Tackling a Shilly-Shally Classifier for Predicting Task Success in Spoken Dialogue Interaction

Alexander Schmitt, Alexander Zgorzelski, Wolfgang Minker

Institute of Information Technology, University of Ulm, Germany
alexander.schmitt,alexander.zgorzelski,wolfgang.minker@uni-ulm.de

10.21437/Interspeech.2011-294

Abstract

Statistical models, which predict that a task with a telephone-based Spoken Dialogue System (SDS) is unlikely to be completed, can be useful to adapt dialogue strategies. They can also trigger the decision to route callers directly to human assistance once it is clear that the SDS cannot automate the call. This paper addresses a number of issues that arise when deploying such models. We show that the predictions of a model are subject to strong variations between several adjacent dialogue steps. As a consequence, we show that the accuracy can be significantly risen when using sequences of equal predictions as basis of the decision-making. Furthermore, we implement a confidence metric that takes into account the certainty of the classifier to determine the optimum decision point.

Index Terms: task completion, task success, interaction parameters, problematic dialogues, combining consecutive predictions, IVR

1. Introduction

Adaptivity in Spoken Dialogue Systems has increasingly attracted the research community's interest in the past years. While there has been a strong endeavor to predict user characteristics such as age, gender, emotional state etc., a prediction of the dialogue outcome has only deficiently been explored. However, this can be of special interest to service providers using Interactive Voice Response (IVR) systems in call center environments. The idea behind predicting the dialogue outcome, i.e. task completion, is to map a set of input variables i quantifying the ongoing interaction at arbitrary decision points p to a target variable belonging to one of the classes $c \in \{completed, not_completed\}$. Once it is detected that the system is unable to help the customer, dialogue strategies can be changed and ultimately users can be routed to human operators who can solve the problem.

Discriminative statistical classifiers such as Rule Learners [1, 2], Support Vector Machines (SVM) [3], and statistical language models (SLM) [4] have been used for classification. The accuracy achieved by the classifiers inherently depends on the characteristic of a dialogue system and can thus hardly be compared to each other. Previous studies specify the average performance values of the classifier when predicting the call outcome at decision points $p_1..p_n$. However, to date it has not been analyzed to which degree the prediction of the classifier at arbitrary points within a dialogue deviate one from each other. The present study thus simulates the deployment of a task completion predictor and its effect on specific dialogues. It can be shown that the predictions of a classifier are not stable between a series of predictions for different exchanges. Under these circumstances, a single prediction can hardly be trusted and an escalation to a human operator should not be considered. The proposed approaches help to overcome this problem and enable higher prediction accuracies.

This contribution is organized as follows: In Section 2 we describe related work. That followed, we describe the employed dataset in Section 3 and apply the model on a specific call in Section 4. In Section 5 we demonstrate to which extent a series of consecutive predictions at several exchanges can enhance the model's accuracy. In Section 6 we introduce a decision metric that allows us to further increase the certainty of the decision. Finally, in Section 7 we summarize our work and draw a conclusion.

2. Related Work

One of the first works that aimed to predict the outcome of a dialogue was presented in 1999 by Langkilde et al. [1] for the AT&T "How May I Help You?" (HMIHY) system. Their model for predicting the outcome at the first dialogue exchange provides a significantly better accuracy (8%) than the baseline, which would be the guess of the majority class. Using different automatic feature sets from the whole dialogue, they already were able to predict the outcome with a 23% higher accuracy than the baseline. In addition to the general question, whether a useful prediction accuracy is achievable, Kim [4] has also shown that online classification, i.e. classifying at arbitrary points in an interaction, can come close to off-line classification performance, i.e. after the entire information is gathered. He proposed using the log likelihood ratio (LLR) of two SLM models to estimate call quality for online call monitoring. By incorporating a LLR threshold value approach enabled him to select bad calls dynamically depending on the situation and react accordingly by bringing in human agents to help the callers.

This study is a follow-up of our continous work on the prediction of task completion. In [5] we confirmed the hypothesis from [6] that a deployment of a task completion classifier can lower handling costs for service providers. Under the assumption that emotions play a decisive role for predicting task completion we added a set of acoustic features to the classifier that has previously only been based on log features, cf. [2]. Due to the low occurrence of emotional speech in the dataset this hypothesis could not be confirmed. Finally, in [3] we enhanced the model by a large number of further interaction parameters (IP) and employed n-grams of IPs as feature vectors. The model is based on a series of SVMs.

