Proceedings of the Institute of Acoustics

UTILISING DATABASE INFORMATION TO IMPROVE SPEECH RECOGNITION PERFORMANCE

P J Durston, D J Attwater, M D Edgington

BT Laboratories, Martlesham Heath, Ipswich IP5 3RE

1 INTRODUCTION

In many telephone-based commercial transactions callers accesses information held in databases through an agent. Touch-tone services (e.g. telephone-banking) are now well established, but are limited at any point to a keypad of responses. Speech recognition allows quicker and more efficient access to complex data sources, which is not possible with touch-tone technology.

Previous work, focused on accessing directory databases, has shown how speech recognition can be used to identify a single entry in a database [1]. Often, in large database applications, more than one recognition must be performed to achieve this. In practice the same database field is often recognised in spelt or spoken form [2] or, through the database, recognised against other fields [3,1]. In either case some strategy for combining multiple recognition events is required [4].

In this paper a predictive framework to aid with the design of such a strategy is presented. Three particular approaches are considered: intersection, union and sequential subsetting. As a starting point, the use of database contents as an information source to set recognition priors and constrain the recogniser search space is considered. The framework then considers how knowledge of the database relationships and isolated field recognition can be used to enable identification of the most appropriate combined recognition strategy for a task. The database is therefore used to design a system capable of accessing its entries. Throughout, this paper is illustrated with practical examples from a real, telephony-based, UK address recognition system. Recognition results obtained using a corpus of ~ 1400 isolated-field (not fluent) responses recorded from a national spread of people speaking their own UK address are presented [5]. The BT STAP recogniser was used for all of the experiments [6].

2 DATABASE CONSTRAINS AND GRAMMAR DESIGN.

Choosing the recogniser's defining grammar highlights a trade off between recognition accuracy and out-of-vocabulary (OOV) detection as the vocabulary size is varied. A totally constrained grammar with only *valid* responses (of which there are V_i - the number of distinct database entries for that field) will offer the best accuracy provided the response falls within the vocabulary. However the caller is unaware of the bounds of the vocabulary. In such a grammar OOV utterances may be recognised as one of the constrained grammar items if OOV rejection is not accurate. On the other hand a totally open grammar (in the extreme permitting any combination of speech sounds) will give the lowest recogniser performance. It will be necessary in this case to parse the recogniser output in order to extract valid vocabulary items. This process can provide some OOV detection. In either case, provided the utterance is requested in the correct context, the database defines the breadth (and statistical weight [7]) of valid utterances.

Within the mathematical framework a recogniser i, is characterised by its defining grammar g_i, the lengths of the candidate list produced n_i, and the probability that given an in vocabulary utterance, a correct candidate is in that list $R(n_i,g_i)$. This may be determined experimentally or predicted using observations from similar recognition tasks [8]. When recognising from a list of vocabulary items (i.e. when grammar perplexity equals the vocabulary size, N_i) the grammar may be approximated as $G_i(N_i)$ - a function only of the vocabulary size. Throughout this paper it is assumed that in accessing the database information the caller gives self-consistent information relating to an entry in the database. It is also assumed, within the framework, that all recognitions of individual database fields are independent statistical events. This is broadly correct, but in some instances, for example a particular caller with an unclear and often mis-recognised voice, or a poor quality speech channel, it is not so.

The issue of grammar overgeneration is now explored using an example from the UK address recognition task.

2.1 Illustration from UK Postcode recognition

UK postcodes consist of a sequence of 5-7 alphanumeric characters with 1.4M valid combinations (i.e. V_{PC} =1.4M). The simplest postcode grammar capable of modelling all currently valid postcodes would allow any combination of letters and digits in a postcode-like form. This overgenerating grammar would produce 1,755M 'postcodes', many more than are currently valid (e.g. QQ1Q 1QQ). As the grammar is tightened so the over-generation factor, $\gamma = N_{PC}/V_{PC}$, falls the recognition performance correspondingly increases as shown in Figure 1a. This process uses postcode knowledge found in the database to eliminate invalid postcodes by deducing rules. For example postcodes always start with one of only 126 letter combinations. It is imperative that valid postcodes are not excluded. The best accuracy achieved to date is shown in Figure 1b. A grammar that contained only (currently) valid postcodes would run prohibitively slow and so the best results used a grammar that exhibited a degree of overgeneration. Once the database had been used to remove invalid postcode results the top1 accuracy increased to 66%. Figure 1b also shows the benefit of obtaining a ranked list of candidates from the recogniser rather than only a single candidate.

