





External Attention LSTM Models for Cognitive Load Classification from Speech

Ascensión Gallardo-Antolín¹✉  and Juan M. Montero² 

¹ Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Avda. de la Universidad, 30, 28911 Leganés, Madrid, Spain
gallardo@tsc.uc3m.es

² Speech Technology Group, ETSIT, Universidad Politécnica de Madrid, Avda. de la Complutense, 30, 28040 Madrid, Spain
juancho@die.upm.es

Abstract. Cognitive Load (CL) refers to the amount of mental demand that a given task imposes on an individual's cognitive system and it can affect his/her productivity in very high load situations. In this paper, we propose an automatic system capable of classifying the CL level of a speaker by analyzing his/her voice. We focus on the use of Long Short-Term Memory (LSTM) networks with different weighted pooling strategies, such as mean-pooling, max-pooling, last-pooling and a logistic regression attention model. In addition, as an alternative to the previous methods, we propose a novel attention mechanism, called external attention model, that uses external cues, such as log-energy and fundamental frequency, for weighting the contribution of each LSTM temporal frame, overcoming the need of a large amount of data for training the attentional model. Experiments show that the LSTM-based system with external attention model outperforms significantly the baseline system based on Support Vector Machines (SVM) and the LSTM-based systems with the conventional weighed pooling schemes and with the logistic regression attention model.

Keywords: Computational Paralinguistics · Cognitive load · Speech · LSTM · Weighted pooling · Attention model

1 Introduction

Cognitive Load (CL) refers to the amount of mental demand that a given task imposes on a subject's cognitive system and it is usually associated to the working memory that refers to the capacity of holding short-term information in the brain [8]. As overload situations can affect negatively the individual's performance, the automatic detection of the cognitive load levels has many applications in real scenarios such as drivers' or pilots' monitoring.

The work leading to these results has been partly supported by Spanish Government grants TEC2017-84395-P and TEC2017-84593-C2-1-R.

Speech-based CL detection systems are particularly interesting since they are non-intrusive and speech can be easily recorded in real applications. In fact, in 2014, an international challenge (Cognitive Load Sub-Challenge inside the INTERSPEECH 2014 Computational Paralinguistics Challenge) was organized with the aim of studying the best acoustic features and classifiers for this task [23]. Following this line of research, this work focuses on the design of an automatic system for CL level classification from speech.

Different features have been proposed for this task, as spectral-related parameters such as, Mel-Frequency Cepstral Coefficients (MFCC) [13, 23], spectral centroid, spectral flux [23], and prosodic cues (intensity, pitch, silence duration, etc.) [2, 24]. For the classifier module itself, Gaussian Mixture Models (GMM) [13] and Support Vector Machines (SVM) [23, 24] are the most common choices.

However, in the last years, the application of deep learning models to speech-related tasks, such as Automatic Speech Recognition (ASR) [21, 22], Language Recognition (LR) [27] or Speech Emotion Recognition (SER) [10, 11, 19] has allowed to increase the performance drastically. As a consequence, nowadays, Deep Neural Networks (DNN) have become the state of the art in this kind of systems. Among all the architectures proposed in the literature for speech-related tasks, Convolutional Neural Networks (CNN) [21], Long Short-Term Memory (LSTM) networks [7] and their combination are the most commonly used. On the one hand, CNNs exhibit the capability of learning optimal speech representations. On the other hand, LSTMs are capable to perform temporal modeling, so they are very suitable for dealing with sequences as it is the case of speech signals.

The so-called attention modeling is a new line of research, complementary to CNNs and LSTMs, that tries to learn the structure of the temporal sequences aiming at determining the relevance of each frame to the task under consideration. Attention models have been successfully proposed for ASR [4], machine translation [17] or SER [10, 11, 19].

