Automatically Detecting Task Switches in Real-time

Using Psycho-Physiological sensors, Regime-Switching Hidden Markov Models and Conditional Random Fields

ABSTRACT

Task switch detection methods often use software-based metrics to detect a developer's current context. This paper deviates from these common techniques by using psycho-physiological sensors to detect patterns in developers' physiological processes. Specifically, these patterns are detected using Regime-switching Hidden Markov Models. The variety of different patterns from the biometric sensors are then combined using a Conditional Random Field to infer whether a task switch is occuring at any given moment in time. Results show that this approach is a viable alternative both for measuring accuracy and efficiency of detecting the beginning of task switches in real-time.

Keywords

Task Switches, Real-time Detection, Hidden Markov Model, Conditional Random Fields, Psycho-Physiological Sensors

1. INTRODUCTION

If there's one thing that annoys programmers, it is being interrupted while being neck-deep in code[1]. A possible explanation for this could be the fact that humans have rather cost-ineffective context switching behaviour. Psychological and Cognitive Science research attribute this behaviour to humans in general. As an example, Stephen Monsell investigated whether people could switch between relatively easy tasks in everyday life. He concluded that after task switches, responses are substantially slower and usually more error-prone [18]. In *The Cost of a Voluntary Task Switch*, Catherine Arrington and Gordon Logan found a similar result: there is a significant cost associated with switching tasks. Especially when comparing the continuation of the same task and switching to another task, they found that task alternations were slower than task repetitions [1]. Following these results, we should be able to interpolate these inferences onto software engineering as well. If the difficulties of task switching are a connate deficiency, then one can only expect *computer programming*, a complex high order task [25], to make these difficulties more apparent. Indeed, when analysing task switch difficulties on developers, we see the same results. Bailey and Konstan investigated the impact of interruptions on developers *during* and *between* primary tasks. They found evidence that interruptions during primary tasks lead to a longer completion time, an increased likelihood to commit errors and increased annoyance and anxiety [2]. Their final conclusion is that the mitigation of these negative effects can be accomplished by postponing the presentation of peripheral information until coarse boundaries (task switches) are reached.

Clearly this gives rise to a new problem: how do we determine the occurence of these boundaries (or task switches)? According to Czerwinski et al., a large portion of task switching is caused by external interruptions [8]. This would imply that accounting for such interruptions is difficult. As a reply, researchers propose tools such as a *Physiologically Attentive User Interface* which analyses heart rate variability and motor activity using electroencephalogram analysis [5]. Although the fusion between psycho-physiological analysis and software engineering research is still rather experimental (judging by the lack of coherent literature), it is a very promising field. Daniel Chen, in his Ph.D. Thesis, confirms that when deploying physiological sensors, it becomes possible to create interfaces which are physiologically responsive to the user's attentive state [6]. After evaluating existing literature and experiments, he states that, until now, there has only been made little use of real-time physiological measures in modeling user interruptability.

The goal of this paper is to utilise such psycho-physiological sensors and employ them in order to detect task switches in real-time. Specifically, by using a photoplethysmograph, an electroencephalograph and a bioelectromagnetic sensor, the data for task switch analysis is obtained. Using common real-time analysis techniques from natural language processing and financial econometrics, this paper tries to see whether there exists any evidence that psycho-physiological sensors are able to detect task switches. The statistical analysis was performed by analysing Bayesian probabilities. This because sample-based testing, and the corresponding assumptions, are not appropriate for the applied measures.

This paper concludes that the econometric approach to detecting task switches through biometric data has relatively good performance and poses a good alternative to existing methods. Specifically, we observe good accuracy for observing the beginning of a task switch and find many possibilities for improving the developed approach in future research.

