Interaction Quality Estimation in Spoken Dialogue Systems Using Hybrid-HMMs

Stefan Ultes
Ulm University
Albert-Einstein-Allee 43
89081 Ulm, Germany
stefan.ultes@uni-ulm.de

Wolfgang Minker
Ulm University
Albert-Einstein-Allee 43
89081 Ulm, Germany
wolfgang.minker@uni-ulm.de

Abstract

Research trends on SDS evaluation are recently focusing on objective assessment methods. Most existing methods, which derive quality for each system-user-exchange, do not consider temporal dependencies on the quality of previous exchanges. In this work, we investigate an approach for determining Interaction Quality for human-machine dialogue based on methods modeling the sequential characteristics using HMM modeling. Our approach significantly outperforms conventional approaches by up to 4.5% relative improvement based on Unweighted Average Recall metrics.

1 Introduction

Spoken Dialogue Systems (SDSs) play a key role in achieving natural human-machine interaction. One reason is that speech is one major channel of natural human communication. Assessing the quality of such SDSs has been discussed frequently in recent years. The basic principles which all approaches underlie have been analyzed by Möller et al. (2009) creating a taxonomy for quality of human-machine interaction, i.e., Quality of Service (QoS) and Quality of Experience (QoE). Quality of Service describes objective criteria like *total number of turns*. The recent shift of interest in dialogue assessment methods towards subjective criteria is described as Quality of Experience, putting the user in the spotlight of dialogue assessment. For QoE, Möller et al. (2009) identified several aspects contributing to a good user experience, e.g., usability or acceptability. These aspects can be combined under the term user satisfaction, describing the degree by which the user is satisfied with the system's performance. By assessing QoE, the hope of the research community is to better measure the human-like quality of an SDS. While this information may be used during the design process, enabling automatically derived user satisfaction within the dialogue management allows for adaption of the ongoing dialogue (Ultes et al., 2012b).

First work on deriving subjective metrics automatically has been performed by Walker et al. (1997) resulting in the PARADISE framework, which is the current quasi-standard in this field. Briefly explained, a linear dependency is assumed between dialogue parameters and user satisfaction to estimate qualitative performance on the dialogue level.

Measuring the performance of complete dialogues does not allow for adapting to the user *during* the dialogue (Ultes et al., 2012b). Hence, performance measures which provide a measurement for each system-user-exchange[1] are of interest. Approaches based on Hidden Markov Models (HMMs) are widely used for sequence modeling. Therefore, Engelbrecht et al. (2009) used these models for predicting the dialogue quality on the exchange level. Similar to this, we presented work on estimating Interaction Quality using HMMs and Conditioned HMMs (Ultes et al., 2012a). In this contribution, we investigate an approach for recognizing the dialogue quality using a hybrid Markovian model. Here, hybrid means combining statistical approaches such as Support Vector Machines with Hidden Markov Models by modeling the observation probability of the HMMs using classification. While this is the first time hybrid approaches are used for estimating Interaction Quality, they are well-known and have been used before for other classification tasks (e.g. (Valstar and Pantic, 2007; Onaran et al., 2011)).

This paper is outlined as follows: Related work on subjective quality measurement on the ex-

[1] A system-user-exchange consists of a system dialogue turn followed by a user dialogue turn

Proceedings of the SIGDIAL 2014 Conference, pages 208–217,
Philadelphia, U.S.A., 18-20 June 2014.

change level is presented in Section 2. All experiments in this work are based on the Interaction Quality metric of the LEGO corpus described in Section 3. We motivate for introducing time dependency and present our own approach on recognizing Interaction Quality using a Markovian model presented in Section 4 and briefly present the classification algorithms used for the experiments in Section 5. Experiments are presented in Section 6 and their results discussion in Section 7.

2 Significant Related Work

Much research on predicting subjective quality measures on an exchange level has been performed hitherto. However, most of this body of work lacks of either taking account of the sequential structure of the dialogue or resulting in insufficient performance.

Engelbrecht et al. (2009) presented an approach using Hidden Markov Models (HMMs) to model the SDS as a process evolving over time. Performance ratings on a 5 point scale ("bad", "poor", "fair", "good", "excellent") have been applied by the users of the SDS during the dialogue. The interaction was halted while the user rated. A HMM was created consisting of 5 states (one for each rating) and a 6-dimensional input vector. While Engelbrecht et al. (2009) relied on only 6 input variables, we will pursue an approach with 29 input variables. Moreover, we will investigate dialogues of a real world dialogue system annotated with quality labels by expert annotators.

Higashinaka et al. (2010) proposed a model for predicting turn-wise ratings for human-human dialogues. Ratings ranging from 1 to 7 were applied by two expert annotators labeling for smoothness, closeness, and willingness. They achieved an UAR[2] of only 0.2-0.24 which is only slightly above the random baseline of 0.14.

