Dear Author

Here are the proofs of your article.

- You can submit your corrections **online,** via **e-mail** or by **fax**.
- For **online** submission please insert your corrections in the online correction form. Always indicate the line number to which the correction refers.
- You can also insert your corrections in the proof PDF and **email** the annotated PDF.
- For **fax** submission, please ensure that your corrections are clearly legible. Use a fine black pen and write the correction in the margin, not too close to the edge of the page.
- Remember to note the **journal title**, **article number**, and **your name** when sending your response via e-mail or fax.
- **Check** the metadata sheet to make sure that the header information, especially author names and the corresponding affiliations are correctly shown.
- **Check** the questions that may have arisen during copy editing and insert your answers/corrections.
- **Check** that the text is complete and that all figures, tables and their legends are included. Also check the accuracy of special characters, equations, and electronic supplementary material if applicable. If necessary refer to the *Edited manuscript*.
- The publication of inaccurate data such as dosages and units can have serious consequences. Please take particular care that all such details are correct.
- Please **do not** make changes that involve only matters of style. We have generally introduced forms that follow the journal's style.
- Substantial changes in content, e.g., new results, corrected values, title and authorship are not allowed without the approval of the responsible editor. In such a case, please contact the Editorial Office and return his/her consent together with the proof.
- If we do not receive your corrections **within 48 hours**, we will send you a reminder.
- Your article will be published **Online First** approximately one week after receipt of your corrected proofs. This is the **official first publication** citable with the DOI. **Further changes are, therefore, not possible.**
- The **printed version** will follow in a forthcoming issue.

Please note

After online publication, subscribers (personal/institutional) to this journal will have access to the complete article via the DOI using the URL:

`http://dx.doi.org/10.1007/s12559-019-09643-1`

If you would like to know when your article has been published online, take advantage of our free alert service. For registration and further information, go to: http://www.link.springer.com.

Due to the electronic nature of the procedure, the manuscript and the original figures will only be returned to you on special request. When you return your corrections, please inform us, if you would like to have these documents returned.

Metadata of the article that will be visualized in OnlineFirst

1	Article Title	**Biological Neuron Coding Inspired Binary Word Embeddings**	
2	Article Sub- Title		
3	Article Copyright - Year	**The Author(s) 2019** **(This will be the copyright line in the final PDF)**	
4	Journal Name	Cognitive Computation	
5	Corresponding Author	Family Name	**Zeng**
6		Particle	
7		Given Name	**Yi**
8		Suffix	
9		Organization	Chinese Academy of Sciences
10		Division	National Laboratory of Pattern Recognition, Institute of Automation
11		Address	Beijing, China
12		Organization	Chinese Academy of Sciences
13		Division	Institute of Automation
14		Address	Beijing, China
15		Organization	Chinese Academy of Sciences
16		Division	Center for Excellence in Brain Science and Intelligence Technology
17		Address	Shanghai, China
18		Organization	University of Chinese Academy of Sciences
19		Division	
20		Address	Beijing, China
21		e-mail	yi.zeng@ia.ac.cn
22	Author	Family Name	**Wang**
23		Particle	
24		Given Name	**Yuwei**
25		Suffix	
26		Organization	Chinese Academy of Sciences
27		Division	Institute of Automation
28		Address	Beijing, China
29		Organization	Chinese Academy of Sciences

	Division	Center for Excellence in Brain Science and Intelligence Technology
	Address	Shanghai, China
	e-mail	yuwei.wang@ia.ac.cn
Author	Family Name	**Tang**
	Particle	
	Given Name	**Jianbo**
	Suffix	
	Organization	Chinese Academy of Sciences
	Division	Institute of Automation
	Address	Beijing, China
	Organization	Chinese Academy of Sciences
	Division	Center for Excellence in Brain Science and Intelligence Technology
	Address	Shanghai, China
	e-mail	
Author	Family Name	**Xu**
	Particle	
	Given Name	**Bo**
	Suffix	
	Organization	Chinese Academy of Sciences
	Division	Institute of Automation
	Address	Beijing, China
	Organization	Chinese Academy of Sciences
	Division	Center for Excellence in Brain Science and Intelligence Technology
	Address	Shanghai, China
	Organization	University of Chinese Academy of Sciences
	Division	
	Address	Beijing, China
	e-mail	

Schedule	Received	12 December 2018
	Revised	
	Accepted	12 April 2019

Abstract

Word embeddings are the semantic representations of the words. They are derived from large corpus and work well on many natural language tasks, with one downside of costing large memory space. In this paper, we propose binary word embedding models based on inspirations from biological neuron coding mechanisms, converting the spike timing of neurons during specific time intervals into binary codes, reducing the space and speeding up computation. We build three types of models to post-process the original dense word embeddings, namely, the homogeneous Poission processing-based rate coding model, the leaky integrate-and-fire neuron-based model, and the Izhikevich's neuron-based model. We test our binary embedding models on word similarity and text classification tasks of five public datasets. The experimental results show that the brain-inspired binary word embeddings (which reduce approximately 68.75% of the space) get similar results to original embeddings for word similarity task while better performance than traditional binary embeddings on text classification task.

