Medical Text Classification Using Hyperdimensional Computing

Caroline MacLaren
Electrical and Computing Engineering Department
Villanova University
cmaclare@villanova.edu

***Abstract*—Hyperdimensional Computing (HDC) is an emerging Artificial Intelligence model that works towards modeling the decision making abilities of the human brain. Through high-dimensional, holographic randomness, HDC proposes a mathematical procedure for pattern recognition and prediction in data sets. This algorithm utilizes the same encoding scheme for both training and testing, and predictions are made based on the similarity of high-dimensional, randomized vectors. HDC is a competitive approach compared with current state of the art Deep Learning algorithms, such as Convolutional Neural Networks (CNN) and Recurrent Neural Network (RNN). HDC is capable of matching similar accuracy as compared to these models, while proving to be more energy efficient, and with utilizing less memory, needing less computing power. With these advantages, comes the ability to run and operate well on edge devices. HDC, although still new, is ushering in a possibility of more sustainable machine learning endeavours.**

With the rise in the potential of HDC, there is a multitude of different fields where it could be of benefit, such as Natural Language Processing (NLP) algorithms. One such area of need is within the realm of health care and medicine. CNN and hybrid approaches currently have applications in medical text classification, but there is not yet any advancements for HDC. This paper proposes the application of a HDC program for medical text classification competitive to current state of the art counterparts, capable of running on a Raspberry Pi 4. With a strong accuracy of 85.246%, HDC's energy conservation in the realm of medical text classification is providing great momentum for better machine learning alternatives.

I. Introduction

A. Background

Artificial intelligence (AI) is the modeling and process of creating intelligent programs for decision-making. AI is modeled to think and act the way the human brain does. Over time, AI has become more sophisticated, accurate, and fast[1]. Making accurate predictions is a challenge with different ML models. As a result, with the rise of machine learning has ushered in deep learning (DL). Deep learning is a subset of machine learning, and is a multi-dimensional neural network scheme, often used for applications of natural language processing (NLP), pattern recognition, prediction analysis, classification, etc.[2]. Such increases in neural network layers has proven to increase accuracy. However, one of the largest issues confronting sophisticated deep learning models is that they require a large amount of memory, large latency, and large amount of computing power[3]. As a result, applications require expensive hardware necessary in order to match such needs[4]. This raises the concern of making AI applications equitable and easily available. More importantly, this restricts the ability of AI applications to be transferable to edge devices.

An edge device, such as a Raspberry Pi 4, is a computer that can process data at a local level and send information to a network or cloud. Essentially, edge devices are inexpensive, memory-efficient, powerful computing devices[5]. However, DL applications, being so memory intensive, are less transferable to edge devices[3]. With the emergence of Hyperdimensional Computing, possibilities of low memory usage, small latency, and less computing power are growing.

B. Introduction to Hyperdimensional Computing

Hyperdimensional Computing is a brain-inspired paradigm used to model the unique decision-making of the human brain, inspired by neural circuits - a network or structure that connects individual neurons and regions together[6]. Neural circuits perform specific functions when activated and are the basis for large scale neural networks, such as circuits in the hippocampus[7]. Hyperdimensional Computing is the mathematical manipulation of high-dimensional, holographic, random vectors to best model how the human brain classifies and predicts information.

Hyperdimensional Computing mimics the vast network of neurons in a circuit, and their ability to create and recognize patterns through pseudo random creation and manipulation of vectors of 10,000 elements. HDC relies on the principle of randomness, thus creating high-dimensional vectors of randomly distributed -1s and 1s[3]. The foundation of the HDC model lies in its mathematical encoding scheme.

The core of the encoding scheme is through use of three fundamental mathematical operations: multiplication, addition, and permutation (MAP)[3]. Hypervectors (HVs) - the high-dimensional, random vector of -1s and 1s - are permuted to create dissimilarity among other unique HVs. Upon permuting, HVs are bound together through multiplication and a pattern begins to form upon summing all these vector sequences. Upon performing this encoding, the training information is stored in associative memory to be compared with against a testing data set, thus determining an accuracy report. The encoding scheme is applicable to both training and testing phases, making the algorithm robust and easily comprehendable. Similarity tests are conducted to then make a prediction of a class.