Hara et al. (2010) derived turn level ratings from overall ratings of the dialogue which were applied by the users *after* the interaction on a five point scale within an online questionnaire. Using n-grams to model the dialogue by calculating n-gram occurrence frequencies for each satisfaction value showed that results for distinguishing between six classes at any point in the dialogue to be hardly above chance.

A more robust measure for user satisfaction has been presented by Schmitt et al. (2011) within their work about Interaction Quality (IQ) for Spoken Dialogue Systems. In contrast to user satisfaction, the labels were applied by expert annotators *after* the dialogue at the exchange level. Automatically derived parameters were used as features for creating a statistical model using static feature vectors. Schmitt et al. (2011) performed IQ recognition on the LEGO corpus (see Section 3) using linear SVMs. They achieved an UAR[2] of 0.58 based on 10-fold cross-validation which is clearly above the random baseline of 0.2. Ultes et al. (2012a) put an emphasis on the sequential character of the IQ measure by applying a Hidden Markov Models (HMMs) and a Conditioned Hidden Markov Models (CHMMs). Both have been applied using 6-fold cross validation and a reduced feature set of the LEGO corpus achieving an UAR[2] of 0.44 for HMMs and 0.39 for CHMMs. While Ultes et al. (2012a) used generic Gaussian Mixture Models to model the observation probabilities, we use confidence distributions of static classification algorithms.

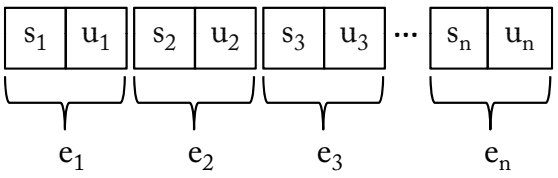

Figure 1: A dialogue may be separated into a sequence of system-user-exchanges where each exchange e_i consists of a system turn s_i followed by a user turn u_i.

3 The LEGO Corpus

For Interaction Quality (IQ) estimation, we use the LEGO corpus published by Schmitt et al. (2012). Interaction Quality is defined similarly to user satisfaction: While the latter represents the true disposition of the user, IQ is the disposition of the user assumed by an expert annotator. Here, expert annotators are people who listen to recorded dialogues after the interactions and rate them by assuming the point of view of the actual person performing the dialogue. These experts are supposed to have some experience with dialogue systems. In this work, expert annotators were "advanced students of computer science and engineering" (Schmitt et al., 2011), i.e., grad students.

The LEGO corpus is based on 200 calls to

[2]Unweighted Average Recall, see Section 6

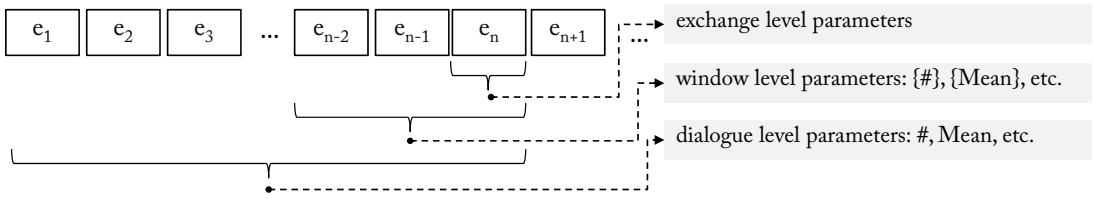

Figure 2: The three different modeling levels representing the interaction at exchange e_n: The most detailed exchange level, comprising parameters of the current exchange; the window level, capturing important parameters from the previous n dialog steps (here $n = 3$); the dialog level, measuring overall performance values from the entire previous interaction.

the "Let's Go Bus Information System" of the Carnegie Mellon University in Pittsburgh (Raux et al., 2006) recorded in 2006. Labels for IQ have been assigned by three expert annotators to 200 calls consisting of 4,885 system-user-exchanges (see Figure 1) in total with an inter-annotator agreement of $\kappa = 0.54$. This may be considered as a moderate agreement (cf. Landis and Koch's Kappa Benchmark Scale (1977)) which is quite good considering the difficulty of the task that required to rate each exchange. For instance, if one annotator reduces the IQ value only one exchange earlier than another annotator, both already disagree on two exchanges. The final label was assigned to each exchange by using the median of all three individual ratings.

IQ was labeled on a scale from 1 (extremely unsatisfied) to 5 (satisfied) considering the complete dialogue up to the current exchange. Thus, each exchange has been rated without regarding any upcoming user utterance. As the users are expected to be satisfied at the beginning, each dialogue's initial rating is 5. In order to ensure consistent labeling, the expert annotators had to follow labeling guidelines (Schmitt et al., 2012).

