Validating Attention Classifiers for Multi-Party Human-Robot Interaction

Mary Ellen Foster
The Interaction Lab, School of Mathematical and Computer Sciences
Heriot-Watt University, EH14 4AS, Edinburgh, UK
M.E.Foster@hw.ac.uk

ABSTRACT

A critical task for a robot designed for interaction in a dynamic public space is estimating whether each of the people in its vicinity is currently seeking the robot's attention. In previous work, we implemented two strategies for estimating the attention-seeking state of customers for a robot bartender—a rule-based classifier derived from the analysis of natural human behaviour, and a set of classifiers trained using supervised learning on a labelled multimodal corpus—and compared the classifiers through cross-validation and in the context of a full-system evaluation. However, because the ground-truth user behaviour was not available, the user study did not fully assess the classifier performance. We therefore carried out a new study validating the performance of all classifiers on a newly recorded, fully labelled test corpus. The highest-scoring trained classifier from the cross-validation study performed very badly on this new test data, while the hand-coded rule and other trained classifiers did much better. We also explored the impact of including information from previous frames in the classifier state: including previous sensor data had a mixed effect, while including the previous attention estimates greatly diminished the performance of all classifiers.

Categories and Subject Descriptors: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems – Evaluation/methodology; I.2.6 [Artificial intelligence]: Learning

Keywords: Social signal processing; supervised learning

1. INTRODUCTION

Human face-to-face communication is a continuous process of exchanging and interpreting multimodal communicative signals [24]. For an robot to participate successfully in this context, it needs more than just the physical skills to perform objective tasks in the world; it also needs the appropriate social skills to understand and respond to the multimodal social signals from its human partners (e.g., gaze, facial expression, and language). The state of the art in input processing areas such as computer vision or speech recognition is to produce a continuous stream of noisy sensor data. In order for this information to be useful for decision-making in an interactive system, all of this continuous, noisy, single-channel information must be combined into a discrete, cross-modal representation to allow the decision-making components to select appropriate behaviour. This is the task of *social signal processing*, a topic that has received increasing attention in recent years—e.g., see [29] for a recent survey.

We consider a robot designed to share a dynamic, multi-party social space, where not all of the participants require attention from the robot at any given time. For such a robot, a crucial task is to estimate *attention seeking*: that is, determining, for each person in the scene, whether that person currently requires attention from the system. Bohus and Horvitz [4, 5] pioneered the use of data-driven methods in this context, by training models designed to predict user engagement based on information from face tracking, pose estimation, person tracking, group inference, along with recognised speech and touch-screen events. A number of more recent systems have also used machine learning to address this task. For example, Li et al. [19] estimated the attention state of users of a robot in a public space, combining person tracking, facial expression recognition, and speaking recognition. Castellano et al. [6] trained a range of engagement classifiers on labelled data extracted from the logs of children interacting with a chess-playing robot. McColl and Nejat [22] automatically classified the social accessibility of people interacting with their robot based on their body pose, while MacHardy et al. [21] classified the engagement states of audience members for an online lecture based on information from facial feature detectors.

Like the above systems, we also take a data-driven approach to this task, making use of the available data in two distinct ways: (1) defining a hand-coded classifier based on rules derived from the observation of natural human behaviour in real bars, and (2) using an annotated corpus of human-robot interactions to train a range of supervised learning classifiers. In a previous study, we compared the classifiers through cross-validation on the training corpus, and also compared the top-performing trained classifier with the rule-based classifier in the context of a user evaluation of the entire system. However, because the ground-truth attention-seeking behaviour of the users in that study is not available, the practical implications are difficult to interpret. In this paper, we therefore test the performance of all of the classifiers (rule-based and trained) on a newly-recorded, fully annotated test corpus. We also examine the impact of incorporating temporal features into the classifier state.

