Call Classification with Hundreds of Classes and Hundred Thousands of Training Utterances ... ... and No Target Domain Data

David Suendermann, Phillip Hunter, and Roberto Pieraccini

SpeechCycle, Inc., New York City, USA
{david, phillip, roberto}@speechcycle.com
http://www.speechcycle.com

Abstract. This paper reports about an effort to build a large-scale call router able to reliably distinguish among 250 call reasons. Because training data from the specific application (Target) domain was not available, the statistical classifier was built using more than 300,000 transcribed and annotated utterances from related, but different, domains. Several tuning cycles including three re-annotation rounds, in-lab data recording, bag-of-words-based consistency cleaning, and recognition parameter optimization improved the classifier accuracy from 32% to a performance clearly above 70%.

1 Introduction

The introduction of natural language processing to automate call routing about ten years ago [Gorin et al., 1997] has led to a strong interest in the development of statistical call classifiers as an enabling technology for interactive voice response (IVR) applications. The goal of a statistical spoken language understanding (SSLU) classifier is that of mapping a natural language utterance—typically a caller's response to an open-ended question—to one of a given set of categories, or classes, of call reasons. Today, SSLU, typically performed by natural language speech recognition followed by a sentence classifier, is often used as a more sophisticated replacement of menu-based systems using dual-tone multi-frequency (DTMF) [itu, 1995] technology (*... push 1 for billing push 2 for sales ...*) or speech-recognition-based directed dialog (*... you can say billing, sales, or ...*). While both DTMF and directed dialog can, in principle, provide very high accuracy routing, these simple solutions are often not practical for several reasons:

– In certain applications, the number of classes can be too large to be handled in a single menu. Even succession of menus hierarchically structured would prove unwieldy with hundreds of classes, not to mention the bad caller experience when five or six menu levels are required to reach the correct routing point.

Table 1. Domains covered by the call classifier with examples and number of classes.

domain	description	examples	classes
TV	cable television support	cable box issues, picture problems, On-Demand and Pay-Per-View orders	79
Internet	broadband internet support	internet and e-mail problems, setup, equipment, security	63
Phone	telephone support	voice mail, caller ID, phone features, dial tone	62
General	everything not covered by the above	billing, orders, appointments	46
Target	everything covered		250

- Even when prompted with a clear menu, callers often describe the reason why they are calling in their own words, and that may not be covered by the rule-based grammar typically used with directed dialog systems.
- For complex domains, callers may not understand or be familiar with the terms used in the menu. For example in response to the prompt: *Do you have a hardware, software, or configuration problem?*, they may respond unexpectedly (*My CD-ROM does not work!*) or choose one of the options at random without really knowing if it applies to their case.

Hence, for very complex scenarios like the one we will discuss in the following, the use of natural language call classification is the only feasible solution. The interactive voice response application described in this paper is designed for the customer service hotline of a large US cable provider and has to process a variety of call reasons belonging to one of the four domains introduced in Table 1.

State-of-the-art call classifiers as those described in [Evanini et al., 2007] are based on a statistical model trained on a large number of sample utterances. In commercial interactive voice response systems, utterance gathering is usually performed in several steps:

1. At first, a few thousand utterances are recorded by a simple collection application which prompts callers to describe the reason they are calling and then, after having recorded the utterance, transfer them to a traditional routing system. Speech recognition and utterance classification are not used during this step. This type of collection is typically limited in time and in volume of calls, since prompting for call reason and then being transferred to a different system which collects the call reason again produces a bad caller experience, and providers are generally averse to impose that to a large number of customers.
2. An initial classifier[1] is built based on the utterances collected in Step 1. The performance of this initial classifier is usually far from satisfactory because of the limited number of samples.

[1] In this work, we use a maximum-likelihood classifier with boosting similar to the one described in [Evanini et al., 2007].

3. The initial classifier from Step 2 is incorporated into the system, and call routing is performed based on its output. In order to limit the negative caller experience produced by a poorly performing classifier, special care is taken in confirming and rejecting low confidence output and following up with properly designed backup directed dialogs. Massive utterance collection is performed at this stage.
4. At reasonable intervals, new classifiers are trained based on the complete set of available utterances iterating over Steps 2 and 3 until the performance reaches a satisfactory level.

The particular challenge we faced in the application described here was due to a customer constraint forcing us to skip Step 1 of the above procedure. So, an initial collection of utterances was not available, and the initial classifier was to be build without appropriate training data from the Target domain. In order to move to Step 2 with a reasonable classifier, we decided to rely on a large amount of data collected from other deployed applications. Moreover, the number of classes was significantly larger than with comparable classification scenarios which usually incorporate less than 100 classes [Evanini et al., 2007]. A preliminary analysis of the call reasons of the Target domain revealed that the number of classes required for this application was 250.