An example of an annotated dialogue is shown in Table 5. It starts off with a good IQ until the system provides some results and then falls drastically as the user input does not correspond to what the system expects. Thus, the system remains in a loop until the user reacts appropriately.

Parameters used as input variables for the IQ model have been derived from the dialogue system modules automatically for each exchange. Furthermore, parameters on three levels have been created: the *exchange level*, the *dialogue level*, and the *window level* (see Figure 2). As parameters like ASRCONFIDENCE (confidence of speech recognition) or UTTERANCE (word sequence recognized by speech recognition) can directly be acquired from the dialogue modules they constitute the *exchange level*. Counts, sums, means, and frequencies of *exchange level* parameters from multiple exchanges are computed to constitute the *dialogue level* (all exchanges up to the current one) and the *window level* (the three previous exchanges).

4 Hybrid-HMM

As Schmitt et al. (2011) model the sequential character of the data only indirectly by designing special features, our approach applies Markovian modeling to directly model temporal dependencies. Temporal dependencies on previous system-user-exchanges are not taken into account by Schmitt et al.; only parameters derived from the current exchange are used. However, we found out that Interaction Quality is highly dependent on the IQ value of the previous exchange. Adding the parameter IQ_{prev} describing the previous IQ value to the input vector to the IQ model consisting of several parameters results in an extended input vector. Calculating the Information Gain Ratio (IGR) of each parameter of the extended input vector shows that IQ_{prev} achieves the highest IGR value of 1.0. In other words, IQ_{prev} represents the parameter which contains the most information for the classification task.

While performing IQ recognition on the extended features set using the annotated IQ values results in an UAR of 0.82, rather using the estimated IQ value results in an UAR of only 0.43. Consequently, other configurations have to be investigated. Here, Markovian approaches offer a self-contained concept of using these temporal dependencies. However, Ultes et al. (2012a) showed that applying neither a classical HMM nor a conditioned HMM yields results outperforming static approaches.

Therefore, in this Section we present a Hybrid-

HMM approach, which is based on the classical HMM and takes advantage of good performing existing static classification approaches. The classical HMM, specifically used for time-sequential data, consists of a set of states S with transition probability matrix A and initial probability vector π over a set of observations B (also called vocabulary) and an observation function b_{q_t} dependent on the state q_t. For calculating the probability $p(q_t|O_t, \lambda)$ of seeing observation sequence $O_t = (o_1, o_2, \ldots, o_t)$ while being in state q_t at time t given the HMM λ, the Forward Algorithm is used:

$$p(q_t = s_j|O_t, \lambda) = \alpha_t(j) = \sum_{i=1}^{|S|} \alpha_{t-1}(i) a_{ij} b_j(o_t) \ . \quad (1)$$

Here, a_{ij} describes the transition probability of transitioning from state s_i to state s_j. To find a suitable model λ, the HMM must be trained, for example, by using the Baum-Welch algorithm. Usually, the observation function b_{q_t} is modeled with Gaussian mixture models (GMMs). For more information on general HMMs, please refer to Rabiner et al. (1989).

For determining the most likely class $\hat{\omega}_t$ at time t, where each state $j \in S$ is associated with one class ω, the following equation is used:

$$\hat{\omega}_t = \arg\max_j \alpha_t(j) \ . \quad (2)$$

For applying an HMM while exploiting existing statistical classification approaches, the observation function $b_j(o_t)$ is modeled by using confidence score distributions of statistical classifiers, e.g., a Support Vector Machine in accordance with Schmitt et al. (2011) (see Section 5). Furthermore, the transition function a_{ij} is computed by taking the frequencies of the state transitions contained in the given corpus. Therefore, an ergodic HMM is used comprising five states with each representing one of the five IQ scores.

Moreover, in SDSs, a system action act is performed at the end of each system turn. This can be utilized by adding an additional dependency on this action to the state transition function a_{ij}. By augmenting Equation 1, this results in

$$\alpha_t(j) = \sum_{i=1}^{|S|} \alpha_{t-1}(i) a_{ij,\mathsf{act}} b_j(o_t) \ . \quad (3)$$

This refinement models differences in state transitions evoked by different system actions, e.g., a different transition probability is expected if a WAIT action is performed compared to a CONFIRMATION. Equation 3 is equal to the belief update equation known from the Partially Observable Markov Decision Process formalism (Kaelbling et al., 1998).

Therefore, two versions of the Hybrid-HMM are evaluated: an action-independent version as in Equation 1 and an action-dependent version as in Equation 3.

5 Classifier Types

For modeling the observation probability $b_j(o_t)$of the hybrid HMM, multiple classification schemes have been applied to investigate the influence of observation distributions with different characteristics on the overall performance.

