

Automatic Cognitive Load Detection from Speech Features

Bo Yin^{1,2}, Natalie Ruiz^{2,3}, Fang Chen³, M. Asif Khawaja^{1,3}

1 School of Electrical Engineering
and Telecommunications

UNSW, Sydney

+61 (2) 9385 4000

bo.yin@student.unsw.edu.au

2 School of Computer Science and
Engineering

UNSW, Sydney

+61 (2) 9385 4000

natr@cse.unsw.edu.au

3 NICTA

Australian Technology Park
Sydney

+61 (2) 9209 4750

{First.Last}@nicta.com.au

ABSTRACT

Cognitive load variations have been found to impact multimodal behaviour, in particular, features of spoken input. In this paper, we present a design and implementation of a user study aimed at soliciting natural speech at three different levels of cognitive load. Some of the speech data produced was then used to train a number of models to automatically detect cognitive load. We describe a classification approach, the cognitive load levels were detected and output as discrete level ranges. The final system achieved a 71.1% accuracy for 3 levels classification in a speaker-independent setting. The ability to detect and manage a user's cognitive load can help us to adapt intelligent interfaces that ensure optimal user performance

Categories and Subject Descriptors

H.1.2 [User-Machine Systems]: Human Information Processing; H.5.2 [Information Interfaces and Presentation]: User Interfaces—Input Devices and Strategies, Interaction Styles

General Terms

Algorithms, Measurement, Performance, Design, Experimentation.

Keywords

Cognitive Load, Speech input features.

1. INTRODUCTION

The richness of multimodal interactive data means that systems could eventually use these features of behaviour to detect cognitive load variations, in an implicit manner. The identification of such indices could help build intelligent interface systems that adapt to difficulties experienced by the user in real-time, e.g. by regulating the pace, volume or format of the output. Adapting the interface to each user, each time can ensure optimal user performance.

Our research goal is the implicit, objective, automated and real-time estimation of a user's cognitive load, suitable for high-complexity, real-time deployment. To do this, it is necessary to first identify and quantify the fluctuations of features in user's multimodal interaction as the experienced cognitive load varies.

OzCHI 2007, 28-30 November 2007, Adelaide, Australia. Copyright the author(s) and CHISIG. Additional copies are available at the ACM Digital Library (<http://portal.acm.org/dl.cfm>) or can be ordered from CHISIG(secretary@chisig.org)

OzCHI 2007 Proceedings, ISBN 978-1-59593-872-5

In this paper, we focus on potential speech feature indices. We present our attempt to induce controlled levels of load and solicit natural speech, and the use of machine learning in the development of a speech classifier that is able to detect 3 levels of load in the speech signal.

2. BACKGROUND

Cognitive load refers to the amount of mental demand imposed by a particular task, and has been associated with the limited capacity of working memory [12] [11]. However, the same task can affect different users in different ways, and can induce levels of perceived cognitive load that vary from one user to another. This is due to a number of reasons, for example, level of domain or interface expertise of the user, their age, mental or physical impediments, etc. The cognitive load experienced by a user in completing a task has a major impact on their ability to learn from the task, and can severely impact their performance, high load detracting from learning. [12]

While the field is relatively new, a number of methods have been used, both in HCI and other domains, in attempts to estimate experienced levels of cognitive load. The goal of this research is to create an objective measure of cognitive load. There are four main methods that are the state of the art: physiological measures, such as galvanic skin response, and heart rate; subjective (self-report) measures, where users rank their experienced level of load on single or multiple rating scales; performance measures, which include testing and error rates [12]; and finally, behavioural measures, which observe feature patterns of interactive behaviour as they are affected in high load situations, such as mouse speed and pressure, linguistic or dialogue patterns, and even text input events and mouse-click events [16]. From these, the objective measures (those that cannot be overtly manipulated by the user) are the most useful and include observations of physiology such as pupil dilatation, some performance measures such as completion times or behavioural measures such as disfluencies or prosodic changes in speech. as cognitive load increases. These can be standardised and allow for comparison across users [15].