In its application, Hyperdimensional Computing's approach results in substantially less energy consumption. Rather than the multi-dimensional application of deep learning models, HDC relies on a larger vector in just one dimension. This, as a result, uses far less memory, thus less computer energy. In some cases, the amount of energy consumed in a deep learning model was over 10 times as much as that of an HDC model[8]. As a result, the time it takes to run an HDC model has also proven to be much faster than that of its DL counterparts. Additionally, HDC requires minimal training data to maintain a strong accuracy report, specifically in classification models. Such advantages can lead to even less energy consumption reports.

C. Current Applications

It should be noted that there are still a limited number of Hyperdimensional Computing applications. However, these have proven to present the expected benefits of HDC models compared to other baseline machine learning models. In [9], a study of a Hyperdimensional Computing application for an Electromyography (EMG) based Hand Gesture Recognition application explores the comparison between HDC and Support Vector Machine (SVM). In most of the subjects, the accuracy of HDC proved to be more accurate than that of the state of the art. More importantly, the hand gesture recognition was capable or providing a 97.8% accuracy reporting while only using $\frac{1}{3}$ of the training data. By providing strong accuracy with little training data, the algorithm can free up even more memory that HDC already uses, thus minimizing the amount of energy the computer uses.

In [4], a Language Recognition application of Hyperdimensional Computing is proposed and implemented. A HDC classification model was compared against a baseline machine learning algorithm to compare the accuracy reports and memory advantages of HDC. Using a set of 21 different European languages, the HVs in this model were associated with every letter of the alphabet. Through use of character binding by a specified n-Gram, the HDC algorithm was able to encode text data for training and testing. The results showed similar and little loss accuracy to the baseline state of the art. More importantly, the memory usage maintained a consistent amount across all n-Grams for the HDC model, whereas the baseline model grew over 20-fold as the n-Grams increased. This study, thus, showcased the energy efficiency of HDC in a text classification model. This model influenced the implementation of my own HDC application in this paper.

Not only have these HDC models proven effective, and in some cases better accuracy in addition to the small memory usage, there is evidence of lesser energy consumption. In the VoiceHD model presented, only 38 kJ of energy was consumed while the Digital Neural Network (DNN) alternative expended 454 kJ[8]. Additionally, this study proposes the option of a hybrid approach, whereby holding the strongest accuracy without as much energy usage in the singular DNN approach. The accuracy remained strong even when using only 40% of the training data.

D. Problem Statement

It has already been discussed how Hyperdimensional Computing has shown remarkable success in maintaining accuracy reports with low energy consumption for NLP algorithms. Much of the current medical text classification algorithms are memory and computer intensive machine learning algorithms - algorithms that are not feasible on edge devices. For many medical applications, information shared is sensitive and needs to be protected due to personal patient privacy and HIPPA regulations. However, due to the memory-intensive models, the alternative to expensive and powerful computers is moving data to and from the cloud. However, the cloud poses a security risk to ensure compliance with the regulations required for health care applications. By providing a new approach, such as Hyperdimensional Computing, this allows for new opportunities for health care applications to reap the benefits of edge devices, while maintaining data security[3].

There are limited, effective medical text classification algorithms. For those that exist and report strong accuracy in classification, they pose the issue of computer energy waste. This paper presents a more accessible, energy-efficient, and accurate machine learning application for medical text classification using the principles of Hyperdimensional Computing. This paper focuses on the classification of cancer hallmarks, which is a term in medical literature to describe the indicators and properties that produce cancer in the human body[10]. This model poses an alternative to the current state of the art that makes medical text classification possible on edge devices.

E. State of the Art for Medical Text Classification

Medical text classification is a growing area of interest, as it has the capability to save time and resources for both doctors and patients. A ML approach to automate such processes for doctor notes and information, while learning context and not relying on a heavy dictionary, can change the scope of day to day medical work. In [11], a medical text classification study is presented using CNN that produces strong competition to current NLP models. In this study, the CNN algorithm was found to be 15% more accurate to other state of the art applications for NLP. This CNN model utilized a Word2Vec algorithm, a mathematical library for associating vectors with words, in coordination with 2 neural layers[11].