In the following, we discuss data resources, design, and test environment of the call classifier including the steps undertaken to face the challenges introduced by this project. The performance of the classifier, measured at each step of the process on a limited set of test utterances obtained during an initial deployment of the application, is reported.

2 Data Recources

As discussed in the introduction, no data specifically collected for the Target domain was available at the beginning of the project, and the customer required automated call routing to be performed in the first deployment. Thus, we decided to rely on a large corpus of transcribed and annotated speech including more than 2 million utterances collected during the deployment of systems designed for the automation of TV, Internet, and Phone sub-domains (for details, see [Acomb et al., 2007]). Only utterances recorded at the initial open prompt in those systems were considered.

In order to preserve the frequency distribution of the categories in each sub-domain, we performed an unbiased selection of the samples by using all the utterances in a given time range of the collection. Table 2 shows the resulting number of utterances in each sub-domain including a transcription of the prompt used for their collection. In contrast, the prompt used in the Target application was:

> Briefly tell me what you're calling about today, for example *I'm having trouble getting online*, or you can say *Give me some choices*.

Table 2. Number of utterances and prompt for a given domain used for the development of the call classifier.

domain	utterances	prompt
TV	32,149	Please briefly describe the reason for your call.
Internet	94,598	Briefly describe the problem, saying something like *I can't send or receive email*. Or you can say *What are my choices?*
Phone	10,819	I'll connect you with someone who can help. To get you to the right place, tell me briefly what you're calling about.

The utterances spoken by callers are affected by the prompt used. Different prompts produce a different distribution on the variety of language used. The prompt for the Internet domain for instance clearly encourages the caller to either say *I can't send or receive email* or *What are my choices?*, whereas the other prompts are entirely open. Very rarely, callers would ask for their choices in this case. This was confirmed looking at the data, where the word *choices* appeared 2,609 times in the Internet domain, whereas there was only 1 occurrence in the TV and Phone domains. Thus, the utterances collected in the different sub-domains poorly reflect the linguistic distribution of the utterances in the Target domain. Another problem with using the available corpora from the sub-domains is due to the different contexts in which the utterances were recorded. In fact, in all those sub-domains, the caller had consciously selected a particular technical support application (TV, Internet, or Phone) before being prompted to speak the reason of the call. Consequently, very often, references to the actual domain are not explicitly mentioned, since it was implied by the initial selection of the caller. As an example, the utterance:

it's not working

may appear as a sample in each sub-domain meaning:

- that *the cable modem is not working* in the TV domain,
- that *there is no Internet connection* in the Internet domain,
- that *something is broken* in the Phone domain.

Instead, when read in the Target domain the very same utterance means that *the reason for calling is completely unclear*. Consequently, by blindly merging the sub-domain corpora, utterances like the one in the above example would appear in several different classes producing an incorrect approximation of the statistical model for the Target domain. We solved this problem by iteratively re-annotating the available corpus according to the Target domain specification.

After the initial call classifier was deployed, we recorded 2991 utterances[2]. These utterances were transcribed and annotated and used as test corpus.

[2] This happened only shortly before the submission deadline of this publication preventing us from collecting more data and further tuning the call classifier.

Table 3. Accuracy of the call classifier from the baseline to the production system.

version	utterances	enhancements	accuracy
I	164,523	live data from applications + sample utterances from designers	32.4%
II	194,077	re-annotation round 1	44.5%
III	302,912	in-house utterance recording + re-annotation round 2	62.8%
IV	242,283	recognizer tuning + re-annotation round 3 + annotation consistency check	71.6%

3 From Baseline to Production

This section reports about our efforts to achieve the best classification performance given the constraints and the challenges described in Section 1. The classification accuracy on the test set of 2991 utterances introduced in Section 2 is reported in Table 3 for each step of the process.

I Just Let It Go

In spite of the arguments pointed out in Section 2 (different prompts, different contexts), we wanted to estimate the baseline performance by blindly merging the sub-domain corpora without any adaptation to the Target domain. However, since only utterances from three domains (TV, Internet, and Phone) were available, application designers were asked to provide a number of example utterances for each class of the missing domain (General). In total, 290 example utterances were produced, annotated, and merged with the rest of the corpus for the initial training. The utterance counts were artificially adjusted to balance the small number of General domain utterances with the large numbers of the other three sub-domains.

II Not Quite There—Let's Get Rid of Major Confusion

By examining the confusion matrix obtained from a test of the baseline classifier (cf. Section 3.I) on several thousand utterances of the same type like the baseline training data, we observed that 30% of the utterances were not only assigned a wrong class but a wrong domain. Consequently, as predicted in Section 2, the context indeed seems to play a significant role in our scenario. Thus, as a first step to get rid of such confusions, classes showing excessive misclassification rates in the confusion matrix were isolated and subject to a first re-annotation round.

