How to Drink from a Fire Hose:
One Person Can Annoscribe 693 Thousand Utterances in One Month

David Suendermann, Jackson Liscombe, Roberto Pieraccini
SpeechCycle Labs
New York, USA
{david, jackson, roberto}@speechcycle.com

Abstract

Transcription and semantic annotation (annoscription) of utterances is crucial part of speech performance analysis and tuning of spoken dialog systems and other natural language processing disciplines. However, the fact that these are manual tasks makes them expensive and slow. In this paper, we will discuss how annoscription can be partially automated. We will show that annoscription can reach a throughput of 693 thousand utterances per person month under certain assumptions.

1 Introduction

Ever since spoken dialog systems entered the commercial market in the mid 1990s, the caller's speech input is subject to collection, transcription, and often also semantic annotation. Utterance transcriptions and annotations (annoscriptions) are used to measure speech recognition and spoken language understanding performance of the application. Furthermore, they are used to improve speech recognition and application functionality by tuning grammars, introducing new transitions in the call flow to cover more of the callers' demands, or changing prompt wording or application logic to influence the speech input. Annoscriptions are also crucial for training statistical language models and utterance classifiers for call routing or other unconstrained speech input contexts (Gorin et al., 1997). Since very recently, statistical methods are used to replace conventional rule-based grammars in every recognition context of commercial spoken dialog systems (Suendermann et al., 2009b). This replacement is only possible by collecting massive amounts of annoscribed data from all contexts of an application. To give the reader an idea of what *massive* means in this case, in (Suendermann et al., 2009b), we used 2,184,203 utterances to build a complex call routing system. In (Suendermann et al., 2009a), 4,293,898 utterances were used to localize an English Internet troubleshooting application to Spanish.

Considering that professional service providers may charge as much as 50 US cents for annoscribing a single utterance, the usage of these amounts of data seems prohibitive since costs for such a project could potentially add up to several million US dollars. Furthermore, one has to consider the average speed of annoscription which rarely exceeds 1000 utterances per hour and person. This means that the turn-around of a project as mentioned above would be several years unless teams of many people work simultaneously. However, the integration of the work of a large team becomes the more tricky the more people are involved. This is especially true for the annotation portion since it requires a thorough understanding of the spoken dialog system's domain and design and very often can only be conducted under close supervision by the interaction designer in charge of the project. Furthermore, there are crucial issues related to intra- and inter-labeler inconsistency becoming more critical the more people work on the same or similar recognition contexts of a given project.

This paper is to show how it is possible to automate large portions of both transcription and annotation while meeting human performance[1] standards. As an example case, we show how the proposed automation techniques can increase annoscription speed to nearly 693 thousand utterances per person and month.

2 Automatic Transcription

2.1 Two Fundamentals

Automatic transcription of spoken utterances may not sound as something new to the reader. In fact, the entire field of automatic speech recognition is about machine transcription. So, why is it worth dedicating a full section to something well-covered in research and industry for half a century? The reason is the demand for *achieving human performance* as formulated in the introduction which, as is also well-known, cannot be satisfied by any of the large-vocabulary speech recognizers ever developed. In order to demonstrate that there is indeed a way to achieve human transcription performance using automatic speech recognition, we would like to refer to two fundamental observations on the performance of speech recog-

[1] In this paper, *performance* stands for *quality* or *accuracy* of transcription or annotation. It does not refer to *speed* or *throughput*.

Proceedings of SIGDIAL 2010: the 11th Annual Meeting of the Special Interest Group in Discourse and Dialogue, pages 257–260,
University of Tokyo, September 24-25, 2010.

nition:

(1) Speech recognition performance can be very high for contexts of constrained vocabulary. An example is the recognition of isolated letters in the scope of a name spelling task as discussed in (Waibel and Lee, 1990) that achieved a word error rate of only 1.1%. In contrast, the word error rate of large-vocabulary continuous speech recognition can be as high as 40 to 65% on telephone speech (Yuk and Flanagan, 1999).

(2) The positive dependence between speech recognition performance and amount of data used to train acoustic and language models, so far, did not reach a saturation point even considering billions of training tokens (Och, 2006).