In general, classification means estimating a class $\hat{\omega}$ to the given observation o by comparing the class-wise probabilities $p(\omega|o)$. In this work, this probability may be used to model the observation probability $b_j(o)$ of the HMM by the posterior probability

$$p(\omega|o) = b_j(o) \quad (4)$$

for $j = \omega$.

As not all classification algorithms provide a posterior probability, it may be replaced by the confidence distribution. A general description of the classification algorithms used in this work are described in the following Section along with a motivation for the feature subset of the LEGO corpus used for estimating the Interaction Quality in this work.

5.1 Support Vector Machine

For a two class problem, a Support Vector Machine (SVM) (Vapnik, 1995) is based on the concept of linear discrimination with maximum margin by defining a hyperplane separating the two classes. The estimated class $\hat{\omega}$ for observation vector $\vec{o}$ is based on the sign of the decision function

$$k(\vec{o}) = \sum_{i=1}^{N} \alpha_i z_i K(\vec{m}_i, \vec{o}) + b \ , \quad (5)$$

where $\vec{m}_i$ represent support vectors defining the hyper plane (together with b), z_i the known class $\vec{m}_i$ belongs to, α_i the weight of $\vec{m}_i$, and $K(\cdot, \cdot)$ a

kernel function. The kernel function is defined as

$$K(\vec{m}, \vec{m}') = \langle \varphi(\vec{m}), \varphi(\vec{m}') \rangle \ , \qquad (6)$$

where $\varphi(\vec{m})$ represents a transformation function mapping $\vec{m}$ into a space Φ of different dimensionality and $\langle \cdot, \cdot \rangle$ defines a scalar product in Φ. By using the kernel function, the linear discrimination may happen in a space of high dimensionality without explicitly transforming the observation vectors into said space.

The SVM implementation which is used in this contribution is *libSVM* (Chang and Lin, 2011). As this algorithm does not provide class probabilities directly, the respective confidence scores are used.

5.2 Naive Bayes

For deriving the posterior probability, the Naive Bayes classifier may be used. It calculates the posterior probability $P(\omega|o)$ of having class ω when seeing the n-dimensional observation vector $\vec{o}$ by applying Bayes rule (Duda et al., 2001):

$$P(\omega|\vec{o}) = \frac{p(\vec{o}|\omega) \cdot P(\omega)}{p(\vec{o})} \ . \qquad (7)$$

In general, observations, i.e., elements of the observation vector, may be correlated with each other and introducing independence assumptions between these elements does usually not reflect the true state of the world. However, correlations are often not very high thus simplifying the Bayes problem has proved to result in reasonable performance. This is utilized by the Naive Bayes classifier by assuming said independence thus calculating

$$p(\vec{o}|\omega) = \prod_{i=1}^{n} p(o_i|\omega) \ . \qquad (8)$$

5.3 Rule Induction

The classification algorithm Rule Induction or Rule Learner is based on the idea of defining rules to assign classes $\hat{\omega}$ to observation vectors $\vec{o}$. In this work, the algorithm RIPPER (Repeated Incremental Pruning to Produce Error Reduction) (Cohen, 1995) is used where each rule consists of conjunctions of $A_n = v$, where A_n is a nominal attribute, or $A_c \geq \theta, A_c \leq \theta$, where A_c is a continuous attribute. Each part of the observation vector $\vec{o}$ is reflected by one of the attributes. The basic process of the algorithm for generating rules is divided into three steps: First, rules are grown by adding attributes to the rule. Second, the rules are pruned. If the resulting rule set is not of sufficient performance, all training examples which are covered by the generated rules are removed from the example set and a new rule is created.

5.4 Feature selection

As stated previously, all experiments are based on the LEGO corpus presented in Section 3. In order to keep the presented results comparable to previous work based on HMM and CHMM (Ultes et al., 2012a), a reduced parameter set is used. Parameters with constant values for most exchanges have been excluded. These would result in rows of zeros during computation of the covariance matrices of the feature vectors, which are needed for HMM and CHMM classification. A row of zeros in the covariance matrix will make it non-invertible, which will cause errors during the computation of the emission probabilities.

