Classification of Alzheimer's Disease with Deep Learning on Eye-tracking Data*

Harshinee Sriram[†] Department of Computer Science The University of British Columbia Vancouver, BC, Canada <u>hsriram@cs.ubc.ca</u> Cristina Conati Department of Computer Science The University of British Columbia Vancouver, BC, Canada <u>conati@cs.ubc.ca</u> Thalia Field Vancouver Stroke Program and Division of Neurology, The University of British Columbia, Vancouver, BC, Canada <u>thalia.field@ubc.ca</u>

ABSTRACT

Existing research has shown the potential of classifying Alzheimer's Disease (AD) from eye-tracking (ET) data with classifiers that rely on task-specific engineered features. In this paper, we investigate whether we can improve on existing results by using a Deep Learning classifier trained end-to-end on raw ET data. This classifier (VTNet) uses a GRU and a CNN in parallel to leverage both visual (V) and temporal (T) representations of ET data and was previously used to detect user confusion while processing visual displays. A main challenge in applying VTNet to our target AD classification task is that the available ET data sequences are much longer than those used in the previous confusion detection task, pushing the limits of what is manageable by LSTM-based models. We discuss how we address this challenge and show that VTNet outperforms the state-of-the-art approaches in AD classification, providing encouraging evidence on the generality of this model to make predictions from ET data.

1 Introduction

In recent years, eye-tracking has been extensively investigated as a source of information for AI agents to determine relevant properties of their users. This research has already generated very encouraging results, showing that eye-tracking (ET) data can be used to train classifiers for predicting user short-term states such as confusion, affect, and mind wandering [26,4,25], as well as long-term properties such as cognitive abilities, personality traits, and health conditions [20,22,41]. Most of these results have been achieved by using traditional Machine Learning (ML) classifiers rather than Deep Learning (DL) methods. This is in part because existing ET datasets are usually relatively small, as acquiring accurate ET data currently requires specialized equipment and collection in a lab setting.

However, there have been some initial attempts to use DL methods to make predictions on user properties from ET data. Most of this existing work converts the ET data into a visual representation (i.e., scanpath or heatmap) that is then analyzed by a CNN-based classifier for prediction [5,10,30]. In contrast, Pusiol et al. [36] trained an RNN model on sequences of ET fixations (namely clusters of raw ET samples associated with

unique points of attention) to distinguish between two developmental disorders.

A DL architecture that leverages both the visual (V) and temporal (T) aspects of ET data, called VTNet, was proposed by Sims and Conati [39]. VTNet includes a GRU and a CNN that operate in parallel, with the GRU taking as direct input raw sequential ET samples, while the CNN processes the corresponding scanpath image, namely a representation of the samples' X and Y coordinates and the transitions between them. This approach was successfully used for classifying user confusion while processing visualizations [26]. The approach also outperformed its GRU and CNN components when they were trained, respectively, on the temporal and visual representation of the ET sequences, showing the value of combining the two representations.

In this paper, we investigate if the VTNet architecture can also improve on the state-of-the-art results in a very different context: leveraging ET data to classify Alzheimer's Disease (AD). There has been increasing interest in devising lightweight classifiers of AD as an initial screening for this condition [15,17,32] because existing assessments tend either to be resource-intensive and time-consuming (e.g., specialized neuroimaging and detailed cognitive assessments), or they are lightweight cognitive screening tools [7] that are not sensitive enough to detect AD or other mild cognitive impairments that can develop into AD. There is existing research that shows the potential of classifying AD from ET data alone or together with language data generated during simple screening tasks [3,20,35]. Most of these works use non-DL classifiers, and they all rely on features engineered based on knowledge of the task at hand.

In this paper, we investigate if VTNet can improve these results when trained end-to-end from raw ET data. In particular, we focus on the previous work that currently has the most reliable results [20] obtained from one of the largest ET datasets for AD screening in the literature (AD dataset from now on), which was collected from participants engaging in three different tasks: a pupil calibration task, a picture description task, and a textual paragraph reading task. The challenge in working with this dataset is that the ET data sequences are much longer than those leveraged in the previous work that used VTNet for classifying confusion. Sequence length is known to be a potentially limiting factor in the effectiveness of LSTM-based models if the length is beyond 1,000 timesteps [27], and the AD dataset involves sequences that have an average length of over 8,000 timesteps, with a maximum of over 26,000 timesteps. We address this issue by first exploring ways to reduce the sequence lengths, and next by augmenting the VTNet architecture with an attention layer, given that attention has been successfully used in Natural Language Processing (NLP) and Computer Vision to allow a model to focus on the most important parts of an input sequence.

Our results show that combining targeted length reduction with the addition of the attention layer allows VTNet to outperform state-of-the-art AD classifiers. These results entail two contributions. The first contribution is a step forward in the quest for accurate lightweight classifiers for AD. The second contribution is that we show the value of the VTNet architecture in a very different classification task, thus providing initial evidence on the generality of this approach for improving the ability of AI agents to leverage ET data to make classifications on relevant properties of their users.

The rest of this paper is organized as follows: Section 2 reviews related work. Sections 3 and 4 describe the AD dataset and the data preprocessing steps. Section 5 summarizes the VTNet model architecture. Section 6 evaluates VTNet with ET sequences of different lengths, while Section 7 evaluates VTNet with attention. Finally, Section 8 concludes the paper and discusses avenues for future work.

2 Related Work

Leveraging ET data for AD classification. Research on ET data as a source of information for AD detection has been inspired by evidence that AD affects the functioning of the eye, causing abnormalities in fixations, saccades, and pupillary responses [15,31,32]. All existing works trained classifiers on ET data features based on knowledge of what was important during the visual tasks designed to test functionalities known to be degraded by AD. For instance, Pavisic et al. [35] trained their classifiers on features relevant to assess one's performance during tasks that tested fixation stability, focus on appearing stimuli and tracking a moving target. Biondi et al. [3] leveraged ET features relevant to reading tasks (e.g. number of repeated fixations on a word). Jang et al. [20] leveraged ET data from tasks such as pupil calibration, picture description, and reading, and trained their classifiers on features defined by looking at specific regions of interest (ROIs) in their tasks (e.g., parts of the picture and paragraph).

All these works reported accuracies above 80%, however, the datasets used in [35] and [3] were rather small in size (57 and 69 datapoints respectively), and these works only reported classification accuracy as their performance metric. On the other hand, Jang et al. [20] leveraged a larger dataset (126 datapoints) and provided an extensive analysis based on several performance metrics. Therefore, we use the data and classifiers in Jang et al. [20] to test and compare the performance of our proposed VTNet approach, which, in contrast to previous approaches, is trained directly on the raw ET samples, hence removing the need to create task-specific engineered features.