Both of these fundamentals can be applied to the transcription task for utterances collected on spoken dialog production systems as follows:

(1) The vocabulary of spoken dialog systems can be rather complex. E.g., the caller utterances used for the localization project mentioned in Section 1 distinguish more than 13,000 types. However, the nature of commercial spoken dialog applications being mostly system-driven strongly constrains the vocabulary in many recognition contexts. E.g., when the prompt reads

> You can say: *recording problems*, *new installation*, *frozen screen*, or *won't turn on*

callers mostly respond things matching the proposed phrases, occasionally altering the wording, and only seldomly using completely unexpected utterances.

(2) The continuous data feed available on high-traffic spoken dialog systems in production processing millions of calls per month can provide large numbers of utterances for every possible recognition context. Even if the context appears to be of a simple nature, as for a yes/no question, the continuous collection of more data will still have an impact on the performance of a language model built using this data.

2.2 How to Achieve Human Performance

Even though we have suggested that the recognition performance in many contexts of spoken dialog systems may be very high, we have still not shown how our observations can be utilized to achieve *human performance* as demanded in Section 1. How would a context-dependent speech recognizer respond when the caller says something completely unexpected such as *let's wreck a nice beach* when asked for the cell phone number? While a human transcriber may still be able to correctly transcribe this sentence, automatic speech recognition will certainly fail even with the largest possible training set. The answer to this question is that the speech recognizer should *not respond at all* in this case but admit that it had trouble recognizing this utterance. Rejection of hypotheses based on confidence scores is common practice in many speech and language processing tasks and is heavily used in spoken dialog systems to avoid mis-interpretation of user inputs.

So, we now know that we can limit automatic transcriptions to hypotheses of a minimum reliability. However, how do we prove that this limited set resembles *human performance*? What is actually *human performance*? Does the human make errors transcribing? And, if so, how do we measure human error? What do we compare it against?

To err is human. Accordingly, there is an error associated with manual transcription which can only be estimated by comparing somebody's transcription with somebody else's due to a lack of ground truth. Preferably, one should have a good number of people transcribe the same speech utterances and than compute the average word error rate comparing every transcription batch with every other producing a reliable estimate of the manual error inherent to the transcription task of spoken dialog system utterances. In order to do so, we compared transcriptions of 258,843 utterances collected from a variety of applications and recognition contexts partially shared by up to six transcribers and found that they averaged at an inter-transcriber word error rate of $\text{WER}_0 = 1.3\%$.

Now, for every recognition context a language model had been trained, we performed automatic speech recognition on held-out test sets of $N = 1000$ utterances producing N hypotheses and their associated confidence scores $P = \{p_1, \ldots p_N\}$. Now, we determined that minimum confidence threshold p_0 for which the word error rate between the set of hypotheses and manual reference transcriptions was not statistically significantly greater than WER_0:

$$p_0 = \arg\min_{p \in P} \text{WER}(V(p)) \;\tilde{\not>}\; \text{WER}_0; \quad (1)$$

$$V(p) = \{\nu_1, \ldots, \nu_K\} : \nu_k \in \{1, \ldots, N\}, p_{\nu_k} \geq p.$$

Statistical significance was achieved when the delta resulted in a p value greater than 0.05 using the χ^2 calculus. For the number of test utterances, 1000, this point is reached when the word error on the test set falls below $\text{WER}_1 = 2.2\%$. This means that Equation 2.2's 'not statistically significantly greater than' sign can be replaced by a regular smaller-than sign as

$$\text{WER} \;\tilde{\not>}\; \text{WER}_0 \Leftrightarrow \text{WER} < \text{WER}_1. \quad (2)$$

This essentially means that there is a chance that the error produced by automatic transcription *is* greater than that of manual transcription, however, on the test set it could not be found to be of significance. Requesting to lower the p value or even demanding that the test set performance falls below the reported manual error can drastically lower the automation rate and, in the latter case, is not even reasonable—how can a machine possibly commit

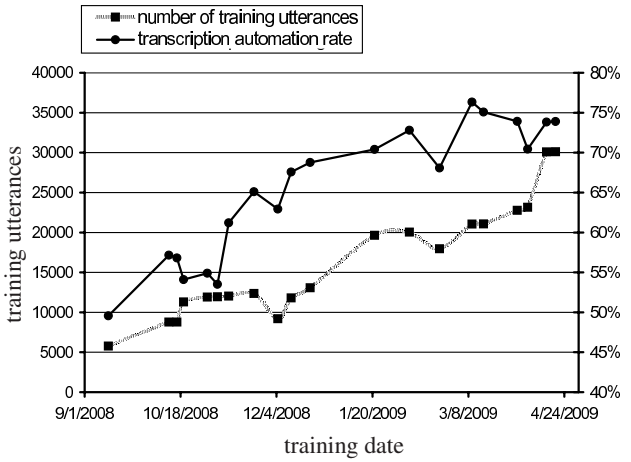

Figure 1: Dependency between amount of training data and transcription automation rate

less errors than a human being as it is trained on human transcriptions?