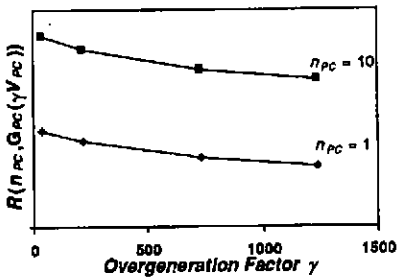

Figure 1a Increase in recognition performance as grammar overgeneration is reduced

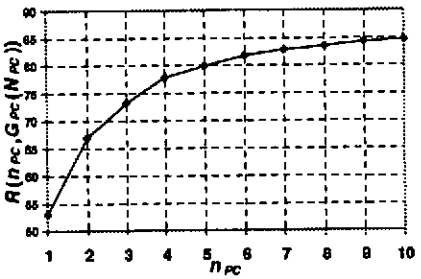

Figure 1b Postcode performance using a slightly overgenerating grammar ($\gamma \sim 8$)

3 COMBINING MULTIPLE RECOGNITIONS

With a large database and the recogniser performance currently available it is often helpful to perform more than one recognition on different fields of the database entry and combine the results. If different recognitions are to be compared it is necessary that the relationship between results is defined. Before comparison can occur the two results must be mapped to a common data representation. The results can then be combined and the resulting candidate(s) propagated. Additionally, results from one recognition, or from previous result combinations, can be used to adjust the priors of subsequent recognitions. This subsetting approach is a powerful means of focussing down on the data entry of interest in large database applications.

3.1 Intersecting recognition results

Intersecting independent recognition results is an effective way of increasing the confidence associated with a proportion of related recognitions. If both recognitions agree and are based on different prior assumptions then the results can be associated with an increased confidence. In addition after intersecting lists the number of results that are propagated is reduced. The disadvantage of this approach is that for a proportion of recognition pairs there will be no intersection. In these instances an additional approach is needed (possibly a union).

3.1.1 Intersecting single candidate recognition

As an example consider a task to identify the first set of letters of a postcode (the outcode letters). The most useful address fields that relate to this are the postcode and the county. The mapping between postcode and outcode letters is a simple truncation whereas for counties (a vocabulary that also includes some major cities) there is a complex relationship, a portion of which is shown in Figure 2a. In addition Figure 2b shows the number of outcode letter sets associated with a county. Combining recognition accuracies for postcode and county shows that there is a 63%

probability that both top1 results will be correctly recognised. To obtain the confidence associated with such an intersection the chance of a false match (due to two mis-recognitions) should be calculated. A false match is indistinguishable from a correct match, although as only the top1 candidates are being intersected in this example the likelihood of a random match is small (and will be determined as 1.7% later in this paper). Thus, if an intersection is found the result is 99.0% accurate; much higher than either of the contributing recognitions.

The chance of a single random match $F_{ijk}(n_i,n_j,1)$ depends on the mapping relationship between the two recognition forms (i and j), the common data representation k, and the two list lengths (n_i and n_j) that are intersected. In the simplest case (e.g. spelt and spoken recognitions) there would be a 1:1 mapping with each vocabulary item in the first vocabulary corresponding, through the common representation, to one and only one item in the second vocabulary. As the relationship becomes more complex, incorporating n:m mappings, then the chance of an accidental match increases.

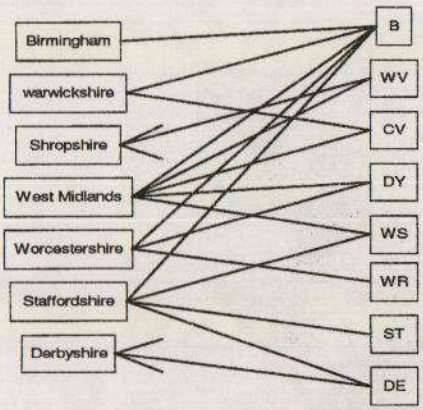

Figure 2a. Portion of relational map between counties and outcode letters in the database