A medical text classification model using CNN for indicators of cancer proved to report strong F scores. Baker *et al*. makes use of a data set of ten classes of different cancer hallmarks. Cancer can be defined by hallmarks, which are properties that describe cell actions and behaviors that sustain cancer in the human body. The ten classes are denoted as positive or negative, whereby the former describes medical abstracts of hallmarks present. In this study, a CNN approach outmatched a current SVM approach by 4% with a total F score of 81%[10].

One recent study in medical text classification highlights two proposed algorithms - a Quad Channel Hybrid Long Short-Term Memory (QC-LSTM) and a Hybrid Bidirectional Quad Recurrent Unit (BiGRU). This paper makes use of the

Hallmark data set, describing three main classes: activating invasion and metastasis, cellular energetics, and tumor promoting inflammation. This study also made use of the AIM data set, which only has two classes: positive and negative. In the first approach (QC-LSTM), a hybrid method of CNN and LSTM, the highest accuracy reported among the three Hallmarks is 75.98%. For the second approach (BiGRU), a hybrid method of CNN and BiGRU, the highest accuracy reported was 74.71%. This study serves as the base comparison of the findings in this paper[12].

There are handfuls of other studies, such as a BiGRU approach to medical text classification found to be 75.72% for the Hallmark data set[13]. However, the three notably mentioned have served critical to the findings of this paper, and are representative examples of the current state of the art for medical text classification. Each of these, although strong in many facets, face the challenges of deep learning models that has been discussed so far. CNN is a critical part of these papers and HDC offers a competitive edge to CNN, overcoming many of the challenges of deep learning models, as will be described in this paper.

II. Hyperdimensional Computing Architecture

A. Data Sets Utilized

To craft a HDC model for text classification, the Hallmark data set from Baker et al.[10] was used for training and testing purposes. Three classes from this data set, activating invasion and metastasis, tumor-promoting inflammations, and deregulating cellular energetics, were used in this HDC application. This data set was taken from around 1,852 medical abstracts and was split into 70%/10%/20% with training, validation, and testing respectively. The validation set was not utilized for the purposes of this algorithm. A breakdown of the data set used is provided in Table I.

Class	Data Set Size	Train	Test
Activating invasion & metastasis	408,465	324,134	84,331
Tumor-promoting inflammations	346,023	271,737	74,286
Deregulating cellular energetics	137,637	110,578	27,059

TABLE I
Data Set Size (in bits)

B. Encoding Scheme

Hyperdimensional Computing relies on a core encoding scheme that takes in a data set and encodes into manipulated hypervectors. First, there is item memory to store the randomized HVs that coorespond to their relevant element. In the case of text processing, characters make up the item memory. For this specific algorithm, every letter of the alphabet, every number, and the characters '%', '-', '+', '/' make up the item memory. Thus, when processing through a sentence, for instance, each time a key is read, its associated HV is pulled for the M.A.P. operation. The mathematical operation can be broken down into the following equation: $nGram = \rho\rho key_1 * \rho key_2 * key_3$. The resultant of the encoder is the sum of all n-Grams.

Dependent upon the decided n-Gram, the encoder will go through *n* keys at a time, permuting each time to create disparity between the individual vectors. For example, with a trigram, the first key is permuted twice and the second key is permuted once, all before being bound together. Upon permuting, the n-Gram HVs are multiplied together to bind the characters, drawing a pattern similar to words. After binding, every n-Gram is summed into one total HV using a sliding window. This process is outlined in Figure 1.

Applying this encoding scheme to medical text classification, patterns among specific technical and medical terms can be drawn and applied to both the training and testing phases of the HDC algorithm. Ultimately, the encoding scheme is at the core of the HDC algorithm.

C. Training Phase

Utilizing the Hallmark data set described in Section 3.1, the three Hallmarks are sent into the encoding module. Upon the processes to encode the text, the resulting sumHV is stored in associative memory. This associative memory stores all of the final trained data and is the sole memory that is used for testing. The training phase can be looking at more closely in Figure 2.

