Interspeech 2024
1-5 September 2024, Kos, Greece

Learning from memory-based models

Rhiannon Mogridge[1], *Anton Ragni*[1]

[1]Speech and Hearing Group, Dept. of Computer Science, University of Sheffield, UK

rmogridge1@sheffield.ac.uk, a.ragni@sheffield.ac.uk

Abstract

Recent work on the CPC2 speech intelligibility task shows promising results using an architecture inspired by memory models from the field of human psychology. This is surprising, given that previous work has shown memory models of this type to be inferior to parametric models, such as transformers, in most modern applications. This paper shows that the difference in performance is reduced or eliminated by using high quality features. Furthermore, we show for the first time that, despite being widely used in the field of human psychology and also for speech and language tasks, this model is a special case of a neural network. Experimental results from different tasks and datasets (CPC2/TIMIT/GoEmotions) confirm that this type of memory model is competitive with equivalently complex parametric models given sufficiently good feature representation, suggesting that high quality features may allow the use of simple, interpretable models without sacrificing performance.

Index Terms: memory models, speech intelligibility

1. Introduction

There is growing interest in incorporating memory into machine learning models, with promising performance demonstrated in fields as diverse as video object detection [1] and language modelling [2, 3, 4]. These memory-augmented models are typically based on purely parametric models, such as Transformers [5], with the memory component added as a small part of the whole. The key question of this paper is: is there a place for machine learning models with memory playing a more central role?

Historically, pure memory models such as k-nearest-neighbour and dynamic time warping have been used for speech and language tasks, but they typically scale poorly with data quantity, and are outperformed by modern, parametric alternatives. They do provide a new option, however: rather than take an existing parametric architecture and incorporate memory, we can take an existing memory model and parametrise it. This alternative option of building memory-based parametric models offers a number of advantages. First, memory-based models can, at least in theory, function with limited data and little or no training, rendering them particularly useful when resources are limited. Second, such models are typically interpretable, and it may be possible to retain this interpretability when adding parameters.

Recently, one such memory-based model showed promise at predicting speech intelligibility [6], ranking second in the 2023 Clarity Prediction Challenge 2 [7] and, as we will show later, fully capable of winning this competition. This result is surprising, given the historic poor-performance of memory-based models, leading us to investigate the possible reasons.

Theories of human cognition, on which such models are usually based, typically assume very good feature representation, which has historically not been the case for automated speech and language tasks. The recent work on powerful self-supervised feature representations, such as those derived from wav2vec [8], HuBERT [9], BERT and Whisper [10], may provide such a 'good' feature representation. Coincidentally, the aforementioned CPC2 model made use of intermediate features extracted from Whisper, which have been found to give good performance in a variety of tasks. Is the lack of good feature representation a reason for the previous poor performance of memory-based models?

In this work, we take a simple, memory-based model developed in the field of human psychology in 1984 [11], and examine its performance as we add parameters. This approach differs from the more common method of taking a parametric model and adding memory. We explore the effect of feature quality on the performance of the memory-based model and its parametric variant over three tasks: a regression task (speech intelligibility prediction); a classification task (frame-based phone classification); and a multi-label classification task (emotion classification of text). We show that the memory-based model explored here is a special case of a neural network, with parameters taken directly from the data without the need for training, which has not been previously recognised, despite being used either directly [12, 13, 14] or as inspiration [15] for machine learning tasks. We further show that a parametric version of this model can meet or exceed the performance of an equivalently complex parametric model, while remaining interpretable.

This paper makes the following contributions:

1. It shows that simple, interpretable, memory-based models with few trained parameters can be competitive with parametric models, given good feature representation. This fact has not been known before with many works stating the contrary.
2. It shows that a memory-based model that has been in use for four decades in a wide variety of fields [11, 16, 17] is in fact a special case of a neural network, which has not been previously recognised.
3. It shows that a simple memory-based model is capable of achieving state-of-the-art performance on the CPC2 speech intelligibility task.

