Incremental Learning of New User Formulations in Automatic Directory Assistance

M. Andorno, L. Fissore, P. Laface*, M. Nigra, C. Popovici, F. Ravera, C. Vair*

* Politecnico di Torino, Italy
{Marco.Andorno,Pietro.Laface}@polito.it
Loquendo, Torino, Italy
(Cosmin.Popovici,Luciano.Fissore,Mario.Nigra,Franco.Ravera,Claudio.Vair}@loquendo.com

Abstract

Directory Assistance for business listings is a challenging task: one of its main problems is that customers formulate their requests for the same listing with great variability. Since it is difficult to reliably predict a priori the user formulations, we have proposed a procedure for detecting, from field data, user formulations that were not foreseen by the designers. These formulations can be added, as variants, to the denominations already included in the system to reduce its failures.

In this work, we propose an incremental procedure that is able to filter a huge amount of calls routed to the operators, collected every month, and to detect a limited number of phonetic strings that can be included as new formulation variants in the system vocabulary.

The results of our experiments, tested on 9 months of calls that the system was unable to serve automatically, show that the incremental procedure, using only additional amount of data collected every month, is able to stay close to the (upper bound) performance of the not incremental one, and offers the possibility of periodically updating the system formulation variants of every city.

1. Introduction

Directory Assistance is the most used service that Telecom operators offer to their customers. This service is expensive because it relies on a multitude of human operators. A strategy for reducing these costs is to reduce the work time of the operators providing information collected by an Automatic Speech Recognizer, another strategy is to rely as much as possible to complete automation. This second strategy has been selected by Telecom Italia, that has deployed since the beginning of year 2001 a nationwide automatic DA system, jointly developed with Loquendo that routinely serves customers asking for residential and business listings. Whenever the automatic system is unable to terminate the transaction with the customer, the call is routed to a human operator. Descriptions of the system have been presented in [1,2].

The analysis of the traffic has shown that about 80% of the DA customer accesses are related to business listings, it is important, thus, to improve the percentage of success of the automatic system for this class of calls.

The Loquendo approach to DA is based on large vocabulary isolated word recognition technology, where the sequence of words of a business listing is concatenated and transcribed as a single word, with possible silences in between. Since the content of the original records in the database does not, typically, match the linguistic expressions used by the callers, a complex processing step, described in Section 3, is needed for deriving a set of possible formulation variants (FVs) from each original records in the listing book.

A large percentage of user expressions, however, still remain uncovered by the FV database. Thus, we have proposed a procedure for detecting, from field data, user formulations that were not foreseen by the designers [2]. These formulations can be added, as variants, to the denominations already included in the system to reduce its failures. In particular we are interested in detecting new formulations for frequently requested listings, for example nicknames of hospitals or other public services, or user requests for the phone number of a popular TV talk show, that of course does not appear in the directory listings.

Our approach is based on partitioning the field data into phonetically similar clusters from which new user formulations can be derived.

Since the system is operational, we may easily collect huge amounts of data from the field, thus in this paper we propose an incremental procedure for learning new user formulations.

The paper is organized as follows: Section 2 gives a short overview of the Loquendo DA system. Section 3 and 4 detail the generation of formulation variants and the evaluation of their coverage. Sections 5 and 6 present our approach for incremental learning of new formulations, while Section 7 is devoted to the conclusions.

2. Loquendo DA system overview

The initial prompt of the system allows the customer to select between residential or business listings, then the city name is collected and finally the desired denomination. For business requests, the address is possibly asked to the user in the following cases: a) the system is not confident about the recognized denomination, b) there are ambiguities in the original database, c) the system cannot find the information in the list of the most frequently requested listings. The search for a business denomination is carried out first on the list of the most important or most frequently requested listings (TopList) that can include up to 25K entries. If the search fails on the TopList, it continues on the list of the denominations associated to the address provided by the user.

3. The rule based approach

The Italian telephone book listings includes more than 25.000.000 records, about 3.500.000 of which are business listings. Each record includes several fields. Of particular interest are three of them: denomination, description, and category. The first two fields contain the denomination/s, and possibly descriptions, related to a given phone number. This information was set up according to the indication of the subscriber, or added afterward by operators for easing their search, without any standard, and thus with a very large variability in the included data for different categories of business. The category field was available only for some old database releases, and it was not filled for several entries. Since this information - tagging a denomination with one of about 1600 categories - was rather precise it has been exploited for the generation of the list of expressions for business listings.

