Recurrent Neural Network analysis of Mass Spectrometry Imaging for cancer classification

A.Panteli
Department of Electrical Engineering, Endhoven University of Technology - a.panteli@student.tue.nl

Abstract—**This paper concerns the ability of recurrent neural networks (RNN) to analyze mass spectrometry imaging (MSI) data in order to correctly classify cancer tissue. It explores whether Long Short-Term Memory (LSTM) units can effectively be used to correctly identify the pattern of cancerous tissue in multidimensional structures. Different LSTM network architectures, configurations and neural network models are investigated and compared. A recommendation is provided, at the end, for classifying cancer tissue with MSI data using recurrent neural networks.**

The code and data used of this implementation can be found at: https://github.com/Baggsy/RNN_implementation.git

Index terms: Recurrent Neural Networks, Mass Spectrometry Imaging (MSI), Imaging Mass Spectrometry (IMS), cancer classification, tumor identification, deep learning, long short-term memory (LSTM),

I. Introduction

NEURAL networks can be a potent tool in many fights [1], [2]; one most fatal, amongst others, being cancer identification. Early diagnosis of the decease can increase the chance of survival in patients [3] and Recurrent Neural Networks (RNN) may have the ability to do so [4]. Conventional machine learning methods of feature extraction and classification, such as Principal Components Analysis (PCA) and Linear Discriminant Analysis (LDA) have been used before on mass spectrometry imaging (MSI) data with promising results [5], [6], [7], [8]. Implementations of Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN) seem to provide improvements on biological tissue, such as lung, pancreatic and gastric cancer, detection [9], [10]. Nevertheless, new advancements in neural networks appear to increase the cancer identification accuracy even more [11]. Their recurrent nature has proven useful in multidimensional sequence processing and pattern extraction [12], [13], [14].

The problem that is faced in this paper is MSI analysis of biological tissue and pattern recognition in multidimensional data. Images can only provide a limited amount of information. MSI data, on the other hand, like the one use in this paper, carry more than 200000 variables per tissue sample scanned. Therefore, the higher dimensionality of MSI data may be more indicative of cancerous tissue than 2-dimensional or 3-dimensional images.To test this hypothesis, MSI data needs to be analysed with a proper tool and conclude whether cancer identification is possible, and useful, or not.

Related work on the main problem explained above, which this paper is focused on, has been done notably using conventional feature extraction and CNN [5] [9]. In the work of T. Boskamp, [5], it was attempted to correctly classify cancer under two different tasks: discriminating between lung and pancreatic cancer (Task LP) and disciminating between lung carcinoma and squamous cell carcinoma (Task ADSQ). The data analysis process proposed uses feature extraction from MSI data and then classification of the those features. In the aforementioned paper, the authors compare two feature extraction methods, namely the discriminating receiver operating characteristic (ROC) values method and the characteristic spectral patterns (CSP) analysis. The CSP method is a variant of the principal component analysis (PCA) method but it is more suited for biological tissue thanks to its non-negative sparse data-model which is easier to interpret from a biological perspective. The ROC method aims at identifying individual mass-to-charge ratio (m/z) values with different statistical distributions. The classification used in the paper of T.Boskamp is linear discriminant analysis (LDA) which computes a linear transformation to separate the groups of extracted feature data. LDA attempts to minimize the ration of class variance, the variance of the mean of the target groups, over the within-class variance, the variance of all spectra in a group and their peers in the same group [5]. As it is concluded in the end of the same paper, cancer classification was successful with both feature extraction methods performing comparably on the ADSQ task. CSP exhibited an edge in cancer detection in the LP task with about 99% accuracy on a core level, taking the whole tissue into account. On the ADSQ task the accuracy of both methods achieved about 80% with a maximum of 86.2% accuracy using CSP at the core level [5].

A second example of MSI data analysis for cancer classification comes from the work of J.Behrmann who used a deep Convolutional Neural Network, IsoteNet [9]. J.Behrmann and associates, use no preprocessing for the data collected after the matrix-assisted laser desorption/ionization imaging mass spectrometry (MALDI-IMS). Their convolutional network consists of one input layer, 4 residual layers of depth 2 stacked on top of each other, one Rectified Linear Unit ReLU, one locally connected network and a dense layer with softmax activation. They use the data obtained by the paper T. Boskamp [5] to compare their neural network approach with the conventional machine learning approach. J.Behrmann, and associates, conclude that the CNN outperforms the feature extraction methods and LDA classification of the T. Boskamp paper [9]. The maximum balanced accuracy reached was

TABLE I
CROSS VALIDATION SCHEME OF THE DATA EVALUATION

Repetition	Training data	Test data
1	S2, S3, S5, S6, S7, S8	S1, S4
2	S1, S3, S4, S5, S6, S8	S2, S7
3	S1, S2, S3, S4, S5, S7	S6, S8
4	S1, S2, S4, S6, S7, S8	S3, S5

S1 to S8 represent the sets that are assigned into 4 repetitions

88.5% ± 0.2 accuracy on the core level in the ADSQ task.

In this paper, the performance of different RNN architectures with a set of predefined parameters is recorded and listed in terms of accuracy and time costs. The data from the J.Behrmann paper which are also used in the T. Boskamp paper [9], will also be used and compared along side the results of the T. Boskamp paper. The RNN best performing model is then improved and compared with the existing, state of the art, cancer classification methods mentioned above. It is then verified that the present improvement of the best Long Short-Term Memory (LSTM) model can outperform the average golden standard technique used today for cancer identification.