DL classifiers leveraging ET data. Recently, several studies have employed DL techniques to make predictions about users from their ET data [1,5,10,30,36,43]. Most of these studies convert the ET data into a visual representation, namely a scanpath or a heatmap (a format that uses color to show the degree of attention to a visual display), which is then used by a CNN-based classifier for prediction. For instance, CNN models have used scanpaths to predict the strategies of participants playing games (e.g., chess [30] and different types of economic games [5]). CNN models trained on heatmaps have been used to classify the ages of toddlers viewing images [10] and the attentional states of participants performing tasks in Augmented Reality [43]. In all these works, the CNN classifiers outperformed a non-DL baseline. Because these approaches do not process the sequences of ET data, they do not face the problem of excessive sequence length.

In contrast to approaches that rely on the visual representation of ET data, Pusiol et al. [36] trained their classifiers on the ET sequences themselves. The data was collected from participants with two different developmental disorders, and this work proposed an RNN-based classifier to distinguish between the two disorders. The training data consisted of ET sequences indicating if a patient was looking at certain face regions (nose, jaw, etc.) of a practitioner who was conducting a diagnostic interview with them, where the sequences were obtained by overlaying the participant's fixation points on the video of the interviewer's face. The resulting sequences were 3,000 timesteps in length. The authors experimented with three window lengths (15, 50, and 250 timesteps), to determine how much sequential information is required for classification and found the RNN trained on the 250 timesteps sequences to be the winning classifier.

Finally, Asish et al. [1] leveraged sequences of raw ET data to classify the distraction level of students in a Virtual Reality classroom. The sequences in the dataset were between 12,000 and 33,000 timesteps long. The authors compared a CNN, an LSTM, and a sequential combination of the two, trained on the data end-to-end, against a Random Forest (RF) classifier trained on summary statistics of the data, with the RF classifier being the winning model.

Leveraging attention to focus on important parts of long sequences. In Natural Language Processing (NLP), several works have used attention to extract important parts from sentences, paragraphs, and utterances for tasks such as speech recognition [6], dialogue act detection and key term extraction [38], and relation extraction [48]. The average length of sequences in these datasets ranges from 36 words per sentence [48] to 120 words per article [38]. Later work in NLP relies on transformer-based architectures to deal with longer sequences (e.g., [9]), but transformers are unsuitable for our work because they are complex models that require large datasets for training.

Attention has also been used in Computer Vision to extract important parts from videos for tasks such as video summarization [21,37], skill level assessment [13], and video question answering [45]. The datasets used in these works originally range from 15,000 timesteps [21] to 540,000 timesteps [37]. To deal with such high sequence lengths, all these approaches reduce the length by decreasing the video sampling frequency [13,21,37,45]. In addition, in [37], the authors use knowledge-based approaches to extract interesting video segments [37], whereas in [13] they uniformly split each video into multiple segments [13].

Combining RNNs and CNNs. There are several works that (like our own) combine the strengths of RNNs and CNNs. Most of these works relate to processing videos [12,18,40,49] and audio [19] using Recurrent Convolutional Networks (RCNs). RCNs typically operate on an input of image sequences (i.e., the frames of a video or spectrograms from audio recordings) where, at each timestep, a CNN extracts visual features from the image and feeds them to the RNN, which models the temporal dynamics of the sequence. RCNs have also been used to predict user states such as emotional valence and intentions from multichannel EEG signals [28,47,50], where the input to the CNN encodes the spatial relationship among the EEG sensors placed on the user's head, along with their values. RCNs combine CNN and RNN at every timestep and therefore do not decouple the temporal from the spatial aspects of the data completely as done in VTNet. By providing a single scanpath to the CNN, VTNet processes the high-level spatial representation of the participant's overall activity related to a task, which complements more local temporal information about potential indicators of AD generated by the GRU from raw sequences.

An alternative way to combine a CNN and an LSTM is to use them in sequence instead of in parallel as VTNet does. Pascanu et al. [34] showed that LSTMs can learn better from high-level features of text embeddings extracted by CNNs, than from the raw embeddings themselves. This approach has been used mainly in NLP for sentiment analysis [2,11,29,44,46], and it does not seem to be useful for our goal of screening participants with AD because our ET data does not have such high dimensionality as word embeddings (each datapoint is a 6-dimensional vector), and reducing the dimensionality may result in a loss of possible patterns for the screening task.

3 Dataset

This paper relies on a dataset originally collected by Jang et al. [20] to build multimodal lightweight screening tools for AD. The data was collected from patients of a specialized memory clinic (either diagnosed with AD or showing early signs of mild cognitive impairments potentially leading to AD) and from control participants from the community (matched with the patient group based on sex and age).

During the study, participants were seated at a testing platform that had a Tobii-Pro X3-120 eye tracker (120 Hz sampling frequency) installed at the bottom of the screen to track gaze coordinates, head distance, and pupil data. The participants were given four tasks to complete: a pupil calibration task, a picture description task, a paragraph reading task, and a memory recall task that did not involve any visual elements. The study's full details are available in [20]. This paper utilizes ET data collected during the first three tasks:

• **Pupil Calibration:** Participants were asked to stare at a still target for 10-15 seconds (Figure 1A) to capture any square-wave jerks that are a hallmark of AD [33].

• **Picture Description:** Participants were asked to verbally describe the Cookie Theft picture (Figure 1B) from the Boston Diagnostic Aphasia Examination [16], a task that has been used extensively for assessing spontaneous speech in a variety of clinical settings [8], including AD [14,23,24].

• **Reading:** Participants were asked to read aloud a standardized paragraph (Figure 1C) from the International Reading Speed Texts (IReST), which is a collection of texts designed to assess reading impairments [42]. The 155-word paragraph described how plants and animals in hot and dry areas adapt to their environment in 9 sentences. The objective of this task was to capture common reading-task deficits associated with AD, such as reduced reading speed and increased word fixations or re-fixations.

Completing these three tasks took an average of 7 minutes. The final ET dataset used in this paper contains 75 control participants (avg. age = 62, std. dev. = 15) and 69 patients (avg. age = 72, std. dev. = 9). The sequence of raw ET samples for each user is represented by a 2D array (see Figure 2A), where the rows are the individual samples collected at 120Hz. Each sample is a 6-dimensional vector consisting of the gaze coordinates (Gx, Gy), the distance (HD) of the left and right eyes from the screen (used to estimate the head's distance from the screen), and the sizes of the left and right pupils (P). Table 1 shows the statistics for the dataset, including the number of datapoints for each task



Figure 1: Tasks used for collecting the AD dataset – (A) Pupil Calibration, (B) Picture Description, and (C) Reading.