However, many of these are unsuitable for our requirements. The main issue with physiological measures is that they are often very obtrusive and require the user to wear lots of cumbersome equipment that can interfere with the task. Also, the high level of data intensity involved in the collection and analysis can be an issue. Usually, the sampling rate needs to be very high (e.g. polling every 15 to 30 milliseconds), and human expertise is required to interpret the resulting patterns. This

means the system loses its ability to make changes to the interface in real-time. Performance measures, on the other hand, are often accurate indicators of load but again, are unsuitable for real-time, automated deployment since they have the disadvantage of being calculated *post-hoc*. They are perhaps more suited to discreet sequential tasks, as seen in traditional learning environments, rather than continuous interaction with an interface [12]. Performance measures are rarely implicit, they require explicit testing of the user, interrupting the task flow and are unsuitable in environments where the tasks may be time-critical.

Though many behavioural measurements are highly task dependent and difficult to apply across domains, certain kinds of behavioural features seem to be up to the task. They can be objective and collected implicitly i.e. while the user is interacting with the system. They may also be domain independent, and thus suitable for such high intensity environments, where other types of measurement are not. In scenarios with multimodal interfaces that include speech input, the data is almost collected for ‘free’ considering it needs to be captured with a certain level of quality for recognition purposes regardless. We hypothesise that various speech features can be used as implicit indices of cognitive load. Prior-art has shown significant variations in levels of spoken disfluency, articulation rate and filler and pause rates [13] in users experiencing low versus high cognitive load. We expect such individual modal features to form part of a greater multimodal suite of index features acting in concert as robust indices of cognitive load.

3. USER STUDY

Our focus is the identification of novel speech signal features, particularly those that have the characteristic of reliably changing in a distinctive way under different experiences of cognitive load. We began by designing an experiment that dealt only in speech (rather than within our entire multimodal range). This enabled us to collect enough speech to achieve statistical power.

Fifteen (7 male and 8 female), random, remunerated, native English speaking subjects were asked to complete a series of reading and comprehension tasks. The general task required subjects to complete two smaller subtasks: firstly, to read a short passage, and secondly, to answer some open ended questions about that passage. We designed a reading and comprehension task to avoid any expertise effect and expected the level of reading and comprehension of adults over 18 to be relatively similar. The texts chosen for this task were designed to induce 3 different levels of load. The Lexile Framework for Reading [17] was used to rate each of these stories by examining its semantic and syntactic complexity using a 600 million-word corpus. The text complexity ratings range from 200 to 1700 Lexiles, reflecting the expected reading level of a first grade student and a graduate student respectively. Some of the features measured by the Lexile Framework are word length, sentence length and word-frequency. The subjects were asked to read aloud at their own pace and their speech was recorded. Once completed, the passages would be taken away and 3 comprehension questions were asked about the reading which they were asked to respond to in full sentences:

- Give a short summary of the story in at least five whole sentences.
- What was the most interesting point in this story?.
- Describe at least two other points highlighted in this story.

Answers to these open-ended questions were recorded, which allowed the reader to answer in an extended fashion, even though they may not have had the correct answer. This was important as it meant we could collect enough speech data at each level of load. The stories were presented randomly to counter rank and order effects. It was expected that the increase in complexity in the readings would affect the speech through the reading and in the comprehension tasks after each reading. The subjects were also asked to rate the difficulty of reading these stories and again the difficulty of answering the comprehension questions, on a 9-point scale, which would then be used to verify whether the appropriate levels of load were induced.

However, to ensure that the tasks were sufficiently demanding and spaced as far apart as possible in terms of designed complexity, an aural dual-task was added to the most difficult level. The subjects were given a headset to wear while they read, and a series of random two digit numbers were played softly in the background, at random intervals. Subjects were required to count how many numbers they heard throughout the reading. The dual task was also applied to the comprehension question and answer section of that task, they were required to monitor how many numbers were heard while they answered the questions.