2. CLASSIFYING CUSTOMER ATTENTION FOR A ROBOT BARTENDER

This work takes place in the context of the socially aware humanoid robot bartender shown in Figure 1. The hardware for the robot bartender consists of two manipulator arms with grippers, mounted to resemble human arms, along with an animatronic talking head capable of producing facial expressions, rigid head motion, and lip-synchronised synthesised speech; full details of the software

A customer attracts the bartender's attention
ROBOT: [Looks at Customer 1] How can I help you?
CUSTOMER 1: A pint of cider, please.
Another customer attracts the bartender's attention
ROBOT: [Looks at Customer 2] One moment, please.
ROBOT: [Serves Customer 1]
ROBOT: [Looks at Customer 2]
Thanks for waiting. How can I help you?
CUSTOMER 2: I'd like a pint of beer.
ROBOT: [Serves Customer 2]

Figure 1: A socially aware robot bartender

architecture and components of the system are presented in [9]. The bartender supports interactions similar to the one shown in Figure 1, in which two customers enter the bar area and each attempt to order a drink from the bartender. Note that when the second customer attempts to attract the bartender's attention while the bartender is in the process of serving the first customer, the bartender reacts by telling the second customer to wait, finishing the transaction with the first customer, and only then serving the second customer. This socially appropriate behaviour is based on the observation of real bartenders interacting with customers in a natural context [20].

In the context of the above bartending scenario, the main role of social signal processing is to estimate *attention seeking*: determining, for each customer in the scene, whether that customer currently requires attention from the system. This information is critical for implementing the socially appropriate behaviour in the sample interaction. User attention is estimated based on the low-level sensor data published on two continuous input channels. The computer vision system [3, 25] tracks the location, facial expressions, gaze behaviour, and body language of all people in the scene in real time, using a set of visual sensors including two calibrated stereo cameras and a Microsoft Kinect [23] depth sensor. The data from the vision system is published as frame-by-frame updates multiple times a second. The other primary input modality in the system is linguistic [26], combining a speech recogniser with a natural-language parser to create symbolic representations of the speech from all users. For speech recognition, we use the Microsoft Speech API together with the directional microphone array of a second Kinect; incremental hypotheses are published constantly, and recognised speech is parsed using a grammar implemented in OpenCCG [31] to extract the syntactic and semantic information.

Concretely, we consider the following low-level sensor features for the task of classifying customer attention:

- The (x, y, z) coordinates of each customer's head, left hand, and right hand as reported by the vision system;
- The angle of each customer's torso in degrees, where 0° indicates that the customer is facing directly forwards; and
- An estimate of whether each customer is currently speaking, derived from the estimated source angle of each speech hypothesis along with the location information from vision.

CVR	Classifies using regression: the target class is binarised, and one regression model is built for each class value [12].
IB1	A nearest-neighbour classifier that uses normalised Euclidean distance to find the closest training instance [2].
J48	Classifies instances using a pruned C4.5 decision tree [27].
JRip	Implements the RIPPER propositional rule learner [8].
LibSVM	Generates a Support Vector Machine using LIBSVM [7].
Logistic	Multinomial logistic regression with a ridge estimator [18].
NaiveBayes	A Naïve Bayes classifier using estimator classes [15].
ZeroR	Baseline classifier; always predicts the most frequent value.

Figure 2: Classifiers considered

Classifier	Accuracy	AUC	Precision	Recall	F
IB1	0.960	0.932	0.957	0.958	0.957
J48	0.924	0.919	0.925	0.925	0.925
JRip	0.911	0.868	0.913	0.914	0.913
CVR	0.921	0.960	0.911	0.912	0.912
Logistic	0.780	0.739	0.727	0.781	0.710
LibSVM	0.790	0.521	0.830	0.790	0.706
NaiveBayes	0.669	0.656	0.726	0.662	0.685
ZeroR	0.780	0.500	0.609	0.780	0.684
Rule	0.655	*na*	0.635	0.654	0.644

Table 1: Cross-validation results, sorted by F score (from [10])

Every time a new frame is published from the vision system, the attention state of every customer in the scene is estimated using the above sensor features, using two classification strategies. The first strategy employed a simple rule derived from the observation of customers in real bars [20]: a customer was defined to be seeking attention exactly when (1) they were close to the bar, and (2) their torso was turned towards the bartender. The second strategy used an annotated corpus of human-robot interactions to train a range of off-the-shelf supervised learning classifiers using the Weka data mining toolkit [13]. To cover a variety of learning strategies, we used the representative classifiers from the Weka primer [1]; the details are given in Figure 2. Note that for all of our experiments, we treat the classifiers as "black boxes" [28], using the default parameter settings given by Weka and looking only at the classified output. We discuss extensions to this approach at the end of the paper.

