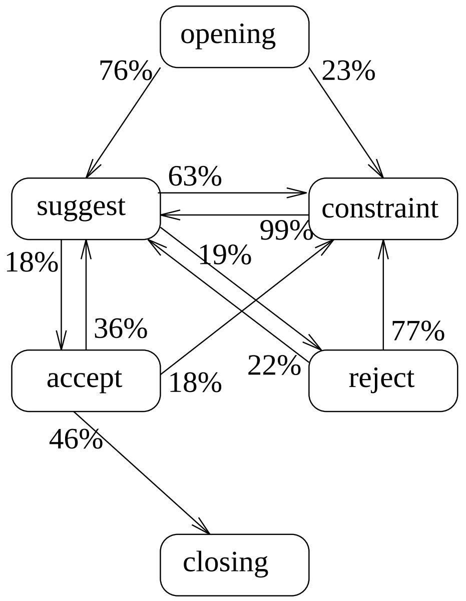

Figure 1. Transition Probabilities

The probability of observing a word in a given dialog state, $b_i(k)$ is given by the frequency of observing that word in that state. It's the emission probability in the Markov Chain.

$$\begin{aligned} b_i(k) &\doteq P(v_k \text{ at } t|q_t = Si) \\ &= \frac{\text{number of times observing } v_k \text{ in } S_i}{\text{number of times in } S_i} \end{aligned}$$

With both transition and emission probabilities, we can compute the probability of being in a given dialog state at a given point of a new dialog. This information can be either combined with other knowledge sources (eg. the probability of a parse given a certain dialog state), or the best state sequence can be used to assign each word the best possible dialog state.

To keep amount of required labeled data small, we clustered the input vocabulary into classes and had the Markov Model operate on the clusters, which yields better generalization; a clustering algorithm clustering words with similar contexts [7, 8] was compared to classes manually designed to suit the task. Though the classes found by the algorithm seemed less intuitive at times, the results showed that it's not necessary to manually build classes.

Table 1 shows the percentage of words assigned with the correct dialog state.

	correct classification
without clustering	69.4%
100 classes	73.6%
200 classes	70.0%
400 classes	73.9%
400 hand build classes	74.1%

Table 1. Classification of the current state

3.2. Modeling Speech Acts with Markov Chains

The experiment of section 3.1 was repeated for the speech acts used in the translation component of JANUS, our speech-to-speech translation system [1, 2].

The training and test data for this experiment was taken form handwritten Interlingua structures used for parser-evaluation. The Interlingua structure contains a semantic representation for a sentence, including speech acts to represent the global *intention.* For this task, 15 speech acts were used. The parser itself sometimes does a poor job identifying the speech act, because they heavily depend on context and the parser only looks at one sentence at a time.

	correct classification
without clustering	61.2%
400 hand build classes	62.3%

Table 2. Classification of the current speech-act

The speech acts *suggest-time* and *state-constraint* were most confusible, because they sometimes only differ by a single word in the previous state:

A: i'm free on Monday *except for ten to twelve*

B: i'm busy on Monday *except for ten to twelve*

The words *except for ten to twelve* are constraints in example A but suggest a time for a meeting in example B.

Here a higher level structure that gives the likelihood of the word *except* in the state *suggest-time* given that the previous state was *suggest-time* will be needed to resolve the ambiguity.

4. UNSUPERVISED TRAINING – HMMS

4.1. State Dependent Monograms

The models derived by training on labeled data will always be suboptimal as they depend on the states chosen and on the consistency of the labeling.

A forward backward training algorithm that converges towards states that optimize the probability of the training utterances yields a model with optimum perplexity reduction. As no lables are required, the amount of training data available is usually substantially larger.

Language model	Perplexity
Monogram Language Model	144
6-state MM build on 15 labeled Dialogs	136
6-state HMM trained on 15 Dialogs	124
6-state HMM trained on 155 Dialogs	94

Table 3. Testset Perplexity of a MM trained on 15 labled dialogs vs. HMM trained 15 dialogs and on 155 dialogs

As shown in figure 2 and table 3, the perplextity reduction for state-dependent monograms in a 6-state model is much larger for unsupervised training than for supervised training, especially as more training data can be used.

