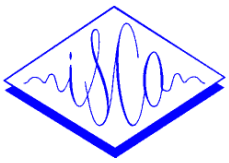

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

A TRELLIS-BASED LANGUAGE MODEL FOR SPEECH RECOGNITION

Nick Waegner and Steve Young
Cambridge University Engineering Department
Trumpington Street
Cambridge CB2 1PZ
United Kingdom

ABSTRACT

This paper discusses tree and trellis-based models as alternatives to conventional n-gram models as the basis for language modelling in automatic speech recognition. The advantage of these models lies in their compactness and the manner in which extended context is used to enhance performance. The latter is confirmed by experiment using models trained and tested on subsets of the Lancaster-Oslo/Bergen corpus.

1. INTRODUCTION

Conventionally automatic speech recognition (ASR) has been viewed as the process of maximising $P(\mathbf{w}/\mathbf{y})$, the likelihood of observing a string of words $\mathbf{w}$ given a string of acoustic observations $\mathbf{y}$. By Bayes' rule,

$$P(\mathbf{w}/\mathbf{y}) = \frac{P(\mathbf{w})P(\mathbf{y}/\mathbf{w})}{P(\mathbf{y})} \tag{1}$$

In maximising $P(\mathbf{w}/\mathbf{y})$, $P(\mathbf{y})$ may be disregarded. Consequently only the maximum of the likelihood $P(\mathbf{w},\mathbf{y}) = P(\mathbf{w})P(\mathbf{y}/\mathbf{w})$ is sought, where $P(\mathbf{w})$ is the a priori probability of the word sequence $\mathbf{w}$ and $P(\mathbf{y}/\mathbf{w})$ is the probability of the acoustic models generating the observations $\mathbf{y}$.

It is clear from the above that a language model, which provides an accurate prediction of the next word in a sequence, is important for the good overall performance of an ASR system. The performance of a language model may be evaluated independently of the acoustic element, using the measure of *perplexity* [1], which is equivalent to the average branching factor at each point of prediction. Given that the language model provides an estimate of the a priori probability $\hat{P}(\mathbf{w}_1^k)$ of a sequence 1 to k of words $\mathbf{w}$, perplexity is given by

$$PP = (\hat{P}(\mathbf{w}_1^k))^{-1/k} \tag{2}$$

This is usually calculated by averaging the sum of the log probabilities,

$$LP = -1/k \log_2 \hat{P}(\mathbf{w}_1^k) = -1/k \sum_{n=1}^{n=k} \log_2 \hat{P}(\mathbf{w}_n) \tag{3}$$

and from this perplexity is given by

$$PP = 2^{LP}. \tag{4}$$

In the following sections, Markov models, trees and trellises are presented as possible language models and their performance evaluated by comparison of perplexity on a task drawn from a tagged corpus.

2. MARKOV MODELLING

To date, the most popular approach to language modelling has been that of Markov chains or *n-grams*, where

$$P(\mathbf{w}_1^k) \approx \prod_{j=1}^{j=k} P(w_j/\mathbf{w}_{j-n}^{j-1}) \tag{5}$$

The conditional probabilities are estimated by maximum likelihood:

$$P(w_j/\mathbf{w}_{j-n}^{j-1}) = \frac{C(w_j,\ldots,w_{j-n})}{C(w_j,\ldots,w_{j-n+1})} \tag{6}$$

where $C(.)$ are counts. Due to constraints on memory and available training data, n is normally no greater than 2. A number of methods are available for smoothing the probability estimates. Here the method of deleted interpolation [1] is favoured for the smoothing of trigrams using unsmoothed trigram and bigram estimates from data :

$$\hat{P}(w_j/\mathbf{w}_{j-2}^{j-1}) = \lambda P(w_j/\mathbf{w}_{j-2}^{j-1}) + (1-\lambda)P(w_j/w_{j-1}) \tag{7}$$

There are a number of possible equivalence classes of trigrams each with an associated λ. These may relate for instance, either to the frequency of occurence of particular unsmoothed trigrams (giving N_f λs, where N_f is the number of frequency *bands*) , or to the particular word w_{j-1}(giving N_v λs, where N_v is the size of the vocabulary).

