FROM RULE-BASED TO STATISTICAL GRAMMARS: CONTINUOUS IMPROVEMENT OF LARGE-SCALE SPOKEN DIALOG SYSTEMS

D. Suendermann, K. Evanini, J. Liscombe, P. Hunter, K. Dayanidhi, R. Pieraccini

SpeechCycle Labs, New York, USA

{david, keelan.evanini, jackson, phillip, krishna, roberto}@speechcycle.com

ABSTRACT

Statistical Spoken Language Understanding grammars (SSLUs) are often used only at the top recognition contexts of modern large-scale spoken dialog systems. We propose to use SSLUs at every recognition context in a dialog system, effectively replacing conventional, manually written grammars. Furthermore, we present a methodology of continuous improvement in which data are collected at every recognition context over an entire dialog system. These data are then used to automatically generate updated context-specific SSLUs at regular intervals and, in so doing, continually improve system performance over time. We have found that SSLUs significantly and consistently outperform even the most carefully designed rule-based grammars in a wide range of contexts in a corpus of over two million utterances collected for a complex call-routing and troubleshooting dialog system.

Index Terms— Statistical Spoken Language Understanding, SSLU, statistical grammars, dialog systems, continuous improvement, very large data sets

1. INTRODUCTION

Today's Third Generation Dialog Systems [1] are often very complex. They may consist of hundreds of dialog states involving extensive dialog structures, have system integration functionality that communicates with backend databases or devices, support multiple input and output modalities, and can sometimes comprise more than 20 minutes in call duration. In order to keep a caller engaged in such environments, the use of human-like speech processing is critical, e.g., the incorporation of natural language understanding, mixed-initiative handling, and dynamic response generation.

Natural language understanding on a large scale was first introduced to automated spoken dialog systems as call classifiers about ten years ago [2]. Here, the caller was asked a general question at the top of the call, such as, "Briefly tell me what you're calling about today." The caller's utterance was transcribed using a speech recognizer, and the caller was routed to a human agent based on a parse of the utterance produced by a semantic classifier. The human agent then interacted with the caller providing services including, e.g., technical problem solving, billing support, or order processing.

Third Generation Dialog Systems, by contrast, are designed to emulate the human agent's role to a far greater degree in the length of interaction and the complexity of the services offered. At the same time, as dialog systems improve, so too do the expectations of callers. Several characteristics of modern dialog system design encourage callers to behave as if they were interacting with a human agent. Such characteristics include open-ended questions at the very beginning of a conversation and offering global commands such as "help" and "repeat" at every point in the dialog. This design encourages callers to say things that are not explicitly prompted by the context prompts in the dialog system. Furthermore, explicit directed dialog prompts in which callers are asked to choose an item from a list often unintentionally elicit out-of-grammar utterances from callers by offering choices that may be incomplete, too vague, or too specific.

How, then, it is possible to satisfy the expectation of natural language understanding at every single moment during the call when the caller's behavior is often unpredictable to an interaction designer? Even listening to hundreds of calls will hardly provide a broad understanding of what exactly is going on at every point in a dialog system that receives millions of calls every month. It is barely possible to satisfy this expectation with the still-common approach of using static, hand-crafted, rule-based grammars [3].

Instead, we propose a method to continuously improve dialog context performance by using caller utterances to tune SSLUs and use them at every dialog recognition context. In the process outlined herein, utterance collection, transcription, annotation, language model and classifier training, baseline testing, and grammar releasing are carried out automatically[1], in an eternally running cycle. The goal is to ensure continual improvement of system behavior and to obtain the highest possible recognition performance reflecting current caller behavior. Our own implementation of this procedure has shown significant recognition improvement over rule-based grammars. This finding was validated on over two million utterances from more than half a million full calls to a complex call-routing and troubleshooting dialog system.

Patent pending.

[1]Transcription and annotation are only partially automated and require human supervision to some extent.