In a previous study we compared all of the classifiers through 10-fold cross-validation against the training corpus. The results of this cross-validation study are reproduced in Table 1, where the groupings in the table reflect differences among the F scores that were significant at the $p < 0.01$ level on a paired T test based on 10 independent cross-validation runs. In a follow-up experiment, the the best-performing trained classifier from the cross-validation study—the IB1 (instance-based) classifier—was compared with the rule-based classifier in the context of an interactive user evaluation of the entire bartender system. The main finding was that the trained classifier changed its estimate of the user's attention state significantly more often than did the rule-based classifier; the trained classifier also tended to detect attention-seeking somewhat more quickly, although that tendency was not found to be significant. The details of the cross-validation and user studies are presented in [10].

3. VALIDATING THE CLASSIFIERS

In the user evaluation summarised above, the ground truth about the customers' actual attention-seeking behaviour was not available. All of the objective metrics used to compare the two classifiers were therefore—necessarily—based solely on the assumption that all customers followed the instructions that they were given: to attract the attention of the bartender and order a drink (as in Figure 1). This

(a) Customer not seeking attention

(b) Customer seeking attention

Figure 3: Sample images from the test data

makes the results of the user study difficult to interpret, as it is impossible to know which of the classifiers actually estimated customer attention more accurately in practice; also, due to the study design, there would have been very few true negative examples. We therefore carried out a new evaluation of the attention classifiers, making use of a newly-recorded test corpus addressing the weaknesses of the previous study: namely, the attention-seeking behaviour of all customers is fully annotated, and the data includes examples of customers who were both seeking and not seeking attention.

The test data is based on six videos, each showing a single customer in front of the bar, as in the sample images in Figure 3. Two different customers were recorded: one who was involved in the human-robot interactions making up the original training corpus, and one who was not. The Elan annotation tool [32] was used to annotate the videos, using the same labels as the original training data: the customer's attention state was labelled as either *NotSeekingAttention* (Figure 3a) or *SeekingAttention* (Figure 3b). The video annotations were synchronised with the frame-by-frame information produced by the JAMES vision system, and a corpus instance was then created from the relevant data in each vision frame, using the annotation for the relevant time stamp as the gold-standard label. In total, the test corpus consisted of 361 instances: 233 labelled as *NotSeekingAttention*, and 128 labelled as *SeekingAttention*.

We then trained each classifier on the full training corpus from the previous study [10], and used each trained classifier to predict labels for every instance in the test data. The results of this test are shown in Table 2, sorted—as in Figure 4—by weighted average F score. As shown by the groupings in the table, the results fell into three broad categories: at the top, the hand-coded rule and the J28, CVR, and NaiveBayes classifiers all had F scores well above the baseline ZeroR classifier, which always chooses the highest-frequency label (*NotSeekingAttention*); the LibSVM classifier exactly reproduced the baseline ZeroR behaviour; while the JRip, Logistic, and IB1 classifiers all did worse than this baseline.

These results contrast strongly with the cross-validation results from Table 1. Firstly, the overall numbers are much lower: while the top performing classifiers from the previous study had scores well above 0.9 on all measures, the top results in this study were in the range of 0.6–0.7. Also, the relative ordering of the classifiers is very different: while the IB1 and JRip classifiers did well on cross-validation, they were both among the lowest-performing classifiers on the test data. On the other hand, the NaiveBayes classifier and the hand-coded rule—which were both near the bottom on the cross-validation study—both scored at or near the top on the test data. Other classifiers such as J48 and CVR did well both in cross-validation and on the test corpus.