All code is publicly available[1].

2. Minerva2

Minerva2 is a global memory model proposed by Hintzman [11] in 1984, created to test theories of human cognition. It has

[1]https://github.com/RhiM1/memory-models

10.21437/Interspeech.2024-1628

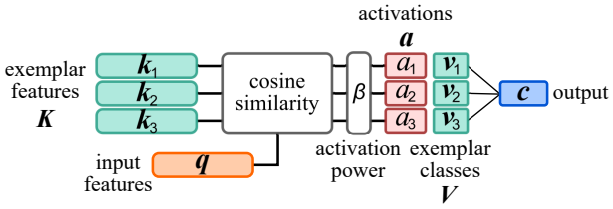

Figure 1: *Minerva 2.*

been widely used for decades in the field of human psychology [11, 16, 17], and has also been adapted for machine learning [12, 13, 14, 15]. It uses a memory composed of previous examples to label new examples. The mechanism by which a new example is labelled, shown in Figure 1, is a non-parametric form of attention [18]. Each example, or *exemplar*, is a tuple of a feature vector $\boldsymbol{k}_n$ and a class representation $\boldsymbol{v}_n$. These can be thought of as non-parametric forms of the keys and values in contemporary attention. The exemplars are packed into a feature matrix, $\boldsymbol{K} = \begin{bmatrix}\boldsymbol{k}_1 & \dots & \boldsymbol{k}_N\end{bmatrix}$, shown in green to the top left of Figure 1, and a class matrix, $\boldsymbol{V} = \begin{bmatrix}\boldsymbol{v}_1 & \dots & \boldsymbol{v}_N\end{bmatrix}$ shown in green towards the right of Figure 1. A new input, $\boldsymbol{q}$, is used directly as a non-parametric query. Let $\tilde{\boldsymbol{q}}$ be the L2-normalised $\boldsymbol{q}$, and $\tilde{\boldsymbol{K}}$ be a version of $\boldsymbol{K}$ where each exemplar (column vector) has been L2-normalised. The predicted class representation, $\boldsymbol{c}$, is given by,

$$\boldsymbol{c} = \boldsymbol{V}\boldsymbol{a}, \quad \text{where} \quad a_i = (\tilde{\boldsymbol{K}}^\top \tilde{\boldsymbol{q}})_i^\beta \tag{1}$$

The predicted class representation is a weighted sum of the exemplar class representations. Unlike conventional attention, there is clear segregation of the queries, keys and values, and the class representations are used directly as the values. Minerva2's form of attention has the benefit that every component has a clearly defined meaning. The downside is that, without any learned parameters, the model is entirely dependent on the feature and class vectors being truly representative.

Minerva2 is a regression model, but the originators of the model used it for classification by comparing the generated label with a 'true' class representation using cosine similarity. Let $\tilde{\boldsymbol{U}}$ be a matrix with column vectors holding the L2-normalised 'true' class representations. The predicted output class is,

$$\hat{y} = \underset{y \in \mathcal{Y}}{argmax}(h_y), \quad \text{where} \quad \boldsymbol{h} = \tilde{\boldsymbol{U}}^\top \tilde{\boldsymbol{c}} \tag{2}$$

where $\mathcal{Y}$ is the set of classes and $\hat{y}$ the predicted class. The 'true' class representations could take various forms, but in the absence of other information, one-hot representations of the class labels provide a simple orthogonal basis.

Minerva has similarities with other non-parametric techniques such as support vector machines (SVM), but as SVMs are not memory models, they fall outside the scope of this work.

2.1. Exemplar set size

Minerva2 works on the assumption that similar feature vectors likely share the same class, and identical feature vectors certainly share the same class. If these assumptions are true, then as the number of exemplars increases, so does the probability of there being an exemplar that is a) very similar to $\boldsymbol{q}$; and b) shares its class. A large exemplar set also includes more exemplars that are *slightly* similar to $\boldsymbol{q}$, but do not share its class. A large exemplar set therefore requires a higher value for the activation power β, which leaves similarities close to 1 (and -1) mostly unchanged, but reduces the magnitude of all other similarities. Provided that the assumptions hold, tuning β while increasing the exemplar set size will improve performance, although with diminishing returns.