62 Keywords separated by ' - ' Word embeddings - Neuron coding - Spiking neural networks

63 Foot note information Yuwei Wang and Yi Zeng have equal contribution to this work and should be regarded as co-first authors

Cognitive Computation
https://doi.org/10.1007/s12559-019-09643-1

Biological Neuron Coding Inspired Binary Word Embeddings

Yuwei Wang[1,2] · Yi Zeng[1,2,3,4] · Jianbo Tang[1,2] · Bo Xu[1,2,3]

Received: 12 December 2018 / Accepted: 12 April 2019

Abstract
Word embeddings are the semantic representations of the words. They are derived from large corpus and work well on many natural language tasks, with one downside of costing large memory space. In this paper, we propose binary word embedding models based on inspirations from biological neuron coding mechanisms, converting the spike timing of neurons during specific time intervals into binary codes, reducing the space and speeding up computation. We build three types of models to post-process the original dense word embeddings, namely, the homogeneous Poission processing-based rate coding model, the leaky integrate-and-fire neuron-based model, and the Izhikevich's neuron-based model. We test our binary embedding models on word similarity and text classification tasks of five public datasets. The experimental results show that the brain-inspired binary word embeddings (which reduce approximately 68.75% of the space) get similar results to original embeddings for word similarity task while better performance than traditional binary embeddings on text classification task.

Q1

Keywords Word embeddings · Neuron coding · Spiking neural networks

Introduction

Word embeddings models can convert both semantic and syntactic information of words into dense vectors, for example, Word2Vec [1] and GloVe [2]. Recently, they attract a lot of attention due to their good performances in various natural language processing tasks, such as language modeling [3], parsing [4], sentence classification [5], and machine translation [6].

However, these dense representations are mostly derived from statistical property of large corpus while are lack of interpretability in each dimension of the word vectors. Several works have tried to transform dense word embeddings into sparse ones to improve the interpretability. Murphy et al. introduced a matrix factorization algorithm named non-negative sparse embeddings (NNSE) on co-occurrence matrix to get sparse, effective and interpretable embeddings [7]. Faruqui et al. defined a L_1 regularized objective function and proposed an post-process optimization algorithm to convert original dense embeddings into sparse or binary embeddings. They call them sparse or binary overcomplete word vector [8]. Sun et al. introduced an algorithm to get sparse embeddings during training Word2Vec model through L_1 regularizer on cost function and regularized dual averaging optimization algorithm [9]. For binary word embeddings, there are also some rounding algorithms on converting dense vectors into discrete integer values to reduce memory. Ling et al. proposed post-processing rounding, stochastic rounding, and auxiliary update vectors algorithms for word embeddings with limited memory, which is named as truncated word embeddings [10]. The interpretability issue in these works is mentioned but not demonstrated clearly. In this paper, we want to improve it via a brain-inspired approach, explaining each dimension of word embeddings based on neuron coding models.

In biological brains, the encoding of information in the areas such as inferior temporal visual cortex, hippocampus, orbitofrontal cortex and insula is with sparse distributed

Yuwei Wang and Yi Zeng have equal contribution to this work and should be regarded as co-first authors

Q2 ✉ Yi Zeng
yi.zeng@ia.ac.cn

Yuwei Wang
yuwei.wang@ia.ac.cn

[1] Institute of Automation, Chinese Academy of Sciences, Beijing, China

[2] University of Chinese Academy of Sciences, Beijing, China

[3] Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China

[4] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China

representation [11]. Many experimental evidences have indicated that biological neural systems use the timing of spikes to encode information [12–14]. The spike trains of cell activities during information transition inspire us to combine traditional word embeddings and neuron coding models into binary embeddings. In this paper, we perform post-process operations on original dense word embeddings to get binary ones with inspirations from biological neuron coding models, and the proposed binary embeddings are with less space occupation and with better interpretability than previous models.

Related Works

Neuron Coding

Neuron coding is concerned with describing the relationship between the stimulus and the neuronal responses [15]. A great many efforts have been dedicated to developing techniques to enable the recording of the brain's electrical activity at different spatial scales, such as single cell spike train recording, local field potential (LFP), and electroencephalogram (EEG) [16]. Neuron coding models mainly concern how neurons encode, transmit, and decode information, and their main focus is to understand how neurons respond to a wide variety of stimuli, and to construct models that attempt to predict responses to other stimuli.

Neurons propagate signals by generating electrical pulses called action potentials: voltage spikes that can travel down nerve fibers. For example, sensory neurons change their activities by firing sequences of action potentials in various temporal patterns, with the presence of external sensory stimuli, such as light, sound, taste, smell and touch [16]. It is known that information about the stimulus is encoded in action potentials and transmitted through connected neurons in our brains.

There are various kinds of hypotheses on neuron coding based on recent neurophysiological findings on biological nervous system, mainly including spike rate coding and spike time coding. For spike rate coding, only the firing rate in an interval is concerned as a measurement for information carried. Rate coding is firstly motivated by the observation of the frog cutaneous receptors by Adrian et al. in 1926 that physiological neurons tend to fire more often for stronger stimuli [17]. Spike rate coding has been the main paradigm in artificial neural networks, such as sigmoidal neurons. Meanwhile, the Poisson-like rate coding is widely used by physiologists to describe how the neurons transmit information. Recently, some neurophysiological results show that efficient processing of information is more likely based on precise timing of action potentials rather than firing rate in some biological neural systems [18–20]. For timing coding hypotheses [21], they mostly concentrate on the timing of individual spikes and the typical ones are the time to first spike [22, 23], rank order coding [20, 24], latency coding [25], and phase coding [26].

In our study, we use Poisson-like coding for spike rate coding and various spiking neuron models for time coding. We try to apply these biological neuron coding hypotheses to build binary word embedding models.

Spiking Neural Network Models

Spiking neural networks (SNNs), which are highly inspired from recent advancement in neuroscience, are often referred as the third generation neural network models [27]. Different from traditional neural networks, SNNs consider the timing of individual spikes as the means of communication and neural computation [21].