[1] As an example, consider a question posted in the programmers' community: `http://programmers.stackexchange.com/questions/46252/how-to-explain-a-layperson-why-a-developer-should-not-be-interrupted-while-neck`

2. METHODOLOGY

The data that is used for the analysis was gathered in an experiment on 10 participants. The experiment focussed on software developers in their natural working environment (a field study). The participants had to work on a primary task and were randomly interrupted with short arithmetic tasks. These peripheral tasks were displayed on another screen, a tablet, next to the main computer screen of the participants. The participants could switch to the arithmetic task whenever they wanted. Shortly after the arithmetic (peripheral) tasks, the users had to fill in a survey regarding their subjective rating of interruptibility at the moment of interruption. The survey also featured short other questions regarding the level of disturbance and the mental load of the primary task that the participants experienced during the interruption. During the experiment, two different sensors where used to track a variety of psycho-physiological variables. The first set of sensors (the NeuroSky Mindband) were used to construct a 1-channel electroencephalograph (EEG) and two variables called attention and meditation [21]. The second set of sensors came in the form of the Empatica E3 wristband and construct both a photoplethysmograph (PPG) and measure Electro Dermal Activity (EDA) as well as other variables such as temperature and accelaration movements. The latter one was not used in this study as the interruptions in the experiment enforced movements of the participants, hence biasing (and self-fulfilling) the use of the accelerometer. Before the experiment started, participants were asked to look at a fishtank video, for approximately two minutes, to try and relax before the experiment began. This was repeated after the experiment as well. The data from the sensors that was gathered during the fishtank video was later used as a benchmark for the sensor data during the experiment.

2.1 Data processing

After the experiment, the raw EEG wave was filtered, smoothed and statistical artifacts were removed. After this, all raw variables were used to calculate a multitude of new variables, we call 'features'. These ranged from calculating the number of peaks per a specified timeperiod to dividing the raw EEG wave into 0.25Hz frequency bins. We collected these frequency bins into the commonly-recognized frequencies *alpha* to *theta* [21]. Apart from this, we also took the Cartesian product of all these frequencies to see whether cross-dependent data improves accuracy in our statistical analysis. As not all variables turned out to contribute to the final statistical model, the final model does not use all features. Using a top-down approach, the least-contributing variables were pruned until all likelihood additions where approximately similar. This means that we pruned the remaining variables based on their total addition to the log likelihood function.

3. STATISTICAL ANALYSIS

3.1 Motivation

The biggest problem with biometric data from psycho- physiological sensors is that it does not contain clear signals. The sensor data is often an extremely noisy process and it's hard to distinguish, let alone quantify, different patterns in the emission spectra of the psycho-physiological sensors. This is because much of the data is under direct influence of unpredictable and unaccountable effects. As an example, the rise in a person's body temperature can be due to an increase in anxiety. However, it could also be due to consuming a hot cup of tea, a sudden change of temperature in the person's environment or even because the person started listening to music [4]. Still, research indicates that body temperature is a very important variable concerning short-term memory, alertness and performance [14]. Exactly for this reason, we investigate whether fluctuations in body temperature have any predictive or confirming power concerning the occurence of task switches. If we are to make any inferences concerning these fluctuations, we need a statistical method that is able to detect such fluctuations, ignoring much of the other noise.

Like biometric data, stock markets feature likewise problems. Stock returns contain not only stochastic trends and fat tailed return distributions but also feature estimation problems such as heteroskedasticity [15]. Like physiological processes, the financial markets feature a highly complex, higher-order process [16]. According to Parra et al., the higher-order statistical properties arising from the non-stationarity of such processes are similar among a variety of natural phenomena [22]. They investigated natural image features, speech sound intensities, stock market variation and MEG alpha activity and found that the same characteristic function (a conditional distribution) can be used to describe all of these processes. By varying the scale of the non-stationary Gaussian process, they infer that it becomes possible to relate a variety of natural signals to one another.

In the spirit of Parra et al., this paper, too, relates socio-economic data to biometric data. In financial econometrics, the distinction between *bull* and *bear* markets[2] or the distinction between different periods of volatility, is often modelled by applying regime-switching hidden markov models (HMM) [16, 17, 9, 11, 12]. Since its proposition by Hamilton (1989), who initially built the algorithm to estimate macro-economic timeseries, the regime-switching HMM has found its application in other sciences as well. An example of this is the prediction of electricity spot prices, as performed by Janczura and Weron (2011). This paper tries to find out if HMMs can provide likewise insights in the real-time detection of regime-switches in biometric data, obtained from psycho-physiological sensors.