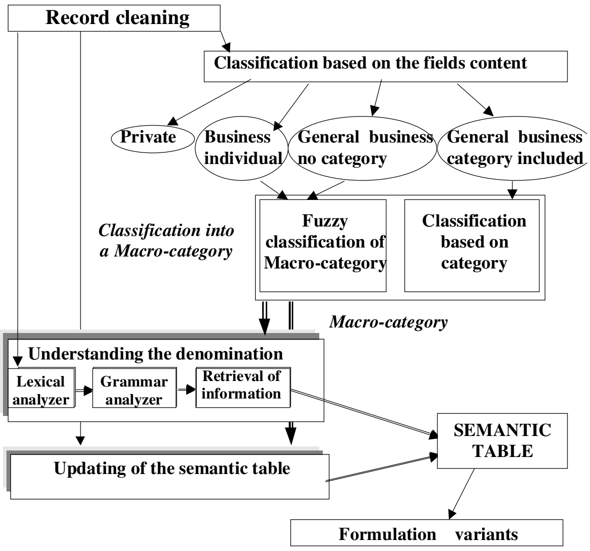
Record cleaning
Classification based on the fields content
Private
Business individual
General business no category
General business category included
Classification into a Macro-category
Fuzzy classification of Macro-category
Classification based on category
Macro-category
Understanding the denomination
Lexical analyzer
Grammar analyzer
Retrieval of information
SEMANTIC TABLE
Updating of the semantic table
Formulation variants

Figure 1*:* Generation of formulation variants

Fig. 1 summarizes the processing steps for generating the formulation variants.

The first step, *record cleaning*, aims at the normalization of the record content by eliminating spurious characters and performing systematic conversions.

The second step classifies the record into the four classes shown in the figure. The records belonging to the *general business classes* are the ones that require most of the efforts due to their complexity and their relevance for the overall DA system performance. To each record of this class is assigned one of 30 macro-categories, obtained from an a priori mapping of the original 1600 categories. The mapping has been designed according to an evaluation of the statistical relevance of a category within the entire database, its expected frequency of use, and the variability of its associated records.

The classification is trivial for records that include the category field. For the other ones, about 20% of the business listings, the macro-category is obtained by means of a fuzzy classifier. The classifier has been trained using the available records including the category field: statistics were collectedabout the frequency of the words occurring in the denomination field for record belonging to the same macro-category. Using these data it is possible to assign to a record the likelihood that it belongs to a given macro-category. In the third step, the denomination field is processed through the lexical and grammatical analysis. A specific grammar has been designed for each macro-category. The lexical analysis produces the terminal symbols, taking also care of mapping keywords typical of a macro-category to a single terminal symbol: the Italian words "bar, ristorante, trattoria, caffè, pizzeria", for example, are mapped to the terminal symbol <ristorazione>. The grammar analyser parses the denomination to detect and extract components with different semantic value.

Pref: 011 *Tel*: 5175296
City: TORINO *Prov*: TORINO
Address: 63, C. VITTORIO EMANUELE II
***Den*: BAR RISTORANTE LA FORCHETTA D'ORO DI MARIO ROSSI**
***Description*: PIZZA**
***Category*: RISTORANTI**

Figure 2*:* Example of some fields in a record

SEMANTIC TABLE		
CATEGORY	**NAME**	**OWNER**
BAR	**LA FORCHETTA D'ORO**	**MARIO ROSSI**
RISTORANTE		
PIZZERIA		
RISTORANTE		

Formulation variants

RISTORANTE LA FORCHETTA D'ORO
BAR LA FORCHETTA D'ORO
BAR RISTORANTE LA FORCHETTA D'ORO
PIZZERIA LA FORCHETTA D'ORO
LA FORCHETTA D'ORO

Figure 3*:* Semantic table and formulation variants

The last step is the semantic interpretation of the denomination performed by searching terminal symbols, grammatical rules, and sequences of non-terminal and terminal symbols.

Every information extracted has an assigned score according to the user model. These scores are used for the generation of the formulation variants. Since the information in the denomination field is sometime incomplete or ambiguous, it is possibly integrated with those included in other fields, such as the description field. The final result is a semantic table that summarizes the semantic content of the record, reducing the variability of the information inserted by the operators in the record fields. An example of semantic table for the record of Fig. 2 is shown in the top half of Fig. 3.

Using the semantic table information it is possible to generate the formulation variants with an associated score. Different generation rules have been devised for each macro-category. The generated formulation variants are shown at the bottom of Fig. 3, ordered by score. The best scoring formulation is also played back to the customer for confirmation.

