Contents lists available at ScienceDirect





Information Sciences

journal homepage: www.elsevier.com/locate/ins

Spike-time encoding as a data compression technique for pattern recognition of temporal data



Neelava Sengupta*, Nikola Kasabov

Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, Auckland, New Zealand

ARTICLE INFO

Article history: Received 28 July 2016 Revised 9 April 2017 Accepted 11 April 2017 Available online 12 April 2017

Keywords: Spike-time encoding Data compression Pattern recognition

ABSTRACT

The human brain's ability to efficiently detect patterns from the continuous streaming sensory stimuli has been a source of constant intrigue for naturalists, and has set the course for the development of research into artificial intelligence. Efficient encoding of such input stimuli into discrete timing of events play a decisive role in the ability of the spiking neurons inside the human brain to compress, transmit and recognise information presented by the external environment. In this article, we introduce the spike-time or temporal encoding paradigm as an efficient general approach to data compression for the purpose of pattern recognition. The data compression through spike-time encoding not only dramatically reduces the volume of data required to capture discriminatory information leading to economical storage and transmission, but can also be used for pattern recognition in streaming data domain. We experimentally show that the spike-time data produced by the temporal encoding techniques achieve comparable (superior in some cases) performance of pattern recognition in comparison to the use of the whole raw data. This article also introduces a generalised background knowledge driven optimisation based temporal encoding framework for encoding time series data and as an illustration of this approach, further formulates a temporal encoding algorithm, namely GAGamma, designed to efficiently compress fMRI data using discrete spike-times. We have evaluated the temporal encoding algorithms on the benchmark Starplus fMRI dataset, and the results demonstrate the temporal encoding algorithm's ability to achieve significant data compression without sacrificing the performance of the pattern recognition in the compressed space. We also show that the GAGamma algorithm provides enhanced flexibility and control (compared to the state of the art temporal encoding algorithms) within the design framework of the temporal encoding problem for fMRI, leading to better quality of spike-time data. The proposed approach, by the use of efficient brain-like encoding mechanism, opens up new possibilities in information compression, communication and pattern recognition and thus is applicable to a range of new applications.

© 2017 Elsevier Inc. All rights reserved.

1. Introduction

The human brain is considered to be the most resourceful and efficient system that can recognise patterns in the continuous stimuli captured by the sensory organs in millisecond resolution. It is also observed that human brain cells,

* Corresponding author.

http://dx.doi.org/10.1016/j.ins.2017.04.017 0020-0255/© 2017 Elsevier Inc. All rights reserved.

E-mail address: neelava.sengupta@aut.ac.nz (N. Sengupta).

when presented with external stimuli, propagates the signal economically, over long distances using electrical impulses or spikes via the synaptic action potentials. Hence, it is imperative that there exists an efficient system, which can transform the massive volume of continuous signal to discrete spikes. In neurobiology, the process of such analog to digital signal transformation is known as neural encoding [4]. It is intriguing that the process of neural encoding not only converts the big streaming continuous data space into a compressed space of spikes, but the brain cells also recognise patterns in such a compressed space. The biological organisation of our brain tends to create signals with a very specific class of distributions, and it is from the perspective of evolution understandable that these distributions are optimised for fast analysis. The most popular hypothesis states that the signal strengths are encoded by the mean firing rate, *i.e.* stronger input signal produces larger volume of neuronal firing on an average in the brain. A range of studies [26,27] across multiple species in the sensory and motor-neuronal system supports the validity of mean firing rate hypothesis. A major drawback of this theory, however, lies in the association of information density with spike density. Determining the spike density in millisecond resolution from a large volume of spikes lead to a level of computational inefficiency. As per an alternate theory on neural encoding, neurons carry information in the precise timing of the spikes. This is known as the temporal encoding or spike-time encoding. Numerous research [15,18] has shown the presence of temporal encoding in different parts of the human brain. Temporal encoding supports the efficient representation of information that is required for very fast processing (in millisecond scale) of the stimulus presented to the human brain. As opposed to the rate coding scheme, high fluctuations in mean firing rate, also known as inter-spike interval (ISI) probability distribution is considered to be informative rather than noise in this scheme. The temporal spike-time representation of the data acts as a lossy compression of information. Most forms of learning, though, could be seen as forms of data compression. In fact, we can, in terms of pattern recognition, only learn something from data when there is redundancy in the data. In many data analysis tasks, the data is preprocessed or recoded in a way that could be seen as a form of data compression. If such a preprocessing does not destroy the patterns of interest, it results in comparative performance of the learning algorithms. The motivation of the temporal encoding, thus, in this context is to reduce large volumes of data into a compressed state with minimal loss and the maximal presence of discriminable information. Examples of data sources where such encoding is useful are high-frequency streaming data such as the pulsar data in radio astronomy and seismic activity data.

From the viewpoint of computational theory, the data encoding problem relates to the concepts of information theory. In the seminal paper of information theory, [35], proposed a mathematically complete theory to quantify transmission of information in a communication channel. A conclusive finding that the amount of information in any object can be estimated as the description length of the object continues to set the stage for the development of communications and data processing. Shannon's information theory is built on a presupposition that the computable information in an object is the characteristic of a random source with known probability distribution of which the object is a part. To realise this idea, Shannon derived the 'entropy' from the first principle, which is the measure of average information emitted by an object when observed. It can be described as the functional mapping of the random variable to a real number.Kolmogorov [24], later proposed an alternate and much generalised notion of information measurement known as algorithmic information theory. Contrary to Shannon's theory, Kolmogorov complexity [5,24] considers information as the property of an object in isolation irrespective of the way the object arose [16]. It describes information as the minimum number of bits from which a message or a file can effectively be reconstructed, *i.e.* the minimum number of bits suffice to store a reproducible file. A computational neuron responsible for emitting spikes from sensory data can be regarded as a logical transmission medium responsible for broadcasting continuous information received from the data source. The two neural coding hypotheses hence can be seen and described in the light of information theory. We observe that the rate coding scheme adheres to Shannon's interpretation of encoding. The inherent assumption of the presence of a random source with a known probability distribution in Shannon's theory is much apposite to the mean firing rate as it relates to the frequency of spikes over time. However, our interest in efficient compression of a large volume of data by a sequence of spike-timings and using it for the purpose of pattern recognition is much more in line with Kolmogorov's notion of object representation by minimal description length using computer programs.