Two stories were provided for the subjects at each level:

Load Level	Lexile Rating	Dual Task
1	925L	No
2	1200L	No
3	1350L	Yes

Table 1: Load Levels for the Reading Study

These six were grouped into two sets. All subjects completed both sets of readings and both sets of comprehension.

4. RESULTS

4.1 OBSERVATIONS

Different subjects had slightly different responses toward the task demand. Few found it quite challenging and were desperate to finish it as quickly as possible, others were quite calm and relaxed through the task. The general consensus was that the story with the highest Lexile ranking and combined dual task was the most difficult to handle, and commented on the level of effort expended in that task. Some claimed to ignore the dual-task, but they too reported an increased load merely due to the background “noise”.

4.2 SUBJECTIVE RATINGS

The methods employed to increase the experienced cognitive load in the tasks were found to be effective. The reported difficulties steadily increase within each group for both the Reading and the Comprehension subtask averages. For the first set of texts, in the Reading subtask, the subjective rating for Level 1 is significantly lower than that for Level 2 (difference of 28.8%), shown by a 2-tailed t-test, $p=0.0003$, <0.05 . The rating for Level 3 is significantly higher than that of Level 2, again shown by a 2-tailed t-test, $p = 0.03$, <0.05 . Similarly in the Reading Task in the second set of stories (same subjects), Level 1 is significantly lower than Level 2 (difference of 20%), shown by a 2-tailed t-test, $p=0.002$, <0.05 . Level 3 is significantly higher than Level 2 (by 36.6%), again shown by a 2-tailed t-test, $p<<0.05$. The comprehension subtasks showed very similar significant increases between the levels of cognitive load, as evidenced by the graph below. The subjective

rating results were encouraging evidence that the subjects' experience of load was changing and the designed difficulty was achieving the desired effect.

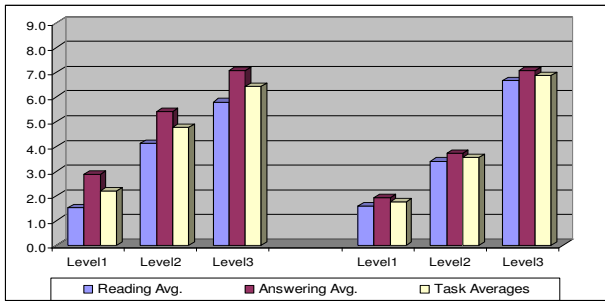


Figure 1: Subjective Ratings for two groups of readings

5. SPEECH BASED COGNITIVE LOAD DETECTION

5.1 CLASSIFICATION TASK

The ultimate purpose of this research is to measure the cognitive load level based on speech features. However, before a theoretical framework can be established to explain the relationship between speech production and cognitive load, a deterministic approach to automated cognitive load measurement is not possible.

Thus, an early approach relies on the classification of speech features using pattern recognition techniques after splitting cognitive load into discrete levels as in the experiment design. In this framework, each level corresponds to an individual class of speech. The classifier system may either use *a priori* knowledge or statistical information extracted from training instances. No prior knowledge is currently available, so we used a statistical classifier system in this research. Using sufficient training instances, the classifier models the designed cognitive load classes, as illustrated in Figure 2, below.

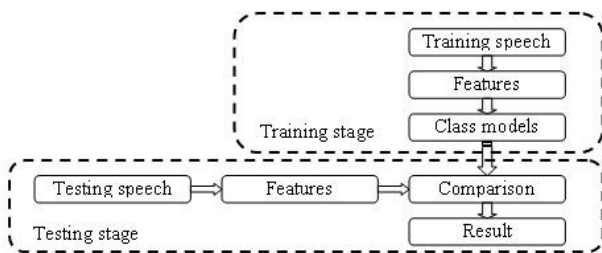


Figure 2 : Typical statistical classification system diagram

The trained class-models are then used to classify a target instance by maximizing likelihood of its input speech features, in order to determine its cognitive load class.