III Still Not Great—We Really Need Live Examples

Although the overall performance already had significantly improved, there was a clear demand for utterances particularly in the General domain, being the one

lacking live training data. The examples provided by the application designers for the initial baseline classifier seemed to be too artificial and different from utterances we would experience in the live system. As a consequence, we set up a platform for recording, transcribing, and annotating calls placed by about 50 subjects recruited internally to the company. A collection system was implemented, and each subject was asked to call it and produce 40 utterances, one for each one of a corresponding number of classes. Utterance categories were randomly distributed among callers with a bias towards those that showed lower performance in the initial experiment. A simple description of each class was provided to the speaker, so as to solicit a reasonably natural response which could also include conversational artifacts such as hesitations, repetitions, linguistic violations, colloquial speech, etc.

A total of 1784 utterances was collected in this step and used in conjunction with the rest of the corpus which included almost 100 times more utterances. To compensate for this count imbalance, the newly collected utterances were split into 67% training and 33% development data. Before merging them with the rest of the corpus, the new utterance counts were inflated by a multiplicative factor learned on the development set. Figure 1 shows the influence of this multiplication factor on the call router accuracy measured on the development set. After including these new utterances, most confused classes were extracted and re-annotated as described in Section 3.II.

IV Let's Tease out The Optimum

As a last step, the training had to be optimized, and the corpus now including more than 300,000 utterances had to be cleaned up as follows:

- speech recognizer tuning (in particular the weighting between acoustic and language model),
- a complete screening of the most frequent utterances among all classes to determine major incorrect annotations still in the data—a third re-annotation round was carried out,
- an annotation consistency cleaning based on bag-of-words matching isolating similar but differently annotated utterances, and
- removal of utterances being over-represented in the data like the one adverted by the Internet prompt in Table 2 (*I can't send or receive email*).

4 Conclusion

Although there was no live training data available for the call classifier's Target domain, we achieved a rather decent accuracy of more than 70% on a task with 250 different classes. This result more than doubled the baseline accuracy and was achieved through a careful re-annotation process involving more than 300,000 utterances and an effort to model the Target behavior by performing in-lab live recordings with 50 people involved.

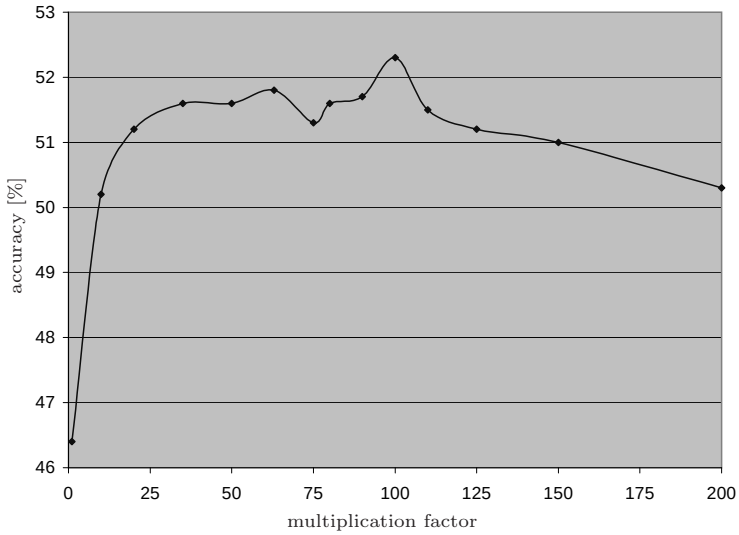

Fig. 1. Influence of the multiplication factor on the call classification accuracy.

After the call classifier will be rolled out in a production system, a considerable number of live utterances will be collected and used to further enhance the performance either by enriching the existing classifier or by completely rebuilding it whatsoever achieves higher scores.

References

[itu, 1995] (1995). Interactive Services Design Guidelines. Technical Report ITU-T Recommendation F.902, ITU, Geneva, Switzerland.

[Acomb et al., 2007] Acomb, K., Bloom, J., Dayanidhi, K., Hunter, P., Krogh, P., Levin, E., and Pieraccini, R. (2007). Technical Support Dialog Systems: Issues, Problems, and Solutions. In *Proc. of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies*, Rochester, USA.

[Evanini et al., 2007] Evanini, K., Suendermann, D., and Pieraccini, R. (2007). Call Classification for Automated Troubleshooting on Large Corpora. In *Proc. of the ASRU*, Kyoto, Japan.

[Gorin et al., 1997] Gorin, A., Riccardi, G., and Wright, J. (1997). How May I Help You? *Speech Communication*, 23(1/2).