1.1. Related work

A significant amount of research has focused on using the biological realism of the spiking neural networks (SNN) for information processing applications akin to traditional neural networks [25]. Under this broad umbrella of SNN, the area of data encoding has been relatively unexplored compared to areas such as neuronal dynamics, network learning behaviours and so on. Human Information Processing Research Laboratory's (Advanced Telecommunication Research Institute) artificial brain (Cellular Automata Machine Brain) project [7] used data encoding as a part of its large-scale brain-like neural architecture. Hardware accelerated implementation of spike encoding for image and video processing was performed in [20]. The literature on the application of spike encoding on the information processing task in data science is restricted to a few algorithms such as temporal-contrast (TC) [22], Hough spiker algorithm (HSA) [19] and Bens spiker algorithm (BSA) [33]. HSA and BSA algorithms are event-driven in nature and can be classified under the temporal encoding paradigm where the time of occurrence of an event (spike) is considered as a unit of information. The TC algorithm, also known as AER encoding, is inspired from the human visual cochlea. The TC algorithm uses a threshold-based method to detect signal contrasts or changes [22]. A user-defined or auto-generated contrast threshold determines the spike events in the TC algorithm. The HSA and BSA algorithm, however, determines a spike event using a deconvolution operation between the

observed signal and a predefined filter. The deconvolution operation in HSA assumes that the convolution function produces a biased converted signal which always stays below the original signal yielding an error [17]. BSA [8], on the other hand, uses a finite reconstruction filter (FIR) for predicted signal generation.

2. A framework for spike-time encoding and compression of temporal data sequences

We formalise the temporal encoding problem for pattern recognition as a data compression problem. The compression function is defined as the map $f : \mathbb{R}^T \to \mathbf{t}^f$, where the encoder $f(\cdot)$ releases a spike at firing times $\mathbf{t}^f := \{t_2^f, t_2^f, \cdots, t_n^f | t_i \in \mathbb{I}^+\}$. The temporal encoding algorithm primarily assumes that the discriminatory information is encoded by the sequence of discrete spike-timings rather than the magnitude and/or the spike density. As a consequence of this assumption, the temporal encoding aims at joint maximisation of information and minimisation of the spike density. Thus, it is in sharp contrast to the rate coding hypothesis. In the next section, we describe a generalised framework for temporal encoding and extend it to formalise a temporal encoding algorithm based on the generalised framework. This formalisation is an extension of the previously published work on this topic [34].

We denote a continuous source signal as $\mathbf{s} \in \mathbb{R}^T$. To simplify the formalisation, we define the encoded spike-train $\mathbf{b} \in \{0, 1\}^T$ as a fixed-length binary sequence of length *T* as opposed to the variable length sequence of spike-timings $\{t_1^f, t_2^f, \cdots, t_n^f | t_i \in \mathbb{I}^+\}$ defined earlier without any loss of generality. *T* denotes the length of the temporal signal to be encoded into spikes. The background knowledge-driven optimisation based encoding algorithm is built on the premise that existing knowledge about the data generation process or in other words the knowledge of the properties of the data generation source can be injected into a predictive system that can generate a predicted signal $\hat{\mathbf{s}}$. For example, the fMRI data generation process behaves like a linear time invariant system, where a spike in the brain cell gives rise to a signal minicking the gamma distribution function [1], whereas the process of EEG data generation can be modelled as a phase varying mixture model of sinusoidal waves or multi-source Gaussian noise model [31]. The notion of knowledge injection is further elaborated in Section 3 using fMRI as an example. If it is possible to formalise the decompression function $\hat{\mathbf{s}}$ from the spike sequence \mathbf{b} , the encoding of data can be formulated as an optimisation problem that minimises the discrepancy between the predicted and the original signal. One way of realising such discrepancy is by minimising the root mean squared error (RMSE) of decompression between the observed signal *s* and the predicted signal $\hat{\mathbf{s}} := f(\mathbf{b}, \Theta)$, Θ being the set of additional parameters required along with \mathbf{b} to describe the prediction function. The optimisation problem can be formulated as:

$$\begin{array}{l} \min_{\mathbf{b},\mathbf{\Theta}} \sqrt{\frac{\sum_{t} (s_{t} - \hat{s}(b_{t},\mathbf{\Theta}))^{2}}{t}} \\ \text{s.t.} \quad b_{t} := \mathbb{I}^{+} \\ 0 \leq b_{t} \leq 1 \\ \sum_{t} b_{t} \leq \alpha \\ \boldsymbol{\beta}^{t} \leq \boldsymbol{\Theta} < \boldsymbol{\gamma} \end{array} \tag{1}$$

The aforementioned optimisation problem is one related to mixed-integer programming optimisation in that a subset of parameter or decision variables to be optimised are integers. A number of solvers exist in the literature that can deal with such problem constraints [2,12,39]. In our implementation, we have used the mixed integer genetic algorithm solver [9,10]. The constraints in Eq. (1) are imposed on the parameters of \hat{s} . The first and the second constraint are used to reduce the search space of the possible values of b_t to {0, 1}. The hyperparameter α is used to control the maximum number of spikes and hence the spike density in the optimal solution. The other sets of hyper-parameters { β , γ } are used to control the upper and lower bounds of the model parameter Θ .

The formulation above for the proposed framework for data encoding is generic, flexible and is driven by knowledgeinjection from the data source. The knowledge-injection component allows the further inclusion of systematic noise as part of \hat{s} , if present. Examples of the inclusion of noise models, such as acoustic noise as part of linear time invariant models of fMRI, is treated in [6,37]. We hypothesise that a sufficiently good choice of $\hat{s}(\mathbf{b}, \Theta)$ preserves and in some cases, enhances the discriminative property of the data in a greatly compressed space. It must also be noted that this formulation adheres to the concept of the non-existence of a universal compression algorithm for all the data sources. The general framework described above can be used to derive specific methods for encoding of special types of data for which background knowledge is available. One such case is fMRI data based on blood-oxygen level dependent response (BOLD). This is further introduced and illustrated in Section 3.