5.2 SPEECH FEATURES

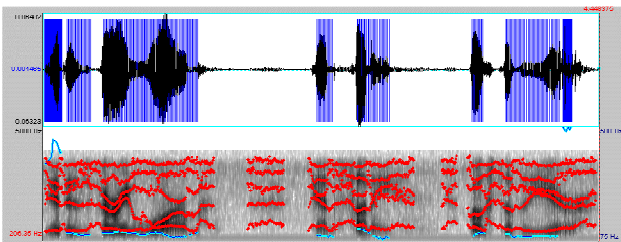


Figure 3: An example of acoustic features

The above figure shows a typical speech utterance and its analysis diagram. The top section depicts the signal in time-domain, while the bottom section depicts the spectrum in frequency domain. In the frequency domain, the line at the bottom indicates the pitch (fundamental frequency), the other lines indicate formants, and the shade of background shows the spectrum. These features provide key information for human auditory perception as well as computer-based speech classification [1].

5.3 Spectrum features

The spectrum feature carries most of the acoustic information and is the major feature in many speech-based recognition tasks. The most popular spectrum feature is Mel-Frequency Cepstral Coefficients (MFCC) [2]. The MFCC extraction algorithm evolved in five steps [3]: (1) Pre-emphasis (2) Spectral analysis (3) Mel-scale filterbank (4) Log (5) Discrete Cosine Transform (DCT). The spectral data is sent to a series of mel-scale filterbanks. The outputs are performed with a natural logarithm. The coefficients are then extracted from the DCT of the log magnitudes.

5.4 Prosodic features

Prosody, which has shown a great potential in emotion and other speech recognition tasks [4, 5], is another important feature in human auditory perception. Specifically, pitch (or fundamental frequency) is used for representing tone; intensity is used for indicating emphasize; which are the two most important characteristics of prosody.

To calculate the pitch value, the autocorrelation function $\phi(\tau)$ of signal $x(n)$ is firstly calculated:

$$\phi(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)x(n+\tau)$$

The pitch then is calculated by tracking the peak of $\phi(\tau)$.

5.5 GMM MODELING

The Gaussian Mixture Model (GMM) maps the feature distributions to a mixture of Gaussian distributions [6]. The idea behind GMM is to model a static distribution of features in the feature space with a series of Gaussian distribution. For example, the distribution of the possibilities of a single dimension feature value is illustrated as the top graph of the following figure:

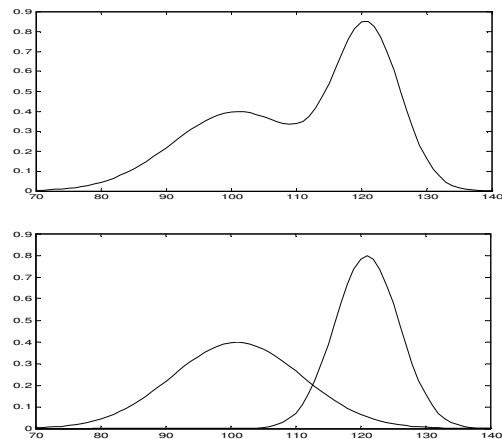


Figure 4: Feature distribution modeled by two Gaussian mixtures

The X-axis is the value of feature. The Y-axis is the possibility of the value. This distribution can be represented as the sum of two Gaussian distributions with different weights, means, and variances, which is shown in the lower diagram.

To model more complicated feature vectors, more Gaussian components are needed. Also, the distribution is not only on single dimension, but on multi-dimensions. Since a single Gaussian distribution can be solely defined with mean and variance, a GMM λ therefore can be parameterized with the weight w , mean vector $\bar{\mu}$, covariance matrix Σ from each of the mixtures. The likelihood of a feature vector \bar{x} to a given GMM λ will be:

$$p(\bar{x} | \lambda) = \sum_{i=1}^K w_i pdf_i(\bar{x})$$

where K is the number of mixtures, $pdf_i(\bar{x})$ is the probability density function of \bar{x} on the i th mixture.