Copyright © 2011 ISCA

3. Dataset and Task Completion Classifier

For our studies we employ a corpus of 104,800 calls from an automated Internet troubleshooter [7]. It helps callers to get back online, recover from e-mail problems, reset passwords, etc. The largest portion of calls concerns the recovery of lost Internet connections. In the present study, we focus on this group comprising 41,422 calls altogether with an average call length of 38.7 exchanges.

Each call was assigned one of the following three class labels [8]:

- 'not_solved'[1]: the problem can be considered as unsolved either because the caller hung up in the middle of the conversation, the user asked for an operator without being offered one, or the system could not solve the problem,
- 'partially_solved'[2]: the problem was partially solved in that the system provided significant hints to the user what the problem was, but, finally, the Internet connectivity was not recovered and the help of an operator was required,
- 'solved'[3]: the problem was solved, i.e. the Internet connectivity was recovered.

The label distribution resulted in 4,931 not solved, 28,486 partially solved and 8,005 solved calls. The decision, whether a call belongs to the class not solved, partially solved or solved is reached through a series of SVMs. For details please refer to [3]. For predicting the outcome of a call a separate SVM model has been created for each of the classification points (at exchanges 6-23) in the SDS. Each SVM has been trained with interaction parameters derived from log files from the current and the two previous system-user exchanges. Examples for such interaction parameters are the confidence of the Automatic Speech Recognition (ASR), the number of unsuccessfully parsed user utterances ($nomatch$), the number of barged-ins, etc. The dimension of the feature space amounts to app. 150 features.

4. Motivation: Applying Models to Specific Calls

During the past years dealing with the prediction of task success the question was risen how a statistical classifier would 'behave' when classifying a specific call at arbitrary points in an ongoing interaction. This motivated us to implement a software framework [9], which displays predictions of discriminative and regressive classifiers jointly with a dialogue flow. It enables a view on the predictions at arbitrary points in the dialogue. For this work, the model presented in [3] has then been applied on specific calls. An example is depicted in Figure 1.

The classifier discriminates between three possible call outcomes: $solved$, $partially_solved$ and not_solved. The chart shows the classifier's confidence of observing a call that belongs to one of the three classes. The bold topmost dots at the 66.6%-confidence line represent the classifier's hypothesis, i.e. the prediction.

[1] according to [8]: $TS : Fs$ "Failed because of the system's behavior" and $TS : Fu$ "Failed because of the user's behavior"

[2] according to [8]: $TS : SN$: "Succeeded in spotting that no solution exists"

[3] according to [8] $TS : S$: "Succeeded (task for which solutions exist)"

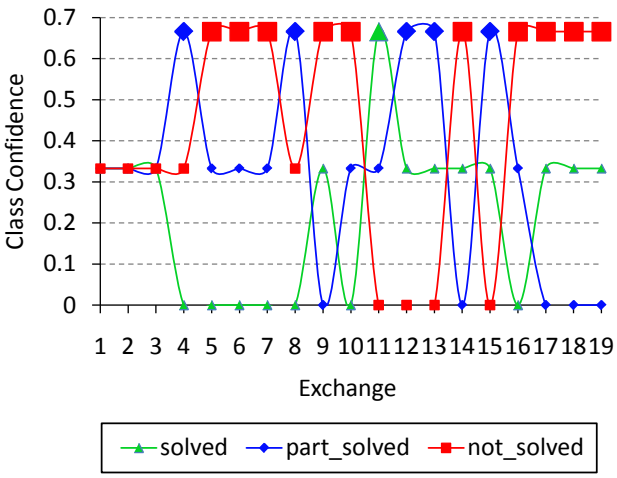

Figure 1: Model chart derived from the Witchcraft Workbench [9] when a task completion prediction model is applied on a specific example call [3]. The lines symbolize the confidence scores of the classifier at each system-user exchange in the dialogue for all three possible call outcomes. The upmost markers at confidence level 0.66 symbolize the classifier's hypothesis. The call depicted here belongs to the class "not_solved".

5. N-Tuples as Decision Rule

A number of observations could be made when applying the model on a large number of calls and analyzing the outcome charts. Firstly, a majority of the calls exhibited such patterns, where the classifier varied strongly between the adjacent system-user exchanges. Secondly, the classifier obviously produces sequences of same predictions. In this chart the class 'not solved' is predicted several times in a row at exchanges 5,6,7 and 9,10 as well as 16,17,18,19. When sequences like these occurred, they have been frequently correct, such as in this example, where the call indeed belongs to the class 'not solved'.