3. GAGamma: a spike-time encoding and compression method for fMRI spatio-temporal data

The fMRI BOLD response is modelled here as a linear time invariant system, which is described by the convolution of the spikes **b** and the haemodynamic response function (HRF), $h(\Theta)$. This operation is characterised by Eqs. (2) and (3).

$$\hat{\mathbf{s}} := \int_{0}^{t} \mathbf{b}(\tau) h(t - \tau) d\tau$$

$$\hat{\mathbf{s}}(\mathbf{b}, \Theta) := \mathbf{b} * h(\Theta)$$
(2)
(3)

$$h(\theta_1, \theta_2) := \frac{1}{\theta_2^{\theta_1} \mathcal{T}(\theta_1)} t^{\theta_1 - 1} e^{-\frac{t}{\theta_2}}$$

$$\tag{4}$$

Numerous mathematical models for HRF have been proposed in the literature [3,13,14]. Majority of the mathematical models for the canonical HRF are found to be some variant of the gamma function. In all our experiments, we have used the gamma distribution function as the HRF model (Eq. (4)). This function is characterised by the parameter set $\Theta := \{\theta_1, \theta_2\}$, where $\theta_1 \in \mathbb{R}^+$ and $\theta_2 \in \mathbb{R}^+$ controls the shape and the scale of the gamma function respectively. By substituting Eqs. (3) and (4) in Eq. (1), the encoding problem can be reduced to solving Eq. (5) and is referred to as GAGamma encoding algorithm hereafter.

$$\min_{\substack{\mathbf{b},\theta_{1},\theta_{2}\\ \mathbf{s},\mathbf{t}.}} \sqrt{\frac{\sum_{t} (s_{t} - \hat{s}(b_{t},\theta_{1},\theta_{2}))^{2}}{t}}{st}}$$

$$b_{t} := \mathbb{I}^{+}$$

$$0 \le b_{t} \le 1$$

$$\sum_{t} b_{t} \le \alpha$$

$$\beta_{1} \le \theta_{1} \le \gamma_{1}$$

$$\beta_{2} \le \theta_{2} \le \gamma_{2}$$
where $\hat{s}_{t}(b_{t},\theta_{1},\theta_{2}) := b_{t} * \frac{1}{\theta_{2}^{\theta_{1}}\mathcal{T}(\theta_{1})} t^{\theta_{1}-1} e^{-\frac{t}{\theta_{2}}}$
(5)

At this point, it is imperative to make the distinction between the GAGamma and the existing HSA and BSA algorithms. Apart from exhibiting similarities in convolution framework, HSA and BSA also resemble GAGamma as methods of stimulus estimation using FIR. Nevertheless, the data encoding approach in HSA and BSA use a deconvolution (of Eq. (2)) approach contrary to the optimisation approach in GAGamma. The knowledge-injection component of GAGamma, as part of formalisation of \hat{s} and the optimisation approach, has two distinct benefits over the deconvolution-based methods:

- We have used a generic Gamma function as the knowledge-injection component to \hat{s} in GAGamma which is driven by the existing knowledge about the fMRI data as opposed to the generic sinusoidal function used as the FIR in BSA. We also argue that this formulation allows the inclusion of additional knowledge about the data source (such as systematic noise) if present, providing greater flexibility in the formulation of the encoding algorithm.
- The optimisation problem formulation in GAGamma jointly optimises for the parameter set Θ and **b**. This formulation thus includes the parameter set Θ of the prediction model \hat{s} and the spike sequence **b** for each individual voxel or feature. In HSA and BSA, the equivalent filter parameters are predetermined for the whole set of voxels and are not learned from the data.
- The constraint $\Sigma_t b_t \leq \alpha$ in GAGamma ensures the flexibility in choosing the desired spike density, hence the ability to control the compression and quality of signal reconstruction. The BSA or HSA algorithm, on the contrary, accommodates no such control in the encoding framework.

4. Experimental results

The experiments described in this article were performed on the publicly available benchmark Starplus fMRI dataset [28] collected by The Centre for Cognitive Brain Imaging, Carnegie Mellon University. The starplus experiment was conducted on a set of 7 subjects. Each subject had undergone multiple trials of the exactly same cognitive experiment. At every trial lasting for 27 s, a set of stimuli were presented to a subject in the following order:

- 1. The first stimulus (picture or sentence) was presented at the beginning for 4 s.
- 2. A blank screen was presented during the interval of 5 8 s.
- 3. The second stimulus (sentence or picture) was presented during the interval of 9 12 s.
- 4. A rest period of 15 s was added after the presentation of the second stimulus.

While the subject performed the cognitive tasks, fMRI images from specific regions of interest (ROI) of the brain were collected at every 500 ms interval. The preprocessed fMRI dataset has been used in a number of pattern recognition studies [29,30,36]. In our study, we have chosen this dataset to build pattern recognition systems that can predict and discriminate between the binary cognitive states of a subject 'seeing a picture' and 'reading a sentence'. We have chosen two subjects (id: 04847 and 07510) randomly and used two spatial ROIs; Calcarine Sulcus ('CALC') and Left Intra-Parietal Sulcus ('LIPL') for our experiments. The choice of the ROI is based on previous work [11] that found these ROIs to be amongst the most discriminatory in the raw continuous data space. The dataset for each subject is composed of 40 samples (trials) of each class, and each sample is made up of 452 and 483 voxels in subject 04847 and 05710 respectively. Each cognitive task lasted for a total of 8 s emitting 16 fMRI images for each class within a trial.

We have used three metrics to evaluate and compare the performance of the encoding techniques and the traditional 'no encoding' (raw data) approach. The evaluation criteria and the baseline encoding techniques are elaborated below:

Bit compression ratio (BCR): We define BCR as the ratio between the average number of bits required to store an encoded dataset and the number of bits required to store a raw dataset, respectively. BCR is directly associated with the relative description lengths (DL) and data type of the datasets. DL of a dataset is described by the length of the dataset represented by the number of values in the dataset. Let us assume a dataset intended for pattern recognition is represented by $D_{raw} := \{x_1, x_2, \dots x_n | type(x_i) = \mathbb{R}\}$ which is transformed by an encoding algorithm to $D_{encoded} := \{y_1, y_2, \dots y_m | type(y_j) = \mathbb{I}^+\}$, where *m* and *n* are the DL of the raw and the encoded data respectively. The BCR is then estimated as:

$$BCR := \frac{m \times sizeof(\mathbb{I}^+)}{n \times sizeof(\mathbb{R})}$$
(6)

The notion of BCR (Eq. (6)) can be analysed from the viewpoint of the Kolmogorov complexity. As described earlier, Kolmogorov's descriptional complexity aims at simpler object representation and simplicity is measured by the DL of the object. Here, the object being a pattern recognition dataset, the objective is to achieve simpler representation of the dataset by performing the encoding operation. This is achieved by minimising the numerator $m \times sizeof(\mathbb{I}^+)$. A compression is said to be achieved, if 0 < BCR < 1 is satisfied. It is also quite evident from Eq. (6) that the data type of the objects present in the dataset contributes significantly to the BCR metric. In our case, the encoded data being represented by positive integers (spike-timings) as opposed to the floating-point numbers in the raw data already contribute significantly to BCR. Additionally, the temporal encoding algorithms aspire to minimise the DL of the object ($m \ll n$), thus achieving a lower BCR.

