A METHOD FOR EVALUATING AND COMPARING USER SIMULATIONS: THE CRAMÉR-VON MISES DIVERGENCE

Jason D. Williams

AT&T Labs – Research, Shannon Laboratory, 180 Park Ave., Florham Park, NJ 07932, USA
jdw@research.att.com

ABSTRACT

Although user simulations are increasingly employed in the development and assessment of spoken dialog systems, there is no accepted method for evaluating user simulations. In this paper, we propose a novel quality measure for user simulations. We view a user simulation as a predictor of the performance of a dialog system, where per-dialog performance is measured with a domain-specific scoring function. The quality of the user simulation is measured as the divergence between the distribution of scores in real dialogs and simulated dialogs, and we argue that the Cramér-von Mises divergence is well-suited to this task. The technique is demonstrated on a corpus of real calls, and we present a table of critical values for practitioners to interpret the statistical significance of comparisons between user simulations.

Index Terms— User simulation, user modelling, dialog simulation, dialog management

1. INTRODUCTION AND BACKGROUND

Recently, researchers have begun applying machine learning techniques to the problem of dialog design. The essential idea is that a human designer provides high-level objectives, and an optimization or planning algorithm determines the detailed plan. This optimization usually requires a user simulation, which is a computer program or model that is intended to be a appropriate substitute for a population of real users. A user simulation consists of a *user behavior model* which generates textual synthetic user responses and a *speech recognition model* which simulates the speech recognition process, possibly introducing errors. Common optimization techniques for dialog systems including Markov decision processes [1, 2, 3, 4, 5] and partially observable Markov decision processes [6, 7, 8, 9, 10] require a user simulation.

In evaluations with real users, dialog systems augmented with machine learning have outperformed reasonable baselines [1]. Unfortunately, evaluations with real users are rarely conducted; instead, it is much more common for machine learning applications to be evaluated *exclusively* with a user simulation, for which no measurement of accuracy or reliability is reported [2, 3, 4, 5, 6, 7, 8, 9, 10]. As such, it is hard to judge whether performance improvements will hold once systems are deployed to real users. To address this problem, we seek a quality measure for user simulations, akin to word error rate (WER) for speech recognition accuracy, perplexity for language modelling, or BLEU score for machine translation.

In past work, Cuayahuitl et al [11] evaluate a user simulations by computing the "dialog similarity" of a real and simulated corpus. Each of these two corpora are viewed as the output of a hidden Markov model (HMM), and the dialog similarity measure is defined as the divergence between the distributions estimated for these two HMMs. This method has the desirable property of producing a scalar-valued distance which can be used to rank order different user simulations. However, casting dialog as the output of an HMM makes strong structural assumptions, and it is not clear how to determine how well the estimated HMMs match the corpora. In addition, it is unclear how to express the relative importance of different dialog elements, such as task completion and dialog length, in a given domain. Finally, to present an evaluation, many details of the HMMs such as their states, transition structures, parameterizations, estimation methods, etc. would need to be discussed, which is cumbersome for practitioners and researchers. In other work, Schatzmann et al [12] propose a broad toolkit of tests for comparing simulated and real dialogs, such as computing the precision and recall of simulated and real user responses. However, the authors do not take up the problem of a single quality measure for a user simulation. In sum, in the field there is no accepted, easily reportable statistic providing an indication of the quality of a user simulation.

This paper is organized as follows. First, section 2 states our assumptions and presents the evaluation procedure. Then, section 3 provides an illustration using real dialog data, and confirms that the evaluation procedure agrees with commonsense intuition. Finally, recognizing that there may be a small number of real dialogs available, section 4 tackles the problem of data sparsity and develops a concise table of critical values for practitioners to easily interpret the reliability of an evaluation. Section 5 then concludes.

2. METHOD

Although past work has argued that the aim of a user simulation is to engage in "realistic" dialogs [12, 11], in practice it is unclear how realism could be implemented as a quantitative metric. Here we take a slightly different view. We believe that the role of a user simulation is to accurately *predict* the performance of a dialog system when it is deployed to a certain user population. More formally:

> For a given dialog system $\mathbb{D}$ and a given user population $\mathbb{U}_0$, the goal of a user simulation $\mathbb{U}_1$ is to accurately predict the performance of $\mathbb{D}$ when it is used by $\mathbb{U}_0$.