The HMM model that we used to estimate the different states is a simple mean model with a switching error term:

$$y_t = \alpha_t + \epsilon_t \tag{1}$$

$$\epsilon_t \sim N(0, \sigma^2_{S_t}), \quad S_t \in \{1, 2\} \tag{2}$$

Eventhough this model is an extreme simplification of reality, all extraordinary deviations will still be absorbed by the error term, which in turn correctly influences the probabilities of the different states. This means that the HMMs try to distinguish between two different error structures (in the Appendix, Figure 6 there is a visualisation for the regime-switching behaviour of this model for the variable 'attention mean difference'). The covariance matrix in the HMMs is calculated using the Hessian matrix (second partial derivatives of log likelihood function). For the calculation of the

[2]This is financial jargon: a 'bull' market refers to a period of increasing prices. Likewise, a 'bear' market refers to a period of decreasing prices.

HMMs, the MATLAB software package 'MS Regress' was used [23].

Apart from extracting different regimes from the biometric data, we also needed a method to estimate the importance of the variables (now transformed into regimes), as well as a method to combine the variables. A simplistic approach to this problem would be the Naives Bayes Classifier (NBC), which assumes independence among features [20]. However, we think that the assumption of independence is not realistic for combining physiological processes. A simple argument for this would be the evidence found for the influence of body temperature on EDA, especially due to circadian factors [3, 30]. To account for these interdependencies, we instead resort to *Conditional Random Fields* (CRF) [28]. This conditional model, often used in natural language processing and with applications in bioinformatics and computer vision, assumes dependencies among the regressors. Therefore, it allows for "*.. the use of rich, global features of the input* [variables]" [28]. Feeding the regimes of the HMMs from the biometric variables to a global CRF, therefore allows us to model multicollinearity as well as time-neighbouring interdependencies among the variables. If any combination of the HMM regimes have a strengthened influence on task-switch detection, the CRF, opposed to the NBC, can detect and exploit this effect. Next to this, the CRF uses a first order discrete-time Markov chain to estimate the different states of the dependent variable (task switches in our case). This gives the added advantage of state-switching pattern recognition: if any regressor changes state right before a task switch happens, the CRF can use such information. For example to recognise post-event patterns that increase the probability that a task switch will happen in the near future. For the calculation of the CRFs, the MATLAB software package 'crfChain' was used [26].

Before estimating *when* exactly a task switch happens, we need to define the duration of a task switch. As our analysis focusses on task switches which are caused by external interruptions only, we defined an interruption as being an "*externally generated, randomly occurring, discrete event that breaks continuity of cognitive focus*" [7]. In our analysis, a task switch therefore happens when a participant shifts attention between one task (primary) and another task (interruption), which has not been his/her main occupation for the recent past [29]. In terms of our raw data, this is the point where the participant clicks 'start' for the short arithmetic task. Unfortunately, in terms of time series prediction, a single isolated point does not encompass enough information for forecasting. We therefore need to extend the task switch to a larger period, such that the CRF can distinguish and learn the characteristics that the participants exhibit when switching tasks. For this, we have defined the period before the task switch as λ and the period after as ρ (see Figure 1). Throughout our statistical analysis, we stick with $\lambda = 10s$, $\rho = 20s$ as we found this to be the most intuitive setup: disturbing a developer more than 10 seconds in advance is not favourable. Likewise, we imagined disturbing a developer more than 20 seconds after the task switch is also suboptimal. Still we found that other combinations of the parameters yield better results. Yet, to avoid over-fitting, we continued with the 10/20 setting for the statistical inference.

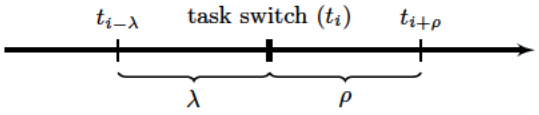

Figure 1: determining task switch duration

4. RESULTS

When using the CRF model, a given set of parameters ($\lambda = a, \rho = b$ for a given a, b) is fixed. The beginning of the task switch is therefore defined as $t_{i-\lambda}$ (see Figure 1). We are especially interested in the efficiency of the model in determining when exactly this moment $t_{i-\lambda}$ takes place. To gain more insight in this problem, we have estimated the efficiency of the model when the learning sample was seperated between participants and when all samples (all participants) where aggregated. The results are displayed in Figure 2. The mean of the aggregated data is $-19.38s$ (σ=$53.76s$).