Although trigrams are both straightforward to implement and have proved highly effective in reducing perplexity in recognition, they only exploit local information contained in the previous two words. It is the possibility of tapping into a more extended context to improve performance, which provides the motivation for tree and trellis-based models described below.

3. TREE MODELS

Decision trees (figure 1) were first proposed as language models for speech recognition by Bahl et al [2]. Adopting the notation of [3], a tree T may be viewed as a set of nodes $T = \{t_0, t_1, \ldots, t_n\}$, with t_0 reserved as the root node. The input to the tree $w_{j-n}, w_{j-n+1}, \ldots, w_{j-1}$ is a string of the

10.21437/ICSLP.1992-70

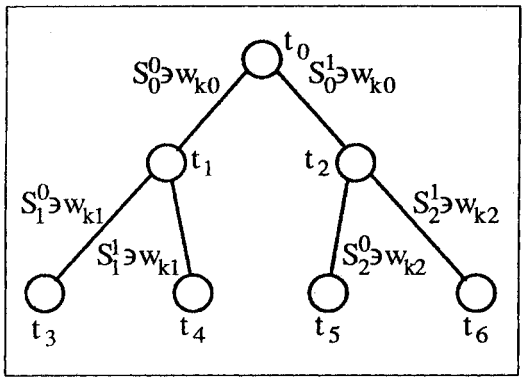

Figure 1. tree

previous n words, and the output from a leaf of the tree is a probability distribution over the possible predicted words $\hat{W}$. Starting from the root node of the tree, at each non-terminal node t_c, a binary function Q_c is performed on w_k, one of the n words of the input. This function takes the form: 'is $w_k \in S_c^j$?', where S_c^j is the jth set of words (for binary branching $0 \leq j < 2$). If the result is true, the left branch from the node is followed otherwise the right branch. This is repeated until a terminal node is encountered.

In growing a tree, the objective lies in minimizing risk for a given cost constraint. Each step consists of splitting some terminal node t into two children in order to maximize a merit function, merit(.), for some test Q, which will lead to a tree which satisfies the global constraints. The merit function employed here is the reduction in class entropy $H(W)$. Given the entropy of a node t,

$$H(Y/t) = -\sum_{w} P(w/t)\log(P(w/t)) \qquad (8)$$

the merit function is,

$$\text{merit}(Q/t) = H(W/t) - \sum_{q=0}^{q=d-1} P(q/t)H(W/t,q) \qquad (9)$$

where $P(q/t)$ is the probability of child q and d is the number of children (in this case, 2).

Due to the computational complexity of growing optimal trees, practical design procedures (deciding on the best splits and hence the sets S_c) are invariably steepest-descent greedy procedures. . That used by IBM in [2] is based on a greedy set growing algorithm, which is computationally intensive. Faster algorithms exists, such as the Flip-Flop algorithm [11]. For the purposes of this work, a clustering-based algorithm due to Chou [6] was employed.

The algorithm is equivalent to K-means clustering and uses a distance measure in this case termed *information divergence*. Vectors consisting of the conditional probabilities of the predicted words, conditioned on a particular instantition of w_k, are iteratively clustered into bins. It can be shown that the optimal partion must satisfy the nearest neighbour criterion as established by the distance metric.

Trees may thus be used to exploit longer term dependencies and relations in the training data to build a more compact model with improved performance as compared to Markov models. However, trees fragment data at an exponential rate, which tends to produce duplicate paths and unnecessarily large structures. Furthermore, the tree is unable to exploit more complex relations in the data, due to the simple questions asked at each node. These issues are considered in the design of the models described in the next section.