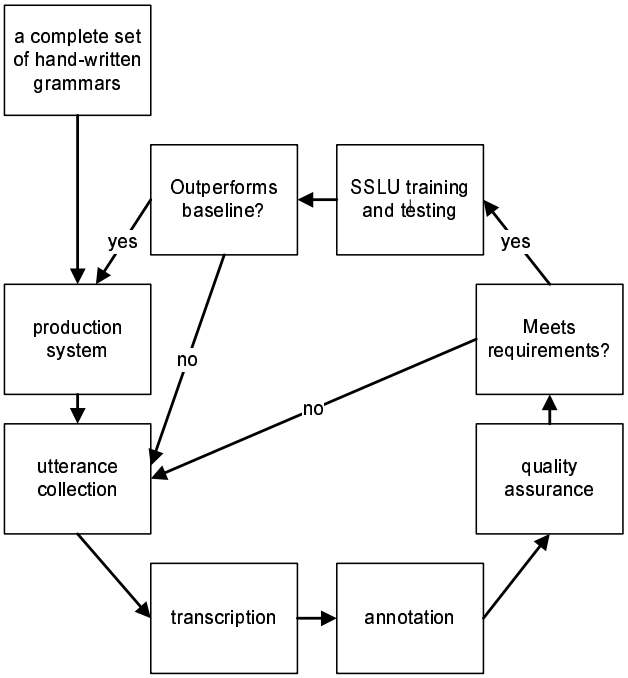

Fig. 1. The continuous grammar improvement cycle.

2. THE CONTINUOUS IMPROVEMENT CYCLE

This section outlines a method for incorporating continuous recognition improvement into every recognition context of a spoken dialog system. Figure 1 shows a high-level view of the continuous improvement cycle.

2.1. The Initial System

For every novel dialog state that requires a new set of semantic parses, a rule-based grammar should be used as a first approximation. In time, given data collected from actual callers, an SSLU that more accurately reflects the distribution of caller utterances can be created using the method described in the following sections.

2.2. Utterance Collection, Transcription, Annotation, and Quality Assurance

First, a random sampling of caller utterances at every recognition state of an in-production dialog system should be collected, transcribed, and annotated for semantic meaning according to the expected parses returned by the system. In order to achieve reliability and consistency among annotations, a rigorous quality assurance procedure must be carried out. Furthermore, in order to facilitate automation of the continuous improvement cycle as much as possible, criteria thresholds should be set (collectively referred to as C) to flag whether it is appropriate to begin training a new SSLU for a given recognition context. Such quality measures and criteria thresholds include (see [4] for details):

- *Completeness*. Only utterances from a date range including a complete set of annotations are considered. This is to make sure that the classes and utterances match the real distribution.
- *Consistency*. Identical utterances (and, optionally, bags-of-words) are required to be assigned to the same semantic class.
- *Congruence*. The parse provided by the initial rule-based grammar for the transcribed utterance must produce the same result as the annotation. Of course, this check is only available when the utterance can be parsed by the rule-based grammar.
- *Coverage*. To assure that the application is able to evaluate the caller response in most of the cases, the grammar coverage should be as high as possible. If an utterance is considered out-of-scope in the current context it gets assigned a garbage class. Examples include noise events, background speech, and cursing. However, reasonable utterances that are not yet covered by the call flow logic also go into the garbage class. If the number of utterances ending up in the garbage class is too high, the issue must be addressed by changing the dialog flow or prompt to accommodate caller behavior and/or by adding new classes to the grammar.
- *Corpus Size*. In order to benchmark grammar performance, a test set of a minimum size must be available. This test set must consist of data never used for training and tuning purposes in order to not bias the results. Also, it is important that the test data is very recent in order to account for trends in the application and caller behavior. The remaining data is split into training and development sets.

2.3. SSLU Training and Testing

Whenever the available data fulfills the above requirements for a recognition context, several tuning tests are performed to arrive at an SSLU that performs best on a development set given several optimization parameters. After building an SSLU for a recognition context, its performance, P_{new}, is compared to the performance of the grammar currently used in production, P_{old}, on the same test set. If P_{new} is significantly better than P_{old} then the new grammar replaces the old one. Additionally, a statistical measure p of the difference between P_{old} and P_{new} is applied consistently to verify that the new grammar is reliably better than the current one. If the new grammar does not outperform the original one then the original grammar is left in production to collect more utterances with which to, potentially, train a more accurate grammar in the future.

2.4. Iteration

Steps 2.2 and 2.3 are carried out in an eternal cycle providing more and more data and producing better and better SSLUs.