Task	Group	Ν	Mean (Std. dev.)	Median	Min	Max
Pupil Calibration	Patient	66	1461 (244)	1405	1151	2317
	Control	71	1369 (253)	1362	106	2049
	Total	137	1403 (249)	1377	106	2317
Picture Description	Patient	67	7906 (4609)	6488	1879	21861
	Control	73	7974 (4659)	7149	950	26103
	Total	140	7948 (4626)	6857	950	26103
Reading	Patient	67	8070 (3650)	7265	375	20712
	Control	73	6459 (1661)	6476	1216	12066
	Total	140	7080 (2719)	6623	375	20712

Table 1: Summary statistics of sequence lengths in the AD dataset

and condition (patients and controls), as well as statistics on the length of the sequences¹.

4 Data preprocessing

The average sequence lengths for the three tasks are 1,403 for Pupil Calibration (std. dev.=249), 7,948 for Picture Description (std. dev.=4626), and 7,080 for Reading (std. dev.=2719), thus they are well above the length of 1,000 timesteps that is known to be suitable for LSTM-based models [27]. To address this issue, we adopted two data preprocessing steps. First, we cyclically split the ET sequences, as was done in [39]. The cyclical splitting process creates four separate datapoints from each original datapoint by assigning samples that are four steps apart to the same new datapoint in a cyclical manner (see Figure 2B). This process preserves the temporal structure of the ET data because there is little change between contiguous samples due to the

Time			Left	eye	Rig	ht ey	е							
(ms) 			[]	$\overline{}$		L								
$\overline{}$	Gх	Gу	HD	Р	HD	Р		0	628.8	398.8	636.0	2.96	632.9	2.89
0	628.8	398.8	636.0	2.96	632.9	2.89		32	616.5	400.3	635.9	2.98	632.9	2.90
8	626.8	408.2	635.9	2.98	632.9	2.89		:	:	720.0	:	2.02	420.1	2.03
14	422.2	411.0	425.0	2.01	422.0	2.07		6040	710.9	739.9	634.1	3.02	630.1	3.03
10	023.3	411.0	035.9	3.01	032.9	2.00		8	626.8	408.2	635.9	2.98	632.9	2.89
24	614.3	405.6	635.9	2.99	632.9	2.89		49	624.0	398.7	635.9	3.00	632.9	2.92
32	616.5	400.3	635.9	2.98	632.9	2.90			:	1	:	1	1	1
49	624.0	398.7	635.9	3.00	632.9	2.92		6048	708.1	737.3	629.0	3.01	624.0	3.02
48	628.8	397.1	635.9	3.00	632.9	2.96	→	16	623.3	411.0	635.9	3.01	632.9	2.88
56	626.8	396.1	635.9	2.99	632.9	2.93		48	628.8	397.1	635.9	3.00	632.9	2.96
	:	:	:	:	:	:			:	:	:	1	-	-
6040	710.9	739.9	634.1	3.02	630.1	3.03		6056	707.9	735.9	631.3	3.02	621.1	3.01
(0.40	700.4	707.0	(00.0	2.04	(04.0	2.00		24	614.3	405.6	635.9	2.99	632.9	2.89
6048	708.1	/3/.3	629.0	3.01	624.0	3.02		56	626.8	396.1	635.9	2.99	632.9	2.93
6056	707.9	735.9	631.3	3.02	621.1	3.01		-	:	1	1	1	1	1
6064	711.9	739.9	633.5	3.01	623.5	3.02		6064	711.9	739.9	633.5	3.01	623.5	3.02
			(A)								(B)			

Figure 2: (A) An example of a datapoint, which is a sequence of ET samples (rows) from a given user. (B) The four distinct datapoints obtained from the datapoint in (A) through cyclical splitting.

high sampling rate while reducing the sequence length by a factor of 4. Additionally, the number of datapoints is increased by the same factor, as a form of data augmentation.

After cyclical splitting, the length of the sequences in the Pupil Calibration task is well below 1,000 (see the "Max" column for Pupil Calibration in Table 1, where the value should be divided by 4). However, this is not the case for the other two tasks. Hence, we experimented with applying a length cutoff to the sequences obtained from cyclical splitting to restrict their maximum length. We chose two cutoff values: the first cutoff value is 1,000, as it ensures that the maximum length of sequences never exceeds the threshold that is typically considered challenging for LSTM models [27]. The second cutoff value is 2000, to examine the effects of a less severe reduction in sequence length by including almost complete information from sequences with mean lengths. Figure 3 shows the distribution of sequence lengths after applying the 2,000-cutoff, which leads to 51 patients and 58 controls for the Picture Description task, and 57 patients and 70 controls for the Reading task having sequences above 1,000 timesteps.



Figure 3: Distribution (Y axis) of sequence lengths (X axis) after applying the 2000 cutoff, for the Picture Description (top) and Reading task (bottom).

 $^{^1\,\}rm These$ are obtained after removing outliers that are 3 std. dev. away from the mean from each task (7 for Pupil Calibration, 4 for Picture Description, and 4 for Reading).



Figure 4: The VTNet architecture.

The application of the cutoff resulted in two variations of the Picture Description and Reading tasks' datasets, with each having a maximum sequence length of, respectively, 1,000 and 2,000 timesteps. We will use these two variations, as well as the dataset with no cutoff applied (all with cyclical splitting) to evaluate the performance of VTNet in distinguishing between patients and controls, as described in the next section.

5 VTNet architecture

The VTNet model architecture was first introduced in [39] and is presented in Figure 4. The model consists of a single-layer GRU sub-model and a two-layer CNN sub-model. The GRU and CNN sub-models operate independently. The GRU processes the sequences of raw ET samples, the CNN processes the corresponding spatial representation, namely the scanpath, that shows where fixations happened and the transitions between them (see as an example the input image to the CNN in Figure 4). The output of the GRU's 256-unit hidden state is concatenated with the 50-element vector output of the CNN to produce a single 306-sized vector. This vector is then passed to a simple neural network with one hidden layer and a SoftMax layer, which generates two outputs that indicate the model's confidence in classifying the input as either AD or control.

The VTNet hyperparameters used for this work are the same as in the original work [39], which discusses how this architecture was designed to be as simple as possible to deal with the limited size typical of ET datasets. The model is trained endto-end as a single entity.

6 Evaluation of VTNet

6.1 Experimental Setup

Our evaluation aims to ascertain how VTNet compares to the best-performing non-DL classifiers from [20] in distinguishing between patients and controls. Therefore, following [20], we evaluate VTNet separately on each of the three tasks (Pupil Calibration, Picture Description, and Reading). For Pupil Calibration, VTNet is evaluated only on the full sequences since the length of the sequences here is less than 1,000 timesteps, as explained in Section 4. For the other two tasks, VTNet is evaluated on the full sequences, as well as on the sequences obtained by applying the length cutoffs of 1,000 and 2,000 timesteps. Hence, we label these three different VTNet models as *VTNet_full, VTNet_1000*, and *VTNet_2000* respectively.