2.2. Equivalence to neural networks

Memory-based models such as Minerva2 are often contrasted with neural networks, but we show here that they are closely related. The ith layer in a FFNN can be represented as,

$$\boldsymbol{h}^{(i)} = \boldsymbol{\phi}^{(2)}(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)}) \tag{3}$$

where the $\boldsymbol{W}^{(i)}$ are the layer weights, the $\boldsymbol{b}^{(i)}$ are the layer biases, and the $\boldsymbol{\phi}^{(i)}$ are the activation functions. Minerva2 for classification can be expressed as a 3-layer FFNN:

$$\boldsymbol{W}^{(1)} = \tilde{\boldsymbol{K}}^\top \qquad \boldsymbol{b}^{(1)} = \boldsymbol{0} \qquad \boldsymbol{\phi}^{(1)}(\boldsymbol{x})_i = x_i^\beta \tag{4}$$

$$\boldsymbol{W}^{(2)} = \boldsymbol{V} \qquad \boldsymbol{b}^{(2)} = \boldsymbol{0} \qquad \boldsymbol{\phi}^{(2)}(\boldsymbol{x}) = \tilde{\boldsymbol{x}} \tag{5}$$

$$\boldsymbol{W}^{(3)} = \tilde{\boldsymbol{U}}^\top \qquad \boldsymbol{b}^{(3)} = \boldsymbol{0} \tag{6}$$

Minerva2 has been in use since 1986, including for speech and language tasks [12, 13, 14] and this, so far as we are aware, is the first demonstration that it is in fact a neural network.

2.3. Parametrising memory-based models

Minerva2's is completely dependent on the exemplar keys and values being suited to the task. This limitation may be alleviated by incorporating a learned feature transformation, allowing the model to a) emphasise relevant information in the features; and b) discard irrelevant information. Let $\boldsymbol{g}$ be an affine transformation, $\boldsymbol{g} : \mathbb{R}^F \rightarrow \mathbb{R}^G$, where F is the input feature dimension and G is some chosen feature embedding size. The similarity comparison between the input query and the exemplar features is performed on the transformed versions, with the activations a_i in Equation 1 replaced by,

$$a_i = (\tilde{\boldsymbol{K}}_{(g)}^\top \tilde{\boldsymbol{q}}_{(g)})_i^\beta \tag{7}$$

where $\boldsymbol{q}_{(g)} = \boldsymbol{g}(\boldsymbol{q})$ and $\boldsymbol{K}_{(g)} = \begin{bmatrix}\boldsymbol{g}(\boldsymbol{k}_1) & \dots & \boldsymbol{g}(\boldsymbol{k}_N)\end{bmatrix}$. We will refer to this parametrised version of Minerva2 as MinervaP. Each class is represented by a one-hot encoding ($\boldsymbol{U}$ in Equation 2), but the exemplar class representations ($\boldsymbol{V}$ in Equation 1) are free parameters learned during training. This allows for specific exemplars to fall between classes. MinervaP can be trained using a suitable activation function and loss function on its output, such as softmax and cross-entropy loss for classification, or mean squared error loss for regression.

3. Experiments

These experiments explore the performance of Minerva2 and MinervaP with different features by comparison with a parametric model. Three tasks were performed, each with three different model architectures: Minerva2, MinervaP, and a baseline FFNN. Each model was tested with three different feature representations for each task. All models used Kaiming initialisation [19], unless otherwise stated. Hyperparameters (learning rate, weight decay, dropout and, for memory-based models, activation power β) were tuned on a development set, with a held-out test set for final results. Further details relating to hyperparameters tuning, development sets and feature extraction are given in the supplementary material. Models were trained using a single NVIDIA RTX 3080 GPU with 10 GB RAM.