It seems obvious to postpone the decision, whether to escalate the call or not, until a certain number of predictions in a row are observed, i.e. an n-tuple of equal predictions. It can be assumed that with increasing tuple size, i.e. double, triple, quadruple, quintuple, etc., the certainty of the classifier and thus the accuracy of the prediction rises. If such a rule is applied, the performance of the model is increased as depicted in Figure 2.

A number of observations can be derived: firstly, with increasing dialogue duration and delay of the actual decision, the accuracy of the model rises for all tuples. This can be explained due to the growing amount of data that is gathered with the growing progress of the dialogue. Secondly, the accuracy is up to 20% higher when a quadruple of equal predictions is used as decision instead of when relying on a single prediction.

A tradeoff has to be accepted when using n-tuples as decision rule: not all calls exhibit e.g. quadruples, i.e. no decision will be reached for those calls. As is depicted in Figure 3, in 11,61% of all calls no decision will apply, when deciding after observing a quadruple.

It seems that the classifier often changes the expected outcome and therefore the course of the dialogue seems to be insecure. The service provider employing such a model has to decide, whether to escalate such calls or whether to take the

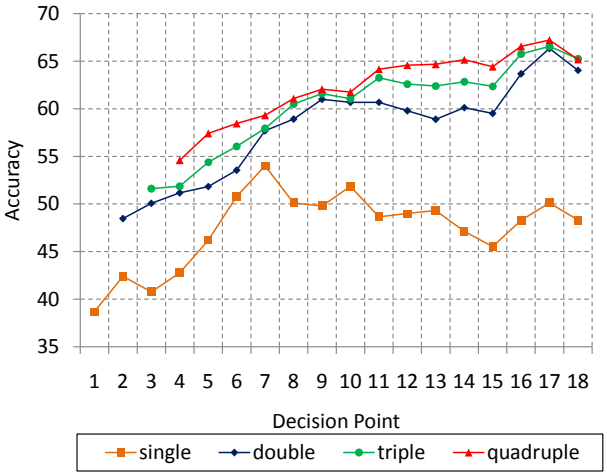

Figure 2: Prediction accuracies of the model when double, triple and quadruples are taken as decision compared to a single prediction (cf. [3]).

higher risk that the call will not end successfully. This comparatively simple approach lacks the flexibility of being able to control the decision point, since the decision is only triggered by the course of the predictions.

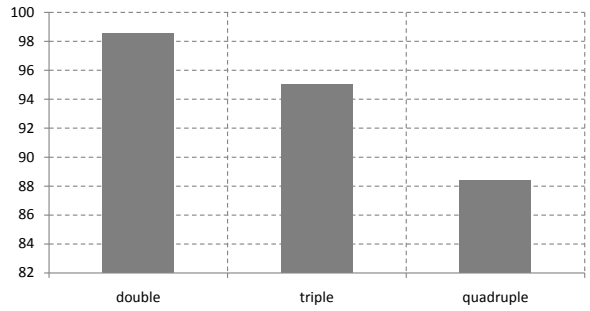

Figure 3: Percentage of calls where a decision is made.

6. Mean Confidence Deviation Metric

We have shown that the n-tuple approach introduced in the previous section already leads to a large gain of prediction accuracy. We followed up on this idea and tried to view the problem from a different perspective. Instead of coming to a decision when a certain amount of the same consecutive predictions are observed, it might be a more elaborated approach to keep track of the *frequency of changes* of the prediction sequences. In other words: The more fluctuating these sequences are, the less likely are those predictions correct, since there seems to be a bigger uncertainty in predicting the correct outcome of this particular call. The confidence of the classifier at an exchange E can mathematically be described by summing up these differences divided by the number of exchanges E that have already been processed, also called Mean Confidence Deviation $MCD1_E$:

$$MCD1_E = \frac{1}{(E-o)+1} \sum_{e=o}^{E} \sum_{c=0}^{C} |x_{e,c} - x_{e-1,c}|,$$