For each task, the performance of the various VTNet models is compared against the best performing model among the non-DL classifiers tested in [20] (called baseline models from now on). It should be noted that the current AD dataset (described in Section 3) is larger than the version used [20] because participant recruitment is ongoing. Thus, we re-trained the non-DL models tested in [20] (Gaussian Naïve Bayes, Random Forest, Logistic Regression) on the current dataset, and selected the best performing model in each task (reported in Table 2) as a baseline for comparison with the VTNet models. All models are evaluated using 10 runs of 10-fold cross-validation (CV), and the results reported in the next section are the average of the 10 runs of 10fold CV. Cross-validation is done across users, ensuring that no user contributes data points to both the training and test sets of a given fold. Cross-validation is also stratified so that the distribution of data points in each fold is kept similar to that of the dataset. For the non-DL models, we use the same hyperparameters as in [20] and we report the same performance metrics, which include:

1. **AUC** (Area Under Curve), which measures the accuracy of the classifier in distinguishing between patients and controls.

2. **Sensitivity** (or true positive rate), indicating the model's ability to detect patients.

3. **Specificity** (or true negative rate), indicating the model's ability to detect controls.

While AUC provides an overall performance measure, sensitivity and specificity are important in medical applications to estimate the likelihood of false negatives and false positives.

We formally compare model performances in each task by running a one-way MANOVA test with classifier type as the factor and the three performance metrics as dependent variables. Post-hoc comparisons are done with Tukey's HSD tests and statistical significance is reported for p < 0.05.

6.2 Results

Table 2 summarizes the performance of all tested models in each task. In Table 2, bold indicates the model with the highest numerical performance, whereas an asterisk indicates whether a specific VTNet model is statistically significantly better than the baseline (best performing) non-DL model. Table 3 summarizes the post-hoc comparisons where the differences for models with different underlines are statistically significant (e.g., Sensitivity of Baseline vs VTNet_full in the Pupil Calibration task), whereas differences for models with the same underline are not (e.g., AUC of Baseline vs VTNet_full in the Pupil Calibration task).

Table 2: Performance of VTNet models trained on ET sequences with different maximum lengths and the corresponding best performing non-DL models, for each task.

Task	Classifier Type	AUC	Sensitivity	Specificity	
		Mean (std. dev.)	Mean (std. dev.)	Mean (std. dev.)	
Pupil Calibration	Gaussian Naïve Bayes	0.71 (0.02)	0.72 (0.02)	0.57 (0.04)	
	VTNet_full	0.70 (0.01)	0.64 (0.01)	0.74 (0.02) *	
Picture Description	Random Forest	0.75 (0.01)	0.65 (0.03)	0.72 (0.03)	
	VTNet_1000	0.67 (0.01)	0.65 (0.02)	0.66(0.01)	
	VTNet_2000	0.63 (0.01)	0.62 (0.02)	0.63 (0.02)	
	VTNet_full	0.58 (0.01)	0.54 (0.02)	0.66 (0.02)	
Reading	Logistic Regression	0.74 (0.03)	0.59 (0.03)	0.77 (0.01)	
	VTNet_1000	0.62 (0.01)	0.63 (0.02) *	0.63 (0.04)	
	VTNet_2000	0.73 (0.01)	0.66 (0.01) *	0.74 (0.01)	
	VTNet_full	0.75 (0.01)	0.68 (0.01) *	0.78 (0.01)	

For the Pupil Calibration task, the MANOVA shows a significant effect of the classifier on both sensitivity and specificity (for sensitivity: $F_{1,18}=136.774$, p<.001, partial $\eta 2=.884$; for specificity: $F_{1,18}=192.823$, p<.001, partial $\eta 2=.915$). Post-hoc comparisons (Table 3A) confirm that the baseline has higher sensitivity than VTNet, whereas VTNet has a higher specificity, with no difference in AUC scores.

For the Picture Description task, the MANOVA shows a significant effect of the classifier on all three performance metrics (for AUC: $F_{3,36}$ =676.844, p<0.001, partial η 2=.983; for sensitivity: $F_{3,36}$ =58.983, p<.001, partial η 2=.831; for specificity:

Table 3: Statistical comparisons of models' performances with Tukey's HSD. Differences for models with the same underlines are not statistically significant, whereas differences for models with different underlines are.

A) Pupil Calibration				
AUC	Baseline > VTNet full			
Sensitivity	Baseline > VTNet_full			
Specificity	<u>VTNet_full</u> > Baseline			
B) Picture D	escription			
AUC	<u>Baseline</u> > <u>VTNet_1000</u> > <u>VTNet_2000</u> > <u>VTNet_full</u>			
Sensitivity	<u>Baseline > VTNet 1000</u> > <u>VTNet 2000</u> > <u>VTNet full</u>			
Specificity	<u>Baseline</u> > <u>VTNet 1000 > VTNet full</u> > <u>VTNet 2000</u>			
C) Reading				
AUC	<u>VTNet_full > Baseline > VTNet_2000</u> > <u>VTNet_1000</u>			
Sensitivity	<u>VTNet_full</u> > <u>VTNet_2000</u> > <u>VTNet_1000</u> > <u>Baseline</u>			
Specificity	<u>VTNet full > Baseline</u> > <u>VTNet 2000</u> > <u>VTNet 1000</u>			

F_{3,36}=25.096, p<.001, partial η 2=.677). Post-hoc comparisons (Table 3B) show that the baseline beats all VTNet models in terms of AUC and specificity. For sensitivity, the baseline and VTNet_1000 have equivalent performance and they outperform the other two VTNet models.

For the Reading task, the MANOVA shows a significant effect of the classifier on all three performance metrics (AUC: $F_{3,36}=125.352$, p<0.001, partial η 2=.913; sensitivity: $F_{3,36}=51.244$, p<.001, partial η 2=.810; specificity: $F_{3,36}=91.483$, p<.001, partial η 2=.884). The post-hoc comparisons (Table 3C) show that all three VTNet models outperform the baseline in terms of sensitivity, with VTNet_full being the winning model. VTNet_full and the baseline are equivalent in specificity, and they outperform the other two VTNet models. For AUC, VTNet_full, VTNet_2000, and the baseline have equivalent performance and they outperform VTNet_1000.

