# A Handsome Set of Metrics to Measure Utterance Classification Performance in Spoken Dialog Systems

**David Suendermann, Jackson Liscombe, Krishna Dayanidhi, Roberto Pieraccini***

SpeechCycle Labs, New York, USA

{david, jackson, krishna, roberto}@speechcycle.com

## Abstract

We present a set of metrics describing classification performance for individual contexts of a spoken dialog system as well as for the entire system. We show how these metrics can be used to train and tune system components and how they are related to Caller Experience, a subjective measure describing how well a caller was treated by the dialog system.

## 1 Introduction

Most of the speech recognition contexts in commercial spoken dialog systems aim at mapping the caller input to one out of a set of context-specific semantic classes (Knight et al., 2001). This is done by providing a *grammar* to the speech recognizer at a given recognition context. A grammar serves two purposes:

- It constraints the lexical content the recognizer is able to recognize in this context (the language model) and
- It assigns one out of a set of possible classes to the recognition hypothesis (the classifier).

This basic concept is independent of the nature of a grammar: it can be a rule-based one, manually or automatically generated; it can comprise a statistical language model and a classifier; it can consist of sets of grammars, language models, or classifiers; or it can be a holistic grammar, i.e., a statistical model combining a language model and a classification model in one large search tree.

Most commercial dialog systems utilize grammars that return a semantic parse in one of these contexts:

- directed dialogs (e.g., yes/no contexts, menus with several choices, collection of information out of a restricted set [*Which type of modem do you have?*]—usually, less than 50 classes)
- open-ended prompts (e.g. for call routing, problem capture; likewise to collect information out of a restricted set [*Tell me what you are calling about today*]—possibly several hundred classes (Gorin et al., 1997; Boye and Wiren, 2007))
- information collection out of a huge (or infinite) set of classes (e.g., collection of phone numbers, dates, names, etc.)

When the performance of spoken dialog systems is to be measured, there is a multitude of objective metrics to do so, many of which feature major disadvantages. Examples include

- **Completion rate** is calculated as the number of completed calls divided by the total number of calls. The main disadvantage of this metric is that it is influenced by many factors out of the system's control, such as caller hang-ups, opt-outs, or call reasons that fall out of the system's scope. Furthermore, there are several system characteristics that impact this metric, such as recognition performance, dialog design, technical stability, availability of back-end integration, etc. As experience shows, all of these factors can have unpredictable influence on the completion rate. On the one hand, a simple wording change in the introduction prompt of a system can make this rate improve significantly, whereas, on the other hand, major improvement of the open-ended speech recognition grammar following this very prompt may not have any impact.
- **Average holding time** is a common term for the average call duration. This metric is often considered to be quite controversial since it is unclear whether longer calls are preferred or dispreferred. Consider the following two incongruous behaviors resulting in longer call duration:
  - The system fails to appropriately treat callers, asking too many questions, performing redundant operations, acting unintelligently because of missing back-end integration, or letting the caller wait in never-ending wait music loops.
  - The system is so well-designed that it engages callers to interact with the system longer.

* Patent pending.

- **Hang-up and opt-out rates.** These metrics try to encapsulate how many callers choose not to use the dialog system, either because they hang up or because they request to speak with a human operator. However, it is unclear how such events are related to dialog system performance. Certainly, many callers may have a prejudice against speaking with automated systems and may hang up or request a human regardless of how well-performing the dialog system is with cooperative users. Furthermore, callers who hang up may do so because they are unable to get their problem solved or they may hang up precisely because their problem was solved (instead of waiting for the more felicitous post-problem-solving dialog modules).

- **Retry rate** is calculated as the average number of times that the system has to re-prompt for caller input because the caller's previous utterance was determined to be Out-of-Grammar. The intuition behind this metric is that the lower the retry rate, the better the system. However, this metric is problematic because it is tied to grammar performance itself. Consider a well-performing grammar that correctly accepts In-Grammar utterances and rejects Out-of-Grammar utterances. This grammar will cause the system to produce retries for all Out-of-Grammar utterances. Consider a poorly designed grammar that accepts everything (incorrectly), even background noise. This grammar would decrease the retry rate but would not be indicative of a well-performing dialog system.