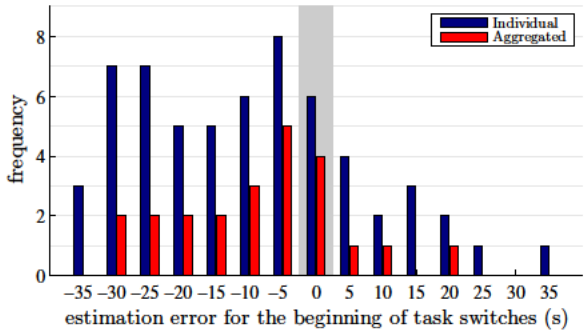

Figure 2: Estimation results for $\lambda = 10s$, $\rho = 20s$

The mean of the individualised data is $-13.85s$ (σ=$23.31s$). From this we can infer that aggregating the data, and ignoring interpersonal differences, does not improve efficiency of the model. Both results seem to be biased towards signaling *earlier* than the actual task switch happens. As discussed later, this can be due to the fact that the experiment utilised a notification to signal the participant that a task switch is necessary. Following this notification, the participants could decide themselves when to switch. This *notification lag* could cause disruptions (hence signals) in the participant's psycho-physiological data, helping the CRF to recognise potential task switches. Another thing to notice is that the *trigger rate* (the rate of signals relative to the total amount of task switches) is 55% (a little over half of the time) for the individual learning and 18% for the aggregated data. This implies that the aggregated data is not only less efficient, it is also more hesitant to trigger when a task switch occurs. This, again, displays a negative influence caused by interpersonal differences.

To examine how well the CRF model estimates the duration of task switches, we loop over every time unit and count the binary estimates for each unit. The confusion matrix of this enumaration is visualised in Figure 3. For the confusion matrix, the $\lambda = 10s, \rho = 20$ setting was used. For each task

		True value		
		Positive	Negative	Total
Model	Positive	**164**	466	630
	Negative	97	**837**	934
	Total	261	1303	1564

Figure 3: Confusion matrix for individual learning

switch, the counting started a minute *before* the task switch and ended a minute *after*. The confusion matrix shows that the model is not good in predicting the length of a task switch. One explanation for this would be that the psycho-physiological data does not match the chosen task switch length $\lambda + \rho$ very well. As estimating the parameters $\lambda + \rho$ is not the main objective of this paper, and because there is no reason for trying to estimate an imposed design decision, we will not explore this matter further.

Apart from estimating the efficiency of the 10/20 setting, it is also relevant to explore the model's behaviour when different parameters are used. To investigate this question, we have estimated the CRF for 20 different parameters settings, each estimating 138 different task switches[3]. The results of this analysis are displayed in Figure 4. In Figure 4, every

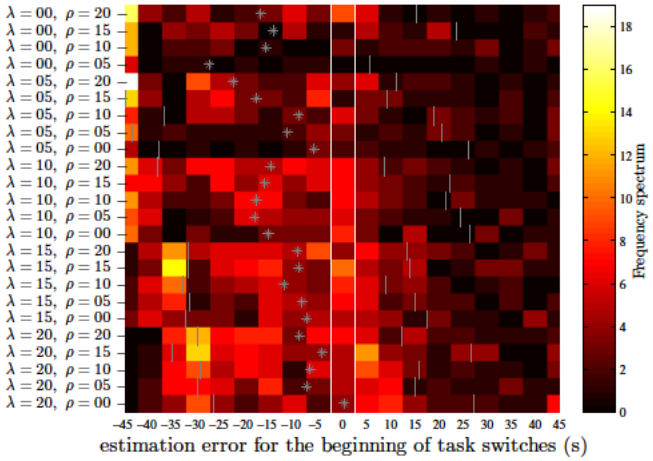

Figure 4: Individualised results for various λ and ρ

row represents the distribution of the estimation error for a different set of the input parameters λ and ρ (see left y-axis). The frequency of observations within each distribution (row) is indicated by the colorbar on the right. A 'warm' (white) color represents the occurence of many estimation errors for a particular time-lag from the actual beginning of the task switch (see x-axis). Likewise, a dark(er) color represents a low frequency of errors for a given time-lag, given the distribution of a particular row. Stars ($*$) denote the mean of each parameter combination. Vertical bars ($|$) denote the (left or right) position of the standard deviation of the estimated mean. From this analysis, we found that the 10/20 setup is not the most efficient setup. Rather, the input parameters $\lambda = 20$ and $\rho = 0$ give the best performance with a mean of $0.35s$ and a standard deviation of $27.02s$.

