Assessing User Expertise in Spoken Dialog System Interactions

Eugénio Ribeiro[1,2], Fernando Batista[1,3], Isabel Trancoso[1,2], José Lopes[4], Ricardo Ribeiro[1,3], and David Martins de Matos[1,2]

[1] L^{2}F – Spoken Language Systems Laboratory – INESC-ID Lisboa
[2] Instituto Superior Técnico, Universidade de Lisboa, Portugal
[3] ISCTE-IUL – Instituto Universitário de Lisboa, Portugal
[4] KTH Speech, Music, and Hearing, Stockholm, Sweden
eugenio.ribeiro@l2f.inesc-id.pt

Abstract. Identifying the level of expertise of its users is important for a system since it can lead to a better interaction through adaptation techniques. Furthermore, this information can be used in offline processes of root cause analysis. However, not much effort has been put into automatically identifying the level of expertise of an user, especially in dialog-based interactions. In this paper we present an approach based on a specific set of task related features. Based on the distribution of the features among the two classes – Novice and Expert – we used Random Forests as a classification approach. Furthermore, we used a Support Vector Machine classifier, in order to perform a result comparison. By applying these approaches on data from a real system, Let's Go, we obtained preliminary results that we consider positive, given the difficulty of the task and the lack of competing approaches for comparison.

Keywords: user expertise · Let's Go · SVM · Random Forest

1 Introduction

The users of a dialog system have different levels of expertise, that is, knowledge of the system's capabilities and experience using it. Thus, identifying the level of expertise of a user is important for a dialog system, since it provides cues for adaptation which can improve dialog flow and the overall user satisfaction. For instance, by identifying a novice user, the system may provide help on the first signs of struggle and adapt its prompts to provide further information. Also, user expertise information can be used to adapt the system's parameters, such as Automatic Speech Recognition (ASR) timeout values, reducing the number of misinterpretations and interruptions. Furthermore, it can be used in offline processes to identify problems caused by lack of expertise, which is important for the development of better dialog systems.

In this article we present an analysis of different features and how they can be used to identify the level of expertise of a user on Let's Go [15] data. The remaining sections are structured as follows: Section 2 presents related work on

user expertise with dialog systems. Section 3 lists relevant feature classes for this task. Section 4 describes the datasets, the specific features extracted, and the classification approaches. Results are presented and discussed in Section 5, and, finally, Section 6 states the achieved conclusions and suggests paths for future work.

2 Related Work

A system that behaves the same way for all users, independently of their expertise, may not provide a truly usable interface for any of them. By knowing the level of expertise of its users, a system could improve the quality of the interaction through adaptation techniques based on that level [13]. However, not much effort has been put into identifying the level of expertise of a user, especially in dialog-based interactions.

Hjalmarsson [9] analyzed dialog dynamics and discussed the utility of creating adaptive spoken dialog systems and individual user models. She suggests that such models can be created using both rule-based and statistical approaches [8]. Given the correct set of rules, rule-based models have good performance on specific tasks. However, they must be handcrafted from the intuition of the designer or experts, which is a time-consuming process. Thus, when annotated data is available, statistical models are a better option. The author suggests Bayesian Networks, Reinforcement Learning, and Decision Trees as promising approaches for the task.

Hassel and Hagen [7] developed an automotive dialog system that adapts to its users' expertise. However, it does not attempt to identify each user's expertise, but rather assumes that every user is a novice and then adapts over time. The adaptation is task-based and controlled by the number of successful attempts and the time since the last execution of that task.

Jokinen and Kanto [10] used user expertise modelling to enable the adaptation of a speech-based e-mail system. They distinguish three levels of expertise – Novice, Competent, and Expert –, a subset of the five proposed by Dreyfus and Dreyfus [4] in their studies about the behaviour of expert systems. The developed models have two components – online and offline. The first tracks the current user session and provides cues for system adaptation, accordingly. The latter is based on statistical event distributions created from all sessions of a user and serves as a starting point for the next session. In terms of features, a small set is used, consisting on the number of timeouts, interruptions, and help requests, as well as the number of times a given task was performed, or a specific system dialog act was invoked.

3 Relevant Features

Since expertise depends on the task being performed, it cannot be identified by large sets of generic acoustic features such as the ones extracted by openSMILE [5]. Thus, a small set of task oriented features must be devised. These

features can be clustered into different categories, according to their origin and what aspects they intend to capture. In the following sections we describe each of these categories.