D. Comparison and Prediction

Upon training the three classes, this associative memory can be compared to an unknown query. Using the testing data set, a testing algorithm was developed to compare an unknown class against all three classes, making a prediction, and then confirming whether or not the prediction was correct. The final accuracy reports how many of the testing queries were correct out of the entire batch. The testing algorithm is similar to that of the training. It reads in the text and encodes it using the encoding scheme in Figure 1. This computed sumHV is then compared against all three sumHVs stored in associative memory.

There are a multitude of different similarity tests that can be used when comparing two vectors. A Hamming distance formula is often used for binary comparison, so this algorithm didn't quite fit the purposes of the algorithm. The Euclidean distance is another route of testing that can determine dissimilarity of vectors. Ultimately, the cosine similarity test was best for the implementation of comparing the unknown class against the associative memory [14]. The cosine angle theorem is as follows:

$$cos(\vec{A}, \vec{B}) = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}||\vec{B}|}$$

By use of this similarity test, the predicted class is that which reports an angle closest to 1, yielding the closest distance between two vectors (i.e. most similar). 122 total samples were tested for the purposes of this algorithm, each around 1,000 bits each. The process of testing is outlined in Figure 2.

Predictions are classified as true positive (TP), true negative (TN), false positive (FP), and false negative (FN) upon the comparison of the prediction to the correct answer. From this, the accuracy, precision, recall, and F score can all be calculated