Classifier	**Accuracy**	**AUC**	**Precision**	**Recall**	**F**
Rule	0.681	*na*	0.694	0.681	0.687
J48	0.648	0.583	0.661	0.648	0.653
CVR	0.598	0.576	0.612	0.598	0.604
NaiveBayes	0.571	0.528	0.638	0.571	0.578
LibSVM	0.645	0.500	0.417	0.645	0.506
ZeroR	0.645	0.500	0.417	0.645	0.506
JRip	0.421	0.350	0.557	0.421	0.432
Logistic	0.438	0.329	0.390	0.438	0.411
IB1	0.349	0.341	0.388	0.349	0.363

Table 2: Classifier performance on the test set, sorted by F score

	NotSeekingAttention			*SeekingAttention*		
Classifier	**Prec**	**Rec**	**F**	**Prec**	**Rec**	**F**
Rule	0.678	0.966	0.796	0.724	0.164	0.268
J48	0.748	0.687	0.736	0.503	0.578	0.538
CVR	0.706	0.648	0.676	0.442	0.508	0.473
NaiveBayes	0.750	0.502	0.602	0.434	0.695	0.535
LibSVM	0.645	1.000	0.785	0.000	0.000	0.000
ZeroR	0.645	1.000	0.785	0.000	0.000	0.000
JRip	0.575	0.395	0.468	0.299	0.469	0.365
Logistic	0.556	0.644	0.596	0.088	0.063	0.073
IB1	0.495	0.395	0.439	0.194	0.266	0.224

Table 3: Per-class precision, recall, and F score

To better understand the performance of the classifiers, we inspected the classifier output on each of the test-data videos. Figure 4 (at the end of the paper) shows the gold-standard annotation for two of the test videos, along with the labels produced by each classifier on those same videos. The light yellow regions correspond to the frames labelled with the *NotSeekingAttention* class, while the dark blue regions correspond to the *SeekingAttention* class. The figure clearly suggests differences among the classifiers: for example, the hand-coded rule selected *SeekingAttention* very rarely; on the other hand, the lowest-performing classifiers (JRip, Logistic, IB1) selected *SeekingAttention* frequently, even in cases (as in Video 2) where the customer never entered this state.

The results in Table 2 reflect weighted averages across both classes. To investigate the above class-specific tendencies more closely, we therefore also computed precision, recall, and F scores separately on the *SeekingAttention* and *NotSeekingAttention* classes; these results are presented in Table 3. In summary, most of the classifiers had higher precision/recall scores on the *NotSeekingAttention* class than on the *SeekingAttention* class, possibly reflecting the fact that this class was the larger in both the training and the test data. The performance on the *SeekingAttention* class varied greatly: ZeroR and LibSVM never selected this class at all; the hand-coded rule and the Logistic and IB1 classifiers had very low recall; while the other classifiers did a much better job at detecting this state.

4. ADDING TEMPORAL CONTEXT

In both the original cross-validation study and in the experiment described above, the input to the classifier consisted only of the sensor data at a given instant, without taking into account any of

the temporal context provided by the interaction. However, real customers switch their attention-seeking state relatively infrequently, so classifying each input frame independently tends to overestimate the number of attention changes. This tendency can be seen in the sample output in Figure 4, where even the best-performing classifiers changed their estimate much more frequently than the gold standard. Table 4 shows the mean number of attention switches per test video produced by each classifier; with the exception of the two classifiers which always select *NotSeekingAttention*, all of the numbers are well above the gold-standard value of 2.0. Note that on the previous user study, stability was the main significant difference between the performance of the hand-coded classifier and the trained IB1 classifier [10]: the hand-coded rule changed its estimate an average of 12.0 times per interaction, while the value for the IB1 classifier was 17.6.

Classifier	Switches
Rule	4.7
J48	10.5
CVR	8.8
NaiveBayes	5.8
LibSVM	0.0
ZeroR	0.0
JRip	11.3
Logistic	5.3
IB1	9.3
Gold standard	*2.0*

Table 4: Stability

If an attention classifier—even one with high overall accuracy—changes its estimate too frequently, the job of the system's interaction manager is made more difficult, in that responding to every change in estimated state is likely to produce undesirable behaviour. In an alternative, unsupervised, POMDP-based approach to interaction management, this issue is addressed by making the POMDP "sticky"; that is, biasing it towards self-transitions [30]. In an effort to improve the stability of the trained classifiers used here, we test two methods of incorporating information from previous frames into the state. We first try adding sensor data from previous frames to the state; we then try adding the classification of previous frames. We do not consider the ZeroR or LibSVM classifiers in this section, as their performance is not affected by either of the manipulations considered here: in all cases, these classifiers still label all instances in the test set as *NotSeekingAttention*.