6.3 Discussion

Based on overall performance (AUC) scores, no VTNet model outperforms the baseline models in any task. We hypothesized as a reason for this result that, despite the undertaken preprocessing steps, the sequences were still too long for the GRU sub-model to process. It is, however, interesting to observe the different trends between the Picture Description and the Reading tasks in terms of VTNet performance with different sequence lengths.

For the Picture Description task, the shorter the sequences the better, with VTNet_1000 outperforming VTNet_2000, which in turns outperforms VTNet_full, in both AUC and sensitivity. We observe the opposite trend in the Reading task. In terms of AUC, there is a non-significant trend of VTNet_full being better than VTNet_2000, which in turn is significantly better than VTNet_1000. There are similar but stronger trajectories for specificity and sensitivity, where the differences are statistically significant. These opposite trends suggest that in the Reading task, behaviors happen toward the end of the task that help distinguish between patients and control. For instance, it might be the case that as patients progress further in the paragraph, their reading impairments become increasingly evident, resulting in more discriminative ET behaviors that are captured

partly with VTNet_2000 and fully with VTNet_full. In contrast, somehow the differences in how patients and controls visually process the Cookie Theft picture may get diluted as the task progresses, thus diminishing the ability of VTNet to discriminate between the two groups when looking at longer sequences. Further clarity on this point could be achieved by doing an offline analysis of gaze patterns at the end of sequences for both tasks. Given that none of our VTNet models outperform the baseline for AUC, we investigate if we can improve their performance by adding an attention layer, as discussed in the next section.

7 Adding an attention layer to VTNet

An attention layer computes the dot products of input sequences and learned weight vectors, producing attention scores that are normalized and used to weight the input sequences. As discussed in Section 2, there is evidence from both NLP and Computer Vision research that adding an attention layer to an LSTM-based model enables the model to focus on the most relevant parts of an input sequence, thus allowing it to capture long-term dependencies more effectively. Hence, in this section, we explore adding a self-attention layer before the GRU sub-model in VTNet. To implement this self-attention layer, we utilized PyTorch's (v1.13.0+cu117) default multi-head attention layer implementation. The dimension of this layer was set to 6 to match the dimensionality of our gaze data (see Figure 2A). The number of parallel attention heads was set to 1. This is because increasing the number of parallel attention heads increases the number of trainable parameters, which can result in overfitting when the dataset is small, as is the case in our work. Moreover, a smaller number of attention heads can also reduce the computational complexity of the model, resulting in more efficient training and evaluation.

7.1 Experimental Setup

To ascertain the effectiveness of augmenting the VTNet architecture with attention, we compare the augmented VTNet models against the original VTNet models and the non-DL baselines from [20]. As we did in Section 6, we perform this comparison for each of the three experimental tasks. For each task, we select the VTNet model that performed the best in the evaluation in Section 6, namely, VTNet_full for Pupil Calibration and Reading, and VTNet_1000 for Picture Description. These models augmented with attention are denoted with the suffix "_att" in the following sections. For example, *VTNet_2000_att* refers to the VTNet model with attention trained on the dataset with a maximum sequence length of 2,000 timesteps. The evaluation process is similar to that described in Section 6.1, where we utilize a one-way MANOVA test with classifier type as the factor and the three performance metrics as dependent variables to compare the relevant models. Similarly, post-hoc comparisons are performed using Tukey's HSD tests and statistical significance is reported for p < 0.05.

7.2 Results

Table 4 summarizes the results of this analysis. For the Pupil Calibration task, the MANOVA shows a significant effect on all three performance metrics (AUC: $F_{2,27}$ =137.134, p<0.001, partial η 2=.910; sensitivity: $F_{2,27}$ =49.424, p<.001, partial η 2=.785; specificity: $F_{2,27}$ =184.747, p<.001, partial η 2=.932). Post-hoc comparisons (Table 5A), show a substantial improvement in performance with VTNet_full_att. This model now beats the baseline with an AUC of 0.78, which is a 9.8% increase (whereas VTNet_full is equivalent to the baseline), For sensitivity, VTNet_full_att is equivalent to the baseline (whereas VTNet_full is worse). For specificity, VTNet_full_att matches the performance of VTNet_full, which already beats the baseline. Furthermore, VTNet_full_att is a very balanced classifier, with 0.71 sensitivity, and 0.75 specificity (whereas VTNet_full has much better specificity than sensitivity with a difference of 10%).

For the Picture Description task, the MANOVA shows a significant effect on all three performance metrics (AUC: F_{2,27}=192.702, p<0.001, partial η 2=.935; sensitivity: F_{2,27}=27.557, p<.001, partial η 2=.671; specificity: F_{2,27}=27.276, p<.001, partial η 2=.669). Post-hoc comparisons (Table 5B) show that VTNet_1000_att outperforms the baseline for both AUC and sensitivity (while VTNet_1000 is either worse or equivalent), with sensitivity being especially impacted by reaching 0.7, which is a 7.7% increase from the baseline. For specificity,

Task	Classifier Type	AUC	Sensitivity	Specificity
		Mean (std. dev.)	Mean (std. dev.)	Mean (std. dev.)
Pupil Calibration	Gaussian Naïve Bayes	0.71 (0.02)	0.72 (0.02)	0.57 (0.04)
	VTNet_full	0.70 (0.01)	0.64 (0.01)	0.74 (0.02) *
	VTNet_full_att	0.78 (0.01) *	0.71 (0.02)	0.75 (0.01) *
Picture Description	Random Forest	0.75 (0.01)	0.65 (0.03)	0.72 (0.03)
	VTNet_1000	0.67 (0.01)	0.65 (0.02)	0.66 (0.01)
	VTNet_1000_att	0.76 (0.01) *	0.70 (0.02) *	0.73 (0.02)
Reading	Logistic Regression	0.74 (0.03)	0.59 (0.03)	0.77 (0.01)
	VTNet_full	0.75 (0.01)	0.68 (0.01) *	0.78 (0.01)
	VTNet_full_att	0.78 (0.01) *	0.70 (0.01) *	0.80 (0.02) *

Table 4: Performance of the most promising VTNet model, its corresponding attention variant, and the baseline non-DL model, for each task.

A) Pupil Calibration				
AUC	VTNet_full_att > Baseline > VTNet_full			
Sensitivity Specificity	<u>Baseline > VTNet_full_att</u> > <u>VTNet_full</u> <u>VTNet_full_att > VTNet_full</u> > <u>Baseline</u>			
B) Picture Des	scription			
AUC	VTNet_1000_att > Baseline > VTNet 1000			
Sensitivity	<u>VTNet_1000_att</u> > <u>Baseline > VTNet_1000</u>			
Specificity	<u>VTNet 1000 att > Baseline</u> > <u>VTNet 1000</u>			
C) Reading				
AUC	<u>VTNet_full_att</u> > <u>VTNet_full</u> > <u>Baseline</u>			
Sensitivity	<u>VTNet_full_att > VTNet_full</u> > <u>Baseline</u>			
Specificity	VTNet_full_att > VTNet_full > Baseline			

Table 5: Statistical comparisons of models' performanceswith Tukey's HSD.