Decoding error: The decoding error metric is the measure of the decompression reliability, *i.e.* the ability to recover the original signal from the compressed spike-timings reliably. We have used the RMSE of signal reconstruction between the original signal *s*, and the predicted signal *ŝ*. The RMSE is given by:

$$RMSE := \sqrt{\frac{\sum_{t} (s - \hat{s}(b_t, \Theta))^2}{t}}$$
(7)

A low RMSE of the signal reconstruction indicates higher preservation of the original data in the spike-timings. However, low RMSE is not necessarily indicative of a better encoding for pattern recognition. For example, an encoder producing better reconstruction error for noisy data may indicate inefficient noise filtering. It must also be noted that the prediction models are built on the spike-time data and have no knowledge of the mapping $\mathbf{s} \rightarrow \mathbf{b}$ being performed beforehand. Hence, although this metric plays an important role in evaluating the robustness of the encoding algorithm with respect to the reconstruction of the raw data, the effect on the quality of pattern recognition performance is unaffected.

Classification performance: From the pattern recognition viewpoint, the classification performance is the most important measure of success. To evaluate the classification performance, we have used the combination of mean classification accuracy and precision. The mean accuracy is estimated from thirty independent runs of 50/50 train/test split of the binary classification data described previously.

Because the data we encode is intended to be used for pattern recognition problems, conservation and possible enhancement of the discriminatory information in the spike-timings is as crucial as efficient compression of the data. This is a distinctly different approach from the existing ones in pattern recognition. In the traditional pattern recognition approach, the volume of the data plays a crucial role in the performance of the pattern recognition algorithms to produce accurate predictions. In the temporal encoding approach, by keeping both compressibility and classification performance as the criteria of evaluation, we are aiming to benefit from the efficient representation of information in the data along with the classification performance. It is thus important to have a balance between compression and conservation of discriminatory information in the encoded data.

Fig. 1 shows a flowchart of the experimental design used in this article. The experimental protocol begins with the raw temporal data **s**. At the first step, the encoding operation is performed on **s** to generate the encoded spike-time data **b**. In the second step, a K-NN based prediction model is learned using a fraction of **b**. The rest of **b** is used to test the performance of the model emitting the prediction performance. The next evaluation criteria relate to the compressibility of the encoding algorithms. To evaluate the compressibility, BCR is calculated by comparing **b** and **s**. Finally, the lossiness of the encoding operation is evaluated by comparing predicted signal \hat{s} produced by the decoding algorithm (corresponding to the encoding algorithm) (seeAppendix A) and the ground truth *s* to emit the decoding error.

K-Nearest Neighbour (K-NN) algorithm and distance functions: As discussed earlier, we have used the non-parametric K-NN algorithm for building the classification model from the data. The class label prediction of a new sample (in our case a spike-train) in K-NN is a majority vote between the neighbours of the new sample, where the sample is assigned to the class label most common among its K nearest neighbours. To evaluate neighbourhood of a sample, it is hence necessary to calculate pairwise distances between the sample to be predicted and the training samples. As we are interested in learning a K-NN model for both raw (in continuous space) and spike-time (discrete space) data, we have used two different distance functions raw and encoded data. For the raw data in the continuous space, we have used the standard Euclidean distance or the L^2 norm as the distance function. On the other hand, for the spike-time dataset, we have used the spike asynchronicity based distance function and is described below:



Fig. 1. Flowchart depicting the evaluation criteria and experimental protocol used in this research.

Spike asynchronicity based distance function: Here we propose a distance function that can capture relative distance between a pair of the spike-train samples. Since we are concerned with using spike-timings as a carrier of information, a useful way to capture similarity between a pair of spike-train samples is to record if the two samples have spiked at the same time instance. We have used mean absolute asynchronicity as the distance function. The mean absolute asynchronicity based distance function between two spike-train samples $\mathbf{b_1} \in \{0, 1\}^{T \times M}$ and $\mathbf{b_2} \in \{0, 1\}^{T \times M}$ is formalised as the mean pairwise Hamming distance between all feature-wise pairs $\mathbf{b_1}^m$ and $\mathbf{b_2}^m$, where *M* is the feature count. As the spike-time data lies in the binary space, the mean pairwise Hamming distance is equivalent to the mean XOR distance between the pairs of spike sequences.

In this study, we have compared and evaluated three different encoding methods. It must be noted that for each encoding or compression algorithm, there's also a decoding algorithm which decompresses the spike-trains into the reconstructed signal **ŝ**.

- GAGamma: The GAGamma encoding method is outlined and described in Section 3. The encoding and decoding principles are given by Eqs. (5) and (3).
- BSA: The BSA encoding and decoding algorithms [33] are formalised in Appendix A (Algorithms 1 and 2), respectively. The BSA encoding algorithm takes a filter function and a threshold value as input along with the signal *s*. The deconvolution approach of BSA begins with a FIR filter, and at every time instant τ calculates two error metrics: $\sum_{k=0}^{p} abs(s(k+\tau) h(k))$ and $\sum_{k=0}^{p} abs(s(k+\tau))$, where *P* is the number of filter taps. If the first error is less than the second error minus the threshold, then the BSA encoder fires a spike and subtracts the filter from the input [33].
- Temporal contrast: The temporal contrast algorithm captures the greater than average changes in the data as spikes. Algorithms 3 and 4 presents the temporal contrast encoding and decoding algorithms respectively. One major characteristic and deviation of temporal contrast algorithm from the temporal encoding framework is its ability to generate spikes with positive and negative polarity. Since we are only interested in the spike-timings, during the classification, we have ignored the polarity of the spikes. The algorithm takes the *factor* \in {0, 1} parameter as input. This parameter controls the estimate of the *threshold*_{TC} variable, which is responsible for determining the spike-timings.