4.1 Previous Sensor Data

The state used in the previous classification experiments included only the sensor data from the current vision frame; we will call this frame f_0. To incorporate some temporal context, we modified the state to add sensor data from the following frames: the immediately preceding frame (f_1), five frames in the past (f_5), and ten frames in the past (f_{10}). To test if these new attributes could help in classification, we first used Correlation-Based Feature Selection (CBF) [14] to select the relevant state features; the result included the full (x, y, z) face position and some of the hand coordinates from f_0, along with the face (x, y) position and the right hand x coordinate from f_1. Note that the attributes selected from f_0 are essentially the same as those selected from the original training data [10]; the additional presence of features from f_1 confirms that the addition of temporal context has the potential to improve classifier performance.

We next re-ran the cross-validation study with the revised states, and also tested the newly trained classifiers against the test data. Table 5 shows the weighted average F score of all trained classifiers from this study, both from 10-fold cross-validation against the training corpus and when run against the test corpus. The overall cross-validation results were similar to those on the original training corpus (Table 1). On the test set, the J48 classifier still had the best overall performance, with a similar F score; the performance of the IB1 classifier improved dramatically, with an F score going from 0.363 to 0.609; while the other classifiers all saw reduced performance on the test set. The final column of Table 5 shows the mean number of times that each classifier changed its estimate per video; these numbers are broadly similar to those in Table 4.

Classifier	F (cv)	F (test)	Switches
J48	0.931	0.614	9.0
CVR	0.926	0.430	7.5
NaiveBayes	0.550	0.485	4.7
JRip	0.921	0.382	9.3
Logistic	0.753	0.418	7.8
IB1	0.878	0.609	11.7

Table 5: Classifier performance with past sensor data

Classifier	F (cv)	F (test, gold)	F (test, est)	Switches
J48	0.975	0.842	0.506	0.0
CVR	0.969	0.864	0.639	12.5
NaiveBayes	0.959	0.845	0.363	1.5
JRip	0.973	0.855	0.506	0.0
Logistic	0.973	0.888	0.542	1.0
IB1	0.980	0.773	0.600	0.7

Table 6: Classifier performance with past classification data

Figure 5 shows the output of all of the newly trained classifiers on the same two sample videos as in Figure 4. Clearly, the addition of the temporal features has caused nearly all of the trained classifiers to select *SeekingAttention* much more frequently than in the original study, even on frames where the customer was not seeking attention, resulting in decreased overall performance for most classifiers; however, the increased amount of context appears to have allowed the IB1 (instance-based) classifier to improve its classification accuracy.

4.2 Previous Classifications

In the preceding section, we investigated the impact of including sensor data from previous frames in the state. Here, we consider another method of modifying the state: including the previous classifier outputs into the state. We first modified the training data in the same way as above, this time by adding the attention label from the f_1, f_5, and f_{10} to the state. As expected, the classification for f_0 depends very strongly on the immediate history; in fact, when we carried out feature selection using CBF on the training data, only the attention label from f_1 was chosen as informative.

The results of this study are presented in Table 6. The first column indicates the F score from 10-fold cross-validation against the revised training corpus; as expected, given the strong predictive power of the previous state, these values are all very high. The next column indicates the F score on the test data where the state is expanded to include the preceding *gold-standard* attention labels; again, as would be expected, these values are generally quite high. However, using the gold-standard labels in this way is an unrealistic test. A better practical assessment of the classifiers is in the next column, which shows the F score when the context includes the previous *estimated* labels—and here the performance is very different. As shown by the sample outputs in Figure 6, the JRip and J48 classifiers choose *NotSeekingAttention* for every frame in the test data, while the Logistic classifier nearly always chose this label; on the other hand, the IB1 classifier labelled nearly every frame as *SeekingAttention*. While the NaiveBayes classifier produced a better spread of estimates, its overall performance was also low; only the CVR classifier had performance close to that found in the preceding studies. The final column of the table shows the number of switches for each classifier; these numbers are generally low, but—due to the above factors—this does not correspond to high-quality output.

