BIDIRECTIONAL QUATERNION LONG SHORT-TERM MEMORY RECURRENT NEURAL NETWORKS FOR SPEECH RECOGNITION

Titouan Parcollet[1,3]*, Mohamed Morchid*[1]*, Georges Linarès*[1]*, and Renato De Mori*[1,2]

[1]Université d'Avignon, LIA, France
[2]McGill University, Montréal, Canada
[3]Orkis, Aix en provence, France

ABSTRACT

Recurrent neural networks (RNN) are at the core of modern automatic speech recognition (ASR) systems. In particular, long short-term memory (LSTM) recurrent neural networks have achieved state-of-the-art results in many speech recognition tasks, due to their efficient representation of long and short term dependencies in sequences of inter-dependent features. Nonetheless, internal dependencies within the element composing multidimensional features are weakly considered by traditional real-valued representations. We propose a novel quaternion long short-term memory (QLSTM) recurrent neural network that takes into account both the external relations between the features composing a sequence, and these internal latent structural dependencies with the quaternion algebra. QLSTMs are compared to LSTMs during a memory copy-task and a realistic application of speech recognition on the Wall Street Journal (WSJ) dataset. QLSTM reaches better performances during the two experiments with up to 2.8 times less learning parameters, leading to a more expressive representation of the information.

Index Terms— Quaternion long-short term memory, recurrent neural networks, speech recognition

1. INTRODUCTION

During the last decade, deep neural networks (DNN) have encountered a wide success in numerous domain applications. In particular, automatic speech recognition systems (ASR) performances have been remarkably improved with the emergence of DNNs. Among them, recurrent neural networks [1] (RNN) have been shown to effectively encode input sequences, increasing the accuracy of neural network based ASR systems [2]. Nonetheless, vanilla RNNs suffer from vanishing/exploding issues [3], or the lack of a memory mechanism to remember patterns in very-long or short sequences. These problems have been alleviated by the introduction of long short-term memory (LSTM) RNN [4] with gates mechanism that allows the model to update or forget information in memory cells, and to select the content cell state to expose in a network hidden state. LSTMs have reached state-of-the art performances in many benchmarks [4, 5], and are widely employed in recent ASR models, with the almost unchanged acoustic input features used in previous systems.

Traditional ASR systems rely on multidimensional acoustic features such as the Mel filter bank energies alongside with the first, and second order time derivatives to characterize time-frames that compose the signal sequence. Considering that these components describe three different views of the same element, neural networks have to learn both the internal relations that exist within these views, and external or global dependencies that exist between the time-frames. Such concerns are partially addressed by increasing the learning capacity of neural network architectures. Nonetheless, even with a huge set of free parameters, it is not certain that both local and global dependencies are properly represented. To address this problem, new quaternion-valued neural networks, based on a high-dimensional algebra, are proposed in this paper.

Quaternions are hyper-complex numbers that contain a real and three separate imaginary components, fitting perfectly to three and four dimensional feature vectors, such as for image processing and robot kinematics [6]. The idea of bundling groups of numbers into separate entities is also exploited by the recent capsule network [7]. With quaternion numbers, LSTMs are conceived to encode latent inter-dependencies between groups of input features during the learning process with less parameters than real-valued LSTMs, by taking advantage of the use of the quaternion *Hamilton product* as the counterpart of the dot product. Early applications of quaternion-valued backpropagation algorithms [8, 9] have efficiently shown that quaternion neural networks can approximate quaternion-valued functions. More recently, neural networks of hyper-complex numbers have received an increasing attention, and some efforts have shown promising results in different applications. In particular, a deep quaternion network [10, 11], a deep quaternion convolutional network [12, 13], or a quaternion recurrent neural network [14] have been successfully employed for challenging tasks such as images, speech and language processing. For speech recognition, in [13], quaternions with only three internal features have been used to encode input

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

speech. An additional internal feature is proposed in this paper to obtain a richer representation with the same number of model parameters.