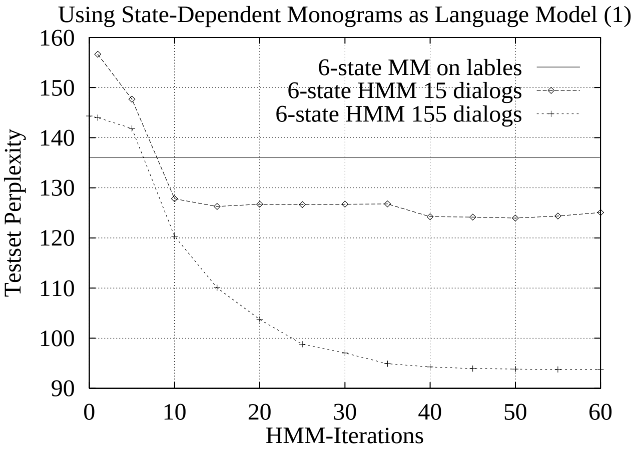

Figure 2. Perplexity reduction of a 6-state Markov Model trained on 15 labled dialogs vs. HMM trained on 15 dialogs and on 155 dialogs

For small numbers of hidden states (3-10), the model learns to build phrases. Each state specializes on a different part of the utterances. A plot of the state activations for an example dialog is given in figure 5 at the end of this paper. Example: Phrases learned by a 6-state hmm:

0: starting a suggestion, opinions
   - I think, I guess, do you think
   - the only time I have

1: days
   - in the morning, in the afternoon
   - Monday, on Monday, Tuesday, the rest of the week

2: yes, no, maybe
   - okay, yeah, yes
   - no, I'm afraid, well

3: times and occupations
   - anytime after two, three to five, at noon, up until three o'clock
   - I have a seminar, i've got a meeting, I'm out of town

4: accept or reject
   - sounds fine to me, that sounds good to me, would be alright, wouldn't be too bad
   - would be bad, that's not too good

5: closing remarks, locations
   - see you later, okay see you then
   - thanks
   - I send you mail regarding the location

For large numbers of hidden states (30-200), the model converges to a cluster-bigram model, where the transition probabilities between states are the transition probabilities between classes. Unlike in cluster algorithms traditionally used for langugage modeling, each word can be in several classes at a time. As the number of hidden states increase the perplexity falls below the bigram perplexity, given that the amount of training data is large enough to estimate all parameters. However the computational effort for the HMM training rises fast, making efficient algorithms mandatory [5]. The perplexity reduction for a 6, 40 and 80 state HMM is shown in figure 3.

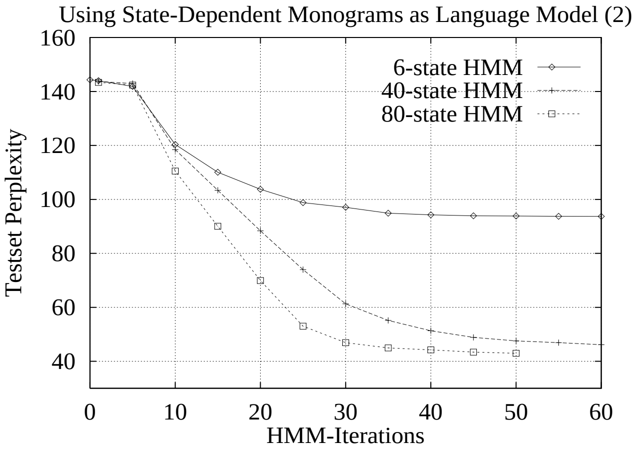

Figure 3. Perplexity of State-Dependent Monograms in an HMM with 6,40 and 80 states trained on 155 Dialogs

4.2. State Dependent Bigrams

While the results on state dependent monograms are encouraging, state dependent bigrams require much more training data. Results on topic-dependent bigram language models reported by other groups [3] were usually obtained on corpora with over 1,000,000 training tokens. Using only the available ESST training data of 50,000 training tokens with a vocabulary size of 1200 words, severe overlearning occured. The testset perplexity was be reduced by 2-3% at best, the training set perplexity by 20-25%.

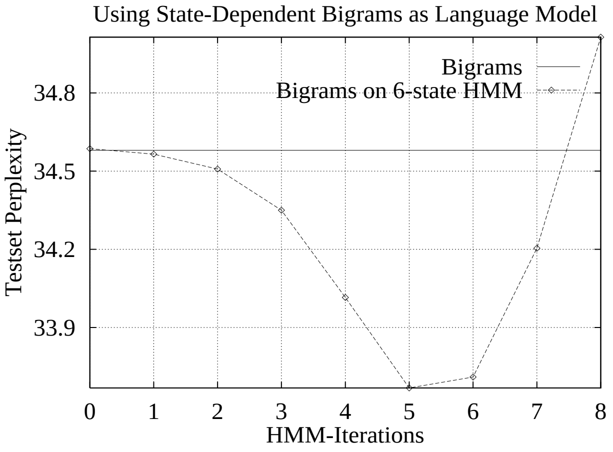

Figure 4. Testset-Perplexity during training for state-dependent Bigrams

The lack in training data could be compensated by combining the HMM-Bigram Model with clustering techniques. Though the small reduction in perplexity yields virtually no reduction in the recognition errorrate, recognition output using this language model is often semantically correct and seems to provide a better basis for speech-to-speech translation than a shorter range language model. This needs to be evaluated further.

5. SUMMARY

Our approach has been successfully used to

- automatically derive dialog state transition diagrams from labled data
- find the best constituent sequence for predefined constituents on the example of dialog states and speech acts
- automatically uncover structure using hidden states which can be used to significantly reduce the monogram perplexity but also to build more complex state-dependent language models.