4. Evaluation of the rule based approach

Several turns of evaluation of the coverage of the user formulations by the FVs have been performed. In a first phase, real user data were collected from the interactions

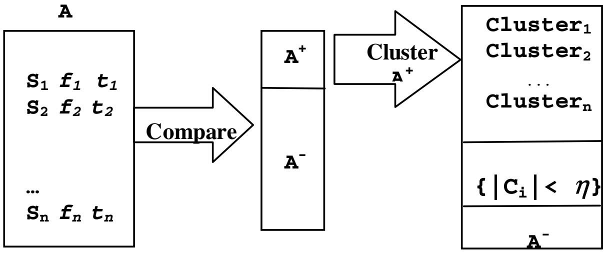

Figure 4: Clustering procedure

with human DA operators located in Turin. Then, other calls to a prototype DA, serving the Catania telephone district, were transcribed. From these preliminary tests it has been verified that the coverage of the original FVs was about 40%. It was thus mandatory to generate more accurate formulations for frequently requested listings, in particular for those presenting high failure rates.

Another large database was then collected from 13 call centers distributed in several regions of Italy. The database includes a total of 20216 transcribed calls, corresponding to the most frequently asked listings, associated to the phone number provided by the human operators.

To generate new, more accurate FVs, the transcribed denominations were analyzed, and generation rules derived, depending on the business category. The FVs that received most attention were those related to hospitals, social services, public utilities, communication and transportation agencies, and the like, because they account for the majority of the calls.

By using the FV rules derived from this new field data, the coverage of the FVs increased from 40% to more than 60%, using an average of 5 FVs per denomination: this also means that many users are rather collaborative, and that the system prompt elicits concise linguistic expressions.

Since the automatic DA system is currently fully operational, new FVs, and possibly rules, are derived whenever the service provider signals consistent anomalies.

5. Automatic learning of formulation variants

Since a large percentage of user expressions still remain uncovered by the FV database, we have proposed a procedure for detecting new user from field data [2]. These formulations can be added, as variants, to the denominations already included in the system to reduce its failures.

The recognition module of the system produces, together with the lexical constrained word hypotheses, the phonetic transcription of each utterance using a phone-looped model.

Our working hypothesis, confirmed by the experimental results was that, collecting a large number of requests for the same listing, there is high probability of obtaining clusters of phonetically similar strings, whose central elements, defined as the string that has the minimum sum of the distance from all the other elements of the cluster, are quite accurate phonetic transcriptions of (possibly new) user formulations.

To assess the capabilities of our approach during May and June 2001 a database was collected including 7.2M phonetic strings related to business listing calls routed to the operators due to system failure to complete its transaction with the customer. The calls in this database are distributed among the listings of 8100 different city names.

It is worth noting that the calls are routed to operators even if the recognized denomination is correctly found in the TopList, but it has not a large enough confidence score, or if it could match the correct denomination in the complete set of listings, but the address is not known to the user.

Since for these calls we don't have the phone number information associated with the phonetic string, but only city name, confirmed to the system by the user, we cluster all the strings associated with the same city name. As reported in [2], our procedure is able to detect a limited number of phonetic strings that are good candidate to become new FVs. In particular, processing this database we obtained a set of 12.3K new formulations covering 184K calls.

6. Incremental learning

Since we may easily collect huge amounts of data from the field, it is interesting, for the sake of efficiency of storage and computation, to evaluate an approach that allows an incremental use of these data for learning new user formulations.

Our approach for deriving new FVs is based on partitioning the field data into phonetically similar clusters. The clusters are grown by means of a furthest neighbor hierarchical cluster procedure on the basis of the frequency of occurrence of identical strings and accounting for the phonetic distance among the included strings. A graphical representation of the procedure is shown in Fig. 4.

Every phonetic string, S_i, in the set **A** to be clustered, has associated its occurrence count f_i, and a time tag. The time tag of a cluster is equal to the time tag of the youngest of its included elements. To compute the distance among a huge number of phonetic strings a beam search matching procedure is used because we are interested in clusters with small dispersion of the included elements [2].

Let's call A^+ the set of strings that have at least one neighbor within a preset distance. These strings are merged producing a set of significant clusters, characterized by high cardinality and small dispersion of the included phonetic strings. The central elements of these clusters are possibly new formulation variants. Another set of clusters, with a number of elements less than a preset threshold η, is kept in the set A^+ as a useful information for the incremental training process, while the remaining strings - without neighbors according to the preset distance – compose the set A^-.

The incremental learning procedure, illustrated in Fig. 5, is carried out according to the following steps:

1. Compute the distance among the phonetic strings in the new set **B**, and detect the set B^- of the strings without any neighbor.
2. Compute the distance among the strings in set **B** and the strings in the old set **A**. Please notice that the distances among the strings in the set **A** are already known. Generate the union of the neighbour sets A^+ and B^+ updating the occurrence frequency of the included strings.