Spiking neuron models are the basis of SNNs, which describe the properties of certain cells in the nervous system that generate spikes across their cell membrane. The most well-known neuron model is Hodgkin-Huxley model (H-H model). In 1952, Hodgkin and Huxley did experiments on the giant axon of squid with the voltage clamp technique, which punctured the cell membrane and allowed to force a specific membrane voltage or current [28]. The model was proposed by the recordings and fitting results, well describing the change of ion channel and neuron behavior after stimulation.

In the H-H model [29], the semipermeable cell membrane separates the interior of the cell from the extracellular liquid and acts as a capacitor. Because of the active ion transportation through the cell membrane, the ion concentration inside the cell is different from that in the extracellular liquid. The Nernst potential generated by the difference in ion concentration is represented by a battery.

The model takes three types of channel into consideration: a sodium channel, a potassium channel, and an unspecific leakage channel with resistance R. From the definition of a capacity $C = Q/v$ where Q is a charge and v is the voltage across the capacitor, thus:

$$C \cdot \frac{dv}{dt} = -\sum_k I_k(t) + I(t) \tag{1}$$

The leakage channel is described by a voltage-independent conductance $g_L = 1/R$. For the sodium channel and the potassium channel, if both of them are open, they transmit currents with a maximum conductance g_{Na} or g_K, respectively. However, the channels are not always open; the probability that a channel is open is described by additional variables m, n, and h. The combined action of m

and h controls the Na^+ channels while the K^+ gates are controlled by n.

$$\sum_k I_k = g_{Na}m^3h(v-E_{Na})+g_K n^4(v-E_K)+g_L(v-E_L) \quad (2)$$

The parameters E_{Na}, E_K, and E_L are empirical parameters and the gating variables m, n, and h are defined by differential equations [28].

In addition to the H-H model, other types of spiking neuron models have been proposed, such as integrate-and-fire models and variants, Izhikevich's neuron model, and spike response model (SRM). Recently, SNN-based models have been applied in variant AI applications, such as character recognition [30, 31], object recognition [32], image segmentation [33], speech recognition [34], robotics [35], knowledge representation [36], and symbolic reasoning [37]. In this paper, we will use leaky integrate-and-fire model and Izhikevich's neuron model to convert the word embeddings into more explainable binary embeddings.

Word Embedding Models Based on Inspirations from Biological Neuron Coding

The Framework

We build unsupervised models for post-processing binary word embeddings based on two types of brain-inspired models, homogeneous Poisson process and spiking neural networks. Based on preprocessed word embeddings, such as Word2Vec and GloVe, these models convert original dense embeddings into the form of binarization. Different from traditional works on binary word representations, our models are inspired by neuroscience which are biologically plausible and more interpretable.

To mimic information transmission in biological brains, we take temporal information into consideration. As Fig. 1 shows, our models combine original dense word embeddings and neural coding algorithms to get the spiking times of neurons during a given period of time. We denote the original dense word embeddings matrix as W, for each element w_{id}, where $i = 1, 2, \cdots, |N|, d = 1, 2, \cdots, |D|$, $|N|$ represents the total number of words and $|D|$ represents the dimensions of each word. For each word, we build a neural model based on the value of each dimension. And during a given time T, we record the membrane potential for each neuron per Δt, via neural coding algorithms which will describe in "Homogeneous Poisson Process-Based Binary Word Embeddings" and "Spiking Neural Networks Based Binary Word Embeddings." Then, spiking times matrix $S^{(i)}$, which contains all neurons' spiking times for the ith word, will be flattened as a vector $f^{(i)}$ with each row concatenated head to tail. The dimensions for $f^{(i)}$ is $|D| \times (T/\Delta t)$. Finally, to make our model more robust, we introduce the tolerance factor tol. We allow a window of $tol \times \Delta t$ to generate a binary bit, and obtain the binary word embeddings in the following way:

$$\mathbf{b}^{(i)} = [\mathcal{T}(\mathbf{f}^{(i)}_{1:1*tol}), \mathcal{T}(\mathbf{f}^{(i)}_{1*tol+1:2*tol}) \cdots, \mathcal{T}(\mathbf{f}^{(i)}_{(k-1)*tol+1:k*tol}), \cdots, \mathcal{T}(\mathbf{f}^{(i)}_{|D|\times(T/\Delta t)-tol:|D|\times(T/\Delta t)})] \quad (3)$$

The $\mathcal{T}(vector)$ operation means that if there are 1s in the vector, then the bit is 1, otherwise it is 0.