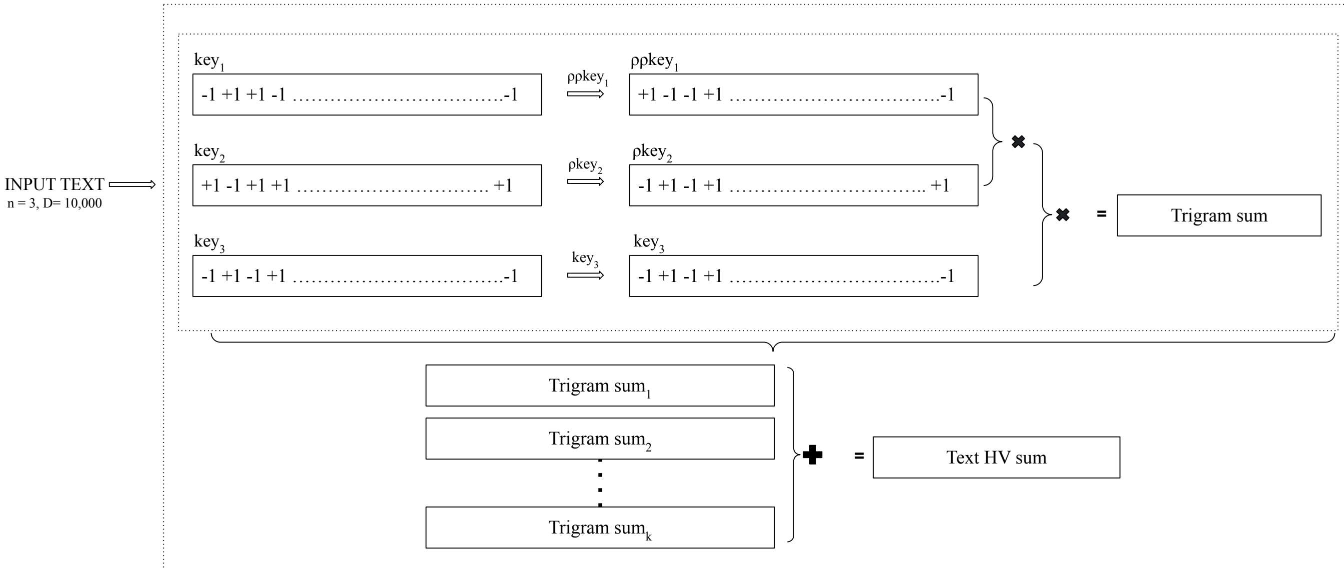

Fig. 1. Encoding Module for HDC

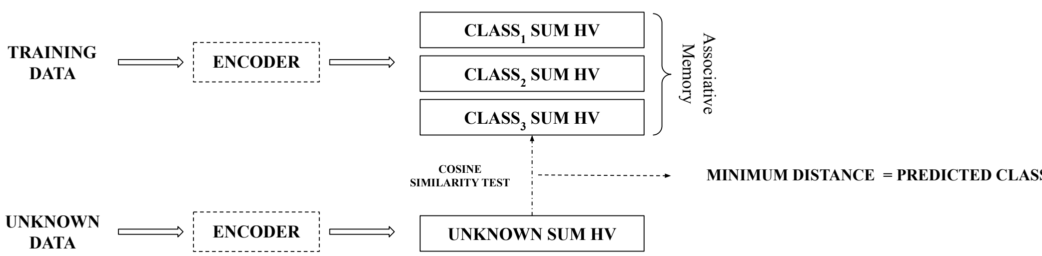

Fig. 2. Predicting an Unknown Class

to determine the strength of the algorithm. Such equations are expressed below [12]:

$$accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$

$$recall = \frac{T_p}{T_p + F_n}$$

$$precision = \frac{T_p}{T_p + F_p}$$

$$Fscore = \frac{2 * precision * recall}{precision + recall}$$

Ultimately, a prediction is compared to the true class to report as either TP, TN, FN, or FP. These indicators are then used to calculate the findings above to describe the strength of the algorithm.

III. Discussion of Results

A. Initial Findings

In the initial algorithm, the accuracy report using trigrams was 85.067%. Table II gives a breakdown of the accuracy findings from the Hallmark data set from the final algorithm. It was found that using a 5-Gram posed the strongest accuracy and F scores. Although the current NLP HDC models tended to perform well with trigrams, the 5-gram likely was strongest for medical text classification due to the vocabulary patterns of medical terms (i.e. the medical text might average words of 5 characters more commonly than the European language study [4]).

Table III illustrates how each n-Gram yielded similar memory usage reportings as well. However, a trigram approach may prove the most successful as it still maintains a strong accuracy reporting, while running much faster than a 5-Gram approach. The time reported for each n-Gram was collected on a Raspberry Pi 4, where about 80% of the total time was taken to train the algorithm.

N-Gram	Accuracy	Precision	Recall	F1 Score
2	81.967%	82.482%	81.967%	82.224%
3	85.246%	85.835%	85.245%	85.539%
4	85.246%	86.564%	85.246%	85.900%
5	87.705%	88.623%	87.705%	88.162%

TABLE II
ACCURACY FINDINGS

N-Gram	Memory Usage	Time
2	78.43 MB	238.792 s
3	77.07 MB	360.383 s
4	77.59 MB	491.267 s
5	76.97 MB	614.495 s

TABLE III
ACCURACY FINDINGS

One of the largest benefits of HDC is that it doesn't require a large amount of training data to maintain a good accuracy for classification. Figure 3 displays how the accuracy for smaller amounts of training data did not have a significantly large affect on accuracy reports. N-Grams of size 5 performed particularly well, maintaining 86% even down to only 40% of the training data. This opens up the opportunity for using only 40% of the training data. For 5-grams, the time to run the algorithm decreased by 53.943%. By minimizing the training data, both the time it takes to train the algorithm and the energy and memory footprint of the algorithm can thus be minimized.

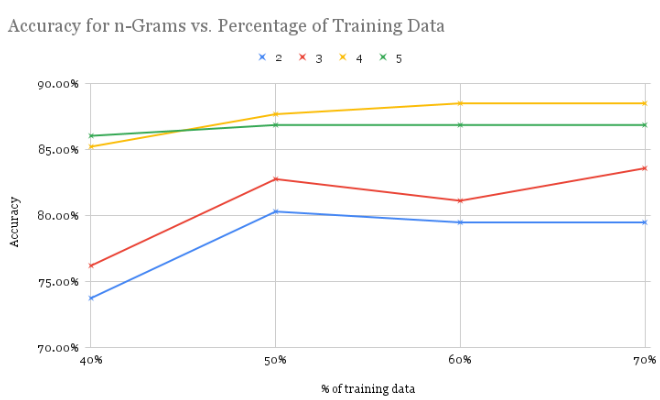

Fig. 3. Accuracy vs. Smaller Amounts of Training Data

It is important to note that some initial trials using all ten classes of the Hallmark data set was run. Using trigrams, the accuracy was found to be 51.951%.