VTNet_1000_att is equivalent to the baseline where VTNet_1000 is worse. As was the case for the Pupil Calibration task, VTNet_1000_att is also balanced with 0.70 sensitivity and 0.73 specificity. In this task, VTNet_1000 is also balanced but with limited accuracies for both measures (0.65 and 0.66).

For the Reading task, the MANOVA shows a significant effect on all three performance metrics (AUC: $F_{2,27}$ =10.484, p<0.001, partial η 2=.437; sensitivity: $F_{2,27}$ =113.574, p<.001, partial η 2=.894; specificity: $F_{2,27}$ =22.288, p<.001, partial η 2=.623). The post-hoc comparisons (Table 5C) show that VTNet_full_att beats the baseline for all measures, and it is either better than (AUC and specificity) or equivalent (sensitivity) to VTNet_full. Interestingly, with 0.70 sensitivity and 0.80 specificity, this VTNet_full_att classifier is not as balanced as its counterparts in the Pupil Calibration and Picture Description tasks. This imbalance is due to a much higher specificity (0.80 compared to 0.75 in Pupil Calibration and 0.73 in Picture Description), whereas sensitivity is around 0.70-0.71 for all three models, showing that the attention layer mostly improves the model's ability to correctly classify control participants.

7.3 Discussion

Our results show that the addition of the attention layer enables the VTNet architecture to outperform the baseline models in all metrics for all tasks, except for sensitivity in the Pupil Calibration task where there is no statistical difference. These results indicate that the attention mechanism enables the GRU sub-model to better focus on critical parts of the ET sequences, despite their length.

One interesting question is whether adding the self-attention layer enhances VTNet's performance regardless of sequence length. To answer this question, we experimented with using VTNet augmented with attention on the confusion dataset from [39], where the ET sequences have a maximum length of 150 timesteps. We found that, with this confusion ET dataset, the VTNet models with and without attention had statistically equivalent performances on all metrics, suggesting that the addition of the attention layer to this architecture is not advantageous when the ET sequences are of manageable length. This is different than what is observed in NLP tasks, where attention helps even with sequences no longer than 120 tokens [6,38,48]. This difference could be due to a variety of reasons, including the nature of the classification task, type of data, and amount of information captured at any time step (e.g., a word arguably has higher information content than an individual raw gaze sample at a particular timestep), calling for further investigation on the relationship between all these factors, sequence lengths, and attention effectiveness.

8 Conclusions and Future Work

In this paper, we investigated whether VTNet, a model originally developed to classify user confusion from their eyetracking (ET) data by processing in parallel a visual and temporal representation of the data, can also improve on the state-of-theart results in classifying Alzheimer's Disease (AD). We addressed the challenge of long ET sequences by combining targeted length reduction with the addition of an attention layer to VTNet and showed that the results outperform the state-of-the-art for AD classification with ET data.

Our work has two contributions: first, the development of more accurate and lightweight classifiers for AD, and second, initial evidence of the generalizability of VTNet for leveraging ET data in different classification tasks.

Moving forward, we plan to experiment with building ensemble classifiers that combine VTNet for ET data and classifiers leveraging language data available in the AD dataset, as was done in Jang et al. [20] with non-DL models. We are especially interested in ascertaining if VTNet can be used to classify AD from speech data in the AD dataset by processing, in parallel, raw speech signals and their corresponding spectrograms, as is done for ET data. Additionally, we plan on testing VTNet on other ET datasets that have been used to predict user states such as learning [22] and affective valence [25].

REFERENCES

- Sarker Monojit Asish, Arun K. Kulshreshth, and Christoph W. Borst. 2022. Detecting distracted students in educational VR environments using machine learning on eye gaze data. *Computers & Graphics* 109, (December 2022), 75–87. DOI:https://doi.org/10.1016/j.cag.2022.10.007
- [2] Mohammad Ehsan Basiri, Shahla Nemati, Moloud Abdar, Erik Cambria, and U. Rajendra Acharya. 2021. ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for sentiment analysis. *Future Generation Computer Systems* 115, (February 2021), 279–294. DOI:https://doi.org/10.1016/j.future.2020.08.005
- Juan Biondi, Gerardo Fernandez, Silvia Castro, and Osvaldo Agamennoni. 2018. Eye-Movement behavior identification for AD diagnosis. DOI:https://doi.org/10.48550/arXiv.1702.00837
- [4] Robert Bixler and Sidney D'Mello. 2015. Automatic Gaze-Based Detection of Mind Wandering with Metacognitive Awareness. In User Modeling, Adaptation and Personalization (Lecture Notes in Computer Science), Springer International Publishing, Cham, 31–43. DOI:https://doi.org/10.1007/978-3-319-20267-9_3
- [5] Sean Anthony Byrne, Adam Peter Frederick Reynolds, Carolina Biliotti, Falco J. Bargagli-Stoffi, Luca Polonio, and Massimo Riccaboni. 2023. Predicting choice behaviour in economic games using gaze data encoded as scanpath images. *Sci Rep* 13, 1 (March 2023), 4722. DOI:https://doi.org/10.1038/s41598-023-31536-5