3.1 Interruptions

Expert users may interrupt the system when they are aware of the complete dialog flow. However, this is not a good approach when system utterances are confirmation prompts which only include the information to be confirmed in the end. In this case, interruptions usually signal a novice user. Furthermore, cases when the system interrupts the user can also reveal a novice user who uses long sentences or pauses that exceed the system's waiting times.

3.2 Delays

Negative delays between the system and user utterances mean that an interruption occurred, which has the previously described implications. On the other hand, long delays suggest that the user is still processing the system's utterance and is unsure about what to say and, thus, may reveal inexperience.

3.3 Durations

Long durations are typically more discriminative than short ones and may suggest inexperience. For instance, a long call usually means that something went wrong during the dialog. Long utterances also suggest inexperience, as they are more prone to recognition errors and interruptions by the system.

3.4 Speech Rate

Speech rate is also a possible indicator of the level of expertise of a user since both high and low speech rates may lead to communication problems. While high speech rates lead to higher error rates in recognition, low speech rates are related to paused speeches, which are more prone to be interrupted by the system. Thus, expert users usually keep a balanced speech rate.

3.5 Help Requests

When a user is new to a system and unsure how it works, he or she typically asks for help, revealing inexperience. This is especially clear in cases when the system offers help and the user immediately accepts it. Unfortunately, some systems do not provide help functionality or the user is not aware it exists.

4 Experimental Setup

This section describes our experimental setup, starting with the used datasets. Next, the used features and their distribution in the training dataset are thoroughly presented. After that, the used classification and evaluation approaches are described.

4.1 Datasets

We explored user expertise on data extracted from interactions with the Let's Go Bus Information System [15], which provides information about bus schedules, through spoken telephonic interaction with a dialog system. This system has been running for many years and has experienced changes over time. Thus, the characteristics of the data differ according to when it was obtained.

In our experiments we used the LEGO [16] corpus. This corpus is a subset of 347 Let's Go calls during the year of 2006. The corpus contains annotations relevant for the user level of expertise identification task, such as barge-in information and durations. In terms of the level of expertise, the original corpus is not annotated. Thus, we annotated each call with that information using two labels – Expert and Novice. Out of the 347 calls, 80 users were labeled as Expert and 235 as Novice. The remaining calls were impossible to annotate since the user did not interact with the system. We used this dataset for analyzing the distribution of different features and as training data for the classification task.

In addition to the 2006 data, we also looked into a small set of 80 calls from 2014 data of the Let's Go corpus. This set was annotated for expertise at KTH using the same two labels – Expert and Novice [11]. The audio files for all but one of calls are accompanied by the logs of the system, which provide important information for feature extraction. Of the 79 calls, 42 users were labeled as Expert and 37 as Novice. The reported Cohen's Kappa for the annotator agreement was 0.73, which is usually considered good. However, we also annotated the 79 calls, labeling 31 users as Expert and 48 as Novice, obtaining an agreement of 0.43 with the original annotation, which is usually considered moderate. We used the 56 calls with agreement to assess the generalization capabilities of our classifiers.

4.2 Features

In Section 3 we defined a set of feature classes that are relevant for identifying the level of expertise of a user. In this section we describe the specific features that we were able to extract from the datasets. Furthermore, for the training dataset, LEGO, we perform an analysis of the distributions of the features among the data, in order to perform a comparison with the previously defined expectations. Table 1 presents these distributions.

Interruptions The LEGO corpus is annotated with user barge-in information. Thus, we were able to extract the number of user barge-ins per dialog. Table 1 shows that novice users are more prone to interrupt the system, with an average of 5 barge-ins per dialog. This was expected, since most of the system utterances in the corpus are of the kind that state information to be confirmed in the final words and, thus, should not be interrupted. However, these statistics did not take the length of the dialog into account. Thus, we calculated the user barge-in rate as the percentage of exchanges containing user barge-ins. The previous results were confirmed, as, on average, novice users barged-in on 16% of the exchanges, while experts only barged-in on 10% of the exchanges. The median

Table 1. Feature distribution among the LEGO dataset in terms of average (μ), median($\tilde{x}$), and standard deviation(σ).