4. TRELLIS MODELS

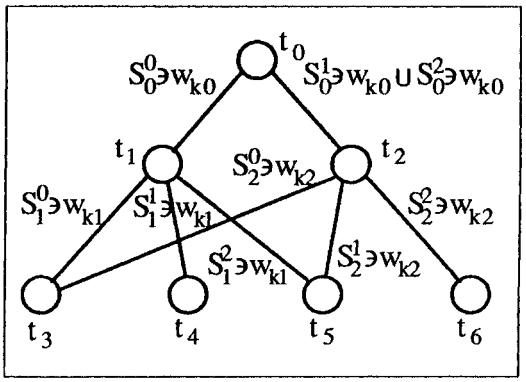

Figure 2. trellis

The algorithm proposed by Chou permits the trees to be grown with more than binary branches. It may also be used to grow hybrid structures, with complex nodes, such as the pylons suggested in Bahl et al [2]. More interesting still is a generalisation to directed acyclical graphs, termed *trellises*. Trellises consist of layers of nodes, whose output bins are combined using the same algorithm as used in the node splitting (figure 2). The latter enables the inference of more complex relations in the data of the form:

$$t = (t_1 \cup \{w_{k1} \in S_1^j\}) \cap \ldots \cap (t_n \cup \{w_{kn} \in S_n^j\}) \qquad (10)$$

where t is a node in a given layer, $t_1, \ldots, t_n$ are nodes in the previous layer, $w_{k1}, \ldots, w_{kn}$ the word on which split is made at each node and $S_1^j, \ldots, S_n^j$ the sets at each node ($0 \leq j < N_s$ where N_s is the number of sets at each node). The splits at each node are chosen independently, in a manner therefore which is not globally optimal. Although a gradient descent technique could be used here, it would considerably increase the training time for the trellis. Despite this, it has been shown [5, 6] that trellises display better storage vs. probability of error performance and do not suffer from the exponential growth of trees as average length increases.

5. EXPERIMENTAL PROCEDURE

5.1. Corpus

For very large language modelling tasks, where the vocabulary is extensive, it is preferable to use parts-of-speech in place of lexical items. Though this often leads to higher perplexity values [7], since for instance, semantic information is lost, the memory requirements of the language model are significantly smaller, as the number of parts-of-speech usually employed is considerably less than the number of

lexical items. Furthermore, this reduces the computational cost of training the models; a consideration of particular importance in the case of trees and trellises.

In order to evaluate the relative performance of the models proposed in the previous sections, each was used to construct a language model for the Lancaster-Oslo/Bergen (LOB) Corpus of tagged text [9]. The latter consists of 500 samples of english texts drawn from 15 different categories. The experiment closely followed the format adopted by Kuhn and de Mori [10], in respect to the division of the data into training, smoothing and testing sets. Three non-overlapping sets of text were drawn from the corpus (table 1).

Text	No. of Samples	No. of words
Training	169	391,294
Smoothing	100	231,297
Testing	100	232,133

Table 1. Division of data

Also a dictionary was compiled from the training set, consisting of word to part-of-speech attachments. The latter gave a total vocabulary of 26,248 lexical items.

Tag	Description	Tag	Description
.	End of Sentence	&FW	Foreign Word
&F0	formula	(	Left Bracket
)	Right Bracket	*'	Begin Quote
*'	End Quote	*-	Dash
,	Comma	AB	Pre-qualifier
AP	Post-determiner	AT	Article
BE	Verb 'be'	CC	Coord. Conj.
CD	Cardinal No.	CD1	'one','1'
CS	Sub. Conjunction	DO	Verb 'do'
DT	Determiner	EX	'there'
HV	Verb 'have'	IN	Preposition
JJ	Adjective	JN	Adj. eg. 'German'
MD	Modal Auxiliary	NC	Cited Word
NN	S. Com. Noun	NNP	eg. 'Englishman'
NNS	Plu. Com. Noun	NNU	Unit (measure)
NP	S. Proper Noun	NPL	S. Loc. Noun
NPS	Plu. Prop. N.	NPT	Titular Noun
NR	Adverbial Noun	OD	Ordinal
PN	Nominal Pronoun	PP	Poss. Det.
PP1	Pers. Pron. 1st	PP2	Pers. Pron. 2nd
PP3	Pers. Pron. 3rd	PPL	Reflex. Pron.
QL	Qualifier	RB	Adverb
RI	Adverb eg. 'near'	TO	'to'
UH	Interjection	VB	Base from verb
VBD	Past tense verb	VBG	Pres. Participle
VBZ	3rd Pers. Sing	WD	WH-determiner
WP	WH-prononun	WRB	Wh-adverb
XNOT	'not','n't'	ZZ	letter