As opposed to these objective measures, there is a subjective measure directly related to the system performance as perceived by the user:

- **Caller Experience**. This metric is used to describe how well the caller is treated by the system according to its design. Caller Experience is measured on a scale between 1 (bad) and 5 (excellent). This is the only subjective measure in this list and is usually estimated based on averaging scores given by multiple voice user interface experts which listen to multiple full calls. Although this metric directly represents the ultimate design goal for spoken dialog systems—i.e., to achieve highest possible user experience—it is very expensive to be repeatedly produced and not suitable to be generated on-the-fly.

  Our former research has suggested, however, that it may be possible to automatically estimate Caller Experience based on several objective measures (Evanini et al., 2008). These measures include the overall number of no-matches and substitutions in a call, operator requests, hang-ups, non-heard speech, the fact whether the call reason could be successfully captured and whether the call reason was finally satisfied. Initial experiments showed a near-human accuracy of the automatic predictor trained on several hundred calls with available manual Caller Experience scores. The most powerful objective metric turned out to be the overall number of no-matches and substitutions, indicating a high correlation between the latter and Caller Experience.

No-matches and substitutions are objective metrics defined in the scope of semantic classification of caller utterances. They are part of a larger set of semantic classification metrics which we systematically demonstrate in Section 2. The remainder of the paper examines three case studies exploring the usefulness and interplay of different evaluation metrics, including:

- the correlation between True Total (one of the introduced metrics) and Caller Experience in Section 3,

- the estimation of speech recognition and classification parameters based on True Total and True Confirm Total (another metric) in Section 4, and

- the tuning of large-scale spoken dialog systems to maximize True Total and its effect on Caller Experience in Section 5.

## 2 Metrics for Utterance Classification

Acoustic events processed by spoken dialog systems are usually split into two main categories: In-Grammar and Out-of-Grammar. In-Grammar utterances are all those that belong to one of the semantic classes processable by the system logic in the given context. Out-of-Grammar utterances comprise all remaining events, such as utterances whose meanings are not handled by the grammar or when the input is non-speech noise.

Spoken dialog systems usually respond to acoustic events after being processed by the grammar in one of three ways:

- The event gets *rejected*. This is when the system either assumes that the event was Out-of-Grammar, or it is so uncertain about its (In-Grammar) finding that it rejects the utterance. Most often, the callers get re-prompted for their input.

- The event gets *accepted*. This is when the system is certain to have correctly detected an In-Grammar semantic class.

Table 1: *Event Acronyms*

| | |
|---|---|
| I | In-Grammar |
| O | Out-of-Grammar |
| A | Accept |
| R | Reject |
| C | Correct |
| W | Wrong |
| Y | Confirm |
| N | Not-Confirm |
| TA | True Accept |
| FA | False Accept |
| TR | True Reject |
| FR | False Reject |
| TAC | True Accept Correct |
| TAW | True Accept Wrong |
| FRC | False Reject Correct |
| FRW | False Reject Wrong |
| FAC | False Accept Confirm |
| FAA | False Accept Accept |
| TACC | True Accept Correct Confirm |
| TACA | True Accept Correct Accept |
| TAWC | True Accept Wrong Confirm |
| TAWA | True Accept Wrong Accept |
| TT | True Total |
| TCT | True Confirm Total |

- The event gets *confirmed*. This is when the system assumes to have correctly detected an In-Grammar class but still is not absolutely certain about it. Consequently, the caller is asked to verify the class. Historically, confirmations are not used in many contexts where they would sound confusing or distracting, for instance in yes/no contexts (“*I am sorry. Did you say NO?*”—“*No!*”—“*This was NO, yes?*”—“*No!!!*”).

Based on these categories, an acoustic event and how the system responds to it can be described by four binary questions:

1. Is the event In-Grammar?
2. Is the event accepted?
3. Is the event correctly classified?
4. Is the event confirmed?

Now, we can draw a diagram containing the first two questions as in Table 2. See Table 1 for all acoustic event classification types used in the remainder of this paper.

Table 2: *In-Grammar? Accepted?*

| | A | R |
|---|---|---|
| I | TA | FR |
| O | FA | TR |

Table 3: *In-Grammar? Accepted? Correct?*

| | A | | R | |
|---|---|---|---|---|
| | C | W | C | W |
| I | TAC | TAW | FRC | FRW |
| O | FA | | TR | |

Extending the diagram to include the third question is only applicable to In-Grammar events since Out-of-Grammar is a single class and, therefore, can only be either falsely accepted or correctly rejected as shown in Table 3.