Here, user population is defined to include the variations expected across users and the variations expected for each individual user, including variations in initiative levels, dialog act frequencies, patience, and so on. For a goal-oriented dialog system, the user population includes the variety and frequency of the tasks that users are trying to accomplish.

We next address performance in a single dialog:

> The performance of a dialog system $\mathbb{D}$ in a particular dialog $d_{(i)}$ can be expressed as a real-valued score $x_{(i)}$, computed by a scoring function $\mathbb{Q}(d_{(i)}) = x_{(i)}$.

The scoring function itself is dependent on the dialog system and is created by its designer. The scoring function takes as input all of the factors that the designer believes are relevant – such as task completion, dialog length, and user satisfaction. The designer may base the scoring function on business requirements [13] or a weighted sum of factors intended to predict user satisfaction such as the PARADISE method [14]. Often, the scoring function is already available since it is required by many machine learning algorithms, where it is sometimes called a *reward function* [2, 8].

Next, these scores can be aggregated into lists:

> A given user population $\mathbb{U}_0$ using dialog system $\mathbb{D}$ will yield a list of scores $\mathcal{S}_0 = (x^0_{(1)}, \ldots, x^0_{(N_0)})$. Similarly, a user simulation $\mathbb{U}_1$ using dialog system $\mathbb{D}$ will yield a list of scores $\mathcal{S}_1 = (x^1_{(1)}, \ldots, x^1_{(N_1)})$.

With these two lists, we can now state the basic intuition of our quality measure for a user simulation:

> A user simulation $\mathbb{U}_1$ may be evaluated by computing a real-valued divergence $D(\mathcal{S}_0||\mathcal{S}_1)$.

In this paper we define a *divergence* $D(\mathcal{X}||\mathcal{Y})$ to be a scalar, non-negative measurement of how well some list $\mathcal{X}$, which are samples from a "true" distribution, is matched by some other list $\mathcal{Y}$, which are samples from a "model" of the truth. If $D(\mathcal{X}||\mathcal{Y}) = 0$, then $\mathcal{Y}$ is taken to be a perfect model of $\mathcal{X}$.

In the limit of an infinite number of dialogs, the sets $\mathcal{S}_0$ and $\mathcal{S}_1$ could be described by probability density functions $p_0(x)$ and $p_1(x)$. In practice, however, collecting real dialogs is expensive and time-consuming, and there may only be $N_0 = 50$ or 100 real dialogs available. Moreover, it seems unlikely that we will know the parametric form of $p_0(x)$ in advance. Thus, an estimate of the density is unlikely to be reliable, and the divergence measure should not depend on a density.

Given this consideration, a natural choice of divergence measure is the *normalized Cramér-von Mises* divergence:

$$D(F_0||F_1) = \alpha \sqrt{\sum_{i=1}^{N_0} (F_0(x^0_{(i)}) - F_1(x^0_{(i)}))^2} \qquad (1)$$

where F_j is the *empirical distribution function* (EDF) of the data $\mathcal{S}_j = (x^j_{(1)}, \ldots, x^j_{(N_j)})$:

$$F_j(x) = \frac{1}{N_j} \sum_{i=1}^{N_j} \begin{cases} 1 & \text{if} \quad x^j_{(i)} < x \\ \frac{1}{2} & \text{if} \quad x^j_{(i)} = x \\ 0 & \text{if} \quad x^j_{(i)} > x \end{cases} \qquad (2)$$

and $\alpha = \sqrt{(12N_0)/(4N_0^2 - 1)}$ is a normalizing constant which scales $D(F_0||F_1)$ to the range $[0, 1]$.

The normalized Cramér-von Mises divergence is based on a family of statistical tests originally developed by Cramér [15] and von Mises [16] which measure agreement between observed sets of real-valued data. Equation 1 is based on a variant of the Cramér-von Mises test studied by Anderson [17], augmented here with a normalization constant α.