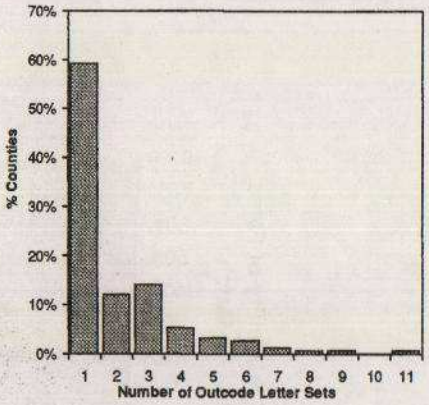

Figure 2b. Number of outcode letter sets associated with UK counties

The accidental match probability is generally inversely proportional to the vocabulary size and so only becomes significant when using small vocabularies (< 200). As the vocabulary grows recognition performance falls, however the accidental match probability falls faster and so becomes less important. For example, intersecting top1 results from a (1.4M) postcode recognition with some other related address field such as road name (of which there are 300K in the UK) would very rarely produce an accidental match.

3.1.2 . Intersecting recognition lists

The benefit of using a list of candidates from a recogniser over a single top1 result has already been noted. When intersecting top1 items high confidences were achieved but at the expense of most of the recognition pairs not intersecting and therefore requiring a different strategy for result propagation. By intersecting two lists the number of intersections and so the effectiveness of the intersection, can be increased.

For the outcode letter identification task two vocabularies are used, labelled P and C, one is set up to recognise outcode letters from postcodes and the other counties. The variables V_P and V_C in this section, are equal to the number of vocabulary items for each field found in the database as the two recognisers have no additional *a priori* constraints.

In order to measure the effectiveness of longer candidate lists the chance of an accidental match (or matches) must be quantified. If the mapping between the two recognition vocabularies is a simple 1:1 correspondence then the chance of m accidental matches can be determined as:

$$F_{ij}(n_i, n_j, 1) = \frac{{}^{V-n_i}P_{n_j}}{{}^{V}P_{n_i}} = \frac{{}^{V-n_i}P_{n_j}}{{}^{V}P_{n_j}} \quad \text{where} \quad {}^{N}P_{r} = \frac{N!}{(N-r)!} \quad \text{and} \quad V = V_1 = V_2$$

In the outcode letter example the relational map is complex and cannot be solved analytically. It is therefore necessary to perform Monte-Carlo simulations in order to determine the accidental match probabilities. It is also necessary to decide in which representation the intersection will be performed. Counties are mapped to outcode letters in this example. In the simulations a random list of varying length is selected from each recogniser vocabulary, mapped to the associated outcode letters and intersected. This was repeated for all combinations of the list lengths up to 10 items each. The proportion of attempts that lead to m matches, $F_{PCP}(n_P, n_C, m)$, was then noted as a function of the two list lengths. The summed results, shown in Figure 3, give the chance of one or more accidental match, $\sum_{m=1} F_{PCP}(n_P, n_C, m)$

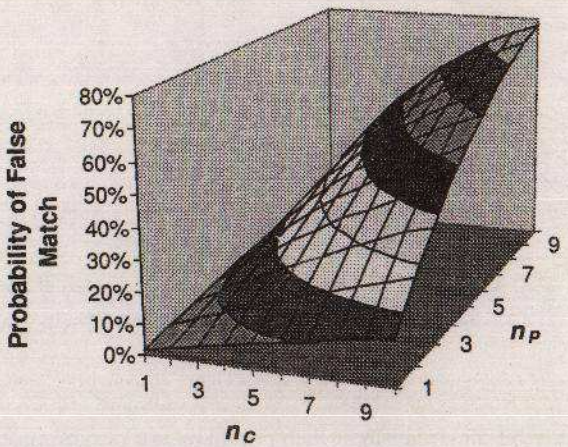

Figure 3 Chance of one or more false match $\sum_{m=1} F_{PCP}(n_P, n_C, m)$ when randomly selecting from 2 vocabularies

When intersecting two top10 lists for the outcode letter task there is a 78% chance of one or more false matches. This reduces the confidence in a single intersection that it is correct. Moreover there is a significant chance of obtaining multiple accidental matches. Two accidental matches occur in 26% of cases with three or more in a further 18%.