In this paper, we propose to adopt the previous findings to cognitive load level classification from speech. As this task has many similarities to SER, our work is mainly based on previous research on emotion classification from speech, especially, [10] and [19]. In particular, we focus on the use of LSTMs in combination with different weighted pooling strategies for CL classification, and we propose an external attention model that tries to take advantage of the benefits offered by attentional schemes, overcoming the need of a large amount of data for their training. Note that one of the main challenges of this kind of tasks is the lack of training data due to the difficulty of collecting and annotating recordings with the appropriate characteristics.

The remainder of this paper is organized as follows: Sect. 2 describes the fundamentals of LSTM with weighted pooling networks, Sect. 3 covers the different weighting schemes used in this work, together to our proposed external attention weighting method. Our results are presented in Sect. 4, followed by some conclusions of the research in Sect. 5.

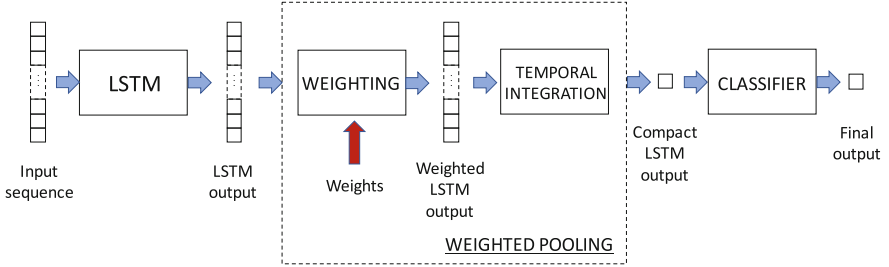


Fig. 1. General scheme of an LSTM with weighted pooling architecture. For simplicity, it is assumed that the LSTM layer is composed by only one LSTM cell.

2 LSTM with Weighted Pooling Networks

Long Short-Term Memory networks are a special kind of Recurrent Neural Networks (RNNs) that have the ability to store information from the past in the so-called memory blocks [7], in such a way that they are capable of learning long-term dependencies, overcoming the vanishing gradient problem. Therefore, LSTM outputs depend on the present and previous inputs, and, for this reason, they are very suitable for modeling temporal sequences, as speech.

The sequence-to-sequence learning carried out by LSTMs can be thought as a transformation of an input sequence of length T , $x = \{x_1, \dots, x_T\}$ into an output sequence $y = \{y_1, \dots, y_T\}$ of the same length, assuming that the classification process is easier in the y -space than in the x -space. However, as in the case of SER, CL classification can be seen as a many-to-one sequence-to-sequence learning problem [10]. Specifically, the input is a sequence of acoustic vectors and the final output must be the predicted CL level for the whole utterance (one single value). For this reason, it is advisable to include an intermediate stage in order to generate a more compact representation of the temporal LSTM output sequence that, in turn, will be the input to the classifier itself [10, 11]. A most common option is the so-called Weighted Pooling (WP) module [19], as shown in Fig. 1. It consists of two different steps: weighting and temporal integration.

A desirable characteristic of WP is the ability for retaining the relevant information regarding the considered task while discarding the non-significant one. This issue can be addressed in the first step, where a weight α_t is computed and assigned to each temporal LSTM output y_t , following a certain criterion. For the CL task, it is reasonable to expect that not all the frames within an utterance reflect the subject's CL state with the same intensity, and therefore, larger weights should be assigned to frames containing significant cues about the speaker's CL, whereas smaller weights should be set to neutral or not relevant frames to the task. Different weighting schemes are discussed in Sect. 3.

In the second step, temporal aggregation, the weighted elements of the LSTM output sequence are somehow combined over time for producing a summarized representation of the information contained in it. For doing this, the most common choice is to perform a simple aggregation operation as follows,

$$z = \sum_{t=1}^T \alpha_t y_t \quad (1)$$

where $y = \{y_1, y_2, \dots, y_T\}$ is the LSTM output sequence, $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_T\}$ is the weight vector and z is the final utterance-level representation.