Further extending the diagram to accomodate the fourth question on whether a recognized class was confirmed is similarly only applicable if an event was accepted, as rejections are never confirmed; see Table 4. Table 5 gives one example for each of the above introduced events for a yes/no grammar.

When the performance of a given recognition context is to be measured, one can collect a certain number of utterances recorded in this context, look at the recognition and application logs to see whether these utterances where accepted or confirmed and which class they were assigned to, transcribe and annotate the utterances for their semantic class and finally count the events and divide them by the total number of utterances. If X is an event from the list in Table 1, we want to refer to $x$ as this average score, e.g., $tac$ is the fraction of total events correctly accepted. One characteristic of these scores is that they sum up to 1 for each of the Diagrams 2 to 4 as for example

$$a + r = 1, \tag{1}$$

$$i + o = 1, \tag{2}$$

$$ta + fr + fa + tr = 1. \tag{3}$$

In order to enable system tuning and to report system performance at-a-glance, the multitude of metrics must be consolidated into a single powerful metric. In the industry, one often uses weights to combine metrics since they are assumed to have different importance. For instance, a False Accept is considered worse than a False Reject since the latter allows for correction in the first retry whereas the former may lead the caller down the wrong path. However, these weights are heavily negotiable and depend on customer, application, and even the recognition context, making it impossible to produce a comprehensive and widely applicable consolidated metric. This

Table 5: *Examples for utterance classification metrics.* This table shows the transcription of an utterance, the semantic class it maps to (if In-Grammar), a binary flag for whether the utterance is In-Grammar, the recognized class (i.e. the grammar output), a flag for whether the recognized class was accepted, a flag for whether the recognized class was correct (i.e. matched the transcription's semantic class), a flag for whether the recognized class was confirmed, and the acronym of the type of event the respective combination results in.

| utterance | class | In-Grammar? | rec. class | accepted? | correct? | confirmed? | event |
|---|---|---|---|---|---|---|---|
| *yeah* | *YES* | 1 | | | | | I |
| *what* | | 0 | | | | | O |
| | | | *NO* | 1 | | | A |
| | | | *NO* | 0 | | | R |
| *no no no* | *NO* | 1 | *NO* | | 1 | | C |
| yes ma'am | *YES* | 1 | *NO* | | 0 | | W |
| | | | | | | 1 | Y |
| | | | | | | 0 | N |
| *i said no* | *NO* | 1 | *YES* | 1 | | | TA |
| *oh my god* | | 0 | *NO* | 1 | | | FA |
| *i can't tell* | | 0 | *NO* | 0 | | | TR |
| *yes always* | *YES* | 1 | *YES* | 0 | | | FR |
| *yes i guess so* | *YES* | 1 | *YES* | 1 | 1 | | TAC |
| *no i don't think so* | *NO* | 1 | *YES* | 1 | 0 | | TAW |
| *definitely yes* | *YES* | 1 | *YES* | 0 | 1 | | FRC |
| *no man* | *NO* | 1 | *YES* | 0 | 0 | | FRW |
| *sunshine* | | 0 | *YES* | 1 | | 1 | FAC |
| *choices* | | 0 | *NO* | 1 | | 0 | FAA |
| *right* | *YES* | 1 | *YES* | 1 | 1 | 1 | TACC |
| *yup* | *YES* | 1 | *YES* | 1 | 1 | 0 | TACA |
| *this is true* | *YES* | 1 | *NO* | 1 | 0 | 1 | TAWC |
| *no nothing* | *NO* | 1 | *YES* | 1 | 0 | 0 | TAWA |

Table 4: *In-Grammar? Accepted? Correct? Confirmed?*

| | | A | | R | |
|---|---|---|---|---|---|
| | | C | W | C | W |
| I | Y | TACC | TAWC | FRC | FRW |
| | N | TACA | TAWA | | |
| O | Y | FAC | | TR | |
| | N | FAA | | | |

is why we propose to split the set of metrics into two groups: good and bad. The sought-for consolidated metric is the sum of all good metrics (hence, an overall accuracy) or, alternatively, the sum of all bad events (overall error rate). In Tables 3 and 4, good metrics are highlighted. Accordingly, we define two consolidated metrics *True Total* and *True Confirm Total* as follows:

$$tt = tac + tr, \tag{4}$$

$$tct = taca + tawc + fac + tr. \tag{5}$$

In the aforementioned special case that a recognition context never confirms, Equation 5 equals Equation 4 since the confirmation terms $tawc$ and $fac$ disappear.