	Novice			Expert		
Feature	μ	$\tilde{x}$	σ	μ	$\tilde{x}$	σ
Interruptions						
# Barge-ins	5.06	3.00	6.79	2.75	2.00	3.15
Barge-in Rate	16.2	15.4	9.9	10.3	9.5	6.9
Delays (s)						
1st Turn	1.52	1.28	3.00	1.32	1.21	2.81
1st Turn (Positive)	2.82	2.18	2.79	1.90	1.49	2.72
Durations (s)						
Utterance	1.81	1.44	3.14	1.19	1.20	0.43
Call	123	104	95	102	76	78
1st Turn	1.81	1.19	2.02	1.72	1.39	1.66
# Exchanges	28.0	23.0	23.4	23.8	20.0	13.8
Speech Rate (phones/s)						
Global	13.7	14.2	3.3	14.8	14.9	1.9
1st Turn	14.3	14.5	4.1	14.8	14.5	2.8
Help Requests						
# Requests	0.27	0.00	0.55	0.00	0.00	0.00

values of 15% and 10% for novice and expert users, respectively, also support the hypothesis. Furthermore, only novice users have barge-in rates over 30%. Information extracted from the first turn is not as discriminative, as around 60% of the users barged-in on the first turn, independently of the class. However, the first system utterance is a fixed introduction, which encourages expert users to barge-in, in order to skip to important parts of the dialog.

On Let's Go 2014, the barge-in information was extracted from the interaction logs.

Delays LEGO annotations include the duration of each user utterance, as well as the time when each exchange starts. However, each exchange starts with the system utterance, for which the duration is not annotated. Thus, we were not able to obtain delay information for most exchanges. The only exception was the first exchange, since the system utterance is fixed and, thus, so is its duration – 10.25 seconds. In this case, we calculated the delay as the difference between the time when the user utterance starts and the time when the system utterance ends. As expected, the results presented in Table 1 suggest that novice users take longer to answer than expert users. Furthermore, when only positive delay values are taken into account, the discrepancy between the two classes is even more evident.

On the 2014 data, we used a similar approach to obtain the first turn delay. In this case, the duration of the first system utterance is 13.29 seconds. The remaining information required to calculate the delay was obtained from the interaction logs.

Durations The LEGO corpus is annotated in terms of duration of user utterances, as well of the whole call. However, a few of the utterances are wrongly annotated. Nonetheless, we were able to compute the average user utterance duration per dialog. As expected, novice users tend to use longer utterances and are much less consistent than expert users. There are no expert users with average utterance durations over 3 seconds. In terms of the whole call, the same conclusions can be drawn, both in terms of time duration and number of exchanges, as novice users have higher values for all the measures. While most calls by expert users last less than 2 minutes, calls by novice users have a wider distribution. As for the duration of the first utterance, on average, novice users still use longer utterances. However, that is not true in terms of median value. Nonetheless, standard deviation for novice users is higher than the average value, which suggests that novice users adopt unpredictable behaviors.

We obtained duration information from 2014 data directly from the audio files, using SoX [1].

Speech Rate We extracted the speech rate in phones per second from each user utterance of the LEGO corpus and used those values to calculate the average speech rate for each dialog. The phones for each utterance were obtained using the neural networks included in the AUDIMUS [12] ASR system. Table 1 shows similar average and median values for both classes, around 15 phones per second. However, expert users are more steady, which leaves the tails of the distribution for novice users only. Looking only at the first user utterance, average and median values are even closer for both classes. Nonetheless, the tails of the distribution are still reserved for novice users only, although the expert users are slightly less steady. The same extraction procedure was applied on 2014 data.

Help Requests From the existing information, we were able to extract the number of help requests detected by the system during each LEGO dialog. As expected, only novice users asked for help, with an average of 0.27 help requests per dialog. 23% of novice users asked for help at least once and up to 3 times. Furthermore, 17% of the novice users asked for help on the first turn.

On the 2014 data, we obtained the number of help requests from the dialog transcriptions, by looking for the help keyword or the zero key on user utterances.

4.3 Classification

Distinguishing between novice and expert users is a binary classification task. From the multiple classification approaches that could be used, we opted Support Vector Machines (SVMs)[3], since it is a widely used approach and typically produces acceptable results, and Random Forest (RF) [2], an approach based on decision trees, which are indicated for this task, given the distribution of our features among the two classes.

To train our SVMs, we took advantage of the Sequential Minimal Optimization (SMO) algorithm [14] implementation provided by the Weka Toolkit [6]. We used the linear kernel and kept the C parameter with its default 1.0 value.

We opted for an RF approach due to its improved performance when compared to a classic decision tree algorithm. We also used the implementation provided by the Weka Toolkit to train our RFs. We used 1000 as the number of generated trees, since it provided a good trade-off between training time and classification accuracy.