The Expectation Maximization (EM) [6] algorithm is the most popular algorithm used to train the GMM, to maximize the likelihood given by:

$$p(X | \lambda) = \prod_{t=1}^T p(\bar{x}_t | \lambda)$$

where $X = \{ \bar{x}_1, \bar{x}_2 \dots \bar{x}_T \}$ is a sequence of training feature vectors.

Compared to the Bayesian Confidence Network used in existing research, the GMM is more suitable to capture the pattern from noisy data, and already received positive results in speech classification area.

5.6 TEMPORAL INFORMATION

Each feature vector only represents the feature data based exactly at the point of calculation. It's a static 'snapshot'. However, from the viewpoint of perception, the dynamic change is also very important for comprehension. To produce the dynamic information from those static feature vectors, some algorithms for capturing temporal information are needed.

5.6.1 Delta Cepstrum

One simple method to capture the temporal information is Delta Cepstrum. Each of the delta cepstrum coefficients is calculated as:

$$\Delta C_i(n) = \frac{\sum_{k=-N}^N k C_i(n+k)}{\sum_{k=-N}^N k^2}$$

where $\Delta C_i(n)$ is the Delta coefficient calculated at frame n for cepstral stream C_i . In most cases, all Delta Cepstrum coefficients were concatenated to the original feature vector to form a new delta enhanced feature vector.

5.6.2 Acceleration (Delta-Delta)

The acceleration is implemented by repeating Delta calculation on pre-calculated Delta coefficients. It provides the second order dynamic information of original features.

The following figure shows an acceleration enhanced feature vector which originally incorporates cepstral and prosodic features.

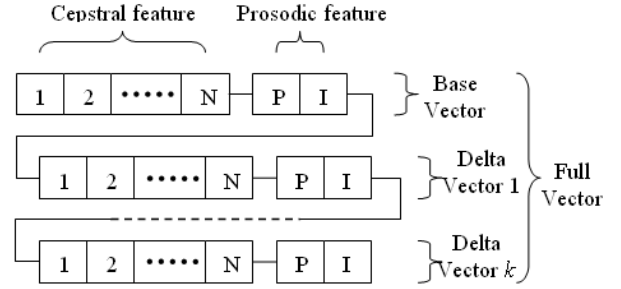


Figure 5: The acceleration enhanced feature vector

5.6.3 Shifted Delta Cepstra (SDC)

An even more advanced algorithm than acceleration is Shifted Delta Cepstra (SDC) [7], which was reported to be superior than acceleration and delta in some situations.

The SDC feature vector is calculated as follows:

$$F_{SDC}(t) = \text{conc} \left(\bar{c}(t+iP+D) - \bar{c}(t+iP-D) \right)_{i=0 \rightarrow k}$$

where $\text{conc}(\cdot)$ means the concatenation operation, $\bar{c}(t)$ is the original feature vector at time t , D , P , k are key parameters for SDC calculation. The advantage of SDC is to provide more long-time relation information than delta or acceleration.

5.7 CHANNEL MISMATCH

One of the key issues in successfully applying the statistical model based classifier is that the speech conditions in the training data and testing data need to be the same. However, due to the short-term distortions, linear channel effects, and other types of interference, some mismatch always occurs between the training and testing data.

To resolve this problem and improve the robustness of system, Cepstral Mean Subtraction (CMS) [8] was deployed in this research. CMS removes any fixed frequency response distortion simply by subtracting the corresponding time average value over the entire speech utterance from each of the cepstral coefficients.

5.8 SPEAKER VARIATION

The variation between different speakers is also a major source of issues for the mismatch problem, especially in a speaker-independent task. To remove the speaker-related characteristics, one possible solution is to map the feature distribution to a unified distribution, namely called Feature Warping [9].

The basic idea of feature warping is the re-mapping of the feature distribution to the ideal distribution assumed by the statistical model over a specified time interval. In case of GMM, a normal distribution is expected. The following figure is an example of the raw signal distribution and the distribution after feature warping.