5. SUMMARY AND FUTURE WORK

We have carried out a series of experiments testing methods for estimating the attention-seeking state of customers for a robot bartender, based on the low-level sensor information. In a previous study, the classifiers were assessed through cross-validation and in the context of a whole-system study; however, because the ground truth data was not available, the previous user study did not give a full picture of the practical usefulness of the classifiers. Here, we carried out a targeted evaluation using newly recorded, fully annotated test data, and found that the relative performance of the classifiers was different. In the previous study, the instance-based IB1 classifier had the highest performance, but on this study, we found that the J48 decision-tree classifier gave the best estimate of the users' attention state. In all cases, even the top-performing classifiers changed their estimate of the customers' attention state much too frequently; in an attempt to address this, we experimented with adding temporal features to the state, but this generally tended to decrease the classification performance without improving stability.

In this study, as in the previous one, we have made a deliberate choice to treat all of the supervised classifiers as black boxes, using the default parameter settings provided by Weka. This is a similar approach to that taken, for example, by Koller and Petrick [16], who compared the off-the-shelf performance of a number of AI planners when applied to tasks derived from natural language generation. However, it is certain that the relative and absolute performance would be significantly affected by appropriate parameter tuning [17], and in future we will explore the space of parameters more fully. We will also investigate other methods for improving the stability of the classification, either by incorporating other features into the classifier state or by implementing methods similar to the "sticky" infinite POMDP [30]. It might also be that improved stability would be achieved by using temporal models such as Hidden Markov Models or Conditional Random Fields, and we will also investigate these approaches. The annotated training and test data will also soon be made publicly available for any other researchers who want to explore classification techniques. Finally, we will explore methods for making improved use of the classifier output in the context of end-to-end interactions with the robot bartender. Here, an advantage of the J48 classifier over the IB1 classifier is that the former is able to estimate the confidence of its classifications, which can be incorporated into the new state representation which retains the uncertainty coming from the underlying input sensors [11].

6. ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007–2013) under grant agreement no. 270435, JAMES: Joint Action for Multimodal Embodied Social Systems (`james-project.eu`).

7. REFERENCES

[1] Weka primer. `http://weka.wikispaces.com/Primer`.

[2] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[3] H. Baltzakis, M. Pateraki, and P. Trahanias. Visual tracking of hands, faces and facial features of multiple persons. *Machine Vision and Applications*, 23(6):1141–1157, 2012.

[4] D. Bohus and E. Horvitz. Dialog in the open world: platform and applications. In *Proceedings of ICMI-MLMI*, 2009.

[5] D. Bohus and E. Horvitz. Learning to predict engagement with a spoken dialog system in open-world settings. In *Proceedings of SIGDial*, 2009.

[6] G. Castellano, I. Leite, A. Pereira, C. Martinho, A. Paiva, and P. McOwan. Detecting engagement in HRI: An exploration of social and task-based context. In *Proceedings of SocialCom*, 2012.

[7] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, 2011.

[8] W. W. Cohen. Fast effective rule induction. In *Proceedings of ICML*, 1995.

[9] M. E. Foster, A. Gaschler, M. Giuliani, A. Isard, M. Pateraki, and R. P. A. Petrick. Two people walk into a bar: Dynamic multi-party social interaction with a robot agent. In *Proceedings of ICMI*, 2012.

[10] M. E. Foster, A. Gaschler, and M. Giuliani. How can I help you? Comparing engagement classification strategies for a robot bartender. In *Proceedings of ICMI*, 2013.

[11] M. E. Foster, S. Keizer, and O. Lemon. Action selection under uncertainty for a socially aware robot bartender. In *Proceedings of HRI*, 2014.

[12] E. Frank, Y. Wang, S. Inglis, G. Holmes, and I. Witten. Using model trees for classification. *Machine Learning*, 32(1):63–76, 1998.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: an update. *SIGKDD Explorations Newsletter*, 11(1):10–18, Nov. 2009.

[14] M. A. Hall. Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of ICML*, 2000.

[15] G. H. John and P. Langley. Estimating continuous distributions in Bayesian classifiers. In *Proceedings of UAI*, 1995.

[16] A. Koller and R. P. A. Petrick. Experiences with planning for natural language generation. *Computational Intelligence*, 27(1):23–40, 2011.

[17] N. Lavesson and P. Davidsson. Quantifying the impact of learning algorithm parameter tuning. In *Proceedings of AAAI*, 2006.

[18] S. le Cessie and J. van Houwelingen. Ridge estimators in logistic regression. *Applied Statistics*, 41(1):191–201, 1992.