where o is an offset that we require since the first three exchanges in our employed dataset contain greetings and introduction prompts that do not contain any predictive power. We thus choose $o = 4$ for the employed dialogue system which is the first exchange in each call that differs from the other calls. c are the possible classes $\in \{0, 1, 2\}$, i.e. $solved$, $partially_solved$ and not_solved. $x_{e,c}$ is the class-wise confidence score (i.e. belonging to class c) at exchange e. It can be seen that in the example dialogue in Figure 1 at exchange four the three sequences split up. This happens with every call, since it is the exchange, where the first real prediction is possible. Because of this split $MCD1_E$ has already the value 0.66 at this point in the depicted call (cf. Figure 5). If, from this exchange on, the prediction sequences remain steady, $MCD1_E$ quickly converges to 0, and every time a confidence change occurs $MCD1_E$ diverges from 0. With this characteristic a decision point can easily be introduced by defining a threshold θ. The decision point is then determined, once the condition $MCD1_E < \theta$ is fulfilled. However, the metric has a decisive disadvantage as can be easily comprehended when looking at Figure 5: early uncertainty will keep the $MCD1_E$ score high. It is thus difficult to fall below θ in later exchanges. The relevance of early uncertainties has to be reduced. We thus further introduced two parameters α and ρ and modified the equation as follows:

$$MCD2_E = \frac{1}{((E-o)+1)^{\alpha}} \sum_{e=o}^{E} \sum_{c=0}^{C} |x_{e,c} - x_{e-1,c}| \cdot \frac{1}{((E+1)-e)^{\rho}},$$

The new variables are:

- An acceleration factor $\alpha \in \{0.5 \ldots 1.5\}$ increases the denominator with increasing call length. Choosing a larger value here forces an earlier decision.
- A reduction factor $\rho \in \{0.5 \ldots 1.0\}$ decreases the relevance of changes of the prediction sequences, which happened long before the current exchange.

We numerically searched for the most effective combination of the input variables and iterated over more than 1200 different combinations of α, ρ and θ. The complexity is increased by the problem of finding a maximum of the three output variables *accuracy*, *average decision point* and *percentage of calls* covered by the metric. Please note that the decision point is not static for all calls, e.g. at exchange number 10. It is dynamically determined once $MCD2_E$ falls below θ. We thus speak of the *average* decision point. This dependency is depicted in Figure 4. If we would aim for the highest possible *accuracy*, the logic consequence is that the decision point has to be set as late as possible and/or the decision has to be only made when the certainty is very high. It is easy to comprehend that a classifier predicting task completion only makes sense when it gets a chance at comparatively early points in a dialogue when still covering a large amount of calls. We think that a reasonable combination is $\theta = 0.24$, $\alpha = 1.45$ and $\rho = 0.9$. Obviously this decision is debatable and depends on the preferences of the service provider and has to be adapted to the specific system. For this configuration, the average decision point is 16 covering 79.14% of all calls with an accuracy of 61.17%. This approach

outperforms the n-tuple approach, e.g. a quadruple decision rule would yield 58.5% with an earliest decision point of 13 and an average decision point of 16 and covers 80.0%.

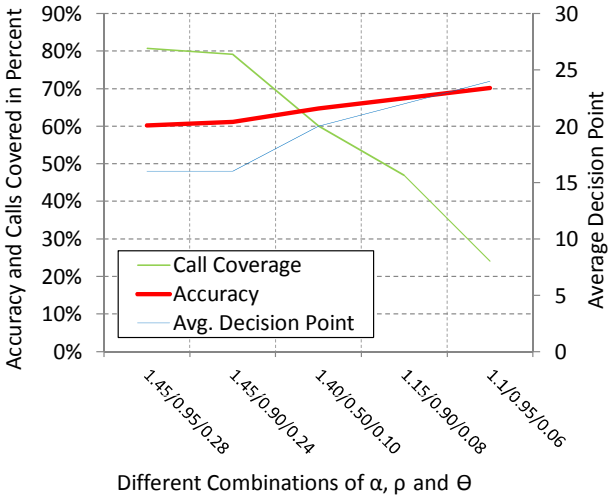

Figure 4: Dependency of accuracy, average decision point and percentage of covered calls for selected example values of acceleration variable α, reduction variable ρ and threshold θ, ordered by accuracy.