Note that it is possible to find a parallelism between this method and the temporal feature integration technique that is part of many parameterization modules in conventional hand-crafted feature-based systems, and whose aim is to obtain segment- or utterance-level representations of sequences of short-time features. Temporal integration has been successfully used in different speech/audio-related tasks, such as SER [5] or acoustic event classification [15, 16]. Well-known methods comprise the computation of the statistics (mean, standard deviation, skewness, ...) of short-time acoustic vectors over longer time scales or their filtering [16]. Although out of the scope of this paper, weighted pooling could be performed by applying any of these techniques instead of the simple aggregation operation in Eq. (1).

3 Weighting Schemes

Several weighting schemes have been proposed in the literature. They can be classified into three categories: fixed, local attention and external attention weights.

3.1 Fixed Weights

This is the most simplistic alternative in which the same weights are used across all the utterances. The most used variants are the following:

- **Mean-pooling.** In this case, it is assumed that all the LSTM frames are equally important and, therefore, the weights α are set to,

$$\alpha_t = \frac{1}{T}, \quad \forall t \quad (2)$$

- **Max-pooling.** Here, it is assumed that the whole LSTM output sequence is optimally represented by its maximum, so the weights follow this expression,

$$\alpha_t = \begin{cases} 1, & y_t = \max\{y\} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

- **Last-pooling.** As in LSTM networks every output relies on previous and present inputs, it can be expected that the last outputs are the most reliable ones since for their computation, the LSTM uses to some extent information from the whole utterance [27]. This is equivalent to take into account only the last M frames of the LSTM output, according to the following expression,

$$\alpha_t = \begin{cases} \frac{1}{M}, & T - M < t \leq T \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

3.2 Local Attention Weights

The aim of this approach is to focus on the frames of the utterance that convey more information about the classification task, therefore, a different weight is assigned to each temporal frame. Although when enough training data is available, it is possible to design more complex attention models, as those described in [10, 11], in this work, we adopt the strategy proposed in [19] where the weights are computed as a simple logistic regression as follows,

$$\alpha_t = \frac{\exp(u^T y_t)}{\sum_{t=1}^{t=T} \exp(u^T y_t)} \quad (5)$$

where u and y are the attention parameters and the LSTM output, respectively. Both, the attention parameters and the LSTM outputs, are obtained in the whole training process of the system.

3.3 External Attention Weights

As mentioned before, the lack of training data prevents the use of complex attention models. Our hypothesis is that, in these cases, the attention model is not going to be properly trained and therefore, it should be more effective to use attention weights derived from external cues.

Previous studies about speech production under cognitive load conditions have shown that the level of CL may affect speech by producing changes in the prosody with respect to the neutral voice. In fact, variations in intensity (energy) [12, 14], fundamental frequency (F_0) [2, 12, 14] and duration [2, 14] are correlated to the speaker's cognitive load. We propose to incorporate the information contained in these prosody-related parameters in the weighted pooling scheme of the LSTM network.

Specifically, we consider the energy (actually, we use the log-energy) and F_0 as external attention signals $e_{ATT}(t)$ with the assumption that frames with high energies and F_0 values are more likely to present a strong content about the subject's CL level. The weights of the external attention model are computed from these signals. For doing this, firstly, e_{ATT} is normalized at utterance-level in the range $[0, 1]$ yielding to a normalized signal \bar{e}_{ATT} , and secondly, the weights are obtained as the result of the softmax transformation applied to the normalized attention signal as follows,

$$\alpha_t = \frac{\exp(\bar{e}_{ATT}(t))}{\sum_{t=1}^{t=T} \exp(\bar{e}_{ATT}(t))} \quad (6)$$

This last operation guarantees that the sum of the weights across all the frames of the utterance is one.

4 Experiments and Results

4.1 Database and Baseline System

To the best of our knowledge, nowadays, there are a few speech databases containing utterances pronounced in different CL conditions by a significant number of speakers and conveniently labeled. One of the databases fulfilling these requirements is the “Cognitive Load with Speech and EGG” (CSLE) database [23, 26] that we have adopted for our experiments. It has been used in the Cognitive Load Sub-Challenge inside the INTERSPEECH 2014 COMPARE [23] whose main objective was the assessment of different speech features and classifiers for the prediction of subjects’ cognitive load from their voice characteristics.