For the comparative evaluation of the encoding methods with the classical 'no encoding' (raw data) method, we have replicated the experiments for the subjects 04847 and 07510. For the GAGamma encoding, we have used two sets of hyper-parameter to demonstrate the tuning capability of the algorithm. In the GAGamma-16 method we used the hyperparameter values [$\alpha = 16$, $\beta = 0$, $\gamma = 10$] (see Eq. (5)) and in GAGamma-3 method we used [$\alpha = 3$, $\beta = 0$, $\gamma = 10$]. The BSA encoding algorithm takes a finite impulse response *filter* and a *threshold*_{BSA} as input. In our experiments, we have used the low pass FIR filter of size 10 and the *threshold*_{BSA} = 0.95. These values are guided by the existing literature present on the application of BSA on brain data [32]. For the Temporal contrast encoding, we have used the hyper-parameter *factor* = 0.6. As a

Table	1
-------	---

Comparative evaluation of the data encoding algorithms applied to subject 04847 and 07510 in the Starplus fMRI dataset.

Subject id	Method	Data type ^a	BCR	Decoding error	Classification performance (K^b)
04847	GAGamma-16	Integer	0.15	0.07	87.41 ± 4.80%(16)
	GAGamma-3	Integer	0.04	0.29	85.02 ± 4.76%(11)
	BSA	Integer	0.08	0.15	84.50 ± 4.47%(3)
	Temporal contrast	Integer	0.06	0.23	54.16 ± 5.77%(1)
	No encoding	Float	1	-	89.55 ± 4.60%(1)
07510	GAGamma-16	Integer	0.15	0.06	$76.00 \pm 5.89(8)$
	GAGamma-3	Integer	0.04	0.27	81.16 ± 7.50(2)
	BSA	Integer	0.04	0.15	74.08 ± 6.71%(8)
	Temporal contrast	Integer	0.05	0.26	52.75 ± 5.84%(2)
	No encoding	Float	1	-	79.11 ± 3.99%(5)
	Random	Integer	0.11	-	52.58 ± 4.79%(1)

^a An integer is assumed to take 8 bits and floating-point number 32 bits.

^b K: Number of nearest neighbours used in K-NN algorithm.

baseline, we have also included a randomly generated spike-train dataset. The random spike-time dataset is created using a Poisson's distribution function ($\lambda = 0.6$). Varying the λ parameter effects the BCR directly for random spike generation. It must be noted that the presented results are non-exhaustive in the hyperparameter space of different encoding methods. In the 'no encoding' method, the raw dataset was created by transforming each multi-dimensional time series (set of images) within a trial into a single static observation by concatenating the feature values across the 16 time intervals similar to the approach employed in [29].

Comparison between 'encoding' and 'no encoding': The advantages of performing encoding as opposed to using the raw data ('no encoding') is well established in Table 1. This table shows the experiments replicated across subjects 04847 and 07510. For every subject, we evaluated three temporal encoding methods. They are the proposed GAGamma, BSA and the temporal contrast encoding. In the GAGamma-X method, X corresponds to the value of the maximum number of allowed spikes, α (Eq. (5)). The temporal encoding methods are compared against the raw data or the 'no encoding' method along with a random spike-train as a baseline. As discussed earlier, the encoding error metric is not applicable for the 'random' and 'no encoding' method. This is because in 'no encoding' and 'random' method we use the raw data and a random spike generator respectively for pattern recognition and thus encoding principles are not applied in these two cases.

It is observed that the encoding operations could compress the data dramatically and thus attaining BCR values between 0.04 and 0.15 *i.e.* an approximate compression of 6 to 25 times compared to the raw data. The most important metric of evaluation, the classification performance column in Table 1 shows that the GAGamma and BSA methods achieve comparable classification performances with respect to the 'no encoding' method, and hence can capture the discriminatory information in the spike-time data well. It should also be noted that for subject 07510 the GAGamma-3 method achieved a classification performance of $81.16 \pm 7.50\%$ as opposed to the $79.11 \pm 3.99\%$ by the raw data and thus outperforming the 'no encoding' method. This, we argue is due to the ability of the encoding algorithms to represent the information in the raw data into the spike-time and thus concisely present to the classifier.

Comparison of the temporal encoding methods: Fig. 3 graphically depicts the quality comparison of the different encoding technique. The plot shows the mean BCR and the classification accuracy of the encoding techniques across the two subjects. The horizontal and vertical error bars are the standard deviations of the BCR and accuracy respectively. We first observe that the GAGamma-16 and GAGamma-3 encodings show superior mean performance in the pattern recognition task compared to other methods. However, our aim is to simultaneously achieve high classification performance and a highly compressed information representation, and thus, the GAGamma-3 and BSA data points residing on the top left quadrant of the plot fares better overall in both respects. From the error bar representations, it can also be seen that the GAGamma method has a negligible deviation on BCR. This is due to the flexibility that the GAGamma method provides to control the spike density by the constraint $\Sigma_t b_t \leq \alpha$ (Eq. (5)) without sacrificing much pattern recognition performance. This can be of significant importance, especially for the storage and transmission of large volumes of streaming data within limited resources, where the encoding operation can precisely tune the compression rate and hence the storage. It is also recognised from the plot that the temporal contrast encoding method fares poorly in this experiment and is no better than a random spike generator.

Table 1 also shows an inverse relationship between the BCR and the decoding error. This is because it requires significantly more effort to accurately represent the seasonal variations in the data using less number of spikes. However, if decoding robustness is of major importance, the GAGamma method can be tuned to maximise spike density and thus will have minimal loss in the signal reconstruction and thus sacrificing the compression. Fig. 2 shows an example of the signal reconstruction done by the GAGamma-16, BSA and temporal contrast decoding algorithms. It can be seen that the signal



Fig. 2. Comparison of signal reconstruction (\$) from a spike sequence by GAGamma-16 decoding algorithm (Eq. (3)), BSA decoding algorithm (Algorithm 2) and Temporal contrast decoding algorithm (Algorithm 4). The true signal is randomly selected from subject 04847 (10th trial and 23rd voxel).