The following sections report on three case studies on the applicability of True Total and True Confirm Total to the tuning of spoken dialog systems and how they relate to Caller Experience.

## 3 On the Correlation between True Total and Caller Experience

As motivated in Section 1, initial experiments on predicting Caller Experience based on objective

Table 6: *Pearson correlation coefficient for several utterance classification metrics on the source data.*

| | A | | R |
|---|---|---|---|
| | C | W | |
| I | 0.394 | -0.160 | -0.230 |
| O | -0.242 | | -0.155 |

$$r(\text{TT}) = 0.378$$

metrics indicated that there is a considerable correlation between Caller Experience and semantic classification metrics such as those introduced in Section 2. In the first of our case studies, this effect is to be deeper analyzed and quantified. For this purpose, we selected 446 calls from four different spoken dialog systems of the customer service hotlines of three major cable service providers. The spoken dialog systems comprised

- a call routing application—cf. (Suendermann et al., 2008),
- a cable TV troubleshooting application,
- a broadband Internet troubleshooting application, and
- a Voice-over-IP troubleshooting application—see for instance (Acomb et al., 2007).

The calls were evaluated by voice user interface experts and Caller Experience was rated according to the scale introduced in Section 1. Furthermore, all speech recognition utterances (4480) were transcribed and annotated with their semantic classes. Thereafter, all utterance classification metrics introduced in Section 2 were computed for every call individually by averaging across all utterances of a call. Finally, we applied the Pearson correlation coefficient (Rodgers and Nicewander, 1988) to the source data points to correlate the Caller Experience score of a single call to the metrics of the same call. This was done in Table 6.

Looking at these numbers, whose magnitude is rather low, one may be suspect of the findings. E.g., $|r(\text{FR})| > |r(\text{TAW})|$ suggesting that False Reject has a more negative impact on Caller Experience than True Accept Wrong (aka *Substitution*) which is against common experience. Reasons for the messiness of the results are that

- Caller Experience is subjective and affected by inter- and intra-expert inconsistency. E.g., in a consistency cross-validation test, we observed identical calls rated by one subject as 1 and by another as 5.

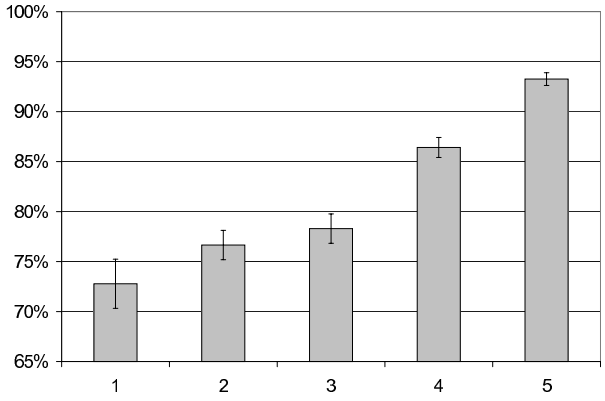


Figure 1: *Dependency between Caller Experience and True Total.*

- Caller Experience scores are discrete, and, hence, can vary by $\pm 1$, even in case of strong consistency.
- Although utterance classification metrics are (almost) objective metrics measuring the percentage of how often certain events happen in average, this average generated for individual calls may not be very meaningful. For instance, a very brief call with a single yes/no utterance correctly classified results in the same True Total score like a series of 50 correct recognitions in a 20-minutes conversation. While the latter is virtually impossible, the former happens rather often and dominates the picture.
- The sample size of the experiment conducted in the present case study (446 calls) is perhaps too small for deep analyses on events rarely happening in the investigated calls.

Trying to overcome these problems, we computed all utterance classification metrics introduced in Section 2, grouping and averaging them for the five distinct values of Caller Experience. As an example, we show the almost linear graph expressing the relationship between True Total and Caller Experience in Figure 1. Applying the Pearson correlation coefficient to this five-point curve yields $r = 0.972$ confirming that what we see is pretty much a straight line. Comparing this value to the coefficients produced by the individual metrics TAC, TAW, FR, FA, and TR as done in Table 7, shows that no other line is as straight as the one produced by True Total supposing its maximization to produce spoken dialog systems with highest level of user experience.