4.4 Evaluation

Since there is no standard partition of the LEGO corpus into training and testing sets, we obtained results using 10-fold cross-validation. Furthermore, we used the data from 2014 to assess the generalization capabilities of our classifiers.

In terms of measures, we use Accuracy and the Kappa Statistic since they are the most indicated measures to evaluate performance and relevance on this task. Accuracy is given by the ratio between the number of correct predictions and the total number of predictions. The Kappa Statistic gives the weighted agreement between the predictions of the classifier and the gold standard, in relation to those of a chance classifier.

5 Results

Since the LEGO dataset is highly unbalanced, we balanced it using the Spread Subsample filter provided by the Weka Toolkit. Still, we performed experiments on both the balanced and unbalanced data. Table 2 presents the results obtained using each set of features independently, as well as different combinations. The **First Turn** set combines the features extracted from the first turn only, while the **Global** set combines the features extracted from the whole dialog. The **All** set combines the two previous sets. The **Selected** set is obtained by applying the Best First feature selection algorithm, provided by the Weka Toolkit, to the **All** set.

Table 2. Results on the unbalanced (Chance = 0.741) and balanced (Chance = 0.500) versions of the LEGO dataset

	Unbalanced		**Balanced**			
	Random Forest		**SVM**		**Random Forest**	
Feature Set	Accuracy	κ	Accuracy	κ	Accuracy	κ
Interruptions	0.702	0.140	0.600	0.200	0.613	0.225
Delays	0.693	0.168	0.494	-0.013	0.519	0.038
Durations	0.790	**0.403**	0.594	0.188	0.744	0.488
Speech Rate	0.686	0.037	0.513	0.025	0.525	0.050
Help Requests	0.741	0.000	0.631	0.263	0.600	0.200
First Turn	0.767	0.321	0.594	0.188	0.713	0.425
Global	0.783	0.377	0.681	0.363	0.769	0.538
All	0.793	0.385	**0.706**	**0.413**	**0.794**	**0.588**
Selected	**0.796**	**0.403**	**0.706**	**0.413**	0.781	0.563

The SVMs classification approach performed poorly on the unbalanced dataset, never surpassing a chance classifier. However, the RF approach achieved 80% accuracy using the **Selected** feature set, which represents an improvement of 6 percentage points. Given the difficulty and subjectivity of the task, the Kappa coefficient of 0.40 should not be disregarded.

On the balanced dataset, both the SVM and RF approaches were able to surpass the chance classifier. Still, similarly to to what happened on the unbalanced dataset, the RF approach performed better. Using all the available features, it achieved 79% accuracy, which represents an improvement of 8 percentage points over the SVM counterpart and 29 percentage points over the chance classifier. The Kappa coefficient of 0.59 is 50% higher than the one obtained for the unbalanced dataset, in spite of facing the same concerns. In this version of the dataset, feature selection did not improve the results.

The **First Turn** feature set is the most relevant for expertise level identification in real time. Using this set, an accuracy of 77% was achieved on the unbalanced dataset, which represents an improvement of 3 percentage points over the chance classifier. On the balanced dataset, the RF approach was able to improve the results of a chance classifier by 21 percentage points and achieve a Kappa coefficient of 0.42. However, the SVM classifier performed poorly. Overall, this means that it is not easy to identify the level of expertise of a user based solely on the first turn of the dialog. Still, a preliminary classification can be obtained to start guiding the system towards user adaptation, and improved as the dialog flows.

In terms of the individual feature sets, duration related features are the most important for the RF approach on both versions of the dataset. On the balanced dataset, interruption and help related features also provide important information. For the SVM approach, the important features remain the same but the order of importance is inverted.

Table 3 presents the results obtained on Let's Go 2014 data by the classifiers trained on the balanced LEGO corpus. We do not show the rows related to feature categories that did not provide relevant results. We can see that, in this case, the SVM approach surpassed the RF one, achieving 66% accuracy and a Kappa coefficient of 0.33, using the **Selected** feature set. This represents an improvement of 11 percentage points over the chance classifier. As for the RF approach, although its accuracy using the **Selected** feature set is just two percentage points below the SVM approach, its Kappa coefficient of 0.22 is much lower and is surpassed, although only slightly, by the 0.23 obtained by using only help related features. Overall, this means that the RF classifiers, which performed better on the LEGO corpus, have less generalization capabilities than the SVM ones. This explains the negative results obtained by the RF classifier using the **Global** feature set, as the differences between both datasets are more noticeable when looking at the dialogs as a whole than when just looking at first turns.