Fig. 3. Plot illustrating the comparison of the quality of the encoding methods with respect to the mean classification accuracy and bit compression ratio across the two subjects. The horizontal and the vertical error bars represent the standard deviation of accuracy and BCR across experiments.

reconstruction by GAGamma is superior from the others as not only can it reconstruct the trends in the signal but also the seasonal variations.

Additionally, GAGamma-16 encoded spikes for the 'seeing a picture', and the 'reading a sentence' stimuli were independently analysed for interpreting the discriminating spatio-temporal influence of the spikes. As described earlier in the experimental protocol, the presentation of a certain stimulus within a trial follows an order, *i.e.* for each stimuli class there exist subclasses of 'presented first' or 'presented second'. To analyse the effect of the first or second presentation of stimuli, we have separated the encoded dataset into four classes, 'picture presented first', 'picture presented second', 'sentence presented first' and 'sentence presented second'. Figs. 4 and 5 show the comparison of the mean spike percentage across the trials for the four subclasses in subject 04847 and subject 07510. The points in the 3D plot correspond to the spatial location of the voxels. Each voxel belongs to two physiologically defined clusters or regions of interest, namely 'CALC' and 'LIPL'. The top row plots are the 'picture' trials, and the bottom row trials are the 'sentence' plots. The columns correspond



(a) 'seeing a picture' first during trial



(b) 'seeing a picture' second during trial



(c) 'reading a sentence' first during trial (d) 'reading a sentence' second during trial

Fig. 4. Comparative analysis of spike frequencies of the subject 04847 seeing picture vs. reading a sentence. Each colour bar represents the mean spike frequency across the trials for the corresponding class.

to the stimulus ('picture' or 'sentence') being presented first or second. The two clusters in each of the 3D plots relate to the two ROI's (top left is 'LIPL' and bottom right is 'CALC') of the brain structure. Functionally, the 'CALC' region is responsible for central and peripheral vision whereas the 'LIPL' region is related to visual attention. In both the subjects, 'reading a sentence second' after 'seeing a picture first' has more spike activity on average across the trials than the other way around, especially in the 'LIPL' region. The mean spike activity in the 'LIPL' is observed to be relatively higher (0.59 and 0.57) when the subjects were 'reading a sentence' than when the subjects were 'seeing a picture' (0.54 and 0.55). A two-sample T-test was conducted between the 'seeing a picture' and the 'reading a sentence' class in the 'LIPL' region for the subjects to validate the previous result. The null hypothesis for the test conducted was the following, H_0 : 'there is no difference between the picture spike activity and sentence spike activity'. The null hypothesis was rejected at 5% significance level with $p = 5.27 \times 10^{-18}$ for subject 04847 and with $p = 7.05 \times 10^{-12}$ for subject 07510. Hence, as per the T-test, the average spike activity across the trials over time for 'seeing a picture' is significantly different from the average spike activity across trials over time for 'reading a sentence'. Further, it must also be noted the sentences shown as part of the experiment are highly visual in nature (e.g. 'It is not true that the dollar is below the plus.') and requires a high image comprehension ability. This result is, therefore, consistent with the experimental results [21] obtained earlier which shows a greater degree of activation and functional connectivity in the 'LIPL' region during cognitive tasks associated with high imagery sentence comprehension. This, in fact, also validates the ability of the proposed encoding algorithm to preserve the useful discriminatory information in the compressed encoded space of data.

Table 2 relates to the reproducibility of the spike-timings produced by the mixed integer genetic algorithm solver for the GAGamma encoding. The genetic algorithm being an evolutionary optimisation solver produces non-reproducible result when on multiple iterations. Nevertheless, a pareto-optimal fitness value is guaranteed on each iteration. To validate the reliability of the GAGamma optimisation, we have conducted ten independent runs of GAGamma encoding in three random voxels (30, 468 and 3429) from trial 12 of subject 04847. Table 2 compares the similarity of the spike-trains produced by the GAGamma encoding using two spike-asynchronicity measures. They are the percentage asynchronicity d_p and Victor Purpura distance d_{vp} respectively. The Victor Purpura distance (d_{vp}) [38] metric is a cost based distance metric.



(c) 'reading a sentence' first during trial (d) 'reading a sentence' second during trial

Fig. 5. Comparative analysis of spike frequencies of the subject 07510 seeing picture vs. reading a sentence. Each colour bar represents the mean spike frequency across the trials for the corresponding class.

Table 2

Average pairwise asynchronicity of three different voxels at the end of ten independent runs of GAGamma encoding.

 Voxel ID
 dp
 dvp

09	30
10	468
11	3429
	3429

The distance is defined by the minimum cost of converting one spike-train into the other using three operations; insertion (cost 1), deletion (cost 1) and shifting a spike by an interval δt (cost $q|\delta t|$). For the smaller value of q the distance metric approximates the spike count difference and hence supports rate coding. A higher penalty value of q, on the contrary, supports the number of non-coincidental spikes and hence temporal encoding. The comparison of the spike synchronicity using d_p and d_{vp} in Table 2 shows that the spike-timings are correctly reproduced approximately 75% of times.

5. Discussion

In this article, we focused on using temporal encoding as a framework to concisely represent large volumes of data by spike-timings and by doing so, preserve the existing discriminatory spatio-temporal information. In this regard, apart from using the existing temporal encoding techniques, we proposed a novel temporal encoding framework and formalised a specific encoding method for fMRI data, called GAGamma. The experimental result on benchmark fMRI dataset shows the superiority of the temporal encoding algorithms such as GAGamma and BSA to succinctly represent the discriminatory information in the compressed encoded spike space without losing any appreciable amount of information. Thus, it achieves comparable or superior pattern recognition performance. We argue that the flexibility of the proposed encoding framework lies in its ability to inject known structure information about the data source and thus provide the compression/encoding algorithms sufficient redundancy to represent the large dataset in an optimally concise manner. We articulated the flexibility of the GAGamma encoding technique to precisely control the compression and the quality of the signal reconstruction by tuning the hyperparameters without much effect on the classification performance.