[3]This is the total amount of task switches that the CRF could recognise from the field data.

This is a big difference with the mean for the 10/20 setting ($\mu = -13.85s, \sigma = 23.31s$). Like Figure 2, Figure 4 also displays the bias the CRF model has towards predicting earlier than the actual task switch. Again this is due to the notification lag, meaning the participant has been signaled before switching tasks. The mean of the attention lag for all the task switches is $45.17s$ ($\sigma = 77.50$) which explains the advantage of increasing λ. In comparison, 74 of the total 138 interruption lags are smaller than 20 seconds (see Appendix, Figure 7). This motivates the increased efficiency of using $\lambda = 20s$. Apart from this, we also observed that individual data was more efficient that aggregated data (compare to Appendix, Figure 8).

4.1 Bayesian inference

To test whether the model has any predictive power, we use Bayesian inference to find out the probability of observing a trigger T (a signal from the model), if there is in fact a task switch S. We then recursively update our prior probabilities as new evidence is released. This means that the outcome of observing a trigger, given a task switch $\mathrm{P}(T|S)$ (read: the probability of a trigger, given a switch) for a certain set of data, will be used as the prior probability $P(T)$ when new evidence is released. This will allow us to test whether the model's predictive power increases by iteratively increasing the amount of available evidence.

Specifically, let $\mathrm{P}(T)$ be the probability that a trigger is observed. Let $\mathrm{P}(S|T)$ be the probability of the occurence of a task switch, given the trigger. Likewise, let $\mathrm{P}(S|\neg T)$ be the probability of observing a switch, given no trigger. The probability of observing a trigger, given the task switch is then:

$$\mathrm{P}(T|S) = \frac{\mathrm{P}(S|T)P(T)}{\mathrm{P}(S|T)\mathrm{P}(T) + \mathrm{P}(S|\neg T)\,(1 - \mathrm{P}(T))} \tag{3}$$

In the same fashion, we also wish to observe the efficiency of given triggers. Therefore, we will also test the probability that a task switch happens if a trigger is fired:

$$\mathrm{P}(S|T) = \frac{\mathrm{P}(T|S)P(S)}{\mathrm{P}(T|S)\mathrm{P}(S) + \mathrm{P}(T|\neg S)\,(1 - \mathrm{P}(S))} \tag{4}$$

The probability will be estimated by enumerating the amount of correct triggers up and until time t_n for all periods t_i where $i = 1, ..., n$. Likewise, we also keep track of the amount of wrong triggers up and until t_n and the total amount of triggers up and until time t_n. Releasing new evidence therefore means increasing n. When this happens, a new posterior probability will be calculated by substituting the last posterior outcome as the new prior in the equations above. For equation 3, this would mean that $\mathrm{P}_{\text{new}}(T|S)$ will be calculated by substituting $\mathrm{P}_{\text{previous}}(T|S)$ as $\mathrm{P}(T)$.

In order to estimate the probabilities, we have calculated the bayesian statistics by randomizing the order in which the evidence was presented to the CRF and then recursively updating the statistics. For this simulation we used individualised learning and the normal 10/20 setting. The results are visualised in Figure 5. The mean of the simulations are denoted by the thick red and blue lines. The initial value of the probabilities $\mathrm{P}(T)$ for equation 3 and $\mathrm{P}(S)$ for equation 4, were both initialised at 0.5. As we can see from Figure 5, the $\mathrm{P}(T|S)$ converges to a probability close to 1.0

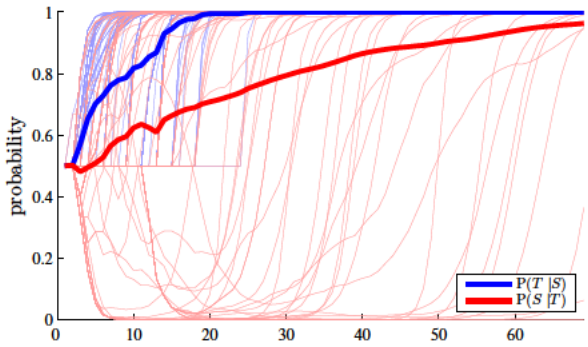

Figure 5: Bayesian inference with randomization

extremely fast. After estimating just 20 task switches, it is pretty clear that the model, on average, will trigger the majority of the time when a task switch occurs. The P($S|T$), however, has more trouble in converging. We see that for many simulation runs, the probability that a trigger correctly predicts a task switch goes to 0 first, converging only after a substantial amount of estimations. Yet, on average, the probability of observing a task switch given a trigger also converges to 1.0 (although slowly). This is due to the fact that for some participants, the model works better than for others. If the randomized order of input presents these participants to the model first, it is likely that the statistics will first take a dive. Later, when the more pursuasive evidence is fed to the model, the probability still converges. This again illustrates the different results among participants.