In order to proceed the accidental match and recogniser match events are treated as statistically independent. The probability of any number of accidental matches with or without a correct recogniser match can then be calculated. When more than one match is encountered it is not obvious which (if any) result should be propagated and so we chose to only propagate results from single matches. (Intersection allows a proportion of all cases to be tagged as high confidence. Allowing multiple matches to propagate could be appropriate if an immediate decision is not required).

Combining the Monte-Carlo results with top$N$ recogniser performance figures ($R_P(n_P,g_P)$ and $R_C(n_C,g_C)$) allows the proportion of single matches to be calculated. The results are shown as a surface plotted against the two list lengths in Figure 4a.

$$P(\textit{single correct match}) = [R(n_P,g_P)\,R(n_C,g_C)]\,[1 - F_{PCP}(n_P,n_C,m)] = \alpha$$
$$P(\textit{single false match}) = [1 - R(n_P,g_P)\,R(n_C,g_C)]\,[F_{PCP}(n_P,n_C,m)] = \beta$$

$$P(single\ match) = \alpha + \beta$$

The morphology of Figure 4a can be understood as follows. As the list length is increased from one candidate the number of single matches increases showing, not surprisingly, that intersecting more candidates increases the likelihood of a single match. However for longer lists there is a far greater chance of an accidental match (or matches) which lowers the number of single matches and caused the curve to fall.

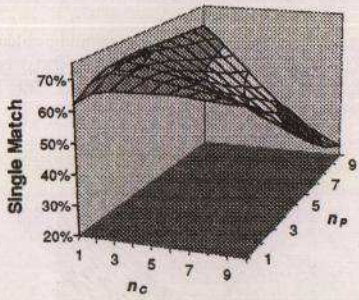

Figure 4a. Percentage of intersections that lead to a single cross-match (correct or accidental)

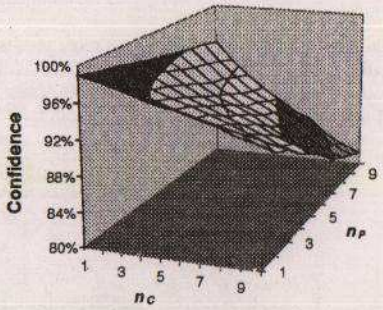

Figure 4b. Confidence that a single match from an intersection is correct

The confidence in a single match being correct (and not accidental) is shown in Figure 4b as a function of list lengths. At large list lengths a match is more likely to be accidental and so the confidence drops.

$$Confidence = \frac{\alpha}{\alpha + \beta}$$

The best operating point is obtained by maximising the number of single match intersections that are correct. Multiplying the two surfaces (Figure 4a and Figure 4b) gives this metric. As the surface gradient in Figure 4b is small in comparison to the first surface, the optimum list lengths are determined as 2 and 3 for county and postcode recognitions respectively. The slight asymmetry is due to different recognition accuracies, vocabulary sizes and the inherent asymmetry of the relational map.

3.1.3 Recognition Experiments

Using a single intersection between postcode and county results obtained using the corpus, yield the figures shown in Table 1 shown together with the theoretical figures (shown in parentheses) derived above. It is indeed found that there is a maximum, following the trends observed above. There is good agreement between theory and experiment for the first two cases. For the final case there is some disagreement in the number of multiple matches, with many more expected in theory than observed. The recogniser did not always return a full list of ten candidates (particular for the outcode letter recognition) accounting for the disagreement.

Letter Pair	County	Single Match		Not One Match		Confidence
n_P	n_C	Correct α	Incorrect β	No matches	Multiple matches	Level $\alpha/(\alpha+\beta)$
1	1	60.4% (61.8%)	1.68% (0.63%)	37.9% (36.5%)	0.00% (1.06%)	97.28% (98.99%)
3	2	72.0% (68.9%)	2.53% (2.17%)	20.8% (21.6%)	4.63% (7.36%)	96.61% (96.94%)
10	10	53.7% (19.3%)	3.57% (4.37%)	5.47% (2.90%)	37.3% (73.4%)	93.75% (81.56%)

Table 1 Experimental and predicted results (in parentheses) for three operating points

3.2 Unioning Recognition Results

Unioning independent candidate lists has the advantage that there is a greatly enhanced probability that the correct information will be propagated in the common form. The only instance when this is not the case is when the spoken utterance fails to be correctly recognised in both cases. The case of unioning outcode letter and county recognitions is shown in Figure 5a.

$$P(\text{in unioned list}) = 1 - [1 - R(n_P,g_P)][1 - R(n_C,g_C)]$$

The disadvantage of a union is that more results must be propagated. The number of results cannot be calculated without knowing how the two lists intersect and so which candidates will appear more than once. Moreover, mapping both recognitions to a common form may cause an increase in the number of items to be propagated. Only in the case of a 1:1 mapping can the number of candidates be analytically determined.