In the future, we intend to use and extend the generalised framework of data encoding proposed here on large radio astronomy streaming data and geophysical data, including multi-sensory earthquake data. The proposed encoding method shows promise in opening up new opportunities in data compression, communication and analysis across different applications including non-invasive brain activity data analysis (EEG, fMRI) [40–41], environmental data analysis [43], remote sensing data analysis [44,45], brain-computer interfaces and so on. The proposed spike-time encoding method is suitable and is intended to be integrated as part of the neuromorphic spiking neural network based pattern recognition system, such as NeuCube [22,23].

Appendix A. Algorithm descriptions

Algorithm 1	BSA	encoding	algorithm.
-------------	-----	----------	------------

```
1: input: s, filter, threshold<sub>BSA</sub>
2: output: b
3: b ⇐ 0
4: L = length(s)
5: F = length(filter)
6: for t = 1: (L - F + 1) do
7:
     e_1 \leftarrow 0
8:
     e_2 \leftarrow 0
      for k = 1 : F do
9:
10:
        e_1 += |s(t+k) - filter(k)|
        e_2 += |s(t+k-1)|
11:
      end for
12.
      if e_1 \leq (e_2 - threshold_{BSA}) then
13:
        b(t) \leftarrow 1
14:
15:
        for k = 1 : F do
           s(i+j-1) = filter(k)
16:
        end for
17.
      end if
18:
19: end for
```

Algorithm 2 BSA decoding algorithm.

1: input: b, filter 2: output: \hat{s} 3: L = length(b)4: F = length(filter)5: **for** t=1:L-F+1 **do** if b(t) == 1 then 6: 7: **for** k = 1 : F **do** $\hat{s}(t+k-1) += filter(k)$ 8: end for ٩· end if 10: 11: end for

Algorithm 3 Temporal contrast encoding algorithm.

1: input: s, factor 2: output: b, threshold_{TC} 3: $L \leftarrow length(s)$ 4: for t = 1 : L - 1 do $diff \leftarrow |s(t+1) - s(t)|$ 5: 6: end for 7: threshold_{TC} \leftarrow mean(diff) + factor \cdot std(diff) 8: $diff \leftarrow [0, diff]$ 9: **for** *t* = 1 : *L* **do if** $diff(t) > threshold_{TC}$ **then** 10. 11. $b(t) \leftarrow 1$ else if $diff(t) < -threshold_{TC}$ then 12: 13: $b(t) \leftarrow -1$ else 14: $b(t) \leftarrow 0$ 15: 16. end if 17: end for

Algorithm 4 Temporal contrast decoding algorithm.

1: input: b, threshold_{TC} 2: output: \hat{s} 3: $\hat{s} \leftarrow 0$ 4: $L \leftarrow length(b)$ 5: for t = 2: *L* do if $\hat{s}(t) > 0$ then 6٠ $\hat{s}(t) \leftarrow \hat{s}(t-1) + threshold_{TC}$ 7: else if $\hat{s}(t) < 0$ then 8. $\hat{s}(t) \leftarrow \hat{s}(t-1) - threshold_{TC}$ 9: else 10. $\hat{s}(t) \leftarrow \hat{s}(t-1)$ 11. end if 12. 13: end for

References

- [1] F.G. Ashby, Statistical Analysis of fMRI Data, MIT press, 2011.
- [2] B. Babu, M. Jehan, Differential evolution for multi-objective optimization, in: Evolutionary Computation, 2003. CEC'03. The 2003 Congress on, 4, IEEE, 2003, pp. 2696–2703.
- [3] G.M. Boynton, S.A. Engel, G.H. Glover, D.J. Heeger, Linear systems analysis of functional magnetic resonance imaging in human v1, J. Neurosci. 16 (13) (1996) 4207–4221.
- [4] E.N. Brown, R.E. Kass, P.P. Mitra, Multiple neural spike train data analysis: state-of-the-art and future challenges, Nat. Neurosci. 7 (5) (2004) 456-461.
- [5] G.J. Chaitin, On the length of programs for computing finite binary sequences, J. ACM (JACM) 13 (4) (1966) 547-569.
- [6] Z. Cho, S. Park, J. Kim, S. Chung, S. Chung, J. Chung, C. Moon, J. Yi, C. Sin, E. Wong, Analysis of acoustic noise in mri, Magn. Reson. Imaging 15 (7) (1997) 815–822.
- [7] H. De Garis, An artificial brain ATR's CAM-Brain project aims to build/evolve an artificial brain with a million neural net modules inside a trillion cell cellular automata machine, New Gen. Comput. 12 (2) (1994) 215–221.
- [8] H. De Garis, N.E. Nawa, M. Hough, M. Korkin, Evolving an optimal de/convolution function for the neural net modules of atr's artificial brain project, in: Neural Networks, 1999. IJCNN'99. International Joint Conference on, 1, IEEE, 1999, pp. 438–443.
- [9] K. Deb, An efficient constraint handling method for genetic algorithms, Comput. Methods Appl. Mech. Eng. 186 (2) (2000) 311-338.
- [10] K. Deep, K.P. Singh, M.L. Kansal, C. Mohan, A real coded genetic algorithm for solving integer and mixed integer optimization problems, Appl. Math. Comput. 212 (2) (2009) 505–518.
- [11] L.-N. Do, H.-J. Yang, A robust feature selection method for classification of cognitive states with fMRI data, in: Advances in Computer Science and its Applications, Springer, 2014, pp. 71–76.
- [12] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, Syst., Man, Cybern., Part B, IEEE Trans. 26 (1) (1996) 29-41.
- [13] K.J. Friston, O. Josephs, G. Rees, R. Turner, Nonlinear event-related responses in fmri, Magn. Reson. Med. 39 (1) (1998) 41-52.
- [14] G.H. Glover, Deconvolution of impulse response in event-related bold fmri 1, Neuroimage 9 (4) (1999) 416–429.
- [15] T. Gollisch, M. Meister, Rapid neural coding in the retina with relative spike latencies, Science 319 (5866) (2008) 1108-1111.
- [16] P. Grunwald, P. Vitányi, Shannon information and kolmogorov complexity, arXiv preprint cs/0410002 (2004).
- [17] F. Hafiz, A.A. Shafie, Encoding of facial images into illumination-invariant spike trains, in: Computer and Communication Engineering (ICCCE), 2012 International Conference on, IEEE, 2012, pp. 132–137.
- [18] R.M. Hallock, P.M. Di Lorenzo, Temporal coding in the gustatory system, Neurosci. Biobehav. Rev. 30 (8) (2006) 1145-1160.
- [19] M. Hough, H. De Garis, M. Korkin, F. Gers, N.E. Nawa, SPIKER: Analog waveform to digital spiketrain conversion in atrs artificial brain (cam-brain) project, in: International Conference on Robotics and Artificial Life, Citeseer, 1999.