The CSLE database contains speech from 26 Australian English speakers recorded at 16 kHz by using a close-talk microphone while performing a set of tasks designed for inducing different levels of cognitive load (low, medium and high, denoted as $L1$, $L2$ and $L3$, respectively). As in the challenge, in this paper, we have considered the following three tasks:

- *Reading Sentence (RS)*. In this case, speakers were asked to read a set of short sentences and recall an isolated letter between them. The degree of cognitive load was objectively assigned according to the number of read sentences before remembering the letter. Each speaker pronounced 75 utterances with a duration of 4 s on average, yielding a total of 1950 speech files.
- *Stroop Time Pressure (STP)*. It is based on the well-known Stroop test [25] where speakers were required to indicate the color of a set of printed words that, in turn, are names of colors. In medium and high load tasks, there was a mismatch between color names and color fonts. In addition, in the case of high load conditions, there was a time constraint for finishing the task. It contains 234 utterances (9 per speaker) with an mean duration of 17 s.
- *Stroop Dual (SD)*. It is similar to the previous task, but in this case, speakers had to execute another simultaneous task (tone counting) in the high load scenario. In total, for this task, 234 utterances (9 per speaker) with an average duration of 21 s were recorded.

The challenge organizers provided a partition of the database into training+development and testing subsets, where it was guaranteed that speakers belong to only one of these subsets (speaker independence). The number of speakers is 18 and 8 in the training+development and testing subsets, respectively. Table 1 shows the details about the database composition.

The baseline system is the one provided by the challenge organizers whose details can be found in [23]. In summary, it uses the standard parameterization adopted in the last Computational Paralinguistics Challenges (6373 characteristics), obtained with the open-source openSMILE feature extractor [6]. The classifier is a linear kernel SVM implemented by using the WEKA toolkit [9].

Following the challenge recommendations, each task was considered separately. This way, for both, the baseline and the LSTM-based models, an independent system per task has been trained with its specific training+development data and evaluated with the corresponding testing data.

Table 1. Composition of the CSLE database. For each task, the number of utterances per subset and cognitive load level are indicated.

Task	Number of utterances				
	Subset	L1	L2	L3	
Reading Sentence	Train+Dev	1350	378	378	594
	Test	600	168	168	264
Stroop Time Pressure	Train+Dev	162	54	54	54
	Test	72	24	24	24
Stroop Dual	Train+Dev	162	54	54	54
	Test	72	24	24	24
Total	Train+Dev	1674	486	486	702
	Test	744	216	216	312

4.2 LSTM-Based Systems Configuration

Figure 2 shows the LSTM architectures with the three main weighting schemes evaluated in this work. In particular, Fig. 2(a) represents the fixed weight approach and its three variants: last-pooling, max-pooling and mean-pooling, Fig. 2(b) shows the system with logistic regression attention weights and Fig. 2(c) depicts our proposal, the LSTM system with external attention model. All systems were implemented with the Tensorflow [1] and Keras [3] packages.

In all cases, the same input acoustic features were used. The feature set consists of $n_B = 64$ log-Mel filterbank energies (log-Mels), computed every 10 ms using a Hamming window of 32 ms long and a mel-scaled filterbank composed of $n_B = 64$ filters by using the Librosa Python toolkit [18]. After feature extraction, mean and standard deviation normalization are applied at utterance-level yielding to a set of normalized log-Mels sequences x_I with $T \times 64$ dimensions, where T is the number of frames of each utterance.

In all architectures, the length of the LSTM input sequences is set to $L = 1024$ for the *RS* task and $L = 2048$ for the *STP* and *SD* tasks, which corresponds to approximately 10 s and 20 s, respectively. Shorter utterances are padded with zeros by using a Masking layer, in such a way that these masked values are not used in further computations. Longer utterances are cut (this is only necessary in a few cases in the *SD* task). The output sequence of the Masking layer is denoted as x and its dimensions are $L \times 64$.