6. ACKNOWLEDGEMENTS

Thanks to everybody in our speech groups in Karlsruhe and at CMU for helpful discussions and patience.

REFERENCES

[1] M. Woszczyna, N. Coccaro, A. Eisele, A. Lavie, A. McNair, T. Polzin, I. Rogina, C.P. Rose, T. Sloboda, M. Tomita, J. Tsutsumi, N. Aoki-Waibel, A. Waibel, and W. Ward, *Recent Advances in Janus, a Speech to Speech Translation System*, EUROSPEECH 1993, volume 2, pp 1295–1299.

[2] M. Woszczyna, N. Aoki-Waibel, F.D. Buo, N. Coccaro, K. Horiguchi, T. Kemp, A. Lavie, A. McNair, T. Polzin, I. Rogina, C.P. Rose, T. Schultz, B. Suhm, M. Tomita, and A. Waibel *Towards Spontaneous Speech Translation*, ICASSP 1994, volume 1, pp 345–349.

[3] R. Kneser, V. Steinbiss *On the Dynamic Adaptation of Stochastic Language Models*, ICASSP 1993, volume 2, pp 586–589

[4] S. Young, *Dialog Structure and Plan Recognition in Spontaneous Spoken Dialog*, EUROSPEECH 1993, volume 2, pp 1169–1172.

[5] T. Kuhn, H. Niemann, E.G Schukat-Talamazzini *Ergodic Hidden Markov Models and Polygrams for Language Modeling* ICASSP 1994, volume 1, pp 357–360

[6] L.R. Rabiner, *a Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, Proceedings of the IEEE, 1989.

[7] S. Finch and N. Chater, *Bootstrapping Syntactic Categories Using Statistical Methods*, Learning of Natural Language: Proceedings of the first SHOE Workshop, 1992, pp 230-235

[8] K. Ries, *Korpusbasierte Techniken zum Lernen von Übersetzung spontan gesprochener Sprache"*, Diplomarbeit, Universität Karlsruhe, 1994

```
            0:  1:  2:  3:  4:  5:
      CAN: ###  .   .   .   .   .
       WE: ###  .   .   .   .   .
     MEET: ###  .   .   .   .   .
       ON:  .   .  ###  .   .   .
      THE:  .  ###  .   .   .   .
   THIRTY:  .  ###  .   .   .   .
    FIRST:  .  ###  .   .   .   .
       IN:  .  ###  .   .   .   .
      THE:  .  ###  .   .   .   .
  MORNING:  .  ###  .   .   .   .
        $:  .   .   .   .   .  ###
     +LS+:  .   .   .   .   .  ###
     +H#+:  .   .  ...  .   .  111
       NO:  .   .  111  .   .  ...
      I’M:  .  ... 111  .   .   .
      OUT:  .  ###  .   .   .   .
       OF:  .  ###  .   .   .   .
     TOWN:  .  ###  .   .   .   .
     FROM:  .  ###  .   .   .   .
      THE:  .  ###  .   .   .   .
   THIRTY:  .  ###  .   .   .   .
    FIRST:  .  ###  .   .   .   .
       TO:  .  ###  .   .   .   .
      THE:  .  ###  .   .   .   .
   SECOND:  .  ###  .   .   .   .
     +UM+:  .   .   .   .   .  ###
     +LS+:  .   .   .   .   .  ###
     +H#+:  .   .   .   .   .  ###
       WE: ###  .   .   .   .   .
    COULD: ###  .   .   .   .   .
     MEET: ###  .   .   .   .   .
       ON:  .   .  ###  .   .   .
      THE:  .  ###  .   .   .   .
    THIRD:  .  ###  .   .   .   .
       IN:  .  ###  .   .   .   .
      THE:  .  ###  .   .   .   .
AFTERNOON:  .  ###  .   .   .   .
  +CLICK+:  .   .   .   .   .   .
 +NONHUM+:  .   .   .   .   .   .
        $:  .   .   .  ;;;  .   .
     +LS+:  .   .   .  ;;;  .   .
      YES:  .   .  ... ;;;  .  ...
   THAT’S:  .   .   .   .  ### ...
     FINE:  .   .   .   .  ### ...
       BY: ...  .   .   .  ### 111
       ME: ...  .   .   .  ... 111
      BYE:  .   .   .   .   .  ###
  +CLICK+:  .   .   .   .   .  ###
        $:  .   .   .   .   .  ###
```

Figure 5. State Activations of an HMM in an Example Dialog. Symbols used: ### = very high activation, 111 = high activation, ;;; = low activation, ... = very low activation, . = no activation;
Possible State Interpretations:
0: starting a suggestion, opinions,
1: days,
2: yes, no, maybe,
3: times and occupations
4: accept and reject,
5: closing remarks.