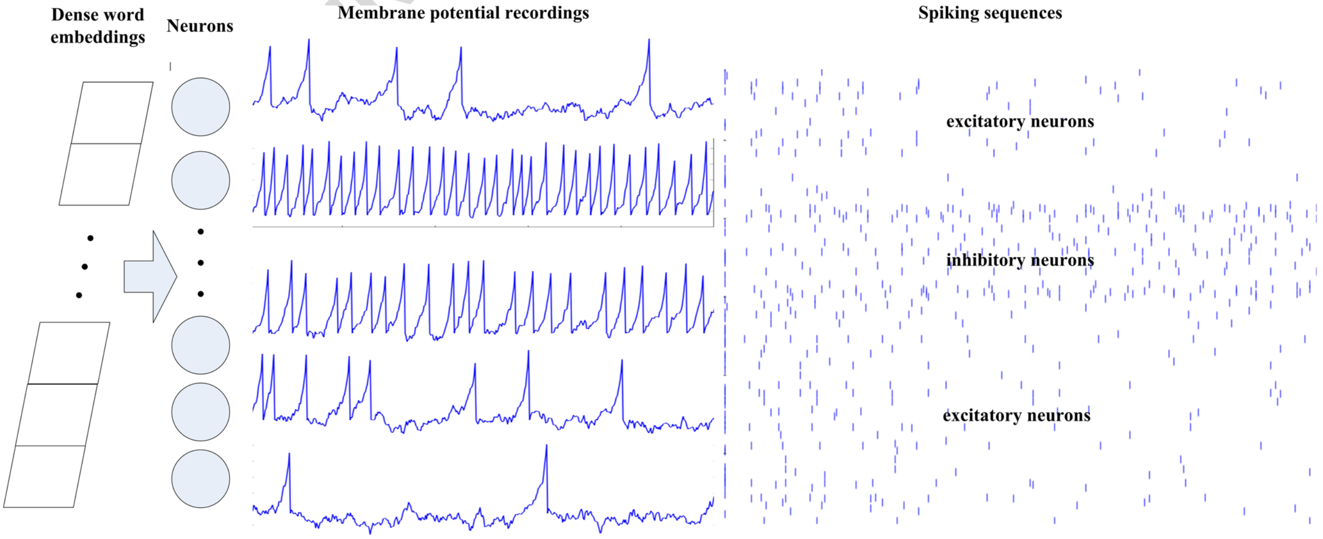

Fig. 1 The framework of neuron coding-based binary embeddings

Homogeneous Poisson Process-Based Binary Word Embeddings

Poisson-like rate coding is a major algorithm to simulate spiking response to stimuli. Biological recordings from medial temporal [38, 39] and primary visual cortex [40] of macaque monkeys have shown good evidence for Poisson process-based coding.

For homogeneous Poisson process, it assumes that for the current spike, there is no dependence at all on preceding spikes and the instantaneous firing rate r is constant over time. Consider that we are given a interval $(0, T)$ and we place a single spike in it randomly. If we pick a subinterval $(t, t + \Delta t)$ of length Δt, the probability that the spike occurred in the subinterval equals $\Delta t/T$. When we place k spikes in $(0, T)$, according to binomial formula, the probability that n of them fall in $(t, t + \Delta t)$ is:

$$P\{n\ spikes\ during\ \Delta t\}=\frac{k!}{(k-n)!n!}(\Delta t/T)^n(1–\Delta t/T)^{k-n} \tag{4}$$

Keeping fire rate $r = k/T$ constant, we increase k and T synchronously. As $k \rightarrow \infty$, the probability becomes:

$$P\{n\ spikes\ during\ \Delta t\} = \frac{(r\Delta t)^n}{n!}e^{-r\Delta t} \tag{5}$$

This is the probability density function for Poisson distribution.

In our homogeneous Poisson process-based binary word embeddings model, we consider each dimension as an independent homogeneous Poisson process and the normalized value of the dimension $w_{id}^{normalized}$ as the constant firing rate. Following the spike generator within the program, for each Δt in the interval $(0, T)$, we compare $w_{id}^{normalized} \cdot \Delta t$ with a random variable x_{random}. Then, we can get the spiking time matrix in this way:

$$w_{id}^{normalized} \cdot \Delta t = \begin{cases} > x_{random} & fire\ a\ spike \\ \leq x_{random} & nothing \end{cases} \tag{6}$$

Spiking Neural Networks Based Binary Word Embeddings

The LIF-Based Binary Word Embedding Model The leaky integrate-and-fire (LIF) neuron model, a simplified version of H-H model, is one of the simplest spiking neuron models [41]. LIF model is widely used because it is biologically realistic and computationally simple to be analyzed and simulated [31, 42, 43].

In the LIF model, as Eq. 7 shows, v is the membrane potential, τ_m is the membrane time constant, and R is the membrane resistance, and for LIF-based word embeddings model, we replace the input current I with the product of the dth dimension value of the ith word and current boost factor I_{boost}.

$$\tau_m\frac{dv}{dt} = -v(t) + R \cdot I_{boost} \cdot w_{id},\ \ if\ v(t) > v_{th},\ v(t) \leftarrow v_r \tag{7}$$

In our LIF-based binary word embedding model, we regard the value $I_{boost} \cdot w_{id}$ as the intensity of current for neurons, and we get the spiking time matrix based on the record of membrane potential v. In addition, we also try to add white noise to the current to improve its robustness.

The Izhikevich Neuron-Based Binary Word Embedding Model The Izhikevich neuron model is not only capable of producing rich firing patterns exhibited by real biological neurons but also computationally simple [44]. The model makes use of bifurcation methodologies [45] to reduce more biophysically accurate H-H neuron model to a simple one of the following form:

$$\frac{dv}{dt}=0.04v(t)^2+5v(t)+140-u(t)+I,\ \frac{du}{dt}=a(bv(t)-u(t)) \tag{8}$$

If $v(t) \geq v_{th}$, then $v(t) \leftarrow c$ and $u(t) \leftarrow u(t) + d$.

In the Izhikevich neuron model, the meaning of v, v_{th}, and v_r are the same as in the LIF model, while u represents the membrane recovery variable and a, b, c, and d are four important hyper-parameters. The parameter a describes the time scale of u, b describes the sensitivity of u to the subthreshold fluctuations of v, and c is used to describe the after-spike reset value of v and is caused by fast high-threshold K^+ conductances. d is used to describe the after-spike reset of u and is caused by slow high-threshold Na^+ and K^+.

As Izhikevich et al. [44] shows, different choices of these four parameters can simulate different types of neurons in the mammalian brains, such as excitatory cortical cells, inhibitory cortical cells, thalamocortical cells, etc. In this paper, we mainly focus on excitatory and inhibitory cortical neurons. According to the intracellular recordings, cortical cells can be divide into different types, for example, regular spiking (RS), intrinsically bursting (IB), and chattering (CH) for excitatory neurons while fast spiking (FS) and low-threshold spiking (LTS) for inhibitory neurons.