The normalized Cramér-von Mises divergence has a host of desirable properties for evaluating user simulations. Because it operates on the empirical distribution function (EDF), it makes no assumptions about the parametric form of $p(x)$ and requires no tuning parameters. In this respect it is preferable to divergence measures which require an estimate of density, such as the Kullback-Leibler divergence. In addition, the Cramér-von Mises family of tests is regarded as having more statistical power than other non-parametric methods for comparing EDFs [18]. Moreover, because a divergence is a scalar, divergences to different user simulations can be rank-ordered, enabling direct comparisons to be made between different user simulations. Also, no assumptions about the structure of dialogs are made, and so it is concise to report. Finally, the normalization provides a common scale of $[0, 1]$ across experiments and domains, independent of N_0. This feature enables scores to be easily interpreted and compared across different experiments and different domains.

Of course, this method has several limitations. First, even if the dialog scores in the sets $\mathcal{S}_0$ and $\mathcal{S}_1$ and co-incident, the true and modelled dialogs may still be quite different in terms of (for example) length or task completion. This is inevitable with any scalar evaluation metric: for example, in speech recognition, identical word error rates may have different numbers of deletions, insertions, and substitutions. Here, the intention is that the domain-specific scoring function $\mathbb{Q}$

Speaker	Transcript	Dialog score
System	AT&T Dialer.	-1
Caller	[silent]	
System	First name and last name?	-1
Caller	Jeannie Smith	
	[JENNY SCHMIDT U S $\sim$ 85]	
System	Jenny Schmidt, USA	-1
Caller	No	
	[NO $\sim$ 95]	
System	First name and last name?	-1
Caller	Jeannie Smith	
	[JEANNE SMITH $\sim$ 98]	
System	Jeanne Smith. Office or cell?	-1
Caller	Office	
	[OFFICE $\sim$ 82]	
System	Office phone.	-1
Caller	[silent]	
System	Dialing.	$+20$
	Total dialog score	$+14$

Table 1. An example conversation with the AT&T dialer. Text in brackets shows ASR recognition result and confidence score, which ranges from 0 to 100. The dialog scoring function is described in the text below and summarized in table 2. Names have been changed for privacy.

weights the relevant factors of the dialog appropriately, such that any aliasing is by definition acceptable. Also, just as evaluation metrics like word error rate do not suggest how a speech recognizer could be improved, we do not expect that our metric will suggest how a user simulation could be improved. Schatzmann's toolkit [12] seems more appropriate for this type of analysis. Finally, our method requires real dialog data from the dialog system being evaluated, but in practice, a user simulation is often used to build (via machine learning) a *new* dialog manager. Theoretically, it is not correct to make claims about the quality of a user simulation interacting with some *new* dialog system for which real dialogs do not exist. Nonetheless, we expect that – all else being equal – a user simulation that is a better performance predictor of *some* dialog system is likely to be a better performance predictor on a new dialog system.

3. EXAMPLE APPLICATION

In this section, we strive to show that the normalized Cramér-von Mises evaluation procedure agrees with common-sense intuition by studying a corpus of real human-computer dialogs. A series of user simulations are created, and it is shown that increasingly accurate user simulations yield decreasing Cramér-von Mises divergences. In other words, it is shown that the Cramér-von Mises divergence correlates well with the qualitative difference between the real environment and the user simulation.

Condition	Dialog score
System transfers caller to the correct destination	20
System transfers caller to the incorrect destination	-20
System hangs up for any reason	-20
Caller hangs up at very first turn	0
Caller hangs up after very first turn	-5
Each system turn	-1

Table 2. The scoring function used for the voice dialer.

The dialog system presented here is a voice dialer application. This application is accessible within the AT&T research lab and receives daily calls. The dialer's vocabulary consists of approximately 30,000 distinct callees across many business units, not just staff in the research lab, and can disambiguate between people with the same name, and between multiple phone listings for the same person (e.g., office and mobile). Table 1 provides the transcript of an example dialog.

The dialer application was selected for this study because it is used by people with real information needs. Since our focus is on user behavior, it would be less desirable to use dialogs collected from paid subjects, who in effect role-play and do not really suffer the consequences of system failures.

The corpus used here, which excludes calls from system developers, consists of 468 calls from 40 distinct callers. These calls were divided into a training set of 320 calls with 1265 caller turns, and a test set of 148 calls with 581 caller turns, with disjoint sets of callers.