As a proof of concept, we ran automatic transcription against the same set of utterances used to determine the manual transcription error, and we found that the average word error rate between manual and automatic annotation was as low as 1.1% for all utterances whose confidence score exceeded the context-dependent threshold trained as described above. In this initial experiment, a total of 60,608 utterances, i.e., 23.4%, had been automated.

2.3 On Automation Rate

Formally, transcription automation rate is the ratio of utterances whose confidence exeeded p_0 in Equation 2.2:

$$\text{transcription automation rate} = \frac{|V(p_0)|}{N} \quad (3)$$

where $|V|$ refers to the cardinality of the set V, i.e., the number of V's members.

The above example's transcription automation rate of 23.4% does not yet sound tremendously high, so we should look at what can be done to increase the automation rate as much as possible. It is predictable that the two fundamentals formulated in Section 2.1 have a large impact on recognition performance and, hence, the transcription automation rate:

(1) In large-scale experiments, we were able to show a significant (negative) correlation between the annotation automation rate and task complexity. Since this study does not fit the present paper's scope, we will refrain from reporting on details at this point.

(2) As an example which influence the amount of training data can have on the transcription automation rate, Figure 1 shows statistics drawn from twenty runs of language model training carried out over the course of seven months while collecting more and more data.

3 Automatic Annotation

Semantic annotation of utterances into one of a final set of classes is a task which may require profound understanding of the application and recognition context the specific utterances were collected in. Examples include simple contexts such as yes/no questions which may be easily manageable also by annotators unfamiliar with the application, high-resolution open prompt contexts with hundreds of technical and highly application-specific classes, or number collection contexts allowing for billions of classes. All these contexts can benefit from two rules which help to significantly reduce an annotator's workload:

(A) **Never do anything twice.** This simple statement means that there should be functionality built into the annotation software or the underlying database that

- lets the annotator process multiple utterances with identical transcription in a single step and
- makes sure that whenever a new utterance shows up with a transcription identical to a formerly annotated one, the new utterance gets assigned the same class automatically.

Figure 2 demonstrates the impact of Rule (A) with two typical examples. The first is a yes/no context allowing for the additional global commands *help*, *hold*, *agent*, *repeat*, and *i don't know*. The other is an open prompt context distinguishing 79 classes.

When using the token/type distinction, the impact of Rule (A) is that annotation effort becomes linear with the number of *types* to work on. While the ratio between types and tokens in a given corpus can be very small (i.e., the automation rate is very high, e.g., 95% in the above yes/no example), this ratio reaches saturation at some point. In the yes/no example, there is only a gradual difference between the automation rates for 10 thousand and 1 million utterances. Hence, at a certain point, the effort becomes virtually linear with the number of *tokens* to be processed.

(B) **Predict as much as possible.** Most of the recognition contexts for which utterances are transcribed and annotated use grammars to implement speech recognition functionality. Many of these

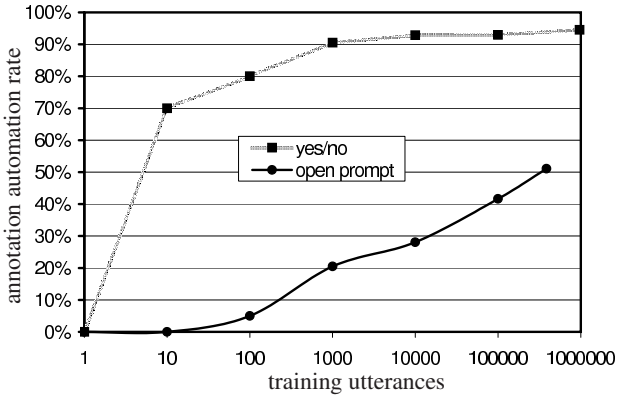

Figure 2: Dependency between number of collected utterances and annotation automation rate based on Rule (A) for two different contexts

