A SCALABLE ARCHITECTURE FOR DIRECTORY ASSISTANCE AUTOMATION

Premkumar Natarajan, Rohit Prasad, Richard M. Schwartz, John Makhoul
{pnataraj, rprasad, schwartz, makhoul}@bbn.com

BBN Technologies
Cambridge, MA 02138, USA

ABSTRACT
We present a novel architecture for providing automated telephone Directory Assistance (DA). The architecture couples a large-vocabulary, statistical n-gram, speech recognition engine with a statistical retrieval system. The use of a statistical n-gram allows for the recognition of unconstrained spoken queries while the statistical retrieval engine allows for an inexact match between a particular spoken query and the training data. Allowing for unconstrained recognition and an inexact match provides the framework for high levels of automation. Once the retrieval engine returns a ranked set of frequently requested telephone numbers (FRN), the rejection module uses a classifier to compute a confidence-like score that is used to make the automation decision. With actual customer calls into an operational, automated DA call center and an FRN set size of 25000 numbers, the new architecture is capable of delivering more than 17% correct automation at a false accept rate of 0.76%.

1. INTRODUCTION
Directory assistance (DA) services provided by the telephone carriers involve a large number of human operators located in many call centers across the country. As such, there is the potential for realizing substantial savings by employing DA automation systems using speech recognition technology. Because of the large number of call volumes, even a small percentage automation can lead to considerable savings. This paper describes a novel architecture for achieving relatively high automation rates. The architecture integrates (a) state-of-the-art large-vocabulary, speaker-independent continuous recognition, (b) statistical information retrieval technology, and (c) an effective rejection algorithm to decide whether to automate a particular DA call or send it to an operator.

2. PROBLEM DESCRIPTION
The typical DA request in the USA is initiated when a caller seeking the telephone number for a particular listing dials 411. Once the connection is established, the system queries the user first for the city/state (also called locality) information and stores the caller’s response. The next system prompt is usually “What Listing?” or a variant thereof. Again the system stores the caller’s response. The two user responses are then forwarded to the operator who searches the telephone number database for the relevant information. The database search usually involves a fast, pattern-matching algorithm that returns a list of possible telephone numbers. The operator quickly scans this list and returns a unique telephone number or, if necessary, asks the user for additional disambiguating information such as the street location.

Business listings typically account for 80% of DA call traffic while residential listings only account for 20% of the calls. Analysis of the released telephone number distribution indicates that a small fraction of business listings account for a substantial fraction of DA call traffic. This small subset of telephone numbers is often referred to as the set of frequently requested numbers or FRNs. For the purposes of this paper, only FRNs are targeted for automation.

3. INITIAL EXPERIMENTS
A simple approach to DA automation would be to build finite-state grammars (FSG) for recognizing listing names and localities and couple the recognition with a simple lookup table to associate valid locality-listing pairs with unique telephone numbers. In this scheme, a query gets an automated response only if the recognized listing-locality pair exists in the lookup table. The direct approach of building the FSG from the information in the DA database is not very effective as the following example (see next page) of real customer queries and relevant parts of the associated database entry indicate (names and locations have been changed to preserve data confidentiality). In this example, none of the user requests corresponds exactly to what is in the database listing.
We could use rules to generate listing phrases from the database information but experimental results show that this approach does not yield sufficient coverage. Given the obvious inadequacy of the DA database as a model for the listing query, we decided for our initial experiments to build a FSG by collecting examples of actual queries (training data) for the FRNs targeted for automation. We used an acoustic phoneme-loop alternate model for rejection and invested significant effort in optimizing the entire system for the application. The FRN set was chosen to cover about 31% of the DA call traffic. The lookup table was generated from locality-listing pairs seen in training data as well as in the database. With this system, we were able to correctly automate 10.5% of the traffic at a corresponding false accept rate of 0.5%. The measurements were made on a test set consisting of 5000 randomly selected customer calls to an operational DA call center.

The graph in Figure 1 illustrates the rate at which the FRN set size grows with traffic coverage that is desired. The fact that the size of the finite-state listing recognition grammar increases rapidly with FRNs presents a computational problem for higher automation levels (unless one designs an annoyingly deep hierarchical call flow that asks a series of disambiguating questions). The problem is compounded by the fact that there are numerous ways in which different callers will ask for the same listing. Measurements on actual customer calls show that about 40% of the users ask for a given listing in one particular way. Even if we allowed for 10 different ways of asking for each listing, the resulting grammar only covers about 66% of the queries. Moreover, callers are often unsure about the physical location of the business and, therefore, an exact combination of the recognized locality and listing is often not present in the lookup table; as a result, the request cannot be automated. Due to these reasons, the FSG approach does not scale well to higher automation rates.