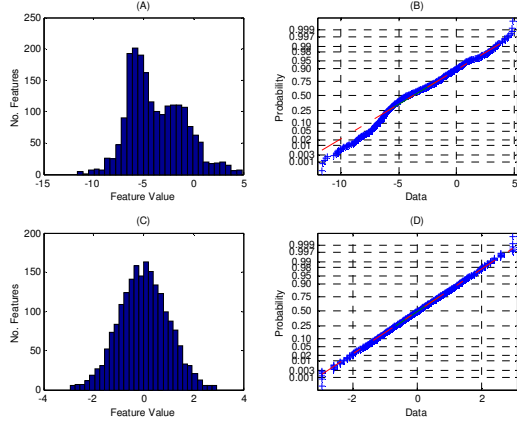


Figure 6: The distribution of a feature in a segment before (A, B) and after (C, D) warping

The warping calculation is done on each of the feature coefficients individually, which assumes different feature coefficients are independent. In the warping process, the mapped value of the current feature value is calculated over a sliding time window:

$$m = ipdf\left(\frac{N + \frac{1}{2} - R}{N}\right)$$

where m is the mapped value, $ipdf()$ is the inverse cumulative distribution function for normal distribution, N is the size of window, R is the ranking (the position in sorted values) of the original value within the current window.

5.9 BACKGROUND MODEL

The straightforward way to model different cognitive load levels is to train individual GMMs for each level from level-specific data. The data collected in this research consists of the Reading and Comprehension subtasks. The amount of data collected from the comprehension tasks was not sufficient for the training of individual models. Although the Reading task had much more data available, surprisingly, it did not show much difference between the tasks at each level of designed cognitive load. There are two pieces of evidence that indicate this. Firstly, when the cognitive load class models are trained solely on the data from the Reading task, the performance of classification of load is close to random classification. In addition, using the Reading task data as well as other data (e.g. part of the Comprehension task data) in the class model training doesn't improve the performance. This does not indicate that there were no differences in the speech data, merely that there were no differences that could be used for training individual GMMs. Differences in other semantic features for example, may still exist, though we have not yet analysed for these. However, a background modeling approach was proposed, where the data from the reading tasks could be used.

A GMM, which is referred to as the background model, was trained from all reading data of all levels. This background model represents the baseline characteristics of speakers. Then the varied cognitive load level models were adapted from the background model using the maximum a-posteriori (MAP) estimation technique [10].

In MAP adaptation, the mean of each mixture in GMM was updated as:

$$\hat{\mu}_i = \frac{N_i}{N_i + \tau} \bar{\mu}_i + \frac{\tau}{N_i + \tau} \mu_i$$

where $\hat{\mu}_i$ is the adapted mean of mixture i , $\bar{\mu}_i$ is the mean of observed adaptation data, μ_i is the mean of background data, N_i is the occupation likelihood of the adaptation data, τ is a weighting of the a priori knowledge to the adaptation data.

During evaluation, the loglikelihood score was calculated by subtracting the loglikelihood scores produced by adapted model and background model:

$$LL_l = \log p(X | \lambda_l) - \log p(X | \lambda_B)$$

where X is the test utterance, λ_l is the l level model, λ_B is the background model. The level which gets the highest loglikelihood score then selected as the classification result.

5.10 A REAL-TIME APPROACH

Since the ultimate purpose of this research is to measure the cognitive load in real-time, an online system was developed. This system can detect the speech and output the cognitive load classification result in real-time. It shows the potential application in a real-world situation.

6. EXPERIMENTS

A GMM based acoustic classifier with all enhanced techniques described in previous sections was developed. The classification task involved three-class hypotheses, corresponding to the three different difficulty levels (which reflect varied cognitive load levels) – L1, L2 and L3, from low to high as seen in Table 1.