In our Izhikevich neuron model-based binary word embedding models, we make use of the combination of excitatory and inhibitory neurons at the rate of 4:1, which is motivated by the rate in mammalian cortex [44]. As mentioned before, for each word, we set $|D|$ neurons and regard the product of the original word embeddings w_{id} and a factor I_{boost} as the the current for the model. We set each neuron to excitatory/inhibitory sub-models, and for

Table 1 Results of word embeddings on the word similarity tasks

–	GloVe			Word2Vec			Average
	WordSim	SimLex	Rare Words	WordSim	SimLex	Rare Words	
Original	56.45	39.20	33.57	61.29	45.73	44.27	46.75
Overcomplete-B	53.23	41.21	*38.90*	41.94	42.71	33.97	41.99
Rude binarization	58.06	*47.24*	34.35	54.84	44.22	34.73	45.57
Poisson	30.16 ± 6.23	27.79 ± 6.27	15.94 ± 4.02	29.03 ± 9.20	27.64 ± 7.62	16.34 ± 4.05	24.48
LIF	*66.13*	42.21	31.69	*62.90*	*55.28*	*59.16*	*52.90*
LIF-noise	51.61 ± 6.31	35.80 ± 5.50	20.87 ± 4.58	50.81 ± 2.80	49.50 ± 3.78	16.79 ± 0.90	37.56
Izh-CH+FS	62.90	40.70	32.45	51.61	41.71	39.31	44.78
Izh-CH+LTS	62.90	42.71	33.59	45.16	39.20	37.40	43.49
Izh-IB+FS	*66.13*	40.20	31.69	51.61	40.20	32.44	43.71
Izh-IB+LTS	62.90	38.19	33.21	54.84	43.22	30.15	43.75
Izh-RS+FS	*66.13*	44.22	30.93	48.39	51.76	32.82	45.71
Izh-RS+LTS	62.90	41.71	30.55	48.39	44.72	38.93	44.53

different dimensions of each word, we get the spike times according to its sub-models.

Experiment Validations

Validation Tasks and Datasets

We evaluate our binary embeddings on word similarity and text classification tasks. The word similarity task has been widely used to measure in which degree the word embeddings can capture the similarity between two words, while the text classification task is a traditional NLP application. In our experiment, all the binary word embedding models are based on two kinds of well-accepted original word embeddings, namely, Word2Vec [1] and GloVe [2].

For word similarity task, we find similar words via Hamming distance, which will be faster than traditional cosine distance for dense embeddings and we evaluate embeddings on three public datasets: (1) **WordSim-353**, it is the most widely used dataset for word similarity test, consisting of 353 pairs of words [46]; (2) **SimLex-999**, it consists of 999 pairs of words and provides a way of measuring how well the word embeddings capture similarity, rather than relatedness or association [47]; (3) **Rare Words**, it consists of 2,034 word pairs proposed by Luong et al. [48], focusing on rare words to complement exiting ones. All these pairs of words are along with human-assigned similarity scores and we check Spearman's rank correlation coefficient between word embeddings and the human labeled ranks.

For the text classification task, we do OR operation on binary embeddings to generate the representation for text and use the *k*-nearest neighbors (kNN) classifier to measure accuracy. We validate our algorithms on two public text datasets: (1) **Search Snippets**, it is a short text dataset collected by Phan et al. [50], which is selected from the results of Web search transaction using predefined phrases of 8 different domains; (2) **Sentiment Analysis**, it is proposed by Socher et al. [49] and is a treebank of sentences annotated with sentiment labels from movie reviews. The sentences in the treebank were split into a train (8544), dev (1101), and test splits (2210). We merge the train and dev part for the kNN classifier and ignore neutral sentences, analyzing performance on only positive and negative class.

Experiment Details and Results

In our experiment, we use the pre-trained GloVe[1] and Word2Vec[2] embeddings, both of which are 300 dimensions. We set three comparative experiments of original embedings, binary embeddings, "Overcomplete-B" derives from Faruqui's work [8], and "Rude Binarization" convert original embeddings into binary ones via simple sign function.

For all the biological neuron coding-inspired models, we set the interval $T = 10$ ms and subinterval $\Delta t = 0.1$ ms. We find the best hyper-parameter through grid-search on word similarity tasks and apply these for both experiment tasks. For Poisson, LIF, and LIF with noise-based model, the tol is 5, while for other models, tol is 10. For LIF and LIF with noise model, $\tau_m = 10$ and $v_{th} = 15$, while for Izhikevich model, $v_{th} = 30$, and other parameters follow [44] for different sub-models. The I_{boost} factors are 100 and

[1] https://nlp.stanford.edu/projects/glove/

[2] https://code.google.com/archive/p/word2vec/

Table 2 Summarized results of two tasks

Methods	Word similarity	Text classification	Average
Overcomplete-B	41.99	57.19	49.59
Rude binarization	45.57	36.12	40.88
Poisson	24.48	46.72	35.60
LIF	*52.90*	46.79	49.84
LIF-noise	37.56	53.51	45.53
Izh-CH+FS	44.78	56.86	50.82
Izh-CH+LTS	43.49	57.87	50.68
Izh-IB+FS	43.71	62.69	53.20
Izh-IB+LTS	43.75	63.83	53.79
Izh-RS+FS	45.71	*70.23*	*57.97*
Izh-RS+LTS	44.53	69.68	57.11

200 for GloVe and Word2Vec respectively. In Addition, for Poisson coding and LIF with noise model, we do 10 times for each, with different random seeds, and Table 1 shows the average and their standard deviation results.