5. LIMITATIONS

The limitations of our approach are mainly due to the wide variety of necessary design decisions. First of all, we employed an extremely simplistic intercept-only model to use in the Regime-switching HMMs. Improving the complexity of the model would probably improve results. Second of all, we restricted ourselves to detecting time switches based on a 5-second unit interval. We did this because of two reasons. First of all, we did not have the computing power to increase the sampling-frequency. Second of all, the simplistic HMM model did not always yield usable results for a high frequency. Another point of discussion is the implicit definition of task switches: we used the moment that the participant pressed 'start' as the point t_i around which our task switch was based $[t_{i-\lambda}, t_{i+\rho}]$. Choosing another metric as the middle of the task switch might improve results. Likewise, we also chose the setting $\lambda = 10, \rho = 20$ to test our models. After our analysis, we have seen that the data might be skewed towards detecting a task switch (too) early due to the existence of a notification-lag. Judging from the superior accuracy of the $\lambda = 20, \rho = 0$ setting, accounting for such notification-lags could also improve results.

6. RELATED WORK

The approach of automatically detecting task switches using Hidden Markov Models is not new. Shen et al. used a similar approach for detecting task switches using software-based classifiers in a HMM-based cost model [27]. They indicate that their Viterbi-based combination method outperforms methods based on '*simple voting*' and the '*likelihood ratio test*'. Eventhough this paper also employes a HMM-based approach towards task switch detection, there are four main differences with the work by Shen et al. First of all, the variables we use are based on psycho-physiological sensors, opposed to software-based metrics. Secondly, this paper uses the forward-backward algorithm opposed to the Viterbi algorithm, to estimate the most probable state for each given time unit. Thirdly, this paper tries to generalize findings by not constricting the parameters λ and ρ to one fixed setup. Shen et al. base their findings solely on the setting $\lambda = 60$ and $\rho = 180$. We consider this setup to be rather unrealistic as interrupting a developer three minutes after a task switch could be considered slow. Lastly, this paper uses a CRF model, not a HMM-based cost function, to combine variables and employs HMMs only for signal extraction. Another task switch detection paper, by Rath et al., studies the factors that influences the approach towards context detection and task switch detection [24]. According to Rath et al., task switch detection is mainly a problem of classification: they assume that if one can detect the current software-based context of the developer, then classifying the current primary task should be trivial. Task switches are therefore assumed to be software-based context switches. As Rath et al. explain, there is a major limitation to such methods of task switch detection: they employ text-based features only. They continue with stating that *ontology-based user task detection* yields better detection-results. This paper, too, disagrees with using text-based features only. This because we tried to analyse not only software-based task switches but also study a wider variety of interruptions, including externally generated, randomly occurring interruptions. Another approach to detecting task switches is presented by Nair et al. [19]. They developed an approach to task switch detection that differs from traditional application-centric computing models. Apart from classifying the context of developers by using text-based features, Nair et al. analysed the metrics associated with the user's interaction with desktop windows. This paper also employs activity-based metrics but, instead of focussing on software, uses external psycho-physiological sensors to automatically classify switches.

7. CONCLUSION

We have shown that combining Regime-Switching Hidden Markov Models and Conditional Random Fields is a good alternative to other methods and yields relatively good results. We observed that the accuracy of the models was better for individualised data than for aggregate data with the best parameter-setting yielding a bias of only 0.35 second to the true start of the switch. We have also shown that the models are able to accurately learn and estimate task switches based on person-specific data. Eventhough the models performed quite well in detecting *when* a task switch occurs, they are not to be used for detecting the *duration* of a task switch. Bayesian statistics confirmed that the models do indeed have both confirming and predictive power for task switches, given a large enough sample size. Still, real-time task switch detection remains difficult. Considering the fact that this paper uses psycho-physiological data only, there is a huge potential for combining various real-time task switch detection methods and models.