The speech data from every subject was allocated as shown in the following figure:

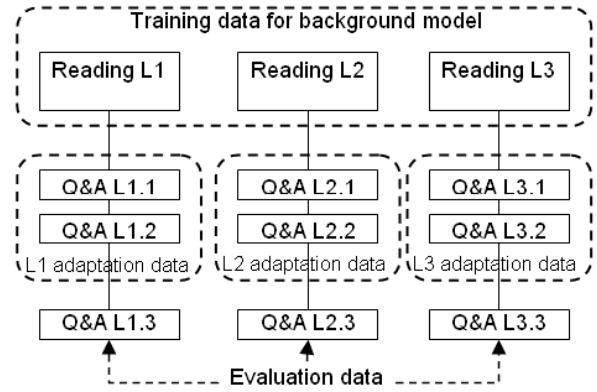


Figure 7 : Data Allocation

All reading speech was used for background model training. The data from the first two comprehension answers from one difficulty level was used to adapt that particular level model. The data from the third comprehension answer was used for evaluation.

In the experiments, different speech features, feature enhancement and normalization algorithms, GMM mixtures, and the effectiveness of background model were investigated. The performances of several typical configurations are shown in the following table.

Table 2: The performances in varied system configurations

ID	SYSTEM CONFIGURATION	Corr%
1	256GMM with Background Model (BM), MFCC	52.2%
2	256GMM with BM, MFCC+Prosodic	59.3%
3	256GMM with BM, MFCC, Prosodic, Delta	62.1%
4	256GMM with BM, MFCC, Prosodic, Acceleration	64.4%
5	256GMM with BM, MFCC, Prosodic, Acceleration, CMS, Feature Warping	71.1%
6	256GMM with BM, MFCC, Prosodic, SDC, CMS, Feature Warping	68.9%
7	512GMM with BM, MFCC, Prosodic, Acceleration, CMS, Feature Warping	62.2%
8	256GMM without BM, MFCC, Prosodic, Acceleration, CMS, Feature Warping	51.1%

The combination of MFCC and prosodic features introduced a significant improvement on the performance. The acceleration algorithm achieved higher performance than delta and SDC, possibly because the SDC largely increased the number of coefficients as well as the feature space, which result in insufficient training data. Similar reason caused the performance drop when GMM mixtures increased. Both CMS and feature warping worked better than expected and further improved the overall performance. The result from the pure GMM system without the background model provides an evidence of the significant improvement introduced by the background model. Among all experiments, the best performance, which is a 71.1% speaker-independent accuracy with 95% confidence interval (-14%, +14%), was achieved by utilizing both MFCC and prosodic features, applying acceleration, CMS, and feature warping techniques.

The confusion matrix in case of the best performance is shown in the following table:

Table 3: The confusion matrix

		Classified as		
		Level 1	Level 2	Level 3
Instances from	Level 1	10	5	0
	Level 2	2	13	0
	Level 3	2	4	9

This confusion matrix clearly demonstrated the classification results at each level performed better than random classification. In all incorrectly classified instances, most were misclassified to the next connected level. There are only two instances were misclassified from Level 3 to Level 1.

7. CONCLUSIONS

In high-intensity, data-laden, real-time system environments, users' cognitive resources can become extraordinarily taxed as they attempt to complete their allotted tasks. This leads to a less efficient work environment, characterized by reduced productivity, fatigue, stress, and errors. The ability to estimate

a user's experienced cognitive load in real-time, and using their implicit interactive data input, will give systems a novel and significant level of flexibility and adaptation when addressing user accessibility. The concept of using speech features to detect changes in load in real time is presented here, and could in future be used as part of an intelligent multimodal interface.

In this paper, we first overviewed a user study design that enabled us to collect speech data under three increasing levels of cognitive load. Analysis of the subjective ratings from the subjects showed that the designed levels of load were actually experienced by the users and consequently could affect their speech production as hypothesized. The reading and comprehension data collected was used to develop a proposed novel and systematic approach to automatically measure the cognitive load levels from speech. By utilizing the modern statistical model based classification techniques, applying advanced feature normalization and enhancement techniques, combining prosodic features and introducing a novel background modeling approach, the proposed system introduced many advantages, such as:

- Automatic cognitive load measuring from speech in real-time;
- Automatic model creation from pre-classified speech without manual labeling or analysis;
- Speaker-independent measurement, there is no need to create a model for each individual subject;
- Utilisation of prosodic features; and
- Novel use of the background model to support the solution.