Table 3. Results on Let's Go 2014 data (Chance = 0.554)

	SVM		**Random Forest**	
Feature Set	Accuracy	κ	Accuracy	κ
Help Requests	0.607	0.268	0.589	**0.232**
First Turn	0.571	0.207	0.538	0.082
Global	0.589	0.153	0.538	-0.018
All	0.643	0.283	0.589	0.103
Selected	**0.661**	**0.327**	**0.643**	0.217

6 Conclusions

In this article we presented an approach for automatically distinguishing novice and expert users based on a specific set of task related features. Given the distributions of the features, a classification approach based on decision trees was indicated. This was confirmed when the RF approach outperformed the widely used SVMs on both versions of the LEGO corpus.

Since this is a relatively unexplored task and the dataset was not previously annotated for expertise, we cannot compare our results with other work. Nonetheless, we believe that the obtained results are positive, since our approach focused on identifying the level of expertise from a single session, without previous information about the user, which is a difficult task.

Furthermore, we were also able to obtain relevant results using features extracted only from the first turn of each dialog. This is important for a fast adaptation of the system to the user's level of expertise, as it provides a preliminary classification of that level, which can be improved as the dialog flows, through the accumulation of the results of all turns.

On the downside, the results obtained on the data from Let's Go 2014 were not as satisfactory, with the RF classifiers revealing less generalization capabilities than the SVM ones.

In terms of future work, we believe that it would be important to obtain more annotated data, in order to train more reliable classifiers, with improved generalization capabilities.

Acknowledgements

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013, by Universidade de Lisboa, and by the EC H2020 project RAGE under grant agreement No 644187.

References

1. SoX - Sound eXchange. `http://sox.sourceforge.net/`, version 14.4.1
2. Breiman, L.: Random Forests. Machine Learning 45(1), 5–32 (2001)

3. Cortes, C., Vapnik, V.: Support-Vector Networks. In: Machine Learning. pp. 273–297 (1995)
4. Dreyfus, H.L., Dreyfus, S.E.: Mind over Machine: The Power of Human Intuition and Expertise in the Era of the Computer. The Free Press, New York, NY, USA (1986)
5. Eyben, F., Weninger, F., Gross, F., Schuller, B.: Recent Developments in openSMILE, the Munich Open-source Multimedia Feature Extractor. In: Proceedings of the 21st ACM International Conference on Multimedia. pp. 835–838 (2013)
6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Exploration Newsletter 11(1), 10–18 (2009)
7. Hassel, L., Hagen, E.: Adaptation of an Automotive Dialogue System to Users' Expertise and Evaluation of the System. Language Resources and Evaluation 40(1), 67–85 (2006)
8. Hjalmarsson, A.: Adaptive Spoken Dialogue Systems (2005), available at `http://www.speech.kth.se/~rolf/NGSLT/gslt_papers_2004/annah_termpaper_05.pdf` on 18/12/2015
9. Hjalmarsson, A.: Towards User Modelling in Conversational Dialogue Systems: A Qualitative Study of the Dynamics of Dialogue Parameters. In: Proceedings of INTERSPEECH 2005. pp. 869–872 (2005)
10. Jokinen, K., Kanto, K.: User Expertise Modeling and Adaptivity in a Speech-Based E-Mail System. In: Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics. pp. 87–94 (2004)
11. Lopes, J., Chorianopoulou, A., Palogiannidi, E., Moniz, H., Abad, A., Louka, K., Iosif, E., Potamianos, A.: The SpeDial Datasets: Datasets for Spoken Dialogue System Analytics. In: Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC) (2016)
12. Meinedo, H., Viveiros, M., Neto, J.a.: Evaluation of a Live Broadcast News Subtitling System for Portuguese. In: Proceedings of INTERSPEECH 2008. pp. 508–511 (2008)
13. Nielsen, J.: Usability Engineering. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)
14. Platt, J.: Fast Training of Support Vector Machines using Sequential Minimal Optimization. In: Advances in Kernel Methods - Support Vector Learning. MIT Press (1998)
15. Raux, A., Bohus, D., Langner, B., Black, A.W., Eskenazi, M.: Doing Research on a Deployed Spoken Dialogue System: One Year of Lets Go! Experience. In: Proceedings of INTERSPEECH 2006. pp. 65–68 (2006)
16. Schmitt, A., Ultes, S., Minker, W.: A Parameterized and Annotated Spoken Dialog Corpus of the CMU Let's Go Bus Information System. In: Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC) (2012)