Table 2. Reduced set of Tags

Instead of using the 135 tags available directly from the LOB corpus, a reduced set of 56 was constructed (table 2). This was done in the interests of computational load once more, however, as many of the original tags were used relatively infrequently, performance of the models was not seriously impaired.

5.2. Training of Models

Trigrams of parts-of-speech or *triPOS* were trained and smoothed, in the manner described in section 2. Deleted interpolation of the unsmoothed triPOS and biPOS probabilities was used for smoothing the triPOS model. Here, the part-of-speech of the word immediately preceeding the predicted one was used to assign the triPOS to a particular smoothing λ.

An important issue in the growing of trees is establishing the correct size. Here, a node was not split if the merit function fell below a threshold or the number of observations arriving at a node was less than a constant. Alternatively, the CART method of pruning using cross-validation [3] could have been adopted, or the iterative technique of [8], however it should be noted that these are only suitable for medium size trees.

In growing the trellis, each node was allowed to split into a maximum of 20 bins. In addition, each layer was allowed to grow by a factor of four, until a maximum of 600 nodes was reached. A maximum of six layers was set for the depth of the Trellis. Consequently, the Trellis language model took the from shown in table 3.

Layer	No. of Nodes
1	1
2	20
3	80
4	240
5	600
6	600

Table 3. Trellis Topology

For the trees and trellises, smoothing is imperative at each node to enable the use of the divergence metric. The scheme based on the *leave-one-out* error measure as suggested in [6] was used. As this is an iterative technique, the bins or children of a node were smoothed with their parent,

$$\tilde{\mu}(t_k) = \lambda \mu_(t_k) + (1-\lambda)\tilde{\mu}(t) \qquad (11)$$

where $\tilde{\mu}(t_k)$ and $\mu_(t_k)$ are the smoothed and unsmoothed distributions of child k respectively and $\tilde{\mu}(t)$ the smoothed parent distribution. In the case of the trellis, the nodes were smoothed with respect to the root node. Having grown the tree and trellis models, they were both finally smoothed using the leave-one-out technique by passing down the training and smoothing data combined.

5.3. Testing

For testing, as in [10], the models were used to predict words of the entire test set, without reference to the tags. In the triPOS case, $P(w_j = w)$, the probability of the predicted word w_j, was given by,

$$\sum_{g_j}(1-ne)\hat{P}(w/g(w))\hat{P}(g_j = g(w)/g_{j-1}, g_{j-2}) + e \qquad (12)$$

where g_k is the part-of-speech at k, $\hat{P}(g_j = g(w)/g_{j-1}, g_{j-2})$ is the smoothed triPOS, n the number of POS, $\hat{P}(w/g(w))$ the estimated probability of tagging word w with part-of-speech $g(w)$ and e a constant to ensure non-zero probability. A similar equation was used to calculate the predicted word probabilities using the trees and trellises,

$$P(w_j = w) = \sum_{g_j} (1 - ne)\hat{P}(w/g(w)).\hat{P}_{tre}(g_j = g(w)) + e \tag{13}$$

where $\hat{P}_{tre}(g_j = g(w))$ is the part-of-speech probability at the leaf of the tree or trellis given the context. In training and testing, a context of 8 preceeding words was employed, deemed sufficient for improved performance, whilst constraining the computational load.