Based on all the above considerations, the contributions of this paper can be summarized as follows: 1) The introduction of a novel model, called bidirectional quaternion long short-term memory neural network (QLSTM)[1], that avoids known RNN problems also present in quaternion RNNs, and shows that QLSTMs achieve top of the line results on speech recognition; 2) The introduction of a novel input quaternion that integrates four views of speech time frames. The model is first evaluated on a synthetic memory copy-task to ensure that the introduction of quaternion into the LSTM model does not alter the basic properties of RNNs. Then, QLSTMs are compared to real-valued LSTMs on a realistic speech recognition task with the Wall Street Journal (WSJ) dataset. The reported results show that the QLSTM outperforms the LSTM in both tasks with a higher long-memory capability on the memory task, a better generalization performance with better word error rates (WER), and a maximum reduction of the number of neural paramaters of 2.8 times compared to real-valued LSTM.

2. QUATERNION ALGEBRA

The quaternion algebra $\mathbb{H}$ defines operations between quaternion numbers. A quaternion Q is an extension of a complex number defined in a four dimensional space as:

$$Q = r1 + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}, \tag{1}$$

where r, x, y, and z are real numbers, and 1, **i**, **j**, and **k** are the quaternion unit basis. In a quaternion, r is the real part, while $x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ with $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$ is the imaginary part, or the vector part. Such a definition can be used to describe spatial rotations. The *Hamilton product* $\otimes$ between two quaternions Q_1 and Q_2 is computed as follows:

$$\begin{aligned} Q_1 \otimes Q_2 =&(r_1r_2 - x_1x_2 - y_1y_2 - z_1z_2)+ \\ &(r_1x_2 + x_1r_2 + y_1z_2 - z_1y_2)\boldsymbol{i}+ \\ &(r_1y_2 - x_1z_2 + y_1r_2 + z_1x_2)\boldsymbol{j}+ \\ &(r_1z_2 + x_1y_2 - y_1x_2 + z_1r_2)\boldsymbol{k}. \end{aligned} \tag{2}$$

The *Hamilton product* is used in QLSTMs to perform transformations of vectors representing quaternions, as well as scaling and interpolation between two rotations following a geodesic over a sphere in the $\mathbb{R}^3$ space as shown in [15].

3. QUATERNION LONG SHORT-TERM MEMORY NEURAL NETWORKS

Based on the quaternion algebra and with the previously described motivations, we introduce the quaternion long short-term memory (QLSTM) recurrent neural network. In a quaternion dense layer, all parameters are quaternions, including inputs, outputs, weights and biases. The quaternion algebra is ensured by manipulating matrices of real numbers [13] to reconstruct the *Hamilton product* from quaternion algebra. Consequently, for each input vector of size N, output vector of size M, dimensions are split in four parts: the first one equals to r, the second to $x\mathbf{i}$, the third one is $y\mathbf{j}$, and the last one equals to $z\mathbf{k}$. The inference process of a fully-connected layer is defined in the real-valued space by the dot product between an input vector and a real-valued $M \times N$ weight matrix. In a QLSTM, this operation is replaced with the *Hamilton product* '$\otimes$' (Eq. 2) with quaternion-valued matrices (*i.e.* each entry in the weight matrix is a quaternion).

Gates are core components of the memory of LSTMs. Based on [16], we propose to extend this mechanism to quaternion numbers. Therefore, the gate action is characterized by an independent modification of each component of the quaternion-valued signal following a component-wise product (*i.e.* in a *split* fashion [17]) with the quaternion-valued gate potential. Let f_t,i_t, o_t, c_t, and h_t be the forget, input, output gates, cell states and the hidden state of a LSTM cell at time-step t. QLSTM equations can be derived as:

$$\begin{aligned} f_t =&\sigma(W_f \otimes x_t + R_f \otimes h_{t-1} + b_f), &(3)\\ i_t =&\sigma(W_i \otimes x_t + R_i \otimes h_{t-1} + b_i), &(4)\\ c_t =&f_t \times c_{t-1} + i_t \times \alpha(W_c \otimes x_t + R_c \otimes h_{t-1} + b_c), &(5)\\ o_t =&\sigma(W_o \otimes x_t + R_o \otimes h_{t-1} + b_o), &(6)\\ h_t =&o_t \times \alpha(c_t), &(7) \end{aligned}$$

with σ and α the sigmoid and tanh *quaternion split activations* [17, 10, 18, 9]. The quaternion weight and bias matrices are initialized following the proposal of [14]. Quaternion bidirectional connections are equivalent to real-valued ones [19]. Consequently, past and future contexts are added together component-wise at each time-step. The full back-propagtion of quaternion-valued recurrent neural network can be found in [14].

4. EXPERIMENTS

This section provides the results for QLSTM and LSTM on the synthetic memory copy-task (Section 4.1), and a description of the quaternion acoustic features (Section 4.2) that are used as inputs during the realistic speech recognition experiment with the Wall Street Journal (WSJ) corpus (Section 4.3).

4.1. Synthetic memory copy-task as a sanity check

The copy task originally introduced by [20] is a synthetic test that highlights how RNN based models manage the long-term memory. This characteristic makes the copy task a powerful benchmark to demonstrate that a recurrent model can learn

[1] Code is available at https://github.com/Orkis-Research/Pytorch-Quaternion-Neural-Networks

long-term dependencies. It consists of an input sequence of a length L, composed of S different symbols followed by a sequence of time-lags or *blanks* of size T, and ended by a delimiter that announces the beginning of the copy operation (after which the initial input sequence should be progressively reconstructed at the output). In this paper, the copy-task is used as a sanity check to ensure that the introduction of quaternions on LSTM models does not harm the basic memorization abilities of the LSTM. The QLSTM is composed of 8K parameters with one hidden layer of size 20, while the LSTM is made of 8.2K parameters with an hidden dimension of 40 neurons. It is worth underlying that due to the nature of the task, the output layer of the QLSTM is real-valued. Indeed, 9 symbols are one-hot encoded ($S = 0, ..., 7$ for the sequence and 8 for the *blank*) and can not be split in four components. Different values of $T = 10, 50, 100$ are investigated alongside with a fixed sequence size of $L = 10$. Models are trained with the Adam optimizer, with an initial learning rate $\lambda = 5{\cdot}10^{-3}$, and without employing any regularization methods. The training is performed on $2,000$ epochs with the cross-entropy used as the loss function. At each epoch, models are fed with a batch of 10 randomly generated sequences.

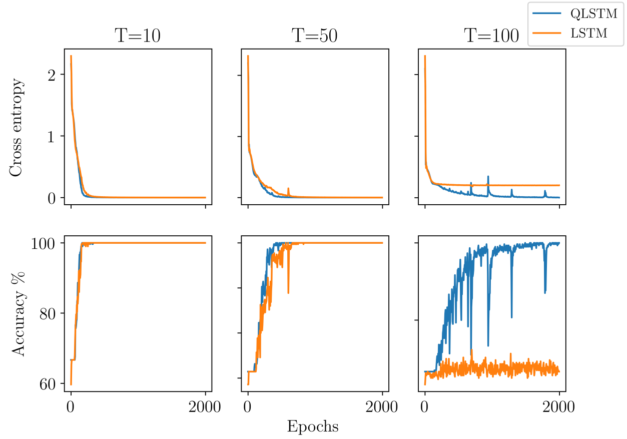

Fig. 1. Evolution of the cross entropy loss, and of the accuracy of both QLSTM (Blue curves) and LSTM (Orange curves) during the synthetic memory copy-task for time lags or *blanks* T of 10, 50 and 100.