3.1. Clarity Prediction Challenge 2

Clarity Prediction Challenge 2 (CPC2) [7] is a speech intelligibility prediction task, with data consisting of tuples of a speech signal and a corresponding measure of intelligibility. For a full description of the task, see [7]. Three different feature representations were derived: **log mel spectrogram**, **XLSR** , and pretrained **Whisper decoder** [10]. For more details, see the supplementary materials. Features were averaged over the time-domain to provide sentence-level features.

All models used a sigmoid activation on the output, subsequently scaled to the range 0-100 to match the range of the correctness scores. The Minerva2 and MinervaP models used 128 exemplars; more than this did not improve performance. The MinervaP models used a feature transformation dimension of 32 and a class representation dimension of 1. The FFNN had hidden dimension 512 and ReLU activations on the hidden layers. For Minerva2 only, a calibration was performed, using the exemplars, to re-scale the output. Training used mean squared error loss, and took approximately 4 minutes for both models.

Table 1: *Speech intelligibility prediction on CPC2.*

Features	Classifier	Learned params	RMSE Dev	RMSE Test
Mel Spec	Minerva2	0	35.17	42.36
	MinervaP	0.01 M	29.63	39.69
	FFNN	0.15 M	28.20	40.60
XLSR	Minerva2	0	33.05	37.33
	MinervaP	0.02 M	24.21	28.59
	FFNN	0.39 M	23.93	27.63
Whisper decoder	Minerva2	0	28.33	29.37
	MinervaP	0.02 M	22.50	24.38
	FFNN	0.39 M	22.40	24.50
beHASPI	CPC2 baseline		-	28.7
Whisper	E011[†]		-	25.1

[†]CPC2 challenge winner - used Whisper encoder features.

CPC2 results are given in Table 1, and are discussed in § 3.4 and 3.7. The development set consists of listeners and enhancement systems seen in the training data; the evaluation set consists of unseen listeners and systems. The RMSE metric gives an indication of how 'wrong' a model's predictions are expected to be; smaller is better.

3.2. TIMIT

Frame-based phone classification was performed on TIMIT [20] using three different feature representations: **log mel spectrogram**, pretrained **wav2vec** [8]; and pretrained **HuBERT** [9]. The Minerva2 and MinervaP models used 14,976 exemplars, selected randomly from the training data and stratified by class (384 per class). The MinervaP model had a feature transformation dimension of 64 and a class representation dimension of 39. The output layer used a softmax activation. The FFNN had hidden dimension 1024. Training used cross-entropy loss, and took around 12 minutes for MinervaP and 9 minutes for the FFNNs. The results given in Table 2 are discussed in §3.4.

3.3. GoEmotions

Emotion classification of text was performed on the GoEmotions dataset [21], a collection of 58,009 Reddit posts with human annotated emotion labels: *negative*, *positive*, *neutral* and *ambiguous*. A single post can have multiple labels. Three different feature representations were used: **LSA** [22]; **Word2vec** [23]; and **BERT** based on MPNet [24].

Table 2: *Frame-based phone classification on TIMIT.*

Features	Classifier	Learned params	Accuracy (%) Dev	Accuracy (%) Test
Mel Spec	Minerva2	0	41.51	40.65
	MinervaP	0.59 M	68.48	67.02
	FFNN	1.19 M	69.64	68.19
Wav2vec	Minerva2	0	50.67	50.83
	MinervaP	0.63 M	82.56	81.21
	FFNN	1.87 M	82.65	81.31
HuBERT	Minerva2	0	73.13	73.02
	MinervaP	0.63 M	88.33	87.50
	FFNN	1.87 M	88.36	87.60

The Minerva2 and MinervaP models use 8,192 exemplars, selected randomly from the training data. The MinervaP model has a feature transformation dimension of 32 and a class representation dimension of 4. The output layer uses a sigmoid activation, and additionally incorporates a learned threshold for each class. The FFNN has hidden dimension 1024. Training used binary cross-entropy loss, and took approximately 4 minutes for both MinervaP and FFNN.