The experiments were also repeated using the cache model suggested by Kuhn and De Mori [10], for the reduction of perplexity. A cache was used over 18 of the most common parts-of-speech and incorporated into the models as follows,

$$\begin{aligned} P(w_j = w) = \; & (1 - d)\textstyle\sum_{g_j}[\lambda_c C(w, i) + (1 - \lambda_c)\hat{P}(w/g(w))] \\ & \times[\hat{P}(g_j = g(w)/g_{j-1}, g_{j-2}) + e] \end{aligned} \tag{14}$$

where $C(w, i)$ is the cache probability for part-of-speech i and word w.

Thus, equations 12, 13 and 14 were used to provide $\hat{P}(w_n)$ required for the calculation of the log probability (equation 3) and subsequently, perplexity (equation 4). The perplexity values on the test set for the various models with and without caching are given in table 4. From these it is clear that a moderate improvement is achievable using the trellis models, with the tree giving only a slightly lower performance, inspite of their more compact size.

Model	Storage	Perplexity	
		No Cache	Cache
TriPOS	3136	368.7	316
Tree	819	371.9	318.6
Trellis	524	362.7	310.7

Table 4. Results

6. CONCLUSIONS

It has been demonstrated that the context discarded in the trigram models may be usefully employed to achieve moderate gains in perplexity using considerably smaller models. Furthermore, the ability of trellis models to capture more complex relations in data than trees, explains their smaller size and improvement in performance, though this is not necessarily guaranteed.

The performance of the models could be enhanced further using models trained on lexical items rather than trees or as in [4] using lemmas (base-forms of lexical items), and combining these with the part-of-speech models by deleted interpolation. The extension of the tree/trellis growing algorithm to such larger feature sets is possible, so too is the extension in the lookback, though in both cases, training complexity also increases.

ACKNOWLEDGEMENT

The authors would like to thank Peter Chou for his helpful suggestions in regard to the design of trellises.

REFERENCES

[1] L. R. Bahl, F. Jelinek and R. L. Mercer. *A maximum likelihood approach to continuous speech recognition.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 5(2):179-190 , March 1983.

[2] L. R. Bahl, P. F. Brown, P. V. Souza and R. L. Mercer. *A tree-based statistical model for natural language model.* IEEE Transactions on Acoustics, Speech and Signal Processing, 37(7):1001-1008 , July 1989.

[3] L. Breiman, J. H. Friedman, R. A. Olshen and C. J. Stone. *Classification and Regression Trees.* Belmont, CA: Wadsworth, 1984.

[4] H. Cerf-Danon and M. El-Bèze. *Three different probabilistic language models: comparison and combination* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing , pages 297-300 , 1991.

[5] P. A. Chou. *Applications of Information Theory to Pattern Recognition and the Design of Decision Trees and Trellises.* Ph.D. dissertation, Stanford University, Stanford, CA, June 1988.

[6] P. A. Chou. *Optimal partitioning for classification and regression trees.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(4):340-354 ,April 1991.

[7] P. Dumouchel, V. Gupta, M. Lennig and P. Mermelstein. *Three probabilistic language models for a large-vocabulary speech recognizer.* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing , pages 340-354 , 1988.

[8] S. B. Gelfand, C. S. Ravishankar and E. J. Delp. *An Iterative Growing and Pruning Algorithm for Classification Tree Design.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(2):163-174 ,February 1991.

[9] S. Johansson, R. Garside, K. Hofland and G. Leech. *The tagged LOB corpus: vertical/horizontal version* Norwegian Computing Centre for the Humanities, Bergen University, June 1986.

[10] R. Kuhn and R. De Mori. *A cache-based natural language model for speech recognition.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(6):520-583 ,June 1990.

[11] A. Nádas, D. Nahamoo, M. Picheny and J. Powell. *An iterative 'flip-flop' algorithm approximation of the most informative split in the construction of decision trees.* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing , pages 565-568 , 1991.