Table 7: *Pearson correlation coefficient for several utterance classification metrics after grouping and averaging.*

| | A | | R |
|---|---|---|---|
| | C | W | |
| I | 0.969 | -0.917 | -0.539 |
| O | -0.953 | | -0.939 |

$$r(\mathrm{TT}) = \mathbf{0.972}$$

## 4 Estimating Speech Parameters by Maximizing True Total or True Confirm Total

The previous section tried to shed some light on the relationship between some of the utterance classification metrics and Caller Experience. We saw that, on average, increasing Caller Experience comes with increasing True Total as the almost linear curve of Figure 1 supposes. As a consequence, much of our effort was dedicated to maximizing True Total in diverse scenarios. Speech recognition as well as semantic classification with all their components (such as acoustic, language, and classification models) and parameters (such as acoustic and semantic rejection and confirmation confidence thresholds, time-outs, etc.) was set up and tuned to produce highest possible scores. This section gives two examples of how parameter settings influence True Total.

### 4.1 Acoustic Confirmation Threshold

When a speech recognizer produces a hypothesis of what has been said, it also returns an acoustic confidence score which the application can utilize to decide whether to reject the utterance, confirm it, or accept it right away. The setting of these thresholds has obviously a large impact on Caller Experience since the application is to reject as few valid utterances as possible, not confirm every single input, but, at the same time, not falsely accept wrong hypotheses. It is also known that these settings can strongly vary from context to context. E.g., in announcements, where no caller input is expected, but, nonetheless utterances like 'agent' or 'help' are supposed to be recognized, rejection must be used much more aggressively than in collection contexts. True Total or True Confirm Total are suitable measures to detect the optimum tradeoff. Figure 2 shows the True Confirm Total graph for a collection context with 30 distinguishable classes. At a confidence value of 0.12, there is a local and global maximum indicating the optimum setting for the confirmation threshold for this grammar context.

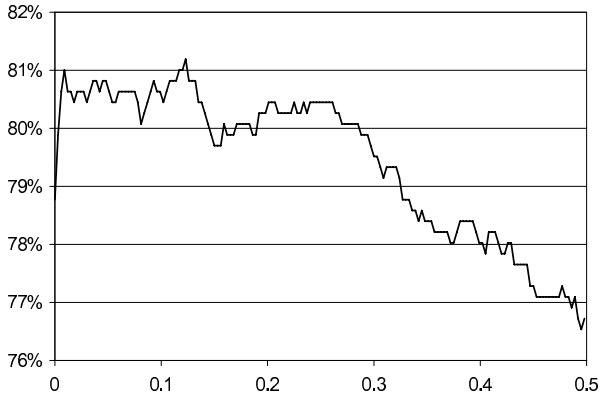


Figure 2: *Tuning the acoustic confirmation threshold.*

### 4.2 Maximum Speech Time-Out

This parameter influences the maximum time the speech recognizer keeps recognizing once speech has started until it gives up and discards the recognition hypothesis. Maximum speech time-out is primarily used to limit processor load on speech recognition servers and avoid situations in which line noise and other long-lasting events keep the recognizer busy for an unnecessarily long time. As it anecdotally happened to callers that they were interrupted by the dialog system, on the one hand, some voice user interface designers tend to chose rather large values for this time-out setting, e.g., 15 or 20 seconds. On the other hand, very long speech input tends to produce more likely a classification error than shorter ones. Might there be a setting which is optimum from the utterance classification point of view?

To investigate this behavior, we took 115,885 transcribed and annotated utterances collected in the