Table 3: *Emotion classification of text on GoEmotions using base and MinervaP models.*

Features	Classifier	Learned params	AUC (%) Dev	AUC (%) Test
LSA	Minerva2	0	61.68	61.65
	MinervaP	0.04 M	69.14	69.55
	FFNN	1.36 M	69.53	70.43
Word2vec	Minerva2	0	75.22	75.60
	MinervaP	0.06 M	82.45	82.56
	FFNN	1.84 M	83.15	83.19
Pretrained BERT	Minerva2	0	79.66	79.61
	MinervaP	0.06 M	85.44	85.61
	FFNN	1.84 M	85.84	86.05

GoEmotions results are given in Table 3. The Area Under ROC Curve (AUC) metric is an average across the four classes, weighted by class frequency. A value of 50% is equivalent to random chance, and higher values indicate better performance.

3.4. Feature and class representation

Improving feature and class representation improves performance for all models across all tasks. More interesting are the relative differences between the models. MinervaP shows a substantial improvement in performance over Minerva2 across all feature representations ($p < 0.01$ in all cases). Whisper, wav2vec and HuBERT have been used effectively for a variety of tasks, such as speech recognition, speaker verification and sentiment analysis [25, 26, 10, 27]. Since one task's relevant information is another task's noise, the ability to transform the features is crucial, allowing the model to retain relevant in-

formation while ignoring irrelevant information. MinervaP's learned feature transformation can reduce the feature dimension from 768 (for Whisper, wav2vec and HuBERT representations) to 64 without sacrificing performance, suggesting that its ability to ignore irrelevant information is useful. The learned feature transform provides most of the improvement over Minerva2, but the learned class representations give a small but consistent additional improvement (from 87.19% to 87.6% for TIMIT HuBERT, for example).

Secondly, while the FFNN outperforms MinervaP on TIMIT ($p < 0.05$ in all cases), the performance difference reduces as feature representation improves, dropping from 1.16% using mel spec features, to just 0.1% using Wav2Vec or HuBERT features. This is also true for the GoEmotions task results: the performance difference drops from 0.88% for the LSA features to 0.63% for the word2vec features to 0.45% for the BERT features. For CPC2, neither model is effective on the evaluation set using mel spec features, but the FFNN performs better on XLSR features, and MinervaP performs better using Whisper features, though neither difference is statistically significant ($p > 0.05$). Statistical comparisons are t-tests on 5 different randomly-initialised repeats of each model.

Memory-based models, such as dynamic time warping for speech recognition, or concatenative speech synthesis, have been competitive in the past, but they have fallen out of use. Likewise, recent comparisons have found that parametric models outperform memory-based models [28]. In contrast, our results show that, as feature representations continue to improve, they can be combined with memory-based models such as MinervaP to make competitive end-to-end models.

3.5. Interpretability

The components of MinervaP are interpretable, allowing us to track how a particular input is classified, which is not possible in a typical FFNN. For illustrative purposes, a small MinervaP model with 64 exemplars was trained using a class dimension of 2. The four 'true' class representations are labelled at the top, bottom, left and right of Figure 2. The L2-normalised learned exemplar representations are shown on the unit circle, coloured by their human-annotated classes. Most are clustered near their class labels, but many of the *ambiguous* exemplars (coloured red) are located near the *neutral* class label. As a result, this small model is almost incapable of classifying *ambiguous* input, but because we can interpret what the model is doing, we can troubleshoot. In this case, using 2-dimensional class representations requires that *positive* and *negative* emotions be considered opposites, which is reasonable, but also that *ambiguous* and *neutral* be considered opposite emotions, which is not.