7. Conclusion and Discussion

In this work it could be shown that predicting task success in an ongoing spoken dialogue interaction brings along uncertainty that has to be dealt with. Certainty, and thus accuracy of the model, can be increased by relying on a series of homonymic hypotheses instead of a single prediction alone. The decision point is thus determined by the first occurrence of such n-tuples with subsequent identical predictions. The employed dataset exhibited an average increase in accuracy of 6.1% for doubles, 10.9% for triples and 11.8% for quadruples. Quid pro quo: In return for the rising accuracy, we cannot come to a decision in all calls. 1.44%, 5.0% and 11.61% of the calls exhibit no doubles, triples or quadruples, respectively at the first possible decision point. However, we have to bear in mind that without a statistical model, even in 100% of the calls there would be no prediction, and thus no decision to which outcome a call will lead. An even higher accuracy can be yielded by introducing the Mean Confidence Deviation metric that keeps track of the classifier's confidences. The metric considers a decision as uncertain, as long as the confidences for the single classes rapidly change during a call. Depending on the acceleration variable α a decision is forced earlier or later in the course of a dialogue. The reduction variable ρ reduces the influence of early uncertainties. Both influence the earliness and the accuracy of the decision as well as the number of calls where a prediction is possible. For future work we are planning to explore the performance of Hidden Markov Models in comparison to Support Vector Machines.

8. Acknowledgement

The research leading to these results has received funding from the Transregional Collaborative Research Centre SFB/TRR 62 "Companion-Technology for Cognitive Technical Systems" funded by the German Research Foundation (DFG).

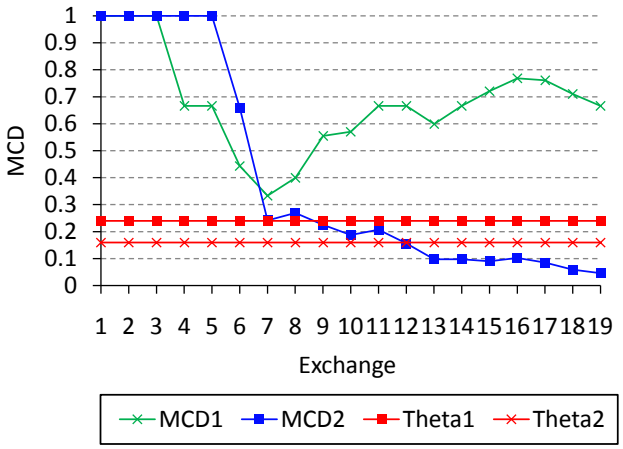

Figure 5: Both metrics applied on the sample call. The threshold $\theta 1$ applies for the $MCD1_E$ metric, $\theta 2$ applies for $MCD2_E$. Applying $MCD2_E$ would trigger a decision at exchange 12, applying $MCD1_E$ would not trigger a decision since the call is considered as unsure according to this metric. Both tresholds have been determined using 629 test calls.

9. References

[1] I. Langkilde, M. Walker, J. Wright, A. Gorin, and D. Litman, "Automatic prediction of problematic human-computer dialogues in how may i help you," in *In Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU99*, 1999, pp. 369–372.

[2] O. Herm, A. Schmitt, and J. Liscombe, "When calls go wrong: How to detect problematic calls based on log-files and emotions?" in *Proc. of the International Conference on Speech and Language Processing (ICSLP) Interspeech 2008*, Sep. 2008, pp. 463–466.

[3] A. Schmitt, M. Scholz, W. Minker, J. Liscombe, and D. Sündermann, "Is it possible to predict task completion in automated troubleshooters?" in *Proc. of the International Conference on Speech and Language Processing (ICSLP)*, Sep. 2010.

[4] W. Kim, "Online call quality monitoring for automating agent-based call centers," in *Proc. of the International Conference on Speech and Language Processing (ICSLP)*, Sep. 2007.

[5] A. Schmitt, C. Hank, and J. Liscombe, "Detecting problematic calls with automated agents," in *4th IEEE Tutorial and Research Workshop Perception and Interactive Technologies for Speech-Based Systems*, Irsee (Germany), Jun. 2008.

[6] E. Levin and R. Pieraccini, "Value-based optimal decision for dialog systems," in *Proc. of Spoken Language Technology Workshop 2006*, Dec. 2006, pp. 198–201.

[7] K. Acomb, J. Bloom, K. Dayanidhi, P. Hunter, P. Krogh, E. Levin, and R. Pieraccini, "Technical support dialog systems:issues, problems, and solutions," in *Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies*. Rochester, NY: ACL, April 2007, pp. 25–31.

[8] "Parameters describing the interaction with spoken dialogue systems," International Telecommunication Union, Geneva, Switzerland, ITU-T Recommendation Supplement 24 to P-Series, 2005, based on ITU-T Contr. COM 12-17 (2009).

[9] A. Schmitt, W. Minker, and N. Sharaf, "Advances in the witchcraft workbench project," in *Proceedings of the SIGDIAL 2010 Conference*. Tokyo, Japan: Association for Computational Linguistics, 2010, pp. 261–264.