Therefore, a feature set consisting of 29 interaction parameters is used for both defining a baseline and for evaluating the Hybrid-HMM. The set consists of the following parameters (for an explanation of the features, please refer to (Schmitt et al., 2012)):

Exchange Level ASRRECOGNITIONSTATUS, ACTIVITYTYPE, ASRCONFIDENCE, ROLEINDEX, ROLENAME, UTD, REPROMPT?, BARGED-IN?, DD, WPST, WPUT

Dialogue Level MEANASRCONFIDENCE, #ASRREJECTIONS, #TIMEOUTS_ASRREJ, #BARGEINS, %ASRREJECTIONS, %TIMEOUTS_ASRREJ, %BARGEINS, #REPROMPTS, %REPROMPTS, #SYSTEMQUESTIONS

Window Level #TIMEOUTS_ASRREJ, #ASRREJECTIONS, #BARGEINS, %BARGEINS, #SYSTEMQUESTIONS, MEANASRCONFIDENCE, #ASRSUCCESS, #RE-PROMPT

For act in Equation 3, the exchange level parameter ACTIVITYTYPE is used which may take one out of the four values "Announcement", "Confirmation", "Question", or "wait". Their distribution within the LEGO corpus is depicted in Figure 3.

6 Experiments and Results

All experiments are conducted using 6-fold cross-validation[3]. This includes the baseline approach

[3] Six folds have been selected as a reasonable trade-off between validity and computation time.

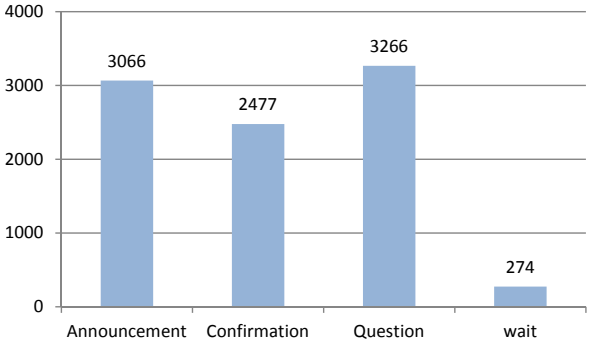

Figure 3: Distribution of the four values for act in Equation 3 in the LEGO corpus. While "wait" occurs rarely, the other three main actions occur at roughly the same frequency.

(also producing the observation probabilities of the Hybrid-HMM approach) and the evaluation of the Hybrid-HMM. For the latter, two phases of cross-validation were applied.

Interaction Quality estimation is done by using three commonly used evaluation metrics: *Unweighted Average Recall (UAR)*, *Cohen's Kappa* (Cohen, 1960) and *Spearman's Rho* (Spearman, 1904). These are also selected as the same metrics have been used in Schmitt et al. (2011) as well.

Recall in general is defined as the rate of correctly classified samples belonging to one class. The recall in UAR for multi-class classification problems with N classes $recall_i$ is computed for each class i and then averaged over all class-wise recalls:

$$UAR = \frac{1}{N} \sum_{i=1}^{N} recall_i \ . \tag{9}$$

Cohen's Kappa measures the relative agreement between two corresponding sets of ratings. In our case, we compute the number of label agreements corrected by the chance level of agreement divided by the maximum proportion of times the labelers could agree. However, Cohen's weighted Kappa is applied as ordinal scores are compared (Cohen, 1968). A weighting factor w is introduced reducing the discount of disagreements the smaller the difference is between two ratings:

$$w = \frac{|r_1 - r_2|}{|r_{max} - r_{min}|} \ . \tag{10}$$

Here, r_1 and r_2 denote the rating pair and r_{max} and r_{min} the maximum and minimum ratings possible.

Table 1: Results for IQ recognition of the statistical classifiers: UAR, κ and ρ for linear SVM, Bayes classification and Rule Induction. σ^2 represents the variances of the confidence scores.

	UAR	κ	ρ	σ^2
SVM (linear)	.495	.611	.774	.020
Bayes	.467	.541	.716	.127
Rule Induction	.596	.678	.790	.131

Correlation between two variables describes the degree by which one variable can be expressed by the other. **Spearman's Rho** is a non-parametric method assuming a monotonic function between the two variables (Spearman, 1904).

6.1 Baseline

As baseline, we adapted the approach of Schmitt et al. (2011). While they focused only on an SVM with linear kernel, we investigate three different static classification approaches. Different classifiers will produce different confidence distributions. These distributions will have different characteristics which is of special interest for evaluating the Hybrid-HMM as will be discussed in Section 7. The confidence characteristics are represented by the variance of the confidence scores σ^2. This variance is used as indicator for how certain the classifier is about its results. If one IQ value has a high confidence while all others have low confidence, the classifier is considered to be very certain. This also results in a high variance. Vice versa, if all IQ values have almost equal confidence indicates high uncertainty. This will result in a low variance.

The classification algorithms, which have been selected arbitrarily, are SVM with linear kernel, Naive Bayes, and Rule Induction (see Section 5). The results in Table 1 show that an SVM with linear kernel (as used by Schmitt et al. (2011)) performs second best with an UAR of 0.495 after Rule Induction with an UAR of 0.596. The results of the SVM differ from the results obtained by Schmitt et al. (UAR of 0.58) as we used a reduced feature set while they used all available features.