This sequence is passed through an LSTM recurrent layer with $n_L = 128$ memory cells and 25% dropout to avoid over-fitting in the training process. The LSTM output, denoted as y , is a sequence of size $L \times 128$. Next, the information contained in y is summarized by using the considered weighting scheme with weights α , yielding to a 128-dimensional vector, z . The length of the weight vector α is L . However, note that when $T < L$, $\alpha_t = 0$, $T < t \leq L$. The vector z is the input of a dense layer with $n_C = 3$ nodes (as the classes of our system are the 3 CL levels, L_1 , L_2 and L_3) with softmax activation producing

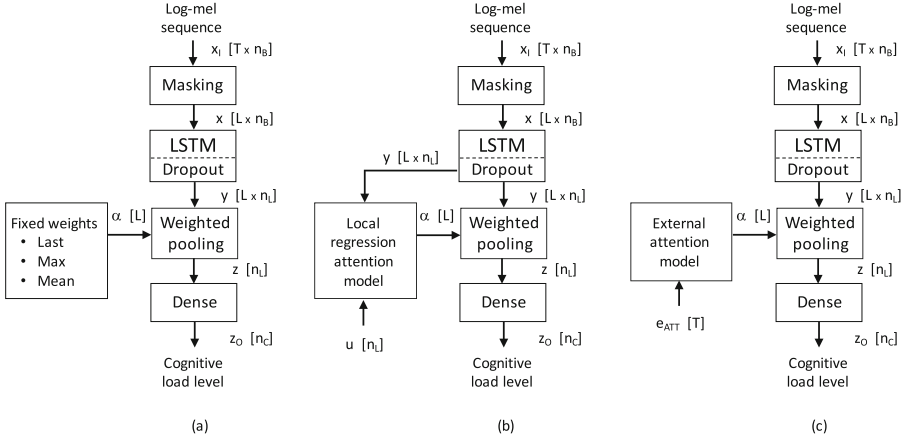


Fig. 2. Different LSTM-based architectures for cognitive load classification. (a) Fixed weights (last-pooling, max-pooling, mean-pooling); (b) Local logistic regression attention model; (c) External attention model. In brackets, the dimension of each variable, where T , L , n_B , n_L and n_C , stand for the number of frames of the input signal, the length of the LSTM input/output sequence, the number of mel filters, the number of LSTM units and the number of classes (CL levels), respectively.

a 3-dimensional output, z_O , representing the probabilities of each class. Finally, the class with the highest probability is assigned to the utterance.

In all cases, the LSTM models were trained using stochastic gradient descent and the Adam method with an initial learning rate of 0.001. We used a batch size of 32 and a maximum number of 60 epochs.

In the logistic regression attention model, the attention parameter vector u has a dimension of $n_L = 128$. All its components were initialized to $1/L$ and then refined during the training stage of the whole system.

In the external attention model, e_{ATT} denotes the external attention signal from which the weights α are derived. In this work, we have considered two alternatives for e_{ATT} . In the first case, it corresponds to the fundamental frequency F_0 of the speech signal computed every 10 ms using a Hamming window of 32 ms long and constraining the maximum F_0 to 500 Hz. In the second case, e_{ATT} is the log-energy of the speech signal extracted every 10 ms using a Hamming window of 32 ms long. Both, F_0 and log-energy were computed with the Librosa Python toolkit [18].

4.3 Results

This Subsection contains the experiments carried out in order to assess the performance of the proposed LSTM-based systems. As the number of instances for each class (CL levels) is unbalanced, results are given in terms of the Unweighted Average Recall (UAR) that is computed as the unweighted mean of the class-specific recalls.

Table 2 contains the results achieved for the baseline system and different LSTM architectures for the three tasks under consideration, *Reading Sentence*, *Stroop Time Pressure* and *Stroop Dual*. The column “Average” refers to the micro-average across the tasks. In the case of the LSTM-based systems, each experiment was run 10 times and therefore, results in Table 2 are the average UAR across the 10 subexperiments and the respective standard deviation.