The results reported in Fig.1 highlight a slightly faster convergence of the QLSTM over the LSTM for all sizes (T). It is also worth noticing that real-valued LSTM failed the copy-task with $T = 100$ while QLSTM succeeded. It is easily explained by the impact of quaternion numbers during the learning process of inter-denpendencies of input features. Indeed, the QLSTM is a smaller (less parameters), but more efficient (dealing with higher dimensions) model than real-valued LSTM, resulting in a higher generalization capability: 20 quaternion neurons are equivalent to $20 \times 4 = 80$ real-valued ones. Overall, the introduction of quaternions in LSTMs do not alter their basics properties, but it provides a higher long-term dependencies learning capability. We hypothesis that such efficiency improvements alongside with a dedicated input representation will help QLSTMs to outperform LSTMs in more realistic tasks, such as speech recognition.

4.2. Quaternion acoustic features

Unlike in [13], this paper proposes to use four internal features in an input quaternion. The raw audio is first split every 10ms with a window of 25ms. Then 40-dimensional log Mel-filter-bank coefficients with first, second, and third order derivatives are extracted using the *pytorch-kaldi*[2] toolkit and the Kaldi s5 recipes [2]. An acoustic quaternion $Q(f,t)$ associated with a frequency band f and a time-frame t is formed as follows:

$$Q(f,t) = e(f,t) + \frac{\partial e(f,t)}{\partial t}\mathbf{i} + \frac{\partial^2 e(f,t)}{\partial^2 t}\mathbf{j} + \frac{\partial^3 e(f,t)}{\partial^3 t}\mathbf{k}. \tag{8}$$

$Q(f,t)$ represents multiple views of a frequency band f at time frame t, consisting of the energy $e(f,t)$ in the filter band at frequency f, its first time derivative describing a slope view, its second time derivative describing a concavity view, and the third derivative describing the rate of change of the second derivative. Quaternions are used to construct latent representations of the external relations between the views characterizing the contents of frequency bands at given time intervals. Thus, the quaternion input vector length is $160/4 = 40$. Decoding is based on Kaldi [2] and weighted finite state transducers (WFST) that integrate acoustic, lexicon and language model probabilities into a single HMM-based search graph.

4.3. Speech recognition with the Wall Street Journal

QLSTMs and LSTMs are trained on both the 14 hour subset 'train-si84', and the full 81 hour dataset 'train-si284' of the Wall Street Journal (WSJ) corpus. The 'test-dev93' development set is employed for validation, while 'test-eval92' composes the testing set. It is important to notice that evaluated LSTMs and QLSTMs are bidirectionals. Architecture models vary in both number of layers and neurons. Indeed the number of recurrent layers L varies from three to four, while the number of neurons N is included in a gap from 256 to $1,024$. Then, one dense layer is stacked alongside with an output dense layer. It is also worth noticing that the number of quaternion units of a QLSTM layer is $N/4$. Indeed, QLSTM neurons are four dimensional (*i.e.* a QLSTM layer that deals with a dimension size of $1,024$ has $1,024/4 = 256$ effective quaternion neurons). Models are optimized with Adam,

[2] pytorch-kaldi is available at https://github.com/mravanelli/pytorch-kaldi

Table 1. Word error rates (WER %) obtained with both training set (WSJ14h and WSJ81h) of the Wall Street Journal corpus. 'test-dev93' and 'test-eval92' are used as validation and testing set respectively. L expresses the number of recurrent layers. Models are bidirectional. Results are from an average of three runs.