Using the model to classify the input text *I didn't know that, thank you for teaching me something today* gives the predicted class representation marked as a black cross near the *positive* label; the human annotators classified this text as *positive* as well. The most similar exemplar to this input is *thank you much at least its just hair still it'll be back*. Examining the most similar exemplars gives information on *why* the input was classified as it was, which is not available from a FFNN.

3.6. Exemplar set size

Figure 3 shows the accuracy (bottom) and optimal value of β (top) of the Minerva2 model predictions for different sized exemplar sets on the TIMIT task. Performance increases as the exemplar set size increases, though with diminishing returns, and substantial computation and memory overheads. As explained in §2.1, the optimal value of β increases with exemplar set size.

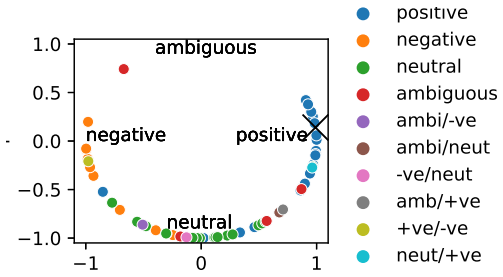

Figure 2: *Learned 2-dimensional exemplar class representations. The black cross is the predicted class representation of the input text "I didn't know that, thank you for telling me".*

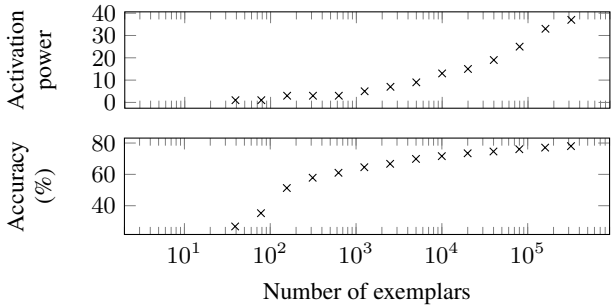

Figure 3: *Minerva2 results on TIMIT test set test set by number of exemplars: accuracy (bottom) and optimal β-value (top).*

3.7. State-of-the-art CPC2 performance

The FFNN and MinervaP perform better on the 2023 CPC2 task than any of the CPC2 challenge submissions. It is particularly interesting that the tiny MinervaP model (with only 20k learned parameters) outperforms substantially larger models using similar features [7, 6]. This suggests that highly parametrised models, such as Long Short-term Memory (LSTM), may be unnecessary and even counter-productive for this task when using high quality features, supporting our conclusion that simple, interpretable models have a place in modern machine learning.

4. Conclusions and further work

Augmenting neural networks with memory has shown promise in several fields. We have shown that an alternative approach - adding parameters to a purely memory-based model - gives a simple, interpretable model that is competitive when using high quality features. It achieves state-of-the-are performance in the CPC2 speech intelligibility task, despite its small size.

We have also demonstrated that the long-standing Minerva2 model, on which our parametric memory model is based, is a neural network with the parameters taken directly from the data. It is capable of performing substantially better than chance *without* training. This presents the intriguing possibility of using Minerva2 as the initialisation of a neural network, which could be further improved with training. The high initial performance could be especially useful for small datasets, and indeed, may be one of the reasons Minerva2 works so well for the small CPC2 dataset. Further exploration of this option is planned.

5. Acknowledgements

This work was supported by the Centre for Doctoral Training in Speech and Language Technologies (SLT) and their Applications funded by UK Research and Innovation [grant number EP/S023062/1]. This work was also supported by Toshiba.

6. References

[1] G. Sun, Y. Hua, G. Hu, and N. Robertson, "Mamba: Multi-level aggregation via memory bank for video object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 2620–2627.

[2] Z. Zhong, T. Lei, and D. Chen, "Training language models with memory augmentation," *arXiv preprint arXiv:2205.12674*, 2022.