LSTM corresponds to the conventional approach where no weighted pooling is applied and only the last frame of the LSTM output is passed through the following dense softmax layer. In the *LSTM+VAD* alternative, a Voice Activity Detector (VAD) is applied to the raw speech signals before the feature extraction in order to remove the silence/noise frames. As can be observed, the use of a VAD produces a decrease in performance. This suggests that silence pauses convey important information for discriminating between different CL levels, as they are related to the rhythm, elocution speed and disfluencies that can be heavily affected by the speaker’s CL state. This result corroborates the observations about the effects of CL on speech production mentioned in, for example, [20].

The fixed weighting schemes evaluated are *Last-pooling* (in this case, the last $M = 200$ frames of the LSTM output were picked and averaged), *Max-pooling* and *Mean-pooling*. All these strategies outperform the conventional LSTM showing that not only the last frame contains relevant information for the task. Among these approaches, *Mean-pooling* achieves the best performance, and therefore, it seems better not to completely discard LSTM frames.

The *Logistic Regression Attention* method outperforms the previous ones, although its results overlap with *Mean-pooling* in the *RS* task and in the average across the three tasks. Nevertheless, it is clear that focusing on frames conveying more CL characteristics can help to improve the performance of the system.

Our proposal, the two *External Attention* approaches, produces the best results for all the tasks in comparison to the rest of LSTM-based systems. Comparing both approaches, using the log-energy as external attention signal slightly outperforms the F_0 alternative. Any case, these results support our hypothesis that the log-energy and F_0 could be used for establishing to some extent the relative importance of the frames for the CL level classification task.

Figure 3 depicts the weights used in the weighted pooling stage of the *Logistic Regression Attention* (top) and the *External Attention* strategy with log-energy (bottom). Contrary to the observations made in [19], in our case, the regression attention weights are very uniform and closely resemble the mean-pooling weights. This justifies the fact that the results achieved by *Mean-pooling* and *Logistic Regression Attention* are rather similar. We hypothesize that one possible reason for this behaviour is the lack of data for adequately training both, the attention and the LSTM model. However, the weights derived from the log-energy in the *External Attention* approach presents a large degree of variation, suggesting that the log-energy becomes a good approximation of the amount of cognitive load content of a speech frame when no enough data is available for training more sophisticated attention models. On average, *External Attention Energy* achieves 9.61% relative error reduction with respect to *Mean-pooling* and 6.85 % with respect to *Logistic Regression Attention*.

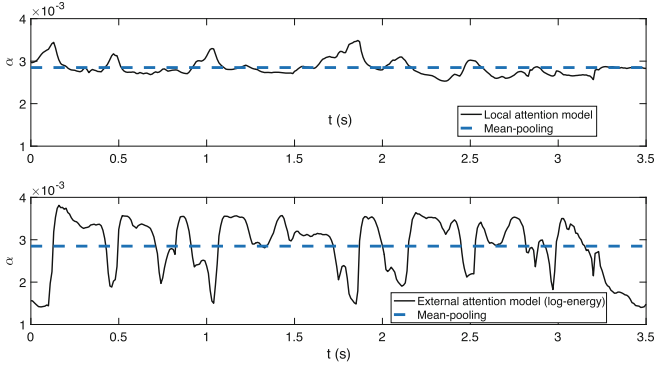


Fig. 3. Attention weights for one utterance belonging to the *Reading Sentence* task. Top: Weights obtained with the local regression attention strategy. Bottom: Weights derived from the log-energy used in the external attention approach.

Regarding the comparison of the LSTM-based systems to the baseline, it can be observed that *Logistic Regression Attention*, *External Attention Energy* and *External Attention F₀* clearly outperforms the SVM-based system for the *RS* task and on average across the three tasks. For the *STP* and *SD* tasks, these systems perform similarly, but these results are not very reliable as the number of test files in both cases is rather small (72 utterances). In summary, *External Attention Energy* achieves a relative error reduction with respect to the baseline of 11.04% and 9.64% for the *RS* task and on average, respectively.