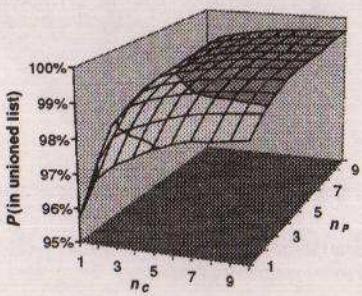

Figure 5a. Percentage of unions where the correct result is propagated

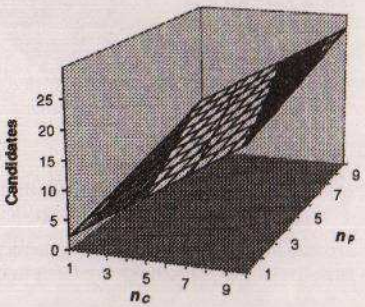

Figure 5b. Approximate number of candidates propagated

Based on the Monte-Carlo approach an approximation to number of candidates after a union of outcode letter and county recognitions is shown in Figure 5b. When using two top10 lists there are ~27 distinct candidates to propagate - more than 20 because of the 1:n map (see Figure 2b).

3.3 Subsetting

In the two approaches above recognitions are performed on complete vocabularies describing a database field and then the results combined in some way. The recognitions could be performed in any order, as they are not interdependent. A complementary technique is to use the results from one recognition to affect the grammar used for subsequent recognitions. This technique, in its most simplistic form, can be used to eliminate all database entries from subsequent grammars that do not match with the first recognition results – in effect guaranteeing an intersection of the two results.

After the first recognition, a list of n_l candidates will be proposed by the recogniser. These may then be mapped via the database to prepare the vocabulary for a second recognition. The second recognition is then performed on a much smaller vocabulary and is correspondingly much more accurate. However with this approach if the first recognition is incorrect then there is no recovery route – the second recognition is bound to identify an incorrect database entry (if OOV rejection is not accurate). The overall accuracy of this approach is then limited by the accuracy of the first recognition and so it is important to perform the most accurate recognition first. The degree of subsetting produced by the first result will have a bearing on the vocabulary size, and so performance, of the following recognition. The degree of subsetting achieved must be weighted against the accuracy of each recognition and the likelihood of excluding the correct entry. The mapping between database fields will affect the order and effectiveness of subsetting in each task separately.

In order to predict performance the average number of vocabulary items in the second vocabulary associated with a single item in the first vocabulary must be known (ε). As an approximation it can be assumed that the subsetted vocabulary grows linearly with the number of candidates from the first recognition (this assumes that there is no overlap between associated items from different recognition candidates). The subsetting performance is thus:

$$P(\textit{in first list}) = R(n_1, G_1(V_1))$$
$$P(\textit{correct}) = R(n_1, G_1(V_1))\, R(n_2, G_2(\varepsilon n_1))$$

As $(\varepsilon n_1) = \gamma V_2$ the second recognition grammar has a very low overgeneration factor ($\gamma << 1$) and thus its performance is correspondingly high. Examples from address recognition show how subsetting can be used to close from a top$N$ candidate list to a single candidate. The number of candidates from the first recognition (n_1) must be adjusted to maximise $P(\textit{correct})$. In the case of using n_1 candidates from a postcode recognition to subset the database and then performing a top1 road name recognition ε is close to unity and so n_1 should be increased and the maximum list length (in practice, 10) used. For subsetting the road name vocabulary using an outcode (all but the final three postcode characters) ε is ~ 270 and so a short outcode candidate list (top1 in practice) was used. It is thus possible to predict the best recognition order for a subsetting task.