Result Analysis Q5

Through analysis from the data shown in Tables 1 and 2 and Fig. 3, we can infer that: (1) We make an exploration on how to generate binary embeddings via biological neuron coding-inspired models (Figs. 2 and 3. The results Q6 show that the SNN-based models show good performance while the Poisson coding-based model reflected rate coding's weakness when transforming dense information into binary bits. Which means, it cannot carry enough information to represent stimuli or patterns. (2) For word

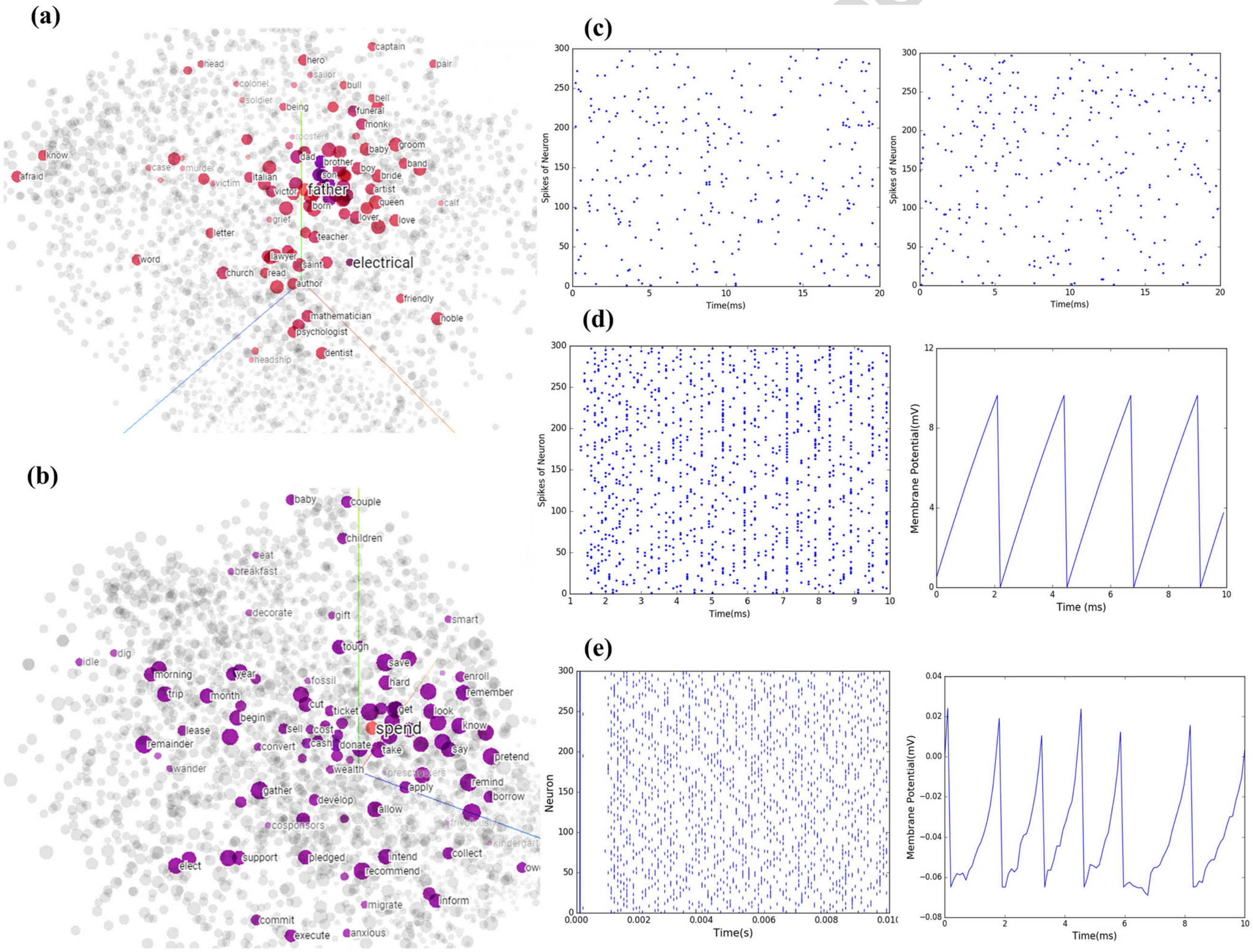

Fig. 2 Results visualization for word similarity tasks. **a**, **b** t-SNE visualization [51] of the words "father" and "spend" in Izh_RS+FS model to show their similar words; **c** spiking matrix of the word "people" in Poisson-based model with different random seeds; **d**, **e** spiking matrix and the 3rd neuron's membrane potential of LIF model and Izh_RS+FS model respectively