6.2 Hybrid-HMM

For evaluating the Hybrid-HMM on Interaction Quality recognition, three aspects are of interest. Most prominent is whether the presented approaches outperform the baseline, i.e., the clas-

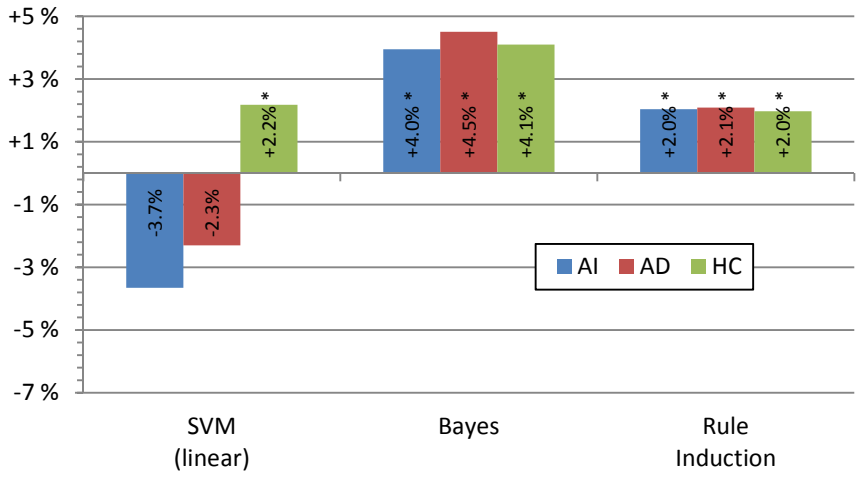

Figure 4: Relative difference of UAR in percent between the baseline performance and the Hybrid-HMM for the action-independent (AI), action-dependent (AD) and handcrafted (HC) transition matrix. Differences marked with an * are significant (Wilcoxon test (Wilcoxon, 1945), $\alpha < 0.05$).

Table 2: Results for the Hybrid-HMM approach: UAR, κ and ρ for the action-independent (AI) and action-dependent (AD) versions.

	UAR		κ		ρ	
	AI	**AD**	**AI**	**AD**	**AI**	**AD**
SVM (linear)	.477	.484	.599	.598	.770	.771
Bayes	.486	.489	.563	.564	.737	.741
Rule Induction	.608	.609	.712	.714	.826	.824

sifier which produces the observation probabilities. Moreover, performance values of action-dependent approaches and action-independent approaches are compared. In addition, the results are analyzed with respect to the characteristic of the confidence distribution.

For producing the confidence scores representing the observation probabilities, the statistical classification algorithms presented in Section 6.1 are used. The initial distribution π for each HMM was chosen in accordance with the annotation guidelines of the LEGO corpus starting each dialogue with an IQ score of 5 resulting in

$$\pi_5 = P(IQ = 5) = 1.0$$
$$\pi_4 = \pi_3 = \pi_2 = \pi_1 = P(IQ \neq 5) = 0.0\ .$$

Results of the experiments with action-dependent (AD) and action-independent (AI) transition function may be seen in Table 2. Again, Rule Induction performed best with Naive Bayes on the second and SVM on the third place.

7 Discussion

While previous work on applying the HMM and CHMM for IQ recognition could not outperform the baseline (Ultes et al., 2012a), Hybrid-HMM experiments show a significant improvement in UAR, Cohen's κ and Spearman's ρ for Naive Bayes and Rule Induction. While performance declines for the linear SVM, this difference has shown to be not significant.

The relative difference of the Hybrid-HMM compared to the respective baseline approaches using an action-dependent and an action-independent transition matrix is depicted in Figure 4. Improvement for the Bayes method was the highest significantly increasing UAR by up to 4.5% relative to the baseline. However, adding action-dependency to the Hybrid-HMM does not show any effect. This may be a result of using ACTIVITYTYPE instead of the actual action. However, using the actual action would result in the need for more data as it contains 45 different values. Significance for all results has been calculated using the Wilcoxon test (Wilcoxon, 1945) by pair-wise comparison of the estimated IQ values of all exchanges. All results except for the decline in SVM performance are significant with $\alpha < 0.05$.

Correlating the confidence variances shown in Table 1 with the improvements of the Hybrid-HMM reveals that for methods with a high variance—and therefore with a greater certainty about the classification result—, an improvement could be accomplished. However, the perfor-

Table 3: Results of Hybrid-HMM with handcrafted transition matrix of the action-independent version.

	UAR	κ	ρ
SVM (linear)	.506	.642	.797
Bayes	.487	.563	.734
Rule Induction	.608	.712	.825

Table 4: Handcrafted transition matrix based on empirical data.

from \ to	1	2	3	4	5
1	0.7	0.3	0	0	0
2	0.25	0.5	0.25	0	0
3	0	0.25	0.5	0.25	0
4	0	0	0.25	0.5	0.25
5	0	0	0	0.3	0.7

mance declined for classification approaches with a low confidence variance, which can be seen as a sign for uncertain classification results.