To illustrate the evaluation method, we begin by creating a scoring function (table 2), which reflects the design priorities of this system. Next, two user behavior models and two ASR models were built. A *handcrafted* user behavior model was designed which assumed that the user is cooperative and patient, always answering questions as requested, and never hanging up. Second, a *stochastic* user behavior model was estimated from the training dialog data. At each system prompt, categories of user responses were counted, including different kinds of cooperative answers, out-of-grammar speech, silence, and hang-ups. These frequency counts were used to form a statistical model of user behavior using simple maximum-likelihood estimation. For example, in the situation s = *the user was asked for a name*, the model for the user's action a is $P(a =$*say first and last names*$|s) = 0.692$, $P(a =$*say name with city and state*$|s) = 0.033$, to $P(a =$*say something out of grammar*$|s) = 0.147$, $P(a =$*remain silent*$|s) = 0.039$, and $P(a =$*hang up*$|s) = 0.089$.

Next, two speech recognition simulations were created. Each speech recognition simulation takes as input the text of the user's speech, and produces as output a (possibly erroneous) text string and a confidence score, which is used by the dialog manager to decide whether to accept or discard the output.

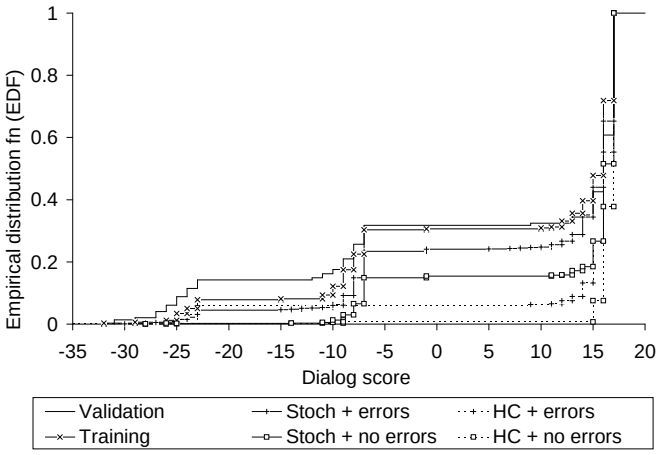

Fig. 1. Empirical distribution function of all user simulations, the training set, and the test set. "HC" is the handcrafted user behavior model, "Stoch" is the stochastic user behavior model estimated from data, and "errors" refer to simulated speech recognition errors.

The first speech recognition simulation made no errors: in-grammar speech was recognized accurately (with the maximum confidence score of 100), silence was correctly identified, and out-of-grammar speech was discarded (via a confidence score of zero). The second speech recognition simulation modelled the errors and confidence scores found in the training set. Error statistics were computed by examining each recognition attempt and determining whether the user's speech a was in-grammar, out-of-grammar, or empty, and also determining whether the recognition outcome $\tilde{a}$ was correct, incorrect, or empty. Counts of each $(a, \tilde{a})$ pair were made and used to compute conditional probabilities $P(\tilde{a}|a)$. For example, when the user said an in-grammar name (action a), the model for the outcome $\tilde{a}$ is $P(\tilde{a} =$*recognized correctly*$|a) = 0.795$, $P(\tilde{a} =$*recognized incorrectly*$|a) = 0.190$, $P(\tilde{a} =$*mistaken for silence*$|a) = 0.015$. In addition, for each $(a, \tilde{a})$ pair, confidence score frequencies were counted and used to simulate confidence scores in simulation.

Each of the two user behavior models (*handcrafted* and *stochastic*) was run with each of the ASR simulations (*with errors* and *without errors*) for 1000 dialogs, and each dialog was scored using the scoring function described in table 2. The EDF for each user behavior model/ASR model pair was then computed and plotted in figure 1. Finally, the normalized Cramér-von Mises divergences from the test set were computed, shown in table 3.

The handcrafted user behavior model with no ASR errors produces the largest Cramér-von Mises divergence; the stochastic user behavior with ASR errors produces the smallest Cramér-von Mises divergence; and the other combinations are between these two. In other words, as the predictive accuracy of the user simulation increases, its Cramér-von Mises divergence decreases. In this experiment, the best and worst user simulations were known in advance by design: the key finding is that the Cramér-von Mises divergence has recovered this ordering, and this result lends support to our claim that the normalized Cramér-von Mises divergence is a suitable quality measure for user simulations.