- [20] T. lakymchuk, A. Rosado-Munoz, M. Bataller-Mompean, J. Guerrero-Martinez, J. Frances-Villora, M. Wegrzyn, M. Adamski, Hardware-accelerated spike train generation for neuromorphic image and video processing, in: Programmable Logic (SPL), 2014 IX Southern Conference on, IEEE, 2014, pp. 1–6.
- [21] M.A. Just, S.D. Newman, T.A. Keller, A. McEleney, P.A. Carpenter, Imagery in sentence comprehension: an fmri study, Neuroimage 21 (1) (2004) 112–124.
 [22] N. Kasabov, N.M. Scott, E. Tu, S. Marks, N. Sengupta, E. Capecci, M. Othman, M.G. Doborjeh, N. Murli, R. Hartono, et al., Evolving spatio-temporal data machines based on the neucube neuromorphic framework: design methodology and selected applications, Neural Netw. 78 (2016) 1–14.
- [23] N.K. Kasabov, Neucube: a spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data, Neural Netw. 52 (2014) 62–76.
- [24] A.N. Kolmogorov, Three approaches to the quantitative definition of information', Probl. Inf. Transm. 1 (1) (1965) 1–7.
- [25] W. Maass, Networks of spiking neurons: the third generation of neural network models, Neural Netw. 10 (9) (1997) 1659–1671.
- [26] Z.F. Mainen, T.J. Sejnowski, Reliability of spike timing in neocortical neurons, Science 268 (5216) (1995) 1503–1506.
- [27] J.H. Maunsell, J.R. Gibson, Visual response latencies in striate cortex of the macaque monkey, J. Neurophysiol. 68 (4) (1992) 1332-1344.
- [28] T. Mitchell, W. Wang, Starplus fMRI data, 2001.
- [29] T.M. Mitchell, R. Hutchinson, M.A. Just, R.S. Niculescu, F. Pereira, X. Wang, Classifying instantaneous cognitive states from fMRI data, American medical informatics association annual symposium, 2003.
- [30] T.M. Mitchell, S.V. Shinkareva, A. Carlson, K.-M. Chang, V.L. Malave, R.A. Mason, M.A. Just, Predicting human brain activity associated with the meanings of nouns, Science 320 (5880) (2008) 1191–1195.
- [31] M.D. Nunez, P.L. Nunez, R. Srinivasan, Handbook of Neuroimaging Data Analysis, Chapman & Hall/CRC.
- [32] N. Nuntalid, K. Dhoble, N. Kasabov, Eeg classification with bsa spike encoding algorithm and evolving probabilistic spiking neural network, in: International Conference on Neural Information Processing, Springer, 2011, pp. 451–460.
- [33] B. Schrauwen, J. Van Campenhout, Bsa, a fast and accurate spike train encoding scheme, in: Proceedings of the international joint conference on neural networks, 4, IEEE Piscataway, NJ, 2003, pp. 2825–2830.
- [34] N. Sengupta, N. Scott, N. Kasabov, Framework for knowledge driven optimisation based data encoding for brain data modelling using spiking neural network architecture, in: Proceedings of the Fifth International Conference on Fuzzy and Neuro Computing (FANCCO-2015), Springer, 2015, pp. 109–118.
 [35] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. 27 (3) (1948) 379–423.
- [36] S.V. Shinkareva, R.A. Mason, V.L. Malave, W. Wang, T.M. Mitchell, M.A. Just, Using fmri brain activation to identify cognitive states associated with perception of tools and dwellings, PLoS One 3 (1) (2008) e1394.
- [37] C.V.R. Sierra, M.J. Versluis, J.M. Hoogduin, H. Duifhuis, Acoustic fmri noise: linear time-invariant system model, IEEE Trans. Biomed. Eng. 55 (9) (2008) 2115–2123.
- [38] J.D. Victor, K.P. Purpura, Metric-space analysis of spike trains: theory, algorithms and application, Netw. 8 (2) (1997) 127-164.
- [39] L. Yiqing, Y. Xigang, L. Yongjian, An improved pso algorithm for solving non-convex nlp/minlp problems with equality constraints, Comput. Chem. Eng. 31 (3) (2007) 153–162.
- [40] N.K. Kasabov, M.G. Doborjeh, Z.G. Doborjeh, Mapping, Learning, Visualization, Classification, and Understanding of fMRI Data in the NeuCube Evolving Spatiotemporal Data Machine of Spiking Neural Networks, IEEE Trans. on Neural Net. and Learn. Sys. 28 (4) (2017) 887–899.
- [41] N Kasabov, E. Capecci, Spiking neural network methodology for modelling, classification and understanding of EEG spatio-temporal data measuring cognitive processes, Inform. Sci. 294 (2015) 565–575.
- [42] M.G. Doborjeh, G.Y. Wang, N.K. Kasabov, K. Nikola, R. Kydd, B. Russell, A spiking neural network methodology and system for learning and comparative analysis of EEG data from healthy versus addiction treated versus addiction not treated subjects, IEEE Trans. on Biom. Engineering 63 (9) (2016) 1830–1841.
- [43] N. Kasabov, V. Feigin, Z.-G. Hou, Y. Chen, L. Liang, R. Krishnamurthi, M. Othman, P. Parmar, Evolving spiking neural networks for personalised modelling, classification and prediction of spatio-temporal patterns with a case study on stroke, Neurocomputing 134 (2014) 269–279.
- [44] P. Boses, N.K. Kasabov, L. Bruzzone, R.N. Hartono, Spiking Neural Networks for Crop Yield Estimation Based on Spatiotemporal Analysis of Image Time Series, IEEE Transactions on Geoscience and Remote Sensing 54 (11) (2016) 6563–6573.
- [45] T. Gao, N. Kasabov, Adaptive cow movement detection using evolving spiking neural network models, Evolving Systems 7 (4) (2016) 277-285.