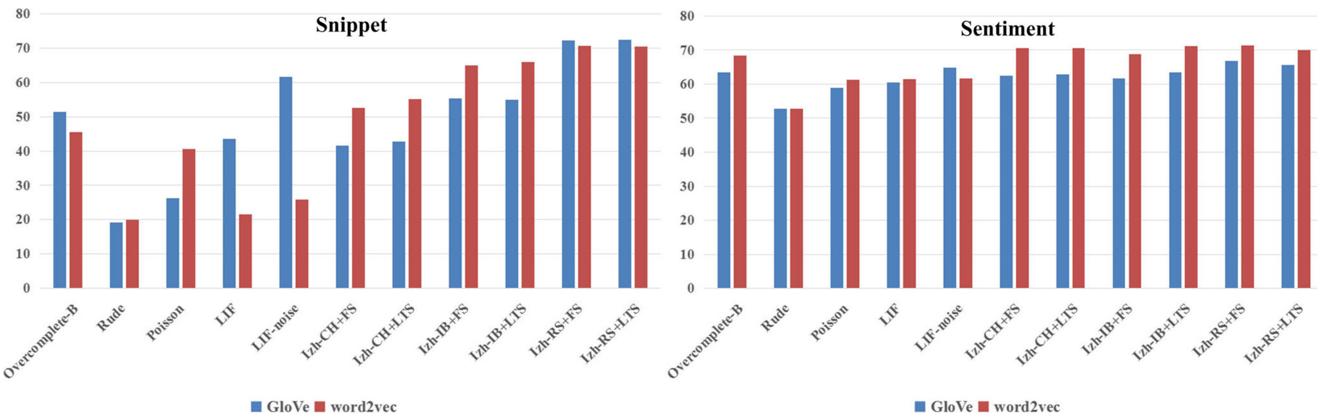

Fig. 3 Results of the text classification task

similarity task, binary word embeddings, especially rude binarization, LIF-based, and Izhkevich-based models which are transformed through dense word embeddings, can get similar results to original ones. (3) The LIF-based binary embeddings model performs well on word similarity tasks while somehow bad on text classification task. This may due to over simplified mechanism of LIF model, making it robust to represent words while lost many semantic information; LIF model with noise can improve the performance of text classification task, while it is unstable and can pull down the word similarity results. (4) The Izhkevich neuron-based binary embedding model gets excellent results on both tasks, especially the combination of RS and FS neuron sub-models is the best one. The model combines the excitatory and inhibitory neurons to mimic the neurons in the biological brain, making a difference when converting the original dense embeddings into binary ones. (5) From the perspective of space occupation, for database of 3 million words (such as the public pre-training Word2Vec vectors) with 300 dimensions takes 3.6 GB in floating point while 1.125 GB as 3000-bit codes ($tol = 10$) for the Izh_RS+FS model, which reduced approximately 68.75% space occupation. For neuron coding-based binary embeddings models, the compression ratio is mainly due to the run time and the tolerance factor tol.

Conclusion

In this paper, we propose three kinds of biological neuron coding-inspired models to generate binary word embeddings, which show better performance and interpretability compared to existing works on word similarity evaluation and text classification task. To the best of our knowledge, this is the first attempt to convert the dense embeddings into binary ones via spike timing, and we have proved its feasibility on some natural language processing applications.

Future Work

Due to the limitation on the performance of supervised SNNs, in this paper, we do post-processing operations on given word embeddings. However, we are looking forward to build SNN-based language model to get brain-inspired word embeddings from the raw corpus. We are trying to adjust the cost function of supervised SNNs and add several biological mechanisms such as STDP to the model to get them. Furthermore, in contrast to excitatory neocortical neurons, which have stereotypical morphological and electrophysiological classes, inhibitory neocortical interneurons have wildly diverse classes with various firing patterns that cannot be classified as FS or LTS [45]. In this paper, we focus on FS and LTS inhibitory neurons for their parameters in Izhikevich's neuron model are easy to get. In the future, we will pay more attention to more detailed types of inhibitory neuron models.

Funding information This study is supported by the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No. XDB32070100), the Beijing Municipality of Science and Technology (Grant No. Z181100001518006), the CETC Joint Fund (Grant No. 6141B08010103), and the Major Research Program of Shandong Province 2018CXGC1503.