Dialogs used to compute EDF $\hat{F}$	$D(F\|\|\hat{F})$
Handcrafted behavior + no ASR errors	0.36
Stochastic behavior + no ASR errors	0.21
Handcrafted behavior + ASR errors	0.20
Stochastic behavior + ASR errors	0.067
Training set (real dialogs)	0.098

Table 3. Cramér-von Mises divergence between the EDF of the test set of dialogs (F) and other corpora.

In addition, the divergence from the held-out test set to the training set is slightly greater than that to the best use simulation, indicating that the predictive accuracy of the best user simulation is within the bounds of sampling error measured with held-out data. Yet this raises an important question: Is the difference between the best and worst user simulations reliable? More generally, what magnitude of difference in divergence is statistically significant? This is the question addressed in the next section.

4. STATISTICAL SIGNIFICANCE

To begin, consider the cumulative distribution functions $P_j(x)$ and probability density functions $p_j(x)$ for the user population and two user simulations. By definition, these describe the true cumulative distribution and probability densities of the user population and the two user simulations in the presence of an infinite number of samples. The normalized Cramér-von Mises divergence on the true distributions is [19]:

$$D^*(P_0||P_j) = \beta \sqrt{\int (P_0(x) - P_j(x))^2 p_0(x) dx} \quad (3)$$

where $\beta = \sqrt{3}$ is a normalization constant that scales the divergence to the range $[0, 1]$.

If this test is applied to each user simulation and it is found that $D^*(P_0||P_1) < D^*(P_0||P_2)$, then it could be concluded that user simulation 1 is better than user simulation 2 (and visa-versa). Since these quantities are exact, there is no chance that an observed difference would be due to noise: any difference is statistically significant. In practice however, we will not have access to $P_j(x)$ nor $p_j(x)$. Rather, we have samples from these distributions $\mathcal{S}_j = (x^j_{(1)}, \ldots, x^j_{(N_j)})$ which are used to compute $D(F_0||F_j)$. The key issue is that $D(F_0||F_j)$ is an *estimate* of $D^*(P_0||P_j)$ and therefore subject to sampling error.

Developing critical values for Cramér-von Mises-type tests is analytically quite difficult [18]. Here we tackle this problem by constructing a simulation experiment. We randomly

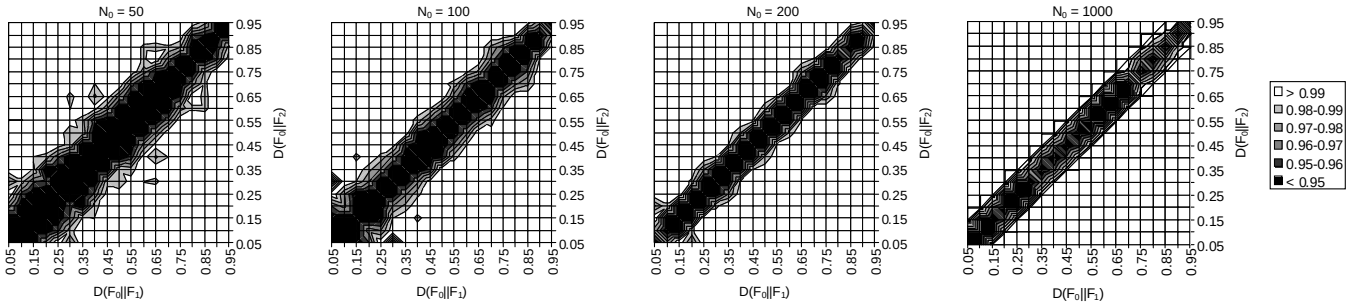

Fig. 2. Measured divergence to user simulation 1 $D(F_0||F_1)$ and user simulation 2 $D(F_0||F_2)$ vs. ordering reliability for $N_1 = N_2 = 1000$ dialogs with each user simulation and various numbers of "real" user dialogs N_0.