Table 1: Annotation automation rates for three different recognition contexts based on Rule (B)

grammar	#symptoms	ann. auto. rate
modem type	43	70.3%
blue/black/snow	10	77.0%
yes/no	10	88.6%

grammars will be rule-based grammars. Even if the grammars are statistical, most often, earlier in time, rule-based grammars had been used in the same recognition context. Hence, we can assume that we are given rule-based grammars for many recognition contexts of the dialog system in question. Per definition, rule-based grammars shall contain canonical rules expressing the relationship between expected utterances in a given context and the semantic classes these utterances are to be associated with. Consequently, whenever for an utterance recorded in the context under consideration there is a rule in the grammar, it provides the correct class for this utterance, and it can be excluded from annotation. These rules can be strongly extended to allow for complex prefix and suffix rules, repetitions, sub-grammars &c. making sure that the majority of utterances will be covered by the rule-based grammars thereby minimizing the annotation effort. Table 1 shows three example grammars of different complexity: One that collects the type of the caller's modem, one for the identification of a TV set's picture color (blue/black/snow), and a yes/no context with global commands. Annotation automation rates for these grammars that were not specifically tuned for maximizing automation but directly taken from the production dialog systems varied between 70.3% and 88.6%.

To never ever touch a formerly annotated utterance type again and to blindly rely on (mayby outdated or erroneous) rule-based grammars to provide baseline annotations may result in annotation mistakes, possibly major ones when frequent utterances are concerned. So, how do we make sure that high annotation performance standards are met?

To answer this question, the authors have developed a set of techniques called C^7 taking care of completeness, consistency, congruence, correlation, confusion, coverage, and corpus size of an annotation set (Suendermann et al., 2008). The mentioned techniques are also useful in the frequent event of changes to the number or scope of annotation classes. This can happen e.g. due to functional changes to the application, changes to prompts, user behavior, or to contexts preceeding the current annotation context. Another frequent reason is the introduction of additional classes to enlarge the scope of the current context[2].

[2]In a specific context, callers may be asked whether they want A, B, or C, but they may respond D. The introduction of a new class D which the application is able to handle requires the re-annotation of all utterances falling into D's scope.

4 693 Thousand Utterances

Finally, we want to return to the initial statement of this paper claiming that one person is able to annoscribe 693 thousand utterances within one month. An approximated automation rate of 80% for transcription and 90% for annotation is possible when there is already a massive database of annoscriptions available to be exploited for automation. These rates result in about 139 thousand transcriptions and 69 thousand annotations outstanding. At a pace of 1000 transcribed or 2000 annotated utterances per hour, the required time would be 139 hours transcription and 35 hours annotation which averages at 40 hours per week[3].

5 Conclusion

This paper has demonstrated how automated annoscription of utterances collected in the production scope of spoken dialog systems can effectively accelerate this conventionally entirely manual effort. When allowing for some overtime, we have shown that a single person is able to produce 693 thousand annoscriptions within one month.

References

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

F. Och. 2006. Challenges in Machine Translation. In *Proc. of the TC-Star Workshop*, Barcelona, Spain.

D. Suendermann, J. Liscombe, K. Evanini, K. Dayanidhi, and R. Pieraccini. 2008. C^5. In *Proc. of the SLT*, Goa, India.

D. Suendermann, J. Liscombe, K. Dayanidhi, and R. Pieraccini. 2009a. Localization of Speech Recognition in Spoken Dialog Systems: How Machine Translation Can Make Our Lives Easier. In *Proc. of the Interspeech*, Brighton, UK.

D. Suendermann, J. Liscombe, K. Evanini, K. Dayanidhi, and R. Pieraccini. 2009b. From Rule-Based to Statistical Grammars: Continuous Improvement of Large-Scale Spoken Dialog Systems. In *Proc. of the ICASSP*, Taipei, Taiwan.

A. Waibel and K.-F. Lee. 1990. *Readings in Speech Recognition*. Morgan Kaufmann, San Francisco, USA.

D. Yuk and J. Flanagan. 1999. Telephone Speech Recognition Using Neural Networks and Hidden Markov Models. In *Proc. of the ICASSP*, Phoenix, USA.

[3]The original title of this paper claimed that one person could annoscribe even one million utterances in a month. However, after receiving multiple complaints about the unlawfulness of a 58-hour workweek, we had to change the title accordingly to avoid disputes with the Department of Labor. Furthermore, as discussed earlier, at the starting point of an annoscription project, automation rates are much lower than later.