   Cluster the union of the sets A^+ and B^+, obtaining the set AB^+ and a new set of clusters. The new clusters typically

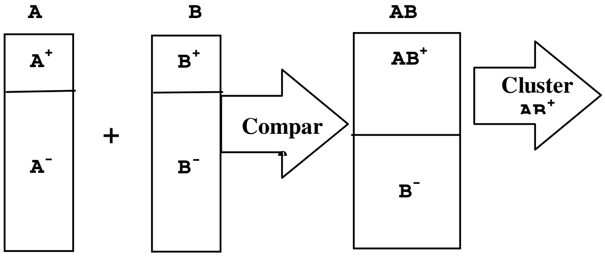

Figure 5: Incremental procedure

3. include old clusters, but they may have a better central element because their cardinality is larger.
4. The information that is kept for the next iteration of the learning procedure is:
   - set $\mathbf{AB}^+$ excluding
     - the strings belonging to significant clusters with very high cardinality. For these clusters we assume that no more data are necessary to possibly improve their central element; only the central element string is kept in $\mathbf{AB}^+$ as the cluster representative updating its occurrence frequency.
     - the strings older than **K** months.
   - set $\mathbf{B}^-$ of the strings without neighbors. The information of the old set $\mathbf{A}^-$ is discarded.

Table 3 compares the results of an experiment (labelled as "Not Incr") using an increasing amount of data given to the cluster procedure as a unique set, with respect to the incremental procedure using the same amount of data. In particular, we used 573K calls routed to the operators - collected in 6 months - for denominations in Torino.

Table 3 presents the number of significant clusters detected (Formulation Variants), the number of calls (strings) included in the clusters, and their percentage with respect to the total number of detected clusters or processed calls respectively.

It is interesting noting that both the number of formulations and the coverage of the calls continue to increase presenting new subset of data to the incremental procedure. Although the incremental procedure is unable to reach the (upper bound) performance of the not incremental one, it is very close to it. Moreover, as can be seen in the last four (shaded) rows of Table 3, using incrementally 3 additional months of data (up to May 02) we recover most of the FVs that were missing in the set incrementally produced using 6 months of data only. Thus, our procedure offers a viable approach for updating periodically the system formulation variants of every city. The proposed procedure is routinely applied to detect the set of user formulations that are most frequently responsible of system failures. There are several reasons for failures that can be corrected by introducing or replacing system FVs:

- the formulation cannot be found in the original record in the written database (nicknames, abbreviations, acronyms, and the like)
- the FV was not foreseen by the rule based approach
- the denomination was not in the list associated to the city name given by the user, but possibly in a nearby hamlet.

Month	Phonetic Strings		Formulation Variants (FVs)			Number of Calls		
	Not Incr	Incr	Not Incr	Incr	Diff (%)	Not Incr	Incr	Diff (%)
Sept 01	90658	90658	483	483	0.00	6699	6699	0.0
Oct 01	202532	111874	1241	1241	0.00	21725	21725	0.0
Nov 01	298843	96311	1811	1786	1.38	36581	36362	0.59
Dec 01	372147	73304	2324	2241	3.57	49305	48584	1.46
Jan 02	483076	110929	3018	2871	4.87	69654	68096	2.23
Feb 02	572941	89865	3658	3395	7.19	87371	84760	2.99
Mar 02		86362		3879	4.45		100762	1.21
Apr 02		80732		4305	3.14		116654	0.87
May 02		31417		4479	2.76		122863	0.75

Table 3: Incremental versus not-incremental approach

For the most frequently requested denominations, the formulations that most frequently failed were included in the TopList of a new release of the DA system replacing some rule-based formulations that never appeared in 9 months.

The new release of the system, with the field adapted vocabulary, has been tested on a single server of a centre in Torino while keeping the old release on another 9 servers of the same centre. Comparing the statistics collected in a few months of the previous and new releases we appreciated a 3% absolute reduction of the calls routed to the human operators.

7. Conclusions[1]

We presented the Loquendo approach to Directory Assistance and proposed an incremental procedure that is able to filter a huge amount of calls routed to the operators, and to detect new user formulations that were not foreseen by the designers of the DA system. These formulations have been used to update the system vocabulary giving a significant reduction of the system failures on a field test set.

We are continuing the monthly collection of data. In particular, we are now able to associate the phone number provided by the human operators to the failed calls. This information provides us another resource toward producing a completely automatic inline procedure for learning the user formulations.

8. References

[1] R. Billi, F. Canavesio, C. Rullent, Automation of Telecom Italia Directory Assistance Service: Field Trials results, Proc. IVTTA 1998, Turin, pp. 11-16, 1998.

[2] C. Popovici, M. Andorno, P. Laface, L. Fissore, M. Nigra, C. Vair, "Learning New User Formulations in Automatic Directory Assistance", Proc. ICASSP 2002, Orlando, FL, pp. 17-20, May 2002.

[1] This work was partially supported by the EU ITS Project SMADA Speech Driven Multi-modal Automatic Directory Assistance