Table 2. Unweighted Average Recalls (UARs) [%] for the baseline system and different LSTM-based classifiers for the Reading Sentence (RS), Stroop Time Pressure (STP) and Stroop Dual (SD) tasks and on Average.

System	RS	STP	SD	Average
SVM [23]	61.50	66.70	56.90	61.60
LSTM	48.87 ± 1.36	55.42 ± 1.02	45.83 ± 4.09	49.61 ± 1.33
LSTM+VAD	45.34 ± 1.79	54.01 ± 2.02	46.60 ± 4.06	46.36 ± 1.51
LSTM Last-Pooling	52.42 ± 1.53	59.57 ± 2.81	46.60 ± 4.11	52.67 ± 1.30
LSTM Max-Pooling	59.87 ± 1.28	53.48 ± 0.98	41.95 ± 1.83	57.54 ± 1.18
LSTM Mean-Pooling	62.99 ± 0.82	60.69 ± 0.67	50.00 ± 2.07	61.61 ± 1.01
LSTM Logistic Regression Attention	63.58 ± 0.48	63.47 ± 0.67	54.59 ± 0.67	62.75 ± 0.59
LSTM External Attention F ₀	65.24 ± 0.95	64.68 ± 0.52	56.35 ± 1.38	64.32 ± 0.83
LSTM External Attention Energy	65.75 ± 0.44	65.97 ± 0.76	59.20 ± 1.03	65.30 ± 0.70

5 Conclusions and Future Work

In this paper, we have developed an automatic system capable of classifying the cognitive load level of a speaker by analyzing his/her voice, based on LSTM models with different weighted pooling strategies. We have evaluated and compared the performance of mean-pooling, max-pooling, last-pooling and a logistic regression attention model. In addition, we have proposed a novel attention mechanism, called external attention model, that uses external cues, such as log-energy and fundamental frequency, for weighting the contribution of each LSTM temporal frame and that it is suitable in situations with scarce training data, as in this case. Experiments have shown that our proposal achieves, on average, a relative error reduction of 9.64% and 6.85% with respect to the baseline SVM and the LSTM with logistic regression attention systems, respectively.

For future work, we plan to extend our research in two directions: to explore different data augmentation techniques for increasing the amount of data for training the LSTM-based system and to study the use of other external cues.

Acknowledgments. We would like to thank Prof. J. Epps for kindly providing the CSLE dataset and Prof. B. Schuller and the rest of the COMPARE 2014 organizers for kindly providing the dataset partition and the baseline system.

References

1. Abadi, M., et al.: TensorFlow: large-scale machine learning on heterogeneous systems. Software (2015). [tensorflow.org](https://www.tensorflow.org)
2. Boril, H., Sadjadi, O., Kleinschmidt, T., Hansen, J.: Analysis and detection of cognitive load and frustration in drivers speech. In: Proceedings of INTERSPEECH 2010, pp. 502–505 (2010)
3. Chollet, F., et al.: Keras: the python deep learning library. Software (2015). <https://github.com/fchollet/keras>
4. Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K., Bengio, Y.: Attention-based models for speech recognition. In: Proceedings of NIPS 2015, pp. 577–585 (2015)
5. Eyben, F., Huber, B., Marchi, E., Schuller, D., Schuller, B.: Real-time robust recognition of speakers' emotions and characteristics on mobile platforms. In: Proceedings of ACII 2015, pp. 778–780 (2015)
6. Eyben, F., Wenginger, F., Groß, F., Schuller, B.: Recent developments in openSMILE, the munich open-source multimedia feature extractor. In: Proceedings of MM 2013, pp. 835–838 (2013)
7. Gers, F.A., Schraudolph, N.N., Schmidhuber, J.: Learning precise timing with LSTM recurrent networks. *J. Mach. Learn. Res.* **3**, 115–143 (2003)
8. van Gog, T., Paas, F.: Cognitive load measurement. In: Seel, N.M. (ed.) *Encyclopedia of the Sciences of Learning*, pp. 599–601. Springer, Boston (2012). <https://doi.org/10.1007/978-1-4419-1428-6>
9. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.: The WEKA data mining software: an update. *SIGKDD Explor.* **11**, 10–18 (2009)
10. Huang, C., Narayanan, S.: Attention assisted discovery of sub-utterance structure in speech emotion recognition. In: Proceedings of INTERSPEECH 2016, pp. 1387–1391 (2016)