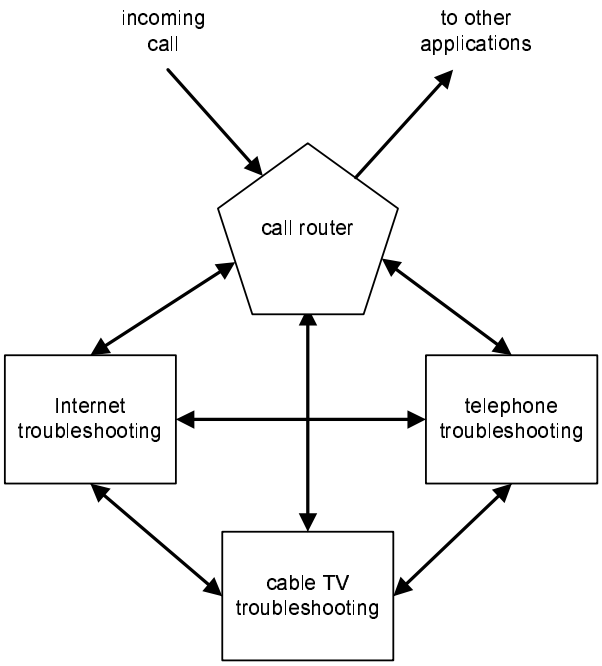

Fig. 2. Breakdown of the target application into individual dialog systems and their connection with each other.

criteria C	
minimum test set size	1,000 utterances
minimum coverage	90%

performance thresholds	
performance: $P = \frac{\text{correctly classified utterances}}{\text{total utterances}}$	$P_{new} - P_{old} > 0$
significance: χ_1^2 test	$p < 0.05$

SSLUs	
language model	trigram + smoothing
classifier	naïve Bayes + boosting

Table 1. Parameter settings.

utterances	2,184,203
calls	533,343
activities	2,021
grammars	145
original average performance (June 2008)	77.97%
average performance to-date	**90.49%**

Table 2. Data resources and grammar performance as of September 2008.

At some point after at least several cycles, we can expect to reach saturation in performance, at which the algorithm would not release subsequent grammars because statistically significant differences in performance will not be found. However, the recognition context should still be incorporated into the continuous improvement cycle as a monitoring device. It is usually the case that caller behavior changes over time for multiple reasons either in utterance distribution or in the ways of describing semantic classes. In effect, then, our continuous improvement cycle would seamlessly and correctly respond to this event.

3. A CASE STUDY: CONTINUOUS IMPROVEMENT CYCLE IMPLEMENTED

3.1. The Target Application

The dialog application used for this research comprises four individual dialog systems interacting with each other. They are implemented in the customer care telephone portal of one of the largest US cable service providers. Figure 2 shows the principal design of this application. When customers call the hotline of the cable provider, they are connected to the top-level call router whose task is to determine the call reason and route the callers to the appropriate destination. This is done by accessing the callers' account information (using their telephone number as an identifier) and then asking either a general opening question such as the one discussed earlier ("Briefly tell me what you're calling about today") or a caller-specific question such as "It looks like you called recently about your account. Are you calling about that now?" Depending on the caller response to the opening question and, potentially, to one or two follow-up questions, the most appropriate routing point is determined, and the call is transferred. If the call is about a technical problem with one ore more of the provider's services (broadband Internet, cable TV, or telephone), the call is connected to one of the three respective troubleshooting dialog systems. If customers face problems with more than one service, they can be interconnected to one of the other troubleshooting dialog systems or back to the call router.

3.2. Settings And Data

Table 1 gives the settings used for the continuous improvement cycle, and Table 2 provides an overview of the data resources used in this research.

3.3. Results

When the first version of the application was launched at the end of June 2008, the average performance of all rule-based grammars was around 78%. This includes directed dialogs, lower performing activities with open prompts, and higher performing standard contexts (such as yes/no), all weighted by their frequencies of use. After three months, almost 2.2 million utterances had been transcribed and annotated and had circulated dozens of times through the grammar improvement cycle. Whenever a grammar significantly outperformed the most recent baseline, it was released and put into production leading to an incremental improvement of

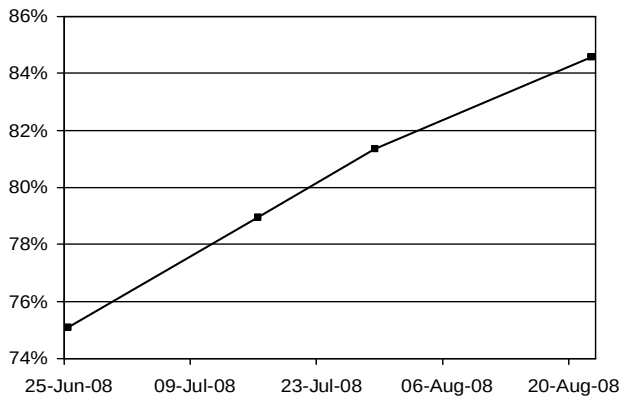

Fig. 3. Performance of different versions in the continuous improvement cycle of the top-level large-vocabulary SSLU.

performance throughout the application. As an example, Figure 3 shows the performance improvement of the top-level large-vocabulary SSLU that distinguishes more than 250 different classes (for details about how annotation can be carried out on such high-resolution SSLUs, see [5]). Almost every two weeks, there was enough data collected in the cycle that a new version could be released. To date, more than 100,000 utterances have been collected for this grammar; nevertheless, its performance does not yet seem to be saturated.