Compliance with Ethical Standards

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems; 2013. p. 3111–9.
2. Pennington J, Socher R, Manning C. Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2014.
3. Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model. J Mach Learn Res. 2003;3:1137–55.
4. Socher R, Bauer J, Manning CD, et al. Parsing with compositional vector grammars. In: Meeting of the association for computational linguistics; 2013. p. 455–65.
5. Kim Y. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882. 2014.
6. Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Advances in neural information processing systems; 2014. p. 3104–12.
7. Murphy B, Talukdar P, Mitchell T. Learning effective and interpretable semantic models using non-negative sparse embedding. In: COLING; 2012. p. 1933–50.
8. Faruqui M, Tsvetkov Y, Yogatama D, et al. Sparse overcomplete word vector representations. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 149101500; 2015.
9. Sun F, Guo J, Lan Y, et al. Sparse word embeddings using l1 regularized online learning. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence; 2016. p. 2915–21.
10. Ling S, Song Y, Dan R. Word embeddings with limited memory. In: Meeting of the association for computational linguistics; 2016. p. 387–92.
11. Rolls ET. Cortical coding. Language Cogn Neurosci. 2017;32(3): 316–29.
12. Hopfield JJ. Pattern recognition computation using action potential timing for stimulus representation. Nature. 1995;376(6535): 33–6.
13. Lestienne R. Determination of the precision of spike timing in the visual cortex of anaesthetised cats. Biol Cybern. 1995;74(1):55–61.
14. Rieke F, Warland D, Rob DRVS, et al. Spikes: exploring the neural code. 1999: MIT Press, Cambridge.
15. Brown EN, Kass RE, Mitra PP. Multiple neural spike train data analysis: state-of-the-art and future challenges. Nat Neurosci. 2004;7(5):456–61.
16. Quiroga RQ, Panzeri S. Principles of neural coding. Boca Raton: CRC Press; 2013.
17. Adrian ED. The impulses produced by sensory nerve endings. J Physiol. 1926;61(1):49–72.
18. Thorpe S, Fize D, Marlot C. Speed of processing in the human visual system. Nature. 1996;381(6582):520–2.
19. Van RR, Thorpe SJ. Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. Neural Comput. 2001;13(6):1255–83.
20. Tiesinga P, Fellous JM, Sejnowski TJ. Regulation of spike timing in visual cortical circuits. Nat Rev Neurosci. 2008;9(2):97.
21. Ponulak F, Kasinski A. Introduction to spiking neural networks: information processing, learning and applications. Acta Neurobiol Exp. 2011;71(4):409.
22. Johansson RS, Birznieks I. First spikes in ensembles of human tactile afferents code complex spatial fingertip events. Nat Neurosci. 2004;7(2):170.
23. Saal HP, Vijayakumar S, Johansson RS. Information about complex fingertip parameters in individual human tactile afferent neurons. J Neurosci: The Official Journal of the Society for Neuroscience. 2009;29(25):8022.
24. Thorpe SJ. Spike arrival times: a highly efficient coding scheme for neural networks. In: Parallel processing in neural systems & computers; 1990.
25. Faisal AA, Selen LP, Wolpert DM. Noise in the nervous system. Nat Rev Neurosci. 2008;9(4):292–303.
26. Laurent G. Dynamical representation of odors by oscillating and evolving neural assemblies. Trends Neurosci. 1996;19(11):489–96.
27. Maas W. Networks of spiking neurons: the third generation of neural network models. Trans Soc Comput Simul Int. 1997;14(4): 1659–71.
28. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol. 1952;117(4):500.
29. Gerstner W, Kistler WM. Spiking neuron models: single neurons, populations, plasticity. Cambridge: Cambridge University Press; 2002.
30. Gupta A, Long LN. Character recognition using spiking neural networks. In: International Joint Conference on Neural Networks; 2007. p. 53–8.
31. Tavanaei A, Maida AS. Bio-inspired spiking convolutional neural network using layer-wise sparse coding and STDP learning. arXiv preprint arXiv:1611:03000. 2016.
32. Cao Y, Chen Y, Khosla D. Spiking deep convolutional neural networks for energy-efficient object recognition. Int J Comput Vis. 2015;113(1):54–66.
33. Azhar H, Iftekharuddin K, Kozma R. A chaos synchronization-based dynamic vision model for image segmentation. In: IEEE International Joint Conference on Neural Networks. IJCNN '05. Proceedings. IEEE; 2005.
34. Loiselle S, Rouat J, Pressnitzer D, et al. Exploration of rank order coding with spiking neural networks for speech recognitionI. In: IEEE International Joint Conference on Neural Networks, 2005 IJCNN '05 Proceedings IEEE; 2005. p. 2076–80.
35. Floreano D, Epars Y, Zufferey JC, et al. Evolution of spiking neural circuits in autonomous mobile robots. Int J Intell Syst. 2006;21(9):1005–24.
36. Crawford E, Gingerich M, Eliasmith C. Biologically plausible, human-scale knowledge representation. Cognit Sci. 2016;40(4):782–821.
37. Stewart TC, Xuan C, Eliasmith C. Symbolic reasoning in spiking neurons: a model of the cortex/basal ganglia/thalamus loop. In: Meeting of the cognitive science society; 2010.
38. O'Keefe LP, Bair W, Movshon JA. Response variability of MT neurons in macaque monkey. Soc Neurosci Abstr. 1997;23: 1125.
39. Bair W, Koch C, Newsome W, et al. Power spectrum analysis of bursting cells in area MT in the behaving monkey. J Neurosci. 1994;14(5):2870–92.
40. Softky WR, Koch C. Cortical cells should fire regularly, but do not. 1992. Q7
41. Burkitt AN. A review of the integrate-and-fire neuron model: I. Biol Cybern. 2006;95(1):1–19.
42. Hu J, Tang H, Tan KC, et al. A spike-timing-based integrated model for pattern recognition. Neural Comput. 2012;25(2):450–72.
43. Hunsberger E, Eliasmith C. Spiking deep networks with LIF neurons. Computer Science. 2015.
44. Izhikevich EM. Simple model of spiking neurons. IEEE Trans Neural Netw. 2003;14(6):1569–72.
45. Izhikevich EM. Dynamical systems in neuroscience: the geometry of excitability and bursting. Cambridge: MIT Press; 2007.
46. Agirre E, Alfonseca E, Hall K, et al. A study on similarity and relatedness using distributional and wordnet-based approaches. In:

Proceedings of human language technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics; 2009. p. 19–27.

47. Hill F, Reichart R, Korhonen A. Simlex-999: evaluating semantic models with (genuine) similarity estimation. Computational Linguistics. 2016.

48. Luong T, Socher R, Manning CD. Better word representations with recursive neural networks for morphology. In: CoNLL; 2013. p. 104–13.

49. Socher R, Perelygin A, Wu JY, et al. Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP); 2013. p. 1642.

50. Phan XH, Nguyen LM, Horiguchi S. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In: Proceedings of the 17th International Conference on World Wide Web. ACM; 2008. p. 91–100.

51. Smilkov D, Thorat N, Nicholson C, et al. Embedding projector: interactive visualization and interpretation of embeddings. arXiv preprint arXiv:1611.05469. 2016.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES:

Q1. Unstructured abstract is not permitted. Thus, headings were removed. Please check if appropriate.

Q2. Yi Zeng has been set as the corresponding author. Please check and advise if correct.

Q3. Figures 1, 2 and 3 contains text below the minimum required font size of 6pts inside the artwork, and there is no sufficient space available for the text to be enlarged. Please provide replacement figure file.

Q4. Figures 1, 2, and 3 contain text below the minimum required font size of 6pts inside the artwork, and there is no sufficient space available for the text to be enlarged. Please provide a replacement file.

Q5. In Tables 1 and 2, please provide the significance of entries in italics in the form of a table footnote; otherwise, please consider setting them to upright.

Q6. Missing citation for Figure 2 was inserted here. Please check if appropriate. Otherwise, please provide citation for Figure 2. Note that the order of main citations of figures in the text must be sequential.

Q7. Please provide complete bibliographic details in reference [40].