B. Optimizations

In an attempt to bolster the accuracy found from the algorithm, the core encoder was manipulated. Rather than utilizing just -1s and 1s, a randomized set of float values from -1 to 1 was instead utilized for the HVs. This was in an attempt to create even more disparity between the HVs for strong differences in comparison. This resulted in only a $> 1\%$ accuracy change from 85.067% to 85.245%.

A retraining functionality was implemented in the algorithm to determine if any accuracy gains were noticeable. The algorithm would run the training data as a test, where if the incorrect classification was made, the query HV would be subtracted from the incorrectly predicted class in associative memory so as to avoid a similar mistake again. Table IV gives a breakdown of the retraining iterations run. As is noticeable, there was not any consistent accuracy gains after multiple iterations of retraining. Iterations of 100 or more could have been run, but the retraining slowed down the entire process of the algorithm and was not proving largely successful.

Iterations	Accuracy
1	86.885%
10	85.246%
25	86.066%
50	85.246%

TABLE IV
RETRAINING FINDINGS FOR TRIGRAMS

C. Comparisons

The HDC model proposed for the three classes of the Hallmark data set was largely successful. With just a small amount of training data, the algorithm was capable of reporting a strong accuracy of 85.246%. This is competitive to some of the other state of the art mentioned. Compared to the hybrid CNN approaches discussed earlier for the Hallmark data set, the HDC model outperformed the Quad Channel Hybrid LSTM model by 10.54% in accuracy. Thus, this model proposes strong evidence of the benefits of an HDC model.

IV. CONCLUSION

Hyperdimensional Computing is a fairly new and efficient model of machine learning that overcomes some of the restrictions that deep learning models currently face, such as large memory usage, computing power, and latency. It proposes a simple implementation, rooted in its installments of pseudo-random, high-dimensional, holographic vectors. Through use of M.A.P., an encoding scheme implemented for both training and testing is the basis for this natural language processing classification.

With the HDC model implemented in this paper, performing at an accuracy reporting of 85.246%, fairing 10.54% better than the current state of the art proposed in [12], there is strong evidence of HDC being a competitive force for medical text classification. More importantly, HDC has the ability of being transferable to edge devices. As a result, this implementation is robust and provides an outlet for medical text classification on edge devices. This fosters the potential of a more green and equitable AI in the realm of medicine. In its application, this model could benefit lower income hospitals and clinics, and could even be accessible to individuals to promote better health literacy.

A. Challenges and Successes

There were some initial challenges confronted with this paper in the early research phase. The work proposed in [13] was intriguing and was the original target of this research. However, the project was not open source and for a good baseline comparison between a CNN and HDC model, it was necessary to compare against an open source project, utilizing the same data sets. As has been noted earlier, there was not much improvement in the accuracy in the retraining phase as might have been expected. Further research on a more complex retraining algorithm may need to be investigated. The final results found were still strong, however. More importantly, they were fast and didn't require a large usage of memory. Ultimately, this paper shows supporting evidence of the possibility of HDC in the realm of medical text classification by presentation of its strong accuracy reports.

B. Future Work

Part of the success of Hyperdimensional Computing is its advances in little energy consumption. It would be worthwhile to investigate how much specific energy (in kJ) this model consumes. With time restraints, this wasn't possible. However, with more research and investigation, this would provide further evidence to the ability of this HDC model developed. Furthermore, running such tests against the hybrid models proposed in [12] would allow for a more robust comparison of the two models.

As mentioned earlier, it would be worthwhile expanding the algorithm to all ten classes of the Hallmark data set, and performing a comparison against the CNN model proposed in [10]. Currently, the accuracy stands only at 51.951% for the ten classes and it would be worth investigating advancements to improve this accuracy. Additionally, a more robust comparison worthwhile would be to compare the AIM data set results of [12]. Determining if HDC can note text documents marked with cancer hallmarks, rather than just which hallmarks were which, could usher a more thorough comparison between these two reports.

It would be worthwhile working towards a more robust medical text classification model outside the realm of cancer hallmarks. The CNN medical text classification model proposed in [13] allowed for an expansive set of classes (26 total) of large disparity and variety. The opportunity to compare an HDC model against this would be a large and worthwhile endeavor.

Acknowledgment

The makings of this paper would not have been possible if it were not for my research advisor Dr. Xun Jiao and assistance from Donging Ma. I am grateful to the Electrical and Computer Engineering Department at Villanova University for allowing me to explore my interests in this research endeavor.