8. REFERENCES

[1] Arrington, C.M., Logan, G.D. 2004. *The Cost of a Voluntary Task Switch*. Psychological Science 2004 15: 610.

[2] Bailey, B.P., Konstan, J.A. 2006. *On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state*. Computers in Human Behavior 22 (2006) 685-708.

[3] Boucsein, W. 2012. *Electrodermal activity - Second Edition*. University of Wuppertal. Springer New York.

[4] Carlsson, F. et al. 2008. *Emotional responses to music: experience, expression, and physiology*. Psychology of Music. Published online 15 October 2008.

[5] Chen, D., Vertegaal, R. 2004. *Using Mental Load for Managing Interruptions in Physiologically Attentive User Interfaces*. CHI 2004, April 24-29, 2004.

[6] Chen, D. 2006. *Towards A Physiological Model of Interruptability*. Thesis of Queen's University Kingston, Ontario, Canada, February 2006.

[7] Corragio, L., 1990. *Deleterious effects of intermittent interruptions on the task performance of knowledge workers: A laboratory investigation*. Unpublished doctoral dissertation, University of Arizona.

[8] Czerwinski, M., Horvitz, E., Wilhite, S. 2004. *A Diary Study of Task Switching and Interruptions*. Microsoft Research. CHI Volume 6, Number 1 2004.

[9] Fuh, C. et al. 2012. *Option Pricing with Markov Switching*. Journal of Data Science 10(2012), 483-509.

[10] Hamilton, J.D. 1989. *A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle*. Econometrica, 57, 357-384.

[11] Hamilton, J.D. 2005. *Regime-Switching Models*. Palgrave Dictionary of Economics - University of California, San Diego.

[12] Ismail, M. T., Isa, Z. 2008. *Identifying Regime Shifts in Malaysian Stock Market Returns*. International Research Journal of Finance and Economics. ISSN 1450-2887 Issue 15. EuroJournals Publishing, Inc. 2008.

[13] Janczura, J., Weron, R. 2011. *Efficient estimation of Markov regime-switching models: An application to electricity spot prices*. Published online: 24 November 2011. Accesible at `http://prac.im.pwr.wroc.pl/~hugo/publ/JanczuraRWeron11_Asta.pdf`.

[14] Johnson, M.P. et al. 1991. *Short-term memory, alertness and performance: a reappraisal of their relationship to body temperature*. J. Sleep Res. (1992) 1, 24-29.

[15] Lamoureux, C. G. and Lastrapes, W. D. 1990. *Heteroskedasticity in Stock Return Data: Volume versus GARCH Effects*. The Journal of Finance, 45: 221âĂŞ229.

[16] Landes, T., Loistl, O. 1992. *Complexity Models in Financial Markets*. Applied Stochastic Models And Data Analysis - VOL. 8, 209-228 (1992).

[17] Liew, C.C., Siu, T.K. 2010. *A hidden Markov regime-switching model for option valuation*. Insurance Mathematics and Economics (Impact Factor: 1.1). 01/2010; 47(3):374-384.

[18] Monsell, S. 2003. *Task switching*. TRENDS in Cognitive Sciences: Vol.7 No.3 March 2003.

[19] Nair, R. et al. 2005. *Frequency-based detection of task switches*. HCI 2005 - Proceedings Vol. 2.

[20] Naive Bayes: explanation of the Naive Bayes Classifier from Princeton University. Accessible at: `http://www.princeton.edu/~achaney/tmve/wiki100k/docs/Naive_Bayes_classifier.html`. Accessed on 20/11/2014.

[21] NeuroSky MindBand Manual. Accesible at: `http://myndplay.com/webimages/mindsetmanual.pdf`. Accessed at 18:28 on 20/11/2014.

[22] Parra et al. 2001. *Higher-order Statistical Properties Arising from the Non-stationarity of Natural Signals*. Advances in Neural Information Processing Systems 13 (Papers from Neural Information Processing Systems - NIPS'2000). MIT Press, Cambridge, MA, 2001: 786-792.

[23] Perlin, M. 2010. *MS Regress - The MATLAB Package for Markov Regime Switching Models*. Available at SSRN: `http://ssrn.com/abstract=1714016`

[24] Rath, A. S. et al. 2010. *Studying the factors influencing automatic user task detection on the computer desktop*. European Conference on Technology En-hanced Learning (EC-TEL 2010), Barcelona, Spain. pp. 292-307.