Models	WSJ14 Dev.	WSJ14 Test	WSJ81 Dev.	WSJ81 Test	Params
$\mathbb{R}$-LSTM-3L-256	12.7	8.6	9.5	6.5	4.0M
$\mathbb{H}$-QLSTM-3L-256	12.8	8.5	9.4	6.5	2.3M
$\mathbb{R}$-LSTM-4L-256	12.1	8.3	9.3	6.4	4.8M
$\mathbb{H}$-QLSTM-4L-256	11.9	8.0	9.1	6.2	2.5M
$\mathbb{R}$-LSTM-3L-512	**11.1**	**7.1**	8.2	5.2	12.2M
$\mathbb{H}$-QLSTM-3L-512	**10.9**	**6.9**	8.1	5.1	5.6M
$\mathbb{R}$-LSTM-4L-512	11.3	7.0	8.1	5.0	15.5M
$\mathbb{H}$-QLSTM-4L-512	11.1	6.8	8.0	4.9	6.5M
$\mathbb{R}$-LSTM-3L-1024	11.4	7.3	7.6	4.8	41.2M
$\mathbb{H}$-QLSTM-3L-1024	11.0	6.9	7.4	4.6	15.5M
$\mathbb{R}$-LSTM-4L-1024	11.2	7.2	**7.4**	**4.5**	53.7M
$\mathbb{H}$-QLSTM-4L-1024	10.9	6.9	**7.2**	**4.3**	18.7M

with vanilla hyper-parameters and an initial learning rate of $5 \cdot 10^{-4}$. The learning rate is progressively annealed using an halving factor of 0.5 that is applied when no performance improvement on the validation set is observed. The models are trained during 15 epochs. All the models converged to a minimum loss, due to the annealed learning rate. Results are from a three folds average.

At first, it is important to notice that reported results on Table 1 compare favorably with equivalent architectures [5] (WER of 11.7% on 'test-dev93'), and are competitive with state-of-the-art and much more complex models based on better engineered features [21] (WER of 3.8% with the 81 hours of training data, and on 'test-eval92'). Table 1 shows that the proposed QLSTM always outperform real-valued LSTM on the test dataset with less neural parameters. Based on the smallest 14 hours subset, a best WER of 6.9% is reported in real conditions (*w.r.t* to the best validation set results) with a three layered QLSTM of size 512, compared to 7.1% for an LSTM with the same size. It is worth mentioning that a best WER of 6.8% is obtained with a four layered QLSTM of size 512, but without consideration for the validation results. Such performances are obtained with a reduction of the number of parameters of 2.2 times, with 5.6M parameters for the QLSTM compared to 12.2M for the real-valued equivalent. This is easily explained by considering the content of the quaternion algebra. Indeed, for a fully-connected layer with $2,048$ input values and $2,048$ hidden units, a real-valued RNN has $2,048^2 \approx 4.2$M parameters, while, to maintain equal input and output dimensions, the quaternion equivalent has 512 quaternions inputs and 512 quaternion hidden units. Therefore, the number of parameters for the quaternion-valued model is $512^2 \times 4 \approx 1$M. Such a complexity reduction turns out to produce better results and have other advantages such as a smaller memory footprint while saving models on budget memory systems. This reduction allows the QLSTM to make the memory more "compact" and therefore, the relations between quaternion components are more robust to unseen documents from both validation and testing data-sets. This characteristic makes our QLSTM model particularly suitable for speech recognition conducted on low computational power devices like smartphones. Both QLSTMs and LSTMs produce better results with the 81 hours of training data. As for the smaller subset, QLSTMs always outperform LSTMs during both validation and testing phases. Indeed, a best WER of 4.3% is reported for a four layered QLSTM of dimension $1,024$, while the best LSTM performed at 4.5% with 2.9 times more parameters, and an equivalently sized architecture.

5. CONCLUSION

This paper proposes to process sequences of traditional and multidimensional acoustic features with a novel quaternion long short-term memory neural network (QLSTM). The paper introduce first a novel quaternion-valued representation of the speech signal to better handle signal sequences dependencies, and a LSTM composed with quaternions to represent in the hidden latent space inter-dependencies between quaternion features. The proposed model has been evaluated on a synthetic memory copy-task and a more realistic speech recognition task with the large Wall Street Journal (WSJ) dataset. The reported results support the initial intuitions by showing that QLSTM are more effective at learning both longer dependencies and a compact representation of multidimensional acoustic speech features by outperforming standard real-valued LSTMs on both experiments, with up to 2.8 times less neural parameters. Therefore, and as for other quaternion-valued architectures, the intuition that the quaternion algebra of the QLSTM offers a better and more compact representation for multidimensional features, alongside with a better learning capability of feature internal dependencies through the *Hamilton product*, have been validated.