[3] W. Wang, L. Dong, H. Cheng, X. Liu, X. Yan, J. Gao, and F. Wei, "Augmenting language models with long-term memory," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[4] Y. Wu, M. N. Rabe, D. Hutchins, and C. Szegedy, "Memorizing transformers," in *2022 International Conference on Learning Respresentations (ICLR)*, 2022.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[6] R. Mogridge, G. Close, R. Sutherland, T. Hain, J. Barker, S. Goetze, and A. Ragni, "Non-intrusive speech intelligibility prediction for hearing-impaired users using intermediate asr features and human memory models," in *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[7] J. Barker, M. Akeroyd, W. Bailey, T. J. Cox, J. F. Culling, J. Firth, S. Graetzer, and G. Naylor, "The 2nd Clarity Prediction Challenge: A machine learning challenge for hearing aid intelligibility prediction," in *ICASSP*, 2024.

[8] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 12 449–12 460.

[9] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," 2021.

[10] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning*. PMLR, 2023, pp. 28 492–28 518.

[11] D. Hintzman, "Minerva 2: a simulation model of human memory," *Behav. Res. Methods Instrum. Comput.*, vol. 16, pp. 96–101, 03 1984.

[12] V. Maier and R. K. Moore, "An investigation into a simulation of episodic memory for automatic speech recognition," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH 2005*. ISCA, 2005, pp. 1245–1248.

[13] V. Maier and R. Moore, "Temporal episodic memory model: an evolution of Minerva2," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH 2007*, 2007, pp. 866–869.

[14] R. Moore and V. Maier, "Preserving fine phonetic detail using episodic memory: Automatic speech recognition using minerva2," in *International Congress of Phonetic Sciences, 2007*, 2007, pp. 197–203.

[15] A. Graves, G. Wayne, M. Reynolds, T. Harley, I. Danihelka, A. Grabska-Barwińska, S. G. Colmenarejo, E. Grefenstette, T. Ramalho, J. Agapiou *et al.*, "Hybrid computing using a neural network with dynamic external memory," *Nature*, vol. 538, no. 7626, pp. 471–476, 2016.

[16] M. R. Dougherty, C. F. Gettys, and E. E. Ogden, "Minerva-dm: A memory processes model for judgments of likelihood." *Psychological Review*, vol. 106, no. 1, p. 180, 1999.

[17] L. Zhang and A. Osth, "Modelling orthographic similarity effects in recognition memory reveals support for open bigram representations of letter coding," *Cognitive Psychology*, vol. 148, p. 101619, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010028523000774

[18] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *ArXiv*, vol. 1409.0473, 2016.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[20] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, "TIMIT acoustic-phonetic continuous speech corpus," *Linguistic Data Consortium*, 1992.

[21] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, "Goemotions: A dataset of fine-grained emotions," *arXiv preprint arXiv:2005.00547*, 2020.

[22] F. Günther, C. Dudschig, and B. Kaup, "Lsafun - an r package for computations based on latent semantic analysis," *Behavior Research Methods*, vol. 47, pp. 930–944, 2015. [Online]. Available: https://link.springer.com/article/10.3758/s13428-014-0529-0

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[24] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "Mpnet: Masked and permuted pre-training for language understanding," in *NeurIPS 2020*. ACM, 2020.

[25] Y. Wang, A. Boumadane, and A. Heba, "A fine-tuned wav2vec 2.0/hubert benchmark for speech emotion recognition, speaker verification and spoken language understanding," *arXiv preprint arXiv:2111.02735*, 2021.

[26] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," *arXiv preprint arXiv:2012.06185*, 2020.

[27] G. Papala, A. Ransing, and P. Jain, "Sentiment analysis and speaker diarization in hindi and marathi using using finetuned whisper: Sentiment analysis in hindi and marathi," *Scalable Computing: Practice and Experience*, vol. 24, no. 4, pp. 835–846, 2023.

[28] J. Sikos and S. Padó, "Frame identification as categorization: Exemplars vs prototypes in embeddingland," in *Proceedings of the 13th International Conference on Computational Semantics-Long Papers*, 2019, pp. 295–306.