4 UK ADDRESS RECOGNITION PERFORMANCE

Using all of the techniques discussed in this paper and combining spoken postcode, county and road name address fields the results, as shown in Figure 6, were obtained for the UK address recognition task. These figures represent the performance obtained for straightforward calls (when the utterances are spoken succinctly with limited background noise) and for all calls (including those containing superfluous speech and/or significant background noise). The techniques allow a portion (43%) of (straightforward callers') postcodes to be identified with an accuracy of 96.7%. A second lower accuracy bracket contains 21% of transcribed postcodes and has an accuracy of 73.3%. Finally the remaining postcodes are labelled as low confidence and are 61.4% correct.

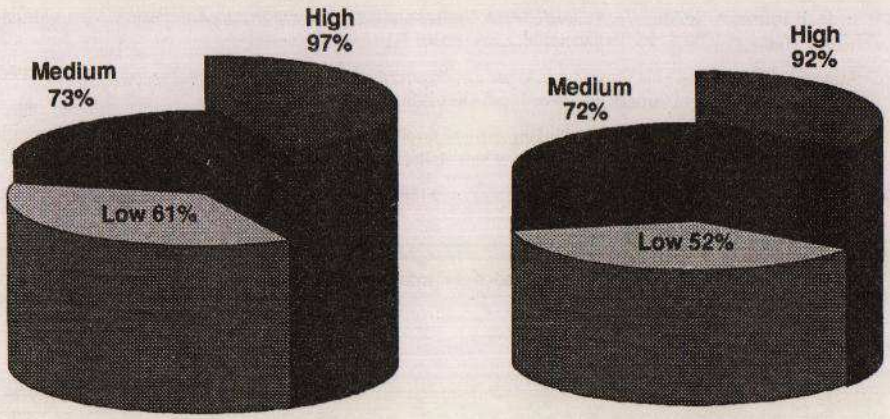

Figure 6 Using the techniques described above address transcriptions can be into three bands banded (as shown by the segment sizes) with confidences as quoted. Figures are shown for both straightforward calls only and all calls in the testing corpus.

5 CONCLUSIONS

When building recogniser vocabularies that match a single item held in a database there is a trade off between a well-constrained grammar giving good recognition performance and a less constrained grammar providing some OOV detection. In large vocabulary recognition task with high perplexity grammars (such as postcodes) some generalisation and so overgeneration is required in order to limit recognition times. On the assumption that the utterance was in vocabulary, results from an overgenerating grammar can be filtered using the database and invalid items removed, thus increasing performance.

Combining recognitions performed on different corroborating fields in the database is a powerful means of increasing both the likelihood of obtaining the correct database entry and increasing its associated confidence. A predictive mathematical framework has been proposed which, following isolated recognition experiments and analysis of the database mappings, can then be used to identify the most appropriate combination strategy for a task.
This paper has shown, for example for UK address recognition, that a portion of caller's addresses can be recognised with an accuracy significantly higher than the recognition accuracy of any one address field.

6 REFERENCES

[1] D Attwater, S Whittaker. *'Issues in large-vocabulary interactive speech systems'* in Speech Technology for Telecommunications. F Westall, R Johnston, A Lewis (eds.), Chapman and Hall, London. pp 465-486 (1997)

[2] M Meyer, H Hild. *'Recognition of Spoken and Spelled Proper Names'* Eurospeech '97, Rhodes, Greece. pp1579-1582 (1997)

[3] F. Seide, A Kellner. *'Towards an Automated Directory Information System'* Eurospeech '97, Rhodes, Greece. pp1327-1330 (1997)

[4] F Seide, B Rüber, A Kellner. *'Improving Speech Understanding by Incorporating Database Constraints and Dialogue History'*. ICSLP '96, Philadelphia, USA. pp1017-1020 (1996)

[5] D Attwater, P J Durston, H R Greenhow. *'What's in an address? Issues in UK address recognition'* Proceedings of 17th annual conference of American Voice Input Output Society. (1998).

[6] F Scahill et.al. *'Speech Recognition - Making it Work for Real'* in Speech Technology for Telecommunications. F Westall, R Johnston, A Lewis (eds.), Chapman and Hall, London. pp323-351 (1997)

[7] H Hild, A Waibel. *'Recognition of Spelled Names over the Telephone'* ICSLP '96, Philadelphia, USA pp346-349 (1996)

[8] A Simons. *'Predictive Assessment for Speaker Independent Isolated Word Recognisers'*. Proceeding of 4th European Conference on Speech Communication and technology, Eurospeech, Madrid, September 1995. ISSN 1018-4047.