4. NEW SYSTEM ARCHITECTURE

The key technical concept in our novel approach is that we allow an inexact match between a spoken query and the training data for FRNs. This inexactness is allowed both in terms of the word-structure of the listing utterance, as well as the validity of the listing and locality pair. To allow for the inexact matches, the DA automation problem is factored into the following three independent, sequential operations:

1. Recognizing the listing and locality utterances.
2. Retrieving the most appropriate FRN, given the recognized listing and locality.
3. Making the automation decision: release the top-choice retrieved FRN, or transfer call to operator.

In contrast to the system described in Section 3, we call this system the Information Retrieval DA (IR-DA) system. The block diagram in Figure 2 graphically illustrates the sequence of operations. Rectangular blocks are operations to be performed and those with curved corners designate statistical models that need to be trained. As shown in the figure, the first operation in the automation system is generating a hypothesized transcription of the listing and locality utterances. Recognition is followed by a retrieval operation that uses the recognized listing and locality to generate a ranked list of FRNs. The final automation decision stage is called the rejection stage. The rejection module estimates a confidence-like score on the top-ranked FRN from the ranked FRN list produced by the retrieval engine.
automation decision is made by comparing the score with a threshold; release the top-choice FRN if the score is above the threshold, else transfer the call to the operator.

It is very important to ensure that the decision strategy used in the third stage is designed to minimize the number of false accepts (FA). An FA occurs when the top-choice FRN is not the correct answer to the user’s query but is still automatically released to the user. The usual quality-of-service (QOS) metric is the ratio of the number of correct accepts (CA) to the number of FA’s, or what we call the CA/FA ratio. We now describe each of the three steps in the process.

5. SPEECH RECOGNITION

The DA query consists of two utterances, a locality utterance and a listing utterance. Listing utterances tend to be more complicated and exhibit more variations than locality utterances. In addition to various forms of the listing name itself, users commonly use one of a large number of prefix/suffix combinations. Occasionally, the listing audio also contains fragments of a conversation that the user may be engaged in with another human during the duration of the DA request. In order to effectively model the variations in listing utterances, we use tri-gram language models with an appropriate back-off [1]. The language model is trained on the available set of transcribed utterances for the FRNs targeted for automation. The language model training data is also augmented with information from the DA database; this augmentation is particularly helpful for FRNs that have sparse training.

For the DA application described in this paper, listing utterances are recognized using the BBN Byblos decoder configured with PTM models in the forward pass and SCTM models in the backward pass [2,3]. Locality recognition, on the other hand, is performed with a much simpler FSG.

6. FRN RETRIEVAL

As mentioned earlier, the key idea behind the new architecture is to allow an inexact match between the query and the training data. Rather than force an exact match, we estimate the probability of a particular inexact match being an acceptable one and use that probability to make an automation decision.

For the purpose of discussing the retrieval strategy, we consider the retrieval query to be made up of the query word being a composite of the listing name itself, users commonly use one of a large number of prefix/suffix combinations. Occasionally, the listing audio also contains fragments of a conversation that the user may be engaged in with another human during the duration of the DA request. In order to effectively model the variations in listing utterances, we use tri-gram language models with an appropriate back-off [1]. The language model is trained on the available set of transcribed utterances for the FRNs targeted for automation. The language model training data is also augmented with information from the DA database; this augmentation is particularly helpful for FRNs that have sparse training.

For the DA application described in this paper, listing utterances are recognized using the BBN Byblos decoder configured with PTM models in the forward pass and SCTM models in the backward pass [2,3]. Locality recognition, on the other hand, is performed with a much simpler FSG.

5. SPEECH RECOGNITION

The DA query consists of two utterances, a locality utterance and a listing utterance. Listing utterances tend to be more complicated and exhibit more variations than locality utterances. In addition to various forms of the listing name itself, users commonly use one of a large number of prefix/suffix combinations. Occasionally, the listing audio also contains fragments of a conversation that the user may be engaged in with another human during the duration of the DA request. In order to effectively model the variations in listing utterances, we use tri-gram language models with an appropriate back-off [1]. The language model is trained on the available set of transcribed utterances for the FRNs targeted for automation. The language model training data is also augmented with information from the DA database; this augmentation is particularly helpful for FRNs that have sparse training.

For the DA application described in this paper, listing utterances are recognized using the BBN Byblos decoder configured with PTM models in the forward pass and SCTM models in the backward pass [2,3]. Locality recognition, on the other hand, is performed with a much simpler FSG.

6. FRN RETRIEVAL

As mentioned earlier, the key idea behind the new architecture is to allow an inexact match between the query and the training data. Rather than force an exact match, we estimate the probability of a particular inexact match being an acceptable one and use that probability to make an automation decision.