- [6] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-Based Models for Speech Recognition. In Advances in Neural Information Processing Systems, Curran Associates, Inc. Retrieved April 3, 2023 from https://papers.nips.cc/paper_files/paper/2015/hash/1068c6e4c8051cfd4e9ea 8072e3189e2-Abstract.html
- [7] Cyndy B. Cordell, Soo Borson, Malaz Boustani, Joshua Chodosh, David Reuben, Joe Verghese, William Thies, Leslie B. Fried, and Medicare Detection of Cognitive Impairment Workgroup. 2013. Alzheimer's Association recommendations for operationalizing the detection of cognitive impairment during the Medicare Annual Wellness Visit in a primary care setting. Alzheimer's & Dementia 9, 2 (2013), 141–150. DOI:https://doi.org/10.1016/j.jalz.2012.09.011
- [8] Louise Cummings. 2019. Describing the Cookie Theft picture: Sources of breakdown in Alzheimer's dementia. *Pragmatics and Society* 10, 2 (July 2019), 153–176. DOI:https://doi.org/10.1075/ps.17011.cum
- [9] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2978–2988. DOI:https://doi.org/10.18653/v1/P19-1285
- [10] Kirsten A. Dalrymple, Ming Jiang, Qi Zhao, and Jed T. Elison. 2019. Machine learning accurately classifies age of toddlers based on eye tracking. *Sci Rep* 9, 1 (April 2019), 6255. DOI:https://doi.org/10.1038/s41598-019-42764-z
- [11] Didan Deng, Zhaokang Chen, and Bertram E. Shi. 2020. Multitask Emotion Recognition with Incomplete Labels. In 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), 592–599. DOI:https://doi.org/10.1109/FG47880.2020.00131
- [12] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. 2625–2634. Retrieved November 2, 2022 from https://openaccess.thecvf.com/content_cvpr_2015/html/Donahue_Long-Term_Recurrent_Convolutional_2015_CVPR_paper.html
- [13] Hazel Doughty, Walterio Mayol-Cuevas, and Dima Damen. 2019. The Pros and Cons: Rank-Aware Temporal Attention for Skill Determination in Long Videos. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 7854–7863. DOI:https://doi.org/10.1109/CVPR.2019.00805
- [14] Kathleen C. Fraser, Jed A. Meltzer, and Frank Rudzicz. 2016. Linguistic Features Identify Alzheimer's Disease in Narrative Speech. Journal of Alzheimer's Disease 49, 2 (January 2016), 407–422. DOI:https://doi.org/10.3233/JAD-150520
- [15] Siobhan Garbutt, Alisa Matlin, Joanna Hellmuth, Ana K. Schenk, Julene K. Johnson, Howard Rosen, David Dean, Joel Kramer, John Neuhaus, Bruce L. Miller, Stephen G. Lisberger, and Adam L. Boxer. 2008. Oculomotor function in frontotemporal lobar degeneration, related disorders and Alzheimer's disease. *Brain* 131, Pt 5 (May 2008), 1268–1281. DOI:https://doi.org/10.1093/brain/awn047
- [16] Harold Goodglass and Edith Kaplan. 1983. The assessment of aphasia and related disorders (2nd ed ed.). Lea & Febiger, Philadelphia.
- [17] Eric L. Granholm, Matthew S. Panizzon, Jeremy A. Elman, Amy J. Jak, Richard L. Hauger, Mark W. Bondi, Michael J. Lyons, Carol E. Franz, and William S. Kremen. 2017. Pupillary Responses as a Biomarker of Early Risk for Alzheimer's Disease. J Alzheimers Dis 56, 4 (2017), 1419–1428. DOI:https://doi.org/10.3233/JAD-161078
- [18] Jingqiu Guo, Yangzexi Liu, Qingyan (Ken) Yang, Yibing Wang, and Shouen Fang. 2021. GPS-based citywide traffic congestion forecasting using CNN-RNN and C3D hybrid model. *Transportmetrica A: Transport Science* 17, 2 (January 2021), 190–211. DOI:https://doi.org/10.1080/23249935.2020.1745927
- [19] Gaurav Gupta, Meghana Kshirsagar, Ming Zhong, Shahrzad Gholami, and Juan Lavista Ferres. 2021. Comparing recurrent convolutional neural networks for large scale bird species classification. *Sci Rep* 11, 1 (August 2021), 17085. DOI:https://doi.org/10.1038/s41598-021-96446-w
- [20] Hyeju Jang, Thomas Soroski, Matteo Rizzo, Oswald Barral, Anuj Harisinghani, Sally Newton-Mason, Saffrin Granby, Thiago Monnerat Stutz da Cunha Vasco, Caitlin Lewis, Pavan Tutt, Giuseppe Carenini, Cristina Conati, and Thalia S. Field. 2021. Classification of Alzheimer's Disease Leveraging Multi-task Machine Learning Analysis of Speech and Eye-Movement Data. Frontiers in Human Neuroscience 15, (2021). Retrieved April 1, 2023 from https://www.frontiersin.org/articles/10.3389/fnhum.2021.716670
- [21] Zhong Ji, Kailin Xiong, Yanwei Pang, and Xuelong Li. 2020. Video Summarization With Attention-Based Encoder–Decoder Networks. *IEEE Transactions on Circuits and Systems for Video Technology* 30, 6 (June 2020), 1709–1717. DOI:https://doi.org/10.1109/TCSVT.2019.2904996

- [22] Samad Kardan and Cristina Conati. 2015. Providing Adaptive Support in an Interactive Simulation for Learning: An Experimental Evaluation. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15), Association for Computing Machinery, New York, NY, USA, 3671–3680. DOI:https://doi.org/10.1145/2702123.2702424
- [23] Sweta Karlekar, Tong Niu, and Mohit Bansal. 2018. Detecting Linguistic Characteristics of Alzheimer's Dementia by Interpreting Neural Models. DOI:https://doi.org/10.48550/arXiv.1804.06440
- [24] Weirui Kong, Hyeju Jang, Giuseppe Carenini, and Thalia Field. 2019. A Neural Model for Predicting Dementia from Language. In Proceedings of the 4th Machine Learning for Healthcare Conference, PMLR, 270–286. Retrieved April 12, 2023 from https://proceedings.mlr.press/v106/kong19a.html
- [25] Sébastien Lallé, Cristina Conati, and Roger Azevedo. 2018. Prediction of Student Achievement Goals and Emotion Valence during Interaction with Pedagogical Agents. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS '18), International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1222–1231.
- [26] Sébastien Lallé, Cristina Conati, and Giuseppe Carenini. 2016. Predicting confusion in information visualization from eye tracking and interaction data. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16), AAAI Press, New York, New York, USA, 2529–2535.
- [27] Shuai Li, Wanqing Li, Chris Cook, Ce Zhu, and Yanbo Gao. 2018. Independently Recurrent Neural Network (IndRNN): Building a Longer and Deeper RNN. 5457–5466. Retrieved April 10, 2023 from https://openaccess.thecvf.com/content_cvpr_2018/html/Li_Independently_ Recurrent_Neural_CVPR_2018_paper.html
- [28] Xiang Li, Dawei Song, Peng Zhang, Guangliang Yu, Yuexian Hou, and Bin Hu. 2016. Emotion recognition from multi-channel EEG data through Convolutional Recurrent Neural Network. In 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 352–359. DOI:https://doi.org/10.1109/BIBM.2016.7822545
- [29] Gang Liu and Jiabao Guo. 2019. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* 337, (April 2019), 325–338.
 DOI:https://doi.org/10.1016/j.neucom.2019.01.078
- [30] Justin Le Louedec, Thomas Guntz, James L. Crowley, and Dominique Vaufreydaz. 2019. Deep learning investigation for chess player attention prediction using eye-tracking and game data. In Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications (ETRA '19), Association for Computing Machinery, New York, NY, USA, 1–9. DOI:https://doi.org/10.1145/3314111.3319827
- [31] Michael R. MacAskill and Tim J. Anderson. 2016. Eye movements in neurodegenerative diseases. Curr Opin Neurol 29, 1 (February 2016), 61–68. DOI:https://doi.org/10.1097/WCO.00000000000274
- [32] Robert J. Molitor, Philip C. Ko, and Brandon A. Ally. 2015. Eye movements in Alzheimer's disease. J Alzheimers Dis 44, 1 (2015), 1–12. DOI:https://doi.org/10.3233/JAD-141173
- [33] Kiyotaka Nakamagoe, Shiori Yamada, Rio Kawakami, Tadachika Koganezawa, and Akira Tamaoka. 2019. Abnormal Saccadic Intrusions with Alzheimer's Disease in Darkness. *Current Alzheimer Research* 16, 4 (April 2019), 293–301. DOI:https://doi.org/10.2174/1567205016666190311102130
- [34] Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. How to Construct Deep Recurrent Neural Networks. DOI:https://doi.org/10.48550/arXiv.1312.6026
- [35] Ivanna M. Pavisić, Nicholas C. Firth, Samuel Parsons, David Martinez Rego, Timothy J. Shakespeare, Keir X. X. Yong, Catherine F. Slattery, Ross W. Paterson, Alexander J. M. Foulkes, Kirsty Macpherson, Amelia M. Carton, Daniel C. Alexander, John Shawe-Taylor, Nick C. Fox, Jonathan M. Schott, Sebastian J. Crutch, and Silvia Primativo. 2017. Eyetracking Metrics in Young Onset Alzheimer's Disease: A Window into Cognitive Visual Functions. Front Neurol 8, (August 2017), 377. DOI:https://doi.org/10.3389/fneur.2017.00377
- [36] Guido Pusiol, Andre Esteva, Scott S. Hall, Michael Frank, Arnold Milstein, and Li Fei-Fei. 2016. Vision-Based Classification of Developmental Disorders Using Eye-Movements. In Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016 (Lecture Notes in Computer Science), Springer International Publishing, Cham, 317–325. DOI:https://doi.org/10.1007/978-3-319-46723-8_37
- [37] Shagan Sah, Sourabh Kulhare, Allison Gray, Subhashini Venugopalan, Emily Prud'Hommeaux, and Raymond Ptucha. 2017. Semantic Text Summarization of Long Videos. In 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), 989–997. DOI:https://doi.org/10.1109/WACV.2017.115
- [38] Sheng-syun Shen and Hung-Yi Lee. 2016. Neural Attention Models for Sequence Classification: Analysis and Application to Key Term Extraction