[19] L. Li, Q. Xu, and Y. K. Tan. Attention-based addressee selection for service and social robots to interact with multiple persons. In *Proceedings of the Workshop at SIGGRAPH Asia*, 2012.

[20] S. Loth, K. Huth, and J. P. De Ruiter. Automatic detection of service initiation signals used in bars. *Frontiers in Psychology*, 4(557), 2013.

[21] Z. MacHardy, K. Syharath, and P. Dewan. Engagement analysis through computer vision. In *Proceedings of CollaborateCom*, 2012.

[22] D. McColl and G. Nejat. Affect detection from body language during social HRI. In *Proceedings of IEEE RO-MAN*, 2012.

[23] Microsoft Corporation. Kinect for Windows. URL `http://www.microsoft.com/en-us/kinectforwindows/`.

[24] L. P. Morency. Modeling human communication dynamics. *IEEE Signal Processing Magazine*, 27(5):112–116, 2010.

[25] M. Pateraki, M. Sigalas, G. Chliveros, and P. Trahanias. Visual human-robot communication in social settings. In *Proceedings of ICRA Workshop on Semantics, Identification and Control of Robot-Human-Environment Interaction*, 2013.

[26] R. P. A. Petrick, M. E. Foster, and A. Isard. Social state recognition and knowledge-level planning for human-robot interaction in a bartender domain. In *Proceedings of AAAI 2012 Workshop on Grounding Language for Physical Systems*, 2012.

[27] R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

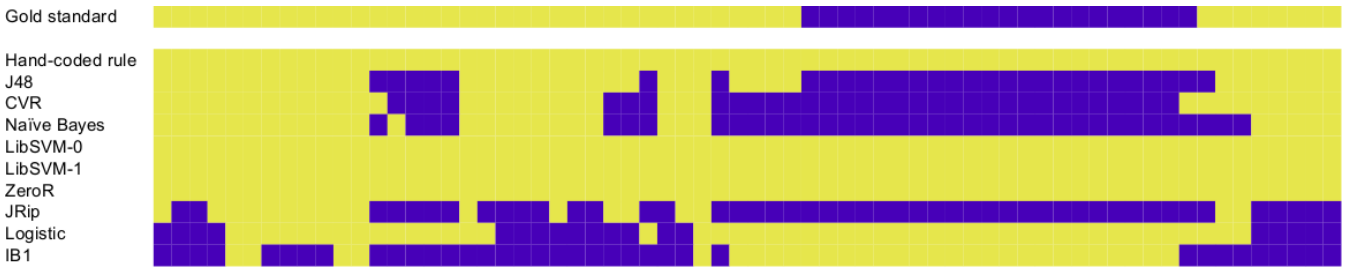
[28] A. Rocha, J. P. Papa, and L. A. A. Meira. How far do we get using machine learning black boxes? *International Journal of Pattern Recognition and Artificial Intelligence*, 26(02):1261001, 2012.

[29] A. Vinciarelli, M. Pantic, D. Heylen, C. Pelachaud, I. Poggi, F. D'Errico, and M. Schroeder. Bridging the gap between social animal and unsocial machine: A survey of social signal processing. *IEEE Transactions on Affective Computing*, 3(1):69–87, 2012.

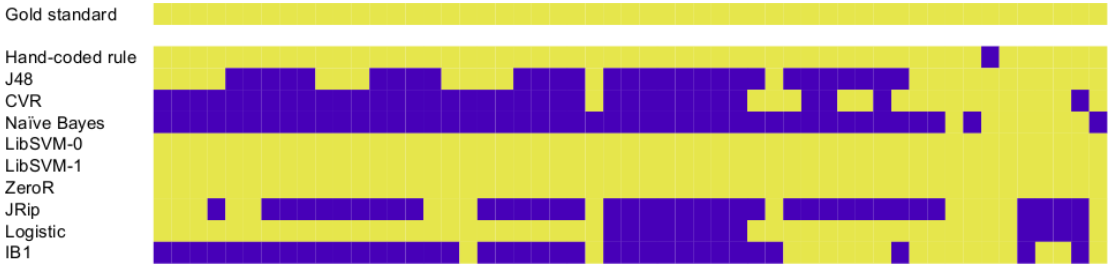
[30] Z. Wang and O. Lemon. A nonparametric Bayesian approach to learning multimodal interaction management. In *Proceedings of SLT*, 2012.