[25] Rogalski, J., Sumarçay, R. 1990. *Acquisition of Programming Knowledge and Skills*. CNRS - Université de Paris 8, URA 1297: 'Psychologie Cognitive du Traitement de l'Information Symbolique'. Psychology of Programming, Academic Press Limited 1990.

[26] Schmidt, M., Swersky K. 2008. *crfChain - A set of Matlab functions for chain-structured conditional random fields with categorical features*. Available at `http://www.cs.ubc.ca/~schmidtm/Software/crfChain.html`

[27] Shen, J. et al. 2007. *Real-Time Detection of Task Switches of Desktop Users*. IJCAI-07, p. 2868-2873.

[28] Sutton, C., McCallum, A. 2012. *An Introduction to Conditional Random Fields for Relational Learning*. Foundations and Trends in Machine Learning 4 (4). 2012.

[29] "Task Switch" as defined by the Cognitive Atlas. Accessible at: `http://www.cognitiveatlas.org/concept/task_switching`. Accessed on 27/11/2014.

[30] Venables, P. H., Christie, M. J. 1980. *Electrodermal activity*. In I. Martin and P. H. Venables (Eds.) Techniques in Psychophysiology (3-67). New York, NY: Wiley.

APPENDIX
A. EXTRA VISUALISATIONS

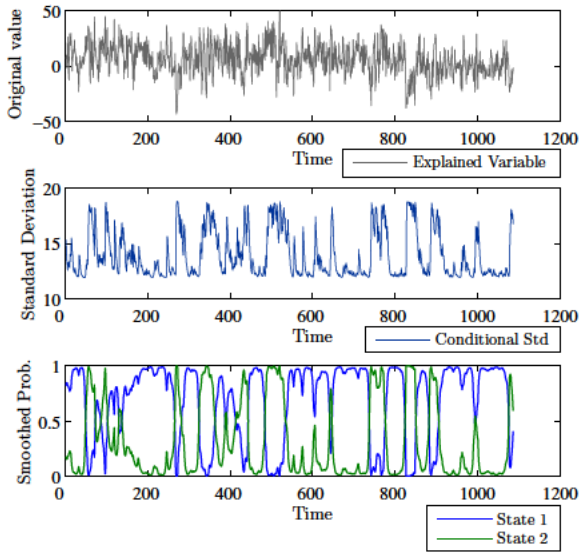

Figure 6: Mean attention difference, 5 second units

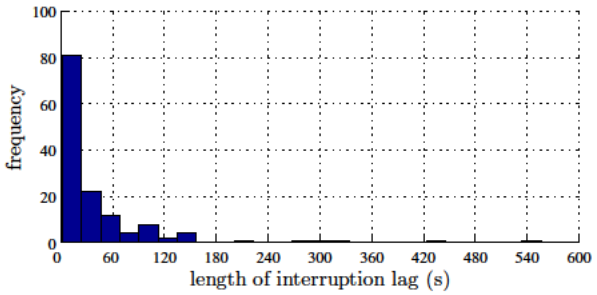

Figure 7: Interruption lag histogram

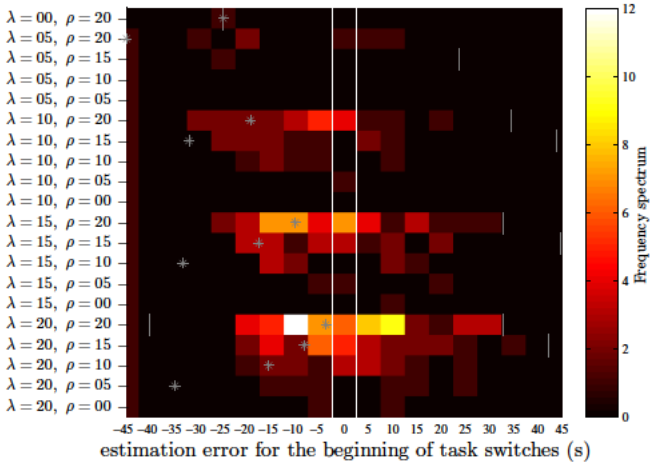

Figure 8: Aggregated results for various λ and ρ