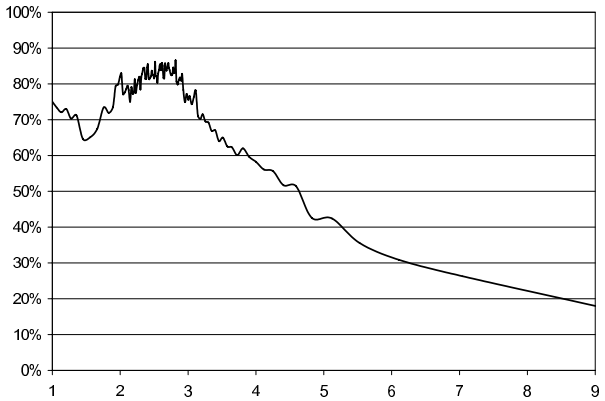


Figure 3: *Dependency between utterance duration and True Total.*

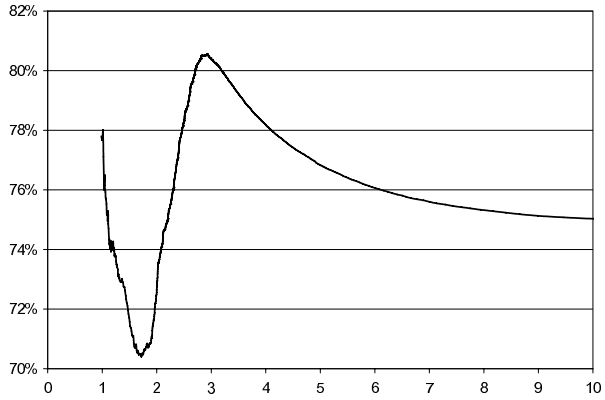


Figure 4: *Dependency between maximum speech time-out and True Total.*

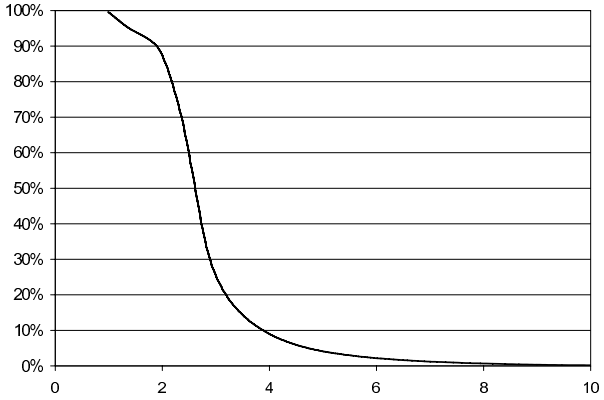


Figure 5: *Percentage of utterances interrupted by maximum speech time-out.*

main collection context of a call routing application and aligned them to their utterance durations. Then, we ordered the utterances in descending order of their duration, grouped always 1000 successive utterances together, and averaged over duration and True Total. This generated 116 data points showing the relationship between the duration of an utterance and its expected True Total, see Figure 3.

The figure shows a clear maximum somewhere around 2.5 seconds and then descends with increasing duration towards zero. Utterances with a duration of 9 seconds exhibited a very low True Total score (20%). Furthermore, it would appear that one should never allow utterances to exceed four second in this context. However, upon further evaluation of the situation, we also have to consider that long utterances occur much less frequently than short ones. To integrate the frequency distribution into this analysis, we produced another graph that shows the average True Total accumulated over all utterances *shorter than a certain duration*. This simulates the effect of using a different maximum speech time-out setting and is displayed in Figure 4. We also show a graph on how many of the utterances would have been interrupted in Figure 5.

The curve shows an interesting down-up-down trajection which can be explained as follows:

- Acoustic events shorter than 1.0 seconds are mostly noise events which are correctly identified since the speech recognizer could not even build a search tree and returns an empty hypothesis which the classifier, in turn, correctly rejects.
- Utterances with a duration around 1.5s are dominated by single words which cannot properly evaluated by the (trigram) language model. So, the acoustic model takes over the main work and, because of its imperfectness, lowers the True Total.
- Utterances with a moderate number of words are best covered by the language model, so we achieve highest accuracy for them ($\approx$3s).
- The longer the utterances continues after 4 seconds, the less likely the language model and classfier are to have seen such utterances, and True Total declines.

Evaluating the case from the pure classifier performance perspective, the maximum speech time-out would have to be set to a very low value (around 3 seconds). However, at this point, about 20% of the callers would be interrupted. The decision whether this optimimum should be accepcted depends on how elegantly the interruption can be designed:

> "I'm so sorry to interrupt, but I'm having a little trouble getting that. So, let's try this a different way."

## 5 Continuous Tuning of a Spoken Dialog System to Maximize True Total and Its Effect on Caller Experience

In the last two sections, we investigated the correlation between True Total and Caller Experience and gave examples on how system parameters can be tuned by maximizing True Total. The present section gives a practical example of how rigorous improvement of utterance classification leads to real improvement of Caller Experience.