generate distributions for a user population $P_0(x)$ and two user simulations $P_1(x)$ and $P_2(x)$. Then, we compute the *true* ordering of the two user simulations as the ordering of $D^*(P_0||P_1)$ and $D^*(P_0||P_2)$. Next, we sample from $P_0(x)$, $P_1(x)$, and $P_2(x)$ to produce $F_0(x)$, $F_1(x)$ and $F_2(x)$, and compute *predicted* ordering of the two user simulations as the ordering of $D(F_0||F_1)$ and $D(F_0||F_2)$. Finally we determine if the predicted ordering agrees with the true ordering, and set an indicator variable q to 1 if the predicted ordering matches true ordering, and to 0 if not. This whole process is repeated M times, and for each iteration m, $D(F_0||F_1)$, $D(F_0||F_2)$, and q are stored as D_1^m, D_2^m, and q^m, respectively. Once the sampling is complete, a plot is constructed which quantizes D_1 and D_2 into square regions. Within each region, the average value of q (notated $\overline{q}$) is computed, which corresponds to the percentage of the time that the sampled data yields the same ordering as the true data. In other words, the end result is a statement of the accuracy of the ordering of 2 user simulations for a given D_1, D_2, N_0, N_1 and N_2.

Concretely, $p_j(x)$ are bi-modal densities represented as the weighted sum of Gaussians, with means sampled uniformly from $[0, 100]$, variances sampled from $[1, 5]$, and weights sampled from $[0, 1]$ and normalized.[1] In these experiments, the number of dialogs from each user simulation is $N_1 = N_2 = 1000$, and $M = 40,000$ iterations are run for each experiment. Experiments were run for various number of dialogs from the "real" user N_0 ranging from 50 to 1000.

Figure 2 shows results. In this figure, black regions indicate $\overline{q} < 0.95$, white regions indicate $\overline{q} > 0.99$, and various shades of gray indicate intermediate values. As the number of real dialogs increases, the (dark) region of low ordering reliability becomes more confined. In addition, the regions of lower probability lie along essentially straight lines parallel to $D_1 = D_2$. This is significant because it implies that the reliability of an ordering is determined mainly by the *difference* between D_1 and D_2, rather than being dependent on their actual values. This result is summarized in table 4, which provides an indication of what differences in divergences are required to conclude an ordering of user simulations is reliable with confidence 90% and 95%.

N_0	$p > 0.90$	$p > 0.95$
50	0.08	0.12
100	0.06	0.09
200	0.05	0.07
500	0.04	0.05
1000	0.03	0.04

Table 4. Difference in normalized Cramér-von Mises divergence between two user simulations required for rank-ordering to be correct for 1000 simulated dialogs and various numbers of real dialogs N_0 with confidence $p > 0.90$ and $p > 0.95$.

Returning to the illustration in section 3, the results in table 4 indicate that, for 100 dialogs in the test set, a difference of 0.06 indicates a 90% ordering accuracy, and a difference of 0.09 indicates a 95% ordering accuracy. This implies that the handcrafted user behavior with no ASR errors is indeed significantly worse than the other user simulations ($p > 0.95$), because $|0.36 - 0.21| = 0.15 > 0.09$. Similarly, the stochastic user behavior with ASR errors is significantly better than the other user simulations ($p > 0.95$). Further, the difference observed between the stochastic user behavior model with ASR errors and the training set does not allow a statistically significant ordering to be inferred ($|0.067 - 0.098| = 0.031 < 0.06$), which is consistent with the hypothesis that the ordering here is due to sampling noise.

5. CONCLUSIONS

This paper has sought to provide system designers and practitioners with a simple, principled method of evaluating and rank-ordering user simulations, based on the normalized Cramér-von Mises divergence. An illustration with a corpus of dialogs collected from real system usage confirms that as the predictive accuracy of a user simulation is improved, the normalized Cramér-von Mises divergence between the real dialogs and the synthetic dialogs decreases. Further, a series of sim-

[1] Additional experimentation (not described here) showed that the findings below were unchanged for larger numbers of modes.

ulation experiments has explored what magnitude of difference in Cramér-von Mises divergences is required to infer a statistically significant rank-ordering, and we have developed a concise table that enables researchers and practitioners to judge whether an observed ordering of two user simulations is statistically significant.

We anticipate that dialog systems will make increasing use of machine learning. Since evaluations with real users will remain expensive, we therefore foresee that evaluations with user simulations will also become more widespread. For these evaluations to be accepted, the quality of the user simulations must themselves be tested in some way. The evaluation metric suggested here is straightforward to apply, concise to report, and easy to interpret, and we hope that it will go some way toward satisfying this need.