While the results for Hybrid-HMM are encouraging, creating a simple handcrafted transition matrix for the action-independent version shown in Table 4 achieved even more promising results as performance for all classifier types could be improved significantly compared to the baseline (see Table 3). The handcrafted matrix was created in a way to smooth the resulting estimates as only transitions from one IQ rating to its neighbors have a probability greater than zero. Drastic changes in the estimated IQ value compared to the previous exchange are thus less likely. The exact values have been derived empirically. By applying this handcrafted transition matrix, even SVM performance with linear kernel could be improved significantly by 2.2% in UAR (see Figure 4) compared to the baseline.

For creating the Interaction Quality scores, annotation guidelines were used resulting in certain characteristics of IQ. Therefore, it may be assumed that the effect of exploiting the dependency on previous states is just a reflection of the guidelines. While this might be true, applying a Hybrid HMM for IQ recognition is reasonable as, despite the guidelines, the IQ metric itself is strongly related to user satisfaction, i.e., ratings applied by users (without guidelines), achieving a Spearman's ρ of 0.66 ($\alpha < 0.01$) (Ultes et al., 2013).

8 Conclusions

As previously published, approaches for recognizing the Interaction Quality of Spoken Dialogue Systems are based on static classification without temporal dependency on previous values, a Hybrid Hidden Markov Model approach has been investigated based on three static classifiers. The Hybrid-HMM achieved a relative improvement up to 4.5% and a maximum of 0.61 UAR. Analyzing the experiments revealed that, while an improvement could be achieved with the Hybrid-HMM approach, handcrafting a transition model achieved even better results as performance for all analyzed classifier types could be improved significantly. Furthermore, applying the Hybrid-HMM approach only yields improved performance if the basic classifier itself has a high confidence about its results.

Further research should be conducted investigating the question how the presented approach as well as the Interaction Quality paradigm in general will generalize for different dialogue domains. As IQ is designed to be domain independent, it may be expected that the Hybrid-HMM will be applicable for different dialogue domains as well.

Finally, it is notable that rule induction outperformed SVM approaches in the baseline by 10 percentage points. While this contribution does not focus on this, analyzing the model may help in understanding the problem of estimating Interaction Quality better, especially since rule-based recognition methods allow easy interpretation.

Acknowledgments

This work was supported by the Transregional Collaborative Research Centre SFB/TRR 62 "Companion-Technology for Cognitive Technical Systems" which is funded by the German Research Foundation (DFG).

References

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. In *Educational and Psychological Measurement*, volume 20, pages 37–46, April.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

William W. Cohen. 1995. Fast effective rule induction. In *Proceedings of the 12th International Con-*

ference on Machine Learning, pages 115–123. Morgan Kaufmann, July.

Richard O. Duda, Peter E. Hart, and David G. Stork. 2001. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, November.

Klaus-Peter Engelbrecht, Florian Gödde, Felix Hartard, Hamed Ketabdar, and Sebastian Möller. 2009. Modeling user satisfaction with hidden markov model. In *SIGDIAL '09: Proceedings of the SIGDIAL 2009 Conference*, pages 170–177, Morristown, NJ, USA. ACL.

Sunao Hara, Norihide Kitaoka, and Kazuya Takeda. 2010. Estimation method of user satisfaction using n-gram-based dialog history model for spoken dialog system. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. ELRA.

Ryuichiro Higashinaka, Yasuhiro Minami, Kohji Dohsaka, and Toyomi Meguro. 2010. Issues in predicting user satisfaction transitions in dialogues: Individual differences, evaluation criteria, and prediction models. In *Spoken Dialogue Systems for Ambient Environments*, volume 6392 of *Lecture Notes in Computer Science*, pages 48–60. Springer Berlin / Heidelberg.

L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1-2):99–134.

J. R. Landis and G. G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, March.

Sebastian Möller, Klaus-Peter Engelbrecht, C. Kühnel, I. Wechsung, and B. Weiss. 2009. A taxonomy of quality of service and quality of experience of multimodal human-machine interaction. In *Quality of Multimedia Experience, 2009. QoMEx 2009. International Workshop on*, pages 7–12, July.

Ibrahim Onaran, N Firat Ince, A Enis Cetin, and Aviva Abosch. 2011. A hybrid svm/hmm based system for the state detection of individual finger movements from multichannel ecog signals. In *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on*, pages 457–460. IEEE.