6. REFERENCES

[1] Larry R. Medsker and Lakhmi J. Jain, "Recurrent neural networks," *Design and Applications*, vol. 5, 2001.

[2] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. Dec. 2011, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

[3] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio, "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning*, 2013, pp. 1310–1318.

[4] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[5] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed, "Hybrid speech recognition with deep bidirectional lstm," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 273–278.

[6] Stephen John Sangwine, "Fourier transforms of colour images using quaternion or hypercomplex, numbers," *Electronics letters*, vol. 32, no. 21, pp. 1979–1980, 1996.

[7] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton, "Dynamic routing between capsules," *arXiv preprint arXiv:1710.09829v2*, 2017.

[8] Paolo Arena, Luigi Fortuna, Luigi Occhipinti, and Maria Gabriella Xibilia, "Neural networks for quaternion-valued function approximation," in *Circuits and Systems, ISCAS'94., IEEE International Symposium on*. IEEE, 1994, vol. 6, pp. 307–310.

[9] Paolo Arena, Luigi Fortuna, Giovanni Muscato, and Maria Gabriella Xibilia, "Multilayer perceptrons to approximate quaternion valued functions," *Neural Networks*, vol. 10, no. 2, pp. 335–342, 1997.

[10] Titouan Parcollet, Mohamed Morchid, Pierre-Michel Bousquet, Richard Dufour, Georges Linarès, and Renato De Mori, "Quaternion neural networks for spoken language understanding," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 362–368.

[11] Titouan Parcollet, Morchid Mohamed, and Georges Linarès, "Quaternion denoising encoder-decoder for theme identification of telephone conversations," *Proc. Interspeech 2017*, pp. 3325–3328, 2017.

[12] Chase J Gaudet and Anthony S Maida, "Deep quaternion networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–8.

[13] Titouan Parcollet, Ying Zhang, Mohamed Morchid, Chiheb Trabelsi, Georges Linarès, Renato de Mori, and Yoshua Bengio, "Quaternion convolutional neural networks for end-to-end automatic speech recognition," in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018.*, 2018, pp. 22–26.

[14] Titouan Parcollet, Mirco Ravanelli, Mohamed Morchid, Georges Linarès, Chiheb Trabelsi, Renato De Mori, and Yoshua Bengio, "Quaternion recurrent neural networks", *arXiv preprint 1806.04418v2*, 2018.

[15] Toshifumi Minemoto, Teijiro Isokawa, Haruhiko Nishimura, and Nobuyuki Matsui, "Feed forward neural network with random quaternionic neurons," *Signal Processing*, vol. 136, pp. 59–68, 2017.

[16] Ivo Danihelka, Greg Wayne, Benigno Uria, Nal Kalchbrenner, and Alex Graves, "Associative long short-term memory," *arXiv preprint 1602.03032*, 2016.

[17] D Xu, L Zhang, and H Zhang, "Learning alogrithms in quaternion neural networks using ghr calculus," *Neural Network World*, vol. 27, no. 3, pp. 271, 2017.

[18] Titouan Parcollet, Mohamed Morchid, and Georges Linares, "Deep quaternion neural networks for spoken language understanding," in *Automatic Speech Recognition and Understanding Workshop (ASRU), 2017 IEEE*. IEEE, 2017, pp. 504–511.

[19] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning*, 2014, pp. 1764–1772.

[20] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] William Chan and Ian Lane, "Deep recurrent neural networks for acoustic modelling," *arXiv preprint arXiv:1504.01482*, 2015.