The overall performance of the application went up to more than 90% within three months of the introduction of the continuous improvement cycle. An important observation in the scope of this research was that for every single one of the grammars whose data met the quality requirements, the SSLU outperformed its rule-based counterpart. This shows once again[2] the advantage of the statistical approach in comparison to the rule-based one that only trusts in human experience and intuition. The following two examples emphasize this finding:

- Let us suppose a caller has trouble with getting online, and she gets transferred to the Internet troubleshooting system which helps her to get connected. At the end of this process, the caller is asked to access a certain website to make sure she is back online. At this activity, she is expected to respond with utterances such as "I am connected", "still no Internet", "repeat the address, please", or one of the global utterances "I need help", "hold on", "repeat", or "agent", etc. The manually tuned rule-based grammar exhibited a reasonably high performance of 90.6%. After collecting almost 8000 utterances for this context, an SSLU was trained and reported a performance of 98.8%. This result was at first considered suspicious since it means a misclassification of only 12 out of 1000 utterances including garbage events. This was deemed impossible, and the grammar was initially not released. However, further investigation into the correctness of the testing procedure showed that this SSLU did indeed perform at a near-human recognition level.
- In another context, a caller has a problem with his digital video recorder (DVR) and is asked what exactly the issue is. He may say "I would like to install my DVR", "I don't know how to record", "my DVR box is frozen", "I cannot turn on my box", and some other global utterances as in the above example. The rule-based grammar performed at 84.9%, which is relatively high for such a context with a large variability among the responses. Since this context is not reached very frequently in the application, there were initially only 1087 utterances available in the first round of the continuous improvement cycle. According to the requirements formulated in Section 3.2, the minimum test size was 1000, so only 87 utterances remained for training. Remarkably, the SSLU built on this sparse data set achieved a performance of 87.8% on the same test set, significantly outperforming the baseline.

[2]The statistical approach has shown to outperform the rule-based one in many natural language processing domains such as parsing [6], part-of-speech tagging [7], and machine translation [8].

4. CONCLUSION

Well-planned large-scale utterance collection, transcription, and annotation, in conjunction with a rigorous quality assurance process, can be used in the scope of a timely and continuous improvement cycle to successively replace rule-based grammars by SSLUs and increase the overall performance of speech recognition in a dialog system significantly and systematically. In this study involving more than 2 million utterances, SSLUs have shown within a few months to outperform rule-based grammars in all contexts, including large-vocabulary open-ended speech, directed dialogs, as well as simple yes/no contexts.

5. REFERENCES

[1] K. Acomb, J. Bloom, K. Dayanidhi, P. Hunter, P. Krogh, E. Levin, and R. Pieraccini, "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, 2007.

[2] A. Gorin, G. Riccardi, and J. Wright, "How May I Help You?" *Speech Communication*, vol. 23, no. 1/2, 1997.

[3] R. Pieraccini and J. Huerta, "Where Do We Go from Here? Research and Commercial Spoken Dialog Systems," in *Proc. of the SIGdial Workshop on Discourse and Dialog*, Lisbon, Portugal, 2005.

[4] D. Suendermann, J. Liscombe, K. Evanini, K. Dayanidhi, and R. Pieraccini, "C^5," in *Proc. of the SLT*, Goa, India, 2008.

[5] D. Suendermann, P. Hunter, and R. Pieraccini, "Call Classification with Hundreds of Classes and Hundred Thousands of Training Utterances ... and No Target Domain Data," in *Proc. of the PIT*, Kloster Irsee, Germany, 2008.

[6] M. Collins, "Three Generative, Lexicalised Models for Statistical Parsing," in *Proc. of the ACL*, Madrid, Spain, 1997.

[7] D. Sündermann and H. Ney, "synther – a New M-Gram POS Tagger," in *Proc. of the NLPKE*, Beijing, China, 2003.

[8] F. Och and H. Ney, "A Systematic Comparison of Various Statistical Alignment Models," *Computational Linguistics*, vol. 29, no. 1, 2003.