The application in question is a combination of the four systems listed in Section 3 which work in an interconnected fashion. When callers access the service hotline, they are first asked to briefly describe their call reason. After up to two follow-up questions to further disambiguate their reason, they are either connected to a human operator or one of the three automated troubleshooting systems. Escalation from one of them can connect the caller to an

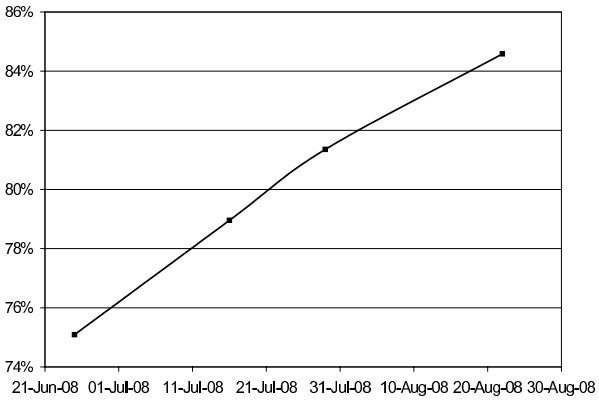


Figure 6: *Increase of the True Total of a large-vocabulary grammar with more than 250 classes over release time.*

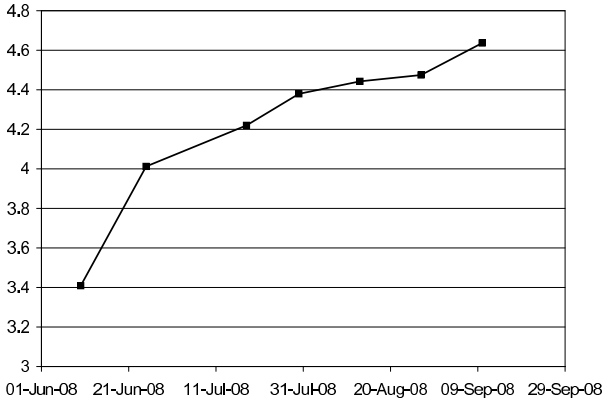


Figure 7: *Increase of Caller Experience over release time.*

agent, transfer the caller back to the call router or to one of the other troubleshooting systems.

When the application was launched in June 2008, its True Total averaged 78%. During the following three months, almost 2.2 million utterances were collected, transcribed, and annotated for their semantic classes to train statistical update grammars in a continuously running process (Suendermann et al., 2009). Whenever a grammar significantly outperformed the most recent baseline, it was released and put into production leading to an incremental improvement of performance throughout the application. As an example, Figure 6 shows the True Total increase of the top-level large-vocabulary grammar that distinguishes more than 250 classes. The overall performance of the application went up to more than 90% True Total within three months of its launch.

Having witnessed a significant gain of a spoken dialog system's True Total, we would now like to know to what extent this improvement manifests itself in an increase of Caller Experience. Figure 7 shows that, indeed, Caller Experience was strongly positively affected. Over the same three month period, we achieved an iterative increase from an initial Caller Experience of 3.4 to 4.6.

## 6 Conclusion

Several of our investigations have suggested a considerable correlation between True Total, an objective utterance classification metric, and Caller Experience, a subjective score of overall system performance usually rated by expert listeners. This observation leads to our main conclusions:

- True Total and several of the other utterance classification metrics introduced in this paper can be used as input to a Caller Experience predictor—as tentative results in (Evanini et al., 2008) confirm.
- Efforts towards improvement of speech recognition in spoken dialog applications should be focused on increasing True Total since this will directly influence Caller Experience.

## References

K. Acomb, J. Bloom, K. Dayanidhi, P. Hunter, P. Krogh, E. Levin, and R. Pieraccini. 2007. Technical Support Dialog Systems: Issues, Problems, and Solutions. In *Proc. of the HLT-NAACL*, Rochester, USA.

J. Boye and M. Wiren. 2007. Multi-Slot Semantics for Natural-Language Call Routing Systems. In *Proc. of the HLT-NAACL*, Rochester, USA.

K. Evanini, P. Hunter, J. Liscombe, D. Suendermann, K. Dayanidhi, and R. Pieraccini:. 2008. Caller Experience: A Method for Evaluating Dialog Systems and Its Automatic Prediction. In *Proc. of the SLT*, Goa, India.

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

S. Knight, G. Gorrell, M. Rayner, D. Milward, R. Koeling, and I. Lewin. 2001. Comparing Grammar-Based and Robust Approaches to Speech Understanding: A Case Study. In *Proc. of the Eurospeech*, Aalborg, Denmark.

J. Rodgers and W. Nicewander. 1988. Thirteen Ways to Look at the Correlation Coefficient. *The American Statistician*, 42(1).

D. Suendermann, P. Hunter, and R. Pieraccini. 2008. 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.

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