As a classification approach, the cognitive load levels were detected and output as three discrete level ranges. In the controlled reading and comprehension experiment, the final system achieved a 71.1% accuracy in a speaker-independent setting, which shows a great potential of real-world application in future. To the best of our knowledge, it is the first time these techniques have been used to resolve cognitive load detection problem.

To improve robustness, other modal indices available through a multimodal interface could also be used to confirm or validate automated estimation results from speech. Customised unification algorithms from a multimodal input fusion analysis module may be possible to implement once these features are identified. The ability to implicitly measure the perceived level of cognitive load through changes in multimodal behaviour, particularly speech, could play a crucial role in human computer interaction design, applications could adapt the output flow and presentation to the current load of the user without intrusive probes. This would allow continuously optimal delivery of content, in a very user-centric way.

8. ACKNOWLEDGMENTS

Many thanks to the subjects who participated in our user study.

9. REFERENCES

- [1] S. Greenberg and T. Arai, "What are the Essential Cues for Understanding Spoken Languages?," *IEICE Transaction on Information & System*, vol. E87-D, pp. 1059, 2004.
- [2] P. Mermelstein, "Distance measures for speech recognition, psychological and instrumental," in *Pattern Recognition and Artificial Intelligence*, C. H. Chen, Ed. New York: Academic, 1976, pp. 374–388.

- [3] B. Milner, "A comparison of front-end configurations for robust speech recognition," *Acoustics, Speech, and Signal Processing, IEEE International Conference on (ICASSP)*, 2002.
- [4] I. Luengo, E. Navas, I. Hernández, and J. Sánchez, "Automatic Emotion Recognition using Prosodic Parameters," *Eurospeech 2005*, 2005.
- [5] B. Yin, E. Ambikairajah, and F. Chen, "Combining Prosodic and Cepstral Features in Language Identification," *IEEE International Conference on Pattern Recognition*, Hong Kong, China, 2006.
- [6] S. Dasgupta, "Learning Mixtures of Gaussians," *Symposium on Foundations of Computer Science*, 1999.
- [7] B. Bielefeld, "Language identification using shifted delta cepstrum," *Fourteenth Annual Speech Research Symposium*, 1994.
- [8] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *Journal of the Acoustical Society of America*, vol. 55, pp. 1304-1312, 1974.
- [9] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," *ODYSSEY-2001*, 2001.
- [10] S. Young, *The HTK Book*: Cambridge University Engineering Department, 2005.
- [11] A. Baddeley, "Working Memory". *Science*, 1992. 255: 556-559.
- [12] F. Paas, et. al., "Cognitive load measurement as a means to advance cognitive load theory". *Educational Psychologist*, 2003, 38, 63-71.
- [13] A. Berthold, & A. Jameson, "Interpreting Symptoms of Cognitive Load in Speech Input" In J. Kay (Ed.), *UM99, User modeling: Proceedings of the Seventh International Conference*. Vienna: Springer Wien New York, pp. 235–244, 1999.
- [14] S. Oviatt, R. Coulston, and R. Lunsford, "When Do We Interact Multimodally? Cognitive Load and Multimodal Communication Patterns". In *Proc. Int. Conf. on Multimodal Interfaces* (2004).
- [15] A.F. Kramer. "Physiological metrics of mental workload: a review of recent progress." In D.L. Damos (Ed.), *Multiple-task performance*. London: Taylor & Francis. pp. 279-328, 1991.
- [16] J. Liu, C. K. Wong, K. K. Hui "An Adaptive User Interface Based on Personalised Learning". *IEEE Intelligent Systems* 18(2):52-57, 2003
- [17] The Lexile Framework For Reading (www.lexile.com)