Lawrence R. Rabiner. 1989. *A tutorial on hidden Markov models and selected applications in speech recognition*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Antoine Raux, Dan Bohus, Brian Langner, Alan W. Black, and Maxine Eskenazi. 2006. Doing research on a deployed spoken dialogue system: One year of letâs go! experience. In *Proc. of the International Conference on Speech and Language Processing (ICSLP)*, September.

Alexander Schmitt, Benjamin Schatz, and Wolfgang Minker. 2011. Modeling and predicting quality in spoken human-computer interaction. In *Proceedings of the SIGDIAL 2011 Conference*, Portland, Oregon, USA, June. Association for Computational Linguistics.

Alexander Schmitt, Stefan Ultes, and Wolfgang Minker. 2012. A parameterized and annotated corpus of the cmu let's go bus information system. In *International Conference on Language Resources and Evaluation (LREC)*.

Charles Edward Spearman. 1904. The proof and measurement of association between two things. *American Journal of Psychology*, 15:88–103.

Stefan Ultes, Robert ElChabb, and Wolfgang Minker. 2012a. Application and evaluation of a conditioned hidden markov model for estimating interaction quality of spoken dialogue systems. In *Proceedings of the 4th International Workshop on Spoken Language Dialog System (IWSDS)*, pages 141–150. Springer, November.

Stefan Ultes, Alexander Schmitt, and Wolfgang Minker. 2012b. Towards quality-adaptive spoken dialogue management. In *NAACL-HLT Workshop on Future directions and needs in the Spoken Dialog Community: Tools and Data (SDCTD 2012)*, pages 49–52, Montréal, Canada, June. ACL.

Stefan Ultes, Alexander Schmitt, and Wolfgang Minker. 2013. On quality ratings for spoken dialogue systems – experts vs. users. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 569–578. ACL, June.

Michel F. Valstar and Maja Pantic. 2007. Combined support vector machines and hidden markov models for modeling facial action temporal dynamics. In *Human-Computer Interaction*, volume 4796 of *Lecture Notes in Computer Science*, pages 118–127. Springer Berlin Heidelberg.

Vladimir N. Vapnik. 1995. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Marilyn Walker, Diane Litman, Candace A. Kamm, and Alicia Abella. 1997. Paradise: a framework for evaluating spoken dialogue agents. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 271–280, Morristown, NJ, USA. ACL.

Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83.

Appendix

Table 5: Example of a dialogue annotated with Interaction Quality for each system-user-exchange. As the system fails to interact properly with the user in the end, the IQ value drops down to one. Utterance represents either the system utterance or the recognized user utterance. (CallID 2061123000 of the LEGO corpus.)

	Utterance	IQ
S:	Welcome to the CMU Let's Go bus information system.	
U:		5
S:	To get help at any time, just say Help or press zero.	
U:		5
S:	What can I do for you?	
U:	PENN AT BUTLER TIME IS IT IS THE CENTRE AT HIGHLAND FROM RUNS THE 61A EIGHT THIRTY 61B CAN'T OAKLAND	5
S:	Leaving from OAKLAND . Is this correct?	
U:	OAKLAND	4
S:	Leaving from OAKLAND . Is this correct?	
U:	*YES*	4
S:	Right	
U:	-	4
S:	Where do you want to go?	
U:	*SQUIRREL HILL OF FIFTH AVENUE*	4
S:	Going to FIFTH AVENUE . Is this correct?	
U:	*YES*	5
S:	Alright	
U:	-	5
S:	When do you wanna travel?	
U:	*SUNDAY*	4
S:	At what time do you wanna travel?	
U:	AT FIVE FOUR VERY ABOUT ELEVEN O'CLOCK	4
S:	*Leaving at 11 a.m. . Is this correct?*	
U:	*YES*	5
S:	Right	
U:	-	5
S:	Hold on. Let me check that for you.	
U:	-	5
S:	OAKLAND and FIFTH AVENUE are both the same stop. Please provide a different start or end point.	
U:	-	3
S:	To ask about a different trip, you can say, start a new query. If you are finished, you can say goodbye.	
U:	*61A*	2
S:	You can say start a new query, or goodbye.	
U:	*FIVE DUQUESNE*	1
S:	To ask about a different trip, you can say, start a new query. If you are finished, you can say goodbye.	
U:	*LAST THE NEXT*	1
S:	You can say start a new query, or goodbye.	
U:	*START WH*	1
S:	To ask about a different trip, you can say, start a new query. If you are finished, you can say goodbye.	
U:	*THE*	1
S:	You can say start a new query, or goodbye.	
U:	*THANK YOU QUERY ELEVEN BACK*	1
S:	To ask about a different trip, you can say, start a new query. If you are finished, you can say goodbye.	
U:	*GOODBYE*	1
S:	Thank you for using the cmu Let's Go Bus Information System. Goodbye.	
U:	-	1