For the purpose of discussing the retrieval strategy, we consider the retrieval query to be made up of the query word being a composite of the listing name itself, users commonly use one of a large number of prefix/suffix combinations. Occasionally, the listing audio also contains fragments of a conversation that the user may be engaged in with another human during the duration of the DA request. In order to effectively model the variations in listing utterances, we use tri-gram language models with an appropriate back-off [1]. The language model is trained on the available set of transcribed utterances for the FRNs targeted for automation. The language model training data is also augmented with information from the DA database; this augmentation is particularly helpful for FRNs that have sparse training.

For the DA application described in this paper, listing utterances are recognized using the BBN Byblos decoder configured with PTM models in the forward pass and SCTM models in the backward pass [2,3]. Locality recognition, on the other hand, is performed with a much simpler FSG.

5. SPEECH RECOGNITION

The DA query consists of two utterances, a locality utterance and a listing utterance. Listing utterances tend to be more complicated and exhibit more variations than locality utterances. In addition to various forms of the listing name itself, users commonly use one of a large number of prefix/suffix combinations. Occasionally, the listing audio also contains fragments of a conversation that the user may be engaged in with another human during the duration of the DA request. In order to effectively model the variations in listing utterances, we use tri-gram language models with an appropriate back-off [1]. The language model is trained on the available set of transcribed utterances for the FRNs targeted for automation. The language model training data is also augmented with information from the DA database; this augmentation is particularly helpful for FRNs that have sparse training.

For the DA application described in this paper, listing utterances are recognized using the BBN Byblos decoder configured with PTM models in the forward pass and SCTM models in the backward pass [2,3]. Locality recognition, on the other hand, is performed with a much simpler FSG.
Finally, the FRN set is ranked in order of decreasing $P(\text{FRN} | Q)$ with the top ranked FRN being the one most likely to be the correct response to the user query.

7. REJECTION

After the retrieval engine returns the top-choice FRN, the rejection component makes the decision as to whether this FRN should be automatically released to the user or the query should be routed to a human operator. To deliver a high quality of service, it is necessary that the CA/FA ratio be as high as possible. The operation is essentially a two-class classification task which we solve using a Generalized Linear Model (GLM) classifier [5] that computes a confidence-like score on a set of features derived from the query.

The two most important features used in the rejection step are the required and allowable word sets. As the name suggests, the required word set for a particular FRN is a set of word tuples, at least one of which must be present in the recognized listing for the query to be associated to that FRN. On the other hand, the allowable word set for a particular FRN is a list of words that are allowable in the recognized listing if the query is to be associated to that FRN. Together, the required and allowable words provide a powerful filtering technique that weed out a large fraction of the false automations. Other features used in the rejection stage are word confidences, speech recognition n-best frequency, and various other scores from the recognizer and the retrieval engine. The top-choice FRN is automatically released to the caller only if the rejection score is above a pre-set threshold. Otherwise, the call is routed to a human operator. The rejection component cuts down the FA’s by a factor of about five while keeping the number of CA’s constant.

8. RESULTS

Table 1 shows the experimental results comparing the FSG system described in Section 3 and the new IR-DA system. The first row of the table lists the results from Section 3. Measured on the same test set, the IR-DA system delivers a CA of 17.7% at an FA of 0.76%, or a QOS CA/FA ratio of 22. By dividing the CA by the percentage FRN coverage in Table 1, we see that the new system demonstrates a higher efficiency in automating FRN traffic coverage. In particular, while the FSG-based system was only able to automate one of every 3 calls asking for an FRN, the IR-DA system is able to automate one out of every 2.3 such calls.

A significant advantage of the new IR-DA system is that one can choose an appropriate operating point (say, a particular QOS number) by varying the rejection threshold. We believe that the new scalable DA automation architecture can be easily extended to higher ranges of automation by increasing the size of the FRN set and by improving the retrieval efficiency even further.

<table>
<thead>
<tr>
<th>System</th>
<th>Percentage traffic covered by FRN set</th>
<th>Correct Automation</th>
<th>False Automation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSG</td>
<td>31%</td>
<td>10.5%</td>
<td>0.50%</td>
</tr>
<tr>
<td>IR-DA</td>
<td>41%</td>
<td>17.7%</td>
<td>0.76%</td>
</tr>
</tbody>
</table>

Table 1: Automation rates for the system that uses only finite-state grammars (FSG) versus the new system which includes information retrieval (IR-DA).

9. CONCLUSIONS

The goal of the directory assistance automation system described in this paper was to achieve the highest automation rates without changing the customer call flow and with minimal impact on the user experience. We presented a novel approach that utilized state-of-the-art technologies in statistical recognition and information retrieval, resulting in high automation rates. The new architecture is scalable to higher ranges of automation by increasing the size of the FRN set and by improving the retrieval efficiency even further. Also, the approach is readily extended to more sophisticated call flows that involve extensive dialogs.

10. REFERENCES