6. ACKNOWLEDGEMENTS

Thanks to Bob Bell for many insightful conversations, to Vincent Goffin for help with the voice dialer code and logs, and to Srinivas Bangalore for helpful comments about the presentation.

7. REFERENCES

[1] S Singh, DJ Litman, M Kearns, and MA Walker, "Optimizing dialogue management with reinforcement leaning: experiments with the NJFun system," *Journal of Artificial Intelligence*, vol. 16, pp. 105–133, 2002.

[2] E Levin, R Pieraccini, and W Eckert, "A stochastic model of human-machine interaction for learning dialogue strategies," *IEEE Trans on Speech and Audio Processing*, vol. 8, no. 1, pp. 11–23, 2000.

[3] K Scheffler and SJ Young, "Automatic learning of dialogue strategy using dialogue simulation and reinforcement learning," in *Proc Human Language Technologies (HLT), San Diego, USA*, 2002, pp. 12–18.

[4] O Pietquin, *A framework for unsupervised learning of dialogue strategies*, Ph.D. thesis, Faculty of Engineering, Mons (TCTS Lab), Belgium, 2004.

[5] J Henderson, O Lemon, and K Georgila, "Hybrid reinforcement/supervised learning for dialogue policies from Communicator data," in *Proc Workshop on Knowledge and Reasoning in Practical Dialogue Systems, IJCAI, Edinburgh*, 2005, pp. 68–75.

[6] N Roy, J Pineau, and S Thrun, "Spoken dialogue management using probabilistic reasoning," in *Proc ACL, Hong Kong*, 2000, pp. 93–100.

[7] B Zhang, Q Cai, J Mao, E Chang, and B Guo, "Spoken dialogue management as planning and acting under uncertainty," in *Proc Eurospeech, Aalborg, Denmark*, 2001, pp. 2169–2172.

[8] JD Williams, *Partially Observable Markov Decison Processes for Spoken Dialogue Management*, Ph.D. thesis, Cambridge University, 2006.

[9] SJ Young, J Schatzmann, K Weilhammer, and H Ye, "The hidden information state approach to dialog management," in *Proc ICASSP, Hawaii*, 2007, pp. IV149–IV152.

[10] TH Bui, M Poel, A Nijholt, and J Zwiers, "A tractable DDN-POMDP approach to affective dialogue modeling for general probabilistic frame-based dialgue systems," in *Proc Workshop on Knowledge and Reasoning in Practical Dialogue Systems, IJCAI, Hyderabad*, 2007, pp. 34–37.

[11] Heriberto Cuayáhuitl, Steve Renals, Oliver Lemon, and Hiroshi Shimodaira, "Human-computer dialogue simulation using hidden markov models," in *Proc ASRU, Puerto Rico*, 2005, pp. 290–295.

[12] J Schatzmann, K Georgila, and SJ Young, "Quantitative evaluation of user simulation techniques for spoken dialogue systems," in *Proc SIGdial Workshop on Discourse and Dialogue, Lisbon*, 2005, pp. 178–181.

[13] E Levin and R Pieraccini, "Value-based optimal decision for dialog systems," in *Proc SLT, Aruba*, 2006, pp. 198–201.

[14] MA Walker, CA Kamm, and DJ Litman, "Towards developing general models of usability with PARADISE," *Natural Language Engineering: Special Issue on Best Practice in Spoken Dialogue Systems*, 2000.

[15] H Cramér, "On the composition of elementary errors. second paper: Statistical applications," *Skandinavisk Aktuarietidskrift*, vol. 11, pp. 171–180, 1928.

[16] R von Mises, *Wahrscheinlichkeitsrechnung und ihre Anwendung in der Statistik und Theoretischen Physik*, F Deuticke, 1931.

[17] TW Anderson, "On the distribution of the two-sample Cramér-von Mises criterion," *Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1148–1159, 1962.

[18] MA Stephens, "Introduction to: Kolmogorov (1933) on the empirical determination of a distribution," in *Breakthrougth in statistics*, S Kotz and NL Johnson, Eds., vol. II, pp. 93–105. Springer Verlag, 1992.

[19] WT Eadie, D Drijard, FE James, MGW Roos, and B Sadoulet, *Statistical Methods in Experimental Physics*, North Holland, 1971.