and Dialogue Act Detection. In Interspeech 2016, ISCA, 2716-2720. DOI:https://doi.org/10.21437/Interspeech.2016-1359

- [39] Shane D. Sims and Cristina Conati. 2020. A Neural Architecture for Detecting User Confusion in Eye-tracking Data. In Proceedings of the 2020 International Conference on Multimodal Interaction (ICMI '20), Association for Computing Machinery, New York, NY, USA, 15–23. DOI:https://doi.org/10.1145/3382507.3418828
- [40] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. 2015. Unsupervised Learning of Video Representations using LSTMs. In Proceedings of the 32nd International Conference on Machine Learning, PMLR, 843–852. Retrieved November 2, 2022 from https://proceedings.mlr.press/v37/srivastava15.html
- [41] Ben Steichen, Cristina Conati, and Giuseppe Carenini. 2014. Inferring Visualization Task Properties, User Performance, and User Cognitive Abilities from Eye Gaze Data. ACM Trans. Interact. Intell. Syst. 4, 2 (July 2014), 11:1-11:29. DOI:https://doi.org/10.1145/2633043
- [42] Susame Trauzettel-Klosinski, Klaus Dietz, and the IReST Study Group. 2012. Standardized Assessment of Reading Performance: The New International Reading Speed Texts IReST. Investigative Ophthalmology & Visual Science 53, 9 (August 2012), 5452–5461. DOI:https://doi.org/10.1167/iovs.11-8284
- [43] Lisa-Marie Vortmann, Jannes Knychalla, Sonja Annerer-Walcher, Mathias Benedek, and Felix Putze. 2021. Imaging Time Series of Eye Tracking Data to Classify Attentional States. Frontiers in Neuroscience 15, (2021). Retrieved April 6, 2023 from https://www.frontiersin.org/articles/10.3389/fnins.2021.664490
- [44] Jin Wang, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2016. Dimensional Sentiment Analysis Using a Regional CNN-LSTM Model. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Berlin, Germany, 225–230. DOI:https://doi.org/10.18653/v1/P16-2037
- [45] Weining Wang, Yan Huang, and Liang Wang. 2020. Long video question answering: A Matching-guided Attention Model. *Pattern Recognition* 102, (June 2020), 107248. DOI:https://doi.org/10.1016/j.patcog.2020.107248
- [46] Shuang Wen and Jian Li. 2018. Recurrent Convolutional Neural Network with Attention for Twitter and Yelp Sentiment Classification: ARC Model for Sentiment Classification. In Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence (ACAI 2018), Association for Computing Machinery, New York, NY, USA, 1–7. DOI:https://doi.org/10.1145/3302425.3302468
- [47] Yilong Yang, Qingfeng Wu, Ming Qiu, Yingdong Wang, and Xiaowei Chen. 2018. Emotion Recognition from Multi-Channel EEG through Parallel Convolutional Recurrent Neural Network. In 2018 International Joint Conference on Neural Networks (IJCNN), 1-7. DOI:https://doi.org/10.1109/IJCNN.2018.8489331
- [48] Bowen Yu, Zhenyu Zhang, Tingwen Liu, Bin Wang, Sujian Li, and Quangang Li. 2019. Beyond Word Attention: Using Segment Attention in Neural Relation Extraction. (2019), 5401–5407. Retrieved April 25, 2023 from https://www.ijcai.org/proceedings/2019/750
- [49] Haiyang Yu, Zhihai Wu, Shuqin Wang, Yunpeng Wang, and Xiaolei Ma. 2017. Spatiotemporal Recurrent Convolutional Networks for Traffic Prediction in Transportation Networks. Sensors 17, 7 (July 2017), 1501. DOI:https://doi.org/10.3390/s17071501
- [50] Dalin Zhang, Lina Yao, Xiang Zhang, Sen Wang, Weitong Chen, Robert Boots, and Boualem Benatallah. 2018. Cascade and Parallel Convolutional Recurrent Neural Networks on EEG-based Intention Recognition for Brain Computer Interface. Proceedings of the AAAI Conference on Artificial Intelligence 32, 1 (April 2018). DOI:https://doi.org/10.1609/aaai.v32i1.11496