[31] M. White. Efficient realization of coordinate structures in Combinatory Categorial Grammar. *Research on Language and Computation*, 4(1):39–75, 2006.

[32] P. Wittenburg, H. Brugman, A. Russel, A. Klassmann, and H. Sloetjes. ELAN: a professional framework for multimodality research. In *Proceedings of LREC*, 2006.

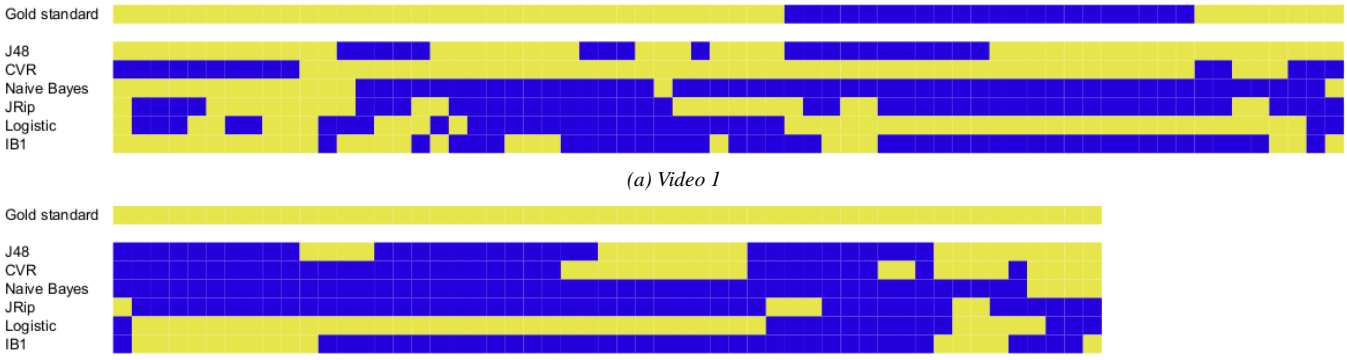


(a) Video 1



(b) Video 2

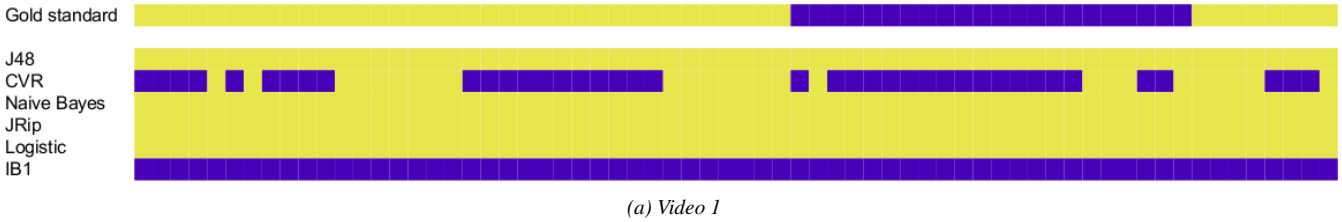
Figure 4: Gold-standard annotations and classifier predictions for two sample videos



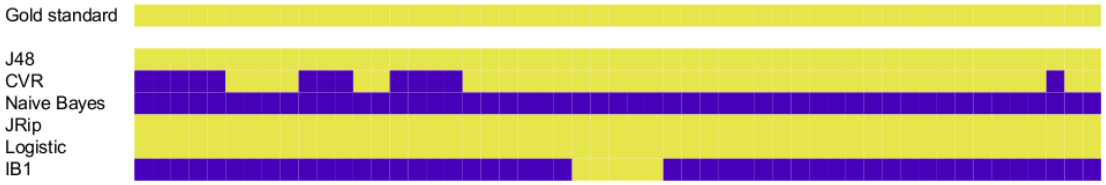
(a) Video 1

(b) Video 2

Figure 5: Gold-standard annotations and classifier predictions for the sample videos, incorporating previous sensor data



(a) Video 1



(b) Video 2

Figure 6: Gold-standard annotations and classifier predictions for the sample videos, incorporating previous classifications