11. Huang, C., Narayanan, S.: Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition. In: Proceedings of ICME 2017, pp. 583–588 (2017)
12. Huttunen, K., Keränen, H., Väyrynen, E., Pääkkönen, R., Leino, T.: Effect of cognitive load on speech prosody in aviation: evidence from military simulator flights. *Appl. Ergon.* **42**(2), 348–357 (2011)
13. Kua, J.M.K., Sethu, V., Le, P., Ambikairajah, E.: The UNSW submission to INTERSPEECH 2014 compare cognitive load challenge. In: Proceedings of INTERSPEECH 2014, pp. 746–750 (2014)
14. Lively, S.E., Pisoni, D.B., Summers, W.V., Bernacki, R.H.: Effects of cognitive workload on speech production: acoustic analyses and perceptual consequences. *J. Acoust. Soc. Am.* **93**(5), 2962–2973 (1993)
15. Ludeña-Choez, J., Gallardo-Antolín, A.: Feature extraction based on the high-pass filtering of audio signals for acoustic event classification. *Comput. Speech Lang.* **30**(1), 32–42 (2015)
16. Ludeña-Choez, J., Gallardo-Antolín, A.: Acoustic event classification using spectral band selection and non-negative matrix factorization-based features. *Expert. Syst. Appl.* **46**(1), 77–86 (2016)
17. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. arXiv preprint [arXiv:1508.04025](https://arxiv.org/abs/1508.04025) (2015)
18. McFee, B., et al.: Librosa: audio and music signal analysis in python. In: Proceedings of SCIPY 2015, pp. 18–25 (2015)
19. Mirsamadi, S., Barsoum, E., Zhang, C.: Automatic speech emotion recognition using recurrent neural networks with local attention. In: Proceedings of ICASSP 2017, pp. 2227–2231 (2017)
20. Müller, C., Großmann-Hutter, B., Jameson, A., Rummer, R., Wittig, F.: Recognizing time pressure and cognitive load on the basis of speech: an experimental study. In: Bauer, M., Gmytrasiewicz, P.J., Vassileva, J. (eds.) *UM 2001. LNCS (LNAI)*, vol. 2109, pp. 24–33. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44566-8_3
21. Qian, Y., Bi, M., Tan, T., Yu, K.: Very deep convolutional neural networks for noise robust speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(12), 2263–2276 (2016)
22. Rao, K., Peng, F., Sak, H., Beaufays, F.: Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In: Proceedings of ICASSP 2015, pp. 4225–4229 (2015)
23. Schuller, B., et al.: The INTERSPEECH 2014 computational paralinguistics challenge: cognitive & physical load. In: Proceedings of INTERSPEECH 2014 (2014)
24. van Segbroeck, M., Travadi, R., Vaz, C., Kim, J., Black, M.P., Potamianos, A., Narayanan, S.S.: Classification of cognitive load from speech using an i-vector framework. In: Proceedings of INTERSPEECH 2014, pp. 751–755 (2014)
25. Stroop, J.R.: Studies of interference in serial verbal reactions. *J. Exp. Psychol.* **18**(6), 643 (1935)
26. Yap, T.F.: Speech production under cognitive load: effects and classification. Ph.D. dissertation, The University of New South Wales, Sydney, Australia (2012)
27. Zazo, R., Lozano-Díez, A., González-Domínguez, J., Toledano, D.T., González-Rodríguez, J.: Language identification in short utterances using long short-term memory (LSTM) recurrent neural networks. *PLoS ONE* **11**(1), e0146917 (2016)