References

[1] I. C. Education, *What is artificial intelligence (ai)?* [Online]. Available: https://www.ibm.com/cloud/learn/what-is-artificial-intelligence.

[2] F. Q. Lauzon, "An introduction to deep learning," in *2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*, 2012, pp. 1438–1439. DOI: 10.1109/ISSPA.2012.6310529.

[3] E. Hassan, Y. Halawani, B. Mohammad, and H. Saleh, "Hyper-dimensional computing challenges and opportunities for ai applications," *IEEE Access*, pp. 1–1, 2021. DOI: 10.1109/ACCESS.2021.3059762.

[4] A. Rahimi, P. Kanerva, and J. M. Rabaey, "A robust and energy-efficient classifier using brain-inspired hyperdimensional computing," in *Proceedings of the 2016 International Symposium on Low Power Electronics and Design*, ser. ISLPED '16, San Francisco Airport, CA, USA: Association for Computing Machinery, 2016, pp. 64–69, ISBN: 9781450341851. DOI: 10.1145/2934583.2934624. [Online]. Available: https://doi.org/10.1145/2934583.2934624.

[5] N. Klingler, *Edge devices for on-device machine learning and computer vision*. [Online]. Available: https://viso.ai/edge-ai/edge-devices/.

[6] P. Martinez and S. G. Sprecher, "Of circuits and brains: The origin and diversification of neural architectures," *Frontiers in Ecology and Evolution*, vol. 8, 2020, ISSN: 2296-701X. DOI: 10.3389/fevo.2020.00082. [Online]. Available: https://www.frontiersin.org/article/10.3389/fevo.2020.00082.

[7] A. Stocco, C. Lebiere, and J. R. Anderson, "Conditional routing of information to the cortex: A model of the basal ganglia's role in cognitive coordination," *Psychological Review*, vol. 117, pp. 541–574, 2010. DOI: 10.1037/a0019077. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3064519/.

[8] M. Imani, D. Kong, A. Rahimi, and T. Rosing, "Voicehd: Hyperdimensional computing for efficient speech recognition," in *2017 IEEE International Conference on Rebooting Computing (ICRC)*, 2017, pp. 1–8. DOI: 10.1109/ICRC.2017.8123650.

[9] A. Rahimi, S. Benatti, P. Kanerva, L. Benini, and J. M. Rabaey, "Hyperdimensional biosignal processing: A case study for emg-based hand gesture recognition," in *2016 IEEE International Conference on Rebooting Computing (ICRC)*, 2016, pp. 1–8. DOI: 10.1109/ICRC.2016.7738683.

[10] S. Baker, A. Korhonen, and S. Pyysalo, "Cancer hallmark text classification using convolutional neural networks," in *Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2016)*, Osaka, Japan: The COLING 2016 Organizing Committee, Dec. 2016, pp. 1–9. [Online]. Available: https://aclanthology.org/W16-5101.

[11] M. Hughes, I. Li, S. Kotoulas, and T. Suzumura, "Medical text classification using convolutional neural networks," *Studies in Health Technology and Informatics*, vol. 235, pp. 246–250, 2017. DOI: 10.3233/978-1-61499-753-5-246. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1704/1704.06841.pdf.

[12] S. K. Prabhakar and D.-O. Won, "Medical text classification using hybrid deep learning models with multihead attention," *Computational Intelligence and Neuroscience*, vol. 2021, no. 16, 2021. DOI: 10.1155/2021/9425655. [Online]. Available: https://www.hindawi.com/journals/cin/2021/9425655/.

[13] L. Qing, W. Linhong, and D. Xuehai, "A novel neural network-based method for medical text classification," *Future Internet*, vol. 11, no. 12, 2019, ISSN: 1999-5903. DOI: 10.3390/fi11120255. [Online]. Available: https://www.mdpi.com/1999-5903/11/12/255.

[14] L. Ge and K. K. Parhi, "Classification using hyperdimensional computing: A review," *CoRR*, vol. abs/2004.11204, 2020. arXiv: 2004.11204. [Online]. Available: https://arxiv.org/abs/2004.11204.