II. METHODOLOGY

A. Establishing baseline performances

To position the work done in this paper in the scientific community 2 baselines have been established to compare the performance of the hypothesis stated earlier, section I. At first, the classification method mentioned in the paper of T. Boskamp [5] was implemented following closely the design choices made in the same work. As a feature extraction method, a simple PCA was implemented with an LDA classifier. The ADSQ task data was used from the same paper, [5], because there is room for improvement on their classification results. Data normalization was set to ”tic” as it was concluded by T. Boskamp that said normalization yielded the highest accuracy overall.

The data was divided into 8 sets and were trained/tested with a cross validation scheme of 4 repetitions consisting of 6 independent training sets and 2 test sets, like in the paper of T. Boskamp, as illustrated in table I. After training the balanced accuracy metric was used to indicate the performance of the method on the two classes, cancer of healthy tissue. Balanced accuracy is not biased by the class proportions and that is beneficial for a binary classifier; because sensitivity of one class equals the specificity of the other. Balanced accuracy is also referred as the area under the curve (AUC) . Therefore, at the end of training, the balanced accuracies of all four repetitions were calculated and their average value was the indicative performance metric of this baseline.

In addition to establishing a working method of the T. Boskamp paper, the J.Behrmann method was also implemented with the same data, to have a neural network reference for the RNN improvement. The code for this implementation was taken directly from the J.Behrmann paper. No adjustments were made to the J.Behrmann method; only the results were retried for further comparisons.

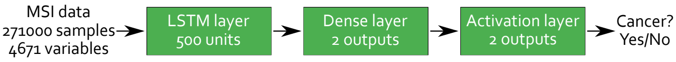

Fig. 1. 1-layer RNN network structure

B. Recurrent Neural Network implementation

To construct the RNN network the tensorflow library from Python was used along with the keras module for faster development. The same data from the two papers mentioned in the subsection II-A was used. As a stepping stone the method followed by the paper of Jinlei Zhang, [11], was first tested. The aforementioned work utilizes an one-layer Long Short-Time memory hidden layer with a dense layer and an activation layer with the softmax activation function, see fig. 1. The labels were transformed into one-hot encoding which is necessary for training of the network in keras. The hyper parameters of this first network design can be found in table II.

Using the same data split as the T. Boskamp and J.Behrmann papers 4 independent models are created and trained, using the keras build-in functions. The only modification that was done on the data was that a validation set was taken out from the training set defined in table I. The resulting cross validation scheme for the RNN development consists of the set training and test data as in the previous methods with the only addition that a validation set is created by randomly removing samples from the training data and assigning it to the validation data, see table III.

During model fitting, the training is temporally stopped from after the training data loss function or the binary accuracy has reached a certain threshold, varying across training. These thresholds have been set after experimentation on the validation data that indicated the limits of model training without over-fitting. After a threshold is met, the training stops and the balanced accuracy of the validation set is measured. Then the training is resume for the next in the list threshold, also known as callback, see table IV. After the last callback, the average balanced accuracy over all callbacks is compared and the highest performing training moment on the validation set is then used to predict the labels on the test data. The value of the balanced accuracy of the test data is considered the indicative performance of the given network.

After the one layer LSTM network implementation the following improvements have been carried out to check whether the performance of the RNN architecture can be further improved:

- Increasing the number of units in the LSTM layer
- Stacking more LSTM layers
- Using bidirectional LSTM layers

It is important to mention, that stacking more LSTM layers was implemented in a very precise order. Firstly, one layer was trained; the weights of which were saved. Then a second layer was constructed and the weights of the first layer were frozen and the layer was set to ”non-trainable”. The first layer, was also set to pass the state and sequence information to the second layer so that the second layer can reconstruct the input and be able to predict the labels upon training the weights of the added layer. This process was repeated with any additional layer until no improvements from the added layers was observed.At the end of the last layer which showed promise for improvement. The weights were saved and the same network work trained as a whole again, with a very small varying learning rate to fine tune its performance.

When constructing a RNN with bidirectional LSTM layers one needs to define how the merging function should be when combining the output of the LSTM training from both directions. For this purpose the performances of merging by taking the average, the maximum value and of concatenation were compared to find which one performed the best.

A core network was determined to be the basis of all modifications and act as reference. Because of the large running time of networks with many units, especially when stacking layers, the standard unit number was determined to be 50. This decision was taken after first running the varying units tests to see which one has the fastest conversion time. 50 units produced, relatively, good accuracy results with the fastest running time; therefore, it is good choice for a standard number of units.

When implementing any change to the original, base network, the other parameters remain the same. This base network consists of one LSTM layer with 50 units, and the other parameters are the same as in table II. At the end of all training, the best network architecture was chosen out of the best performing parameter settings. Although incremental steps in improving the performance of the networks architectures were taken it is important to mention that this strategy might not be optimal. Like in any gradient descent problem an algorithm might saturate at a local maximum/minimum point in the search space. That is because the data complexity might be such that at a certain number of LSTM layers the performance may not increase further but at a significant addition of layers, the network structure might be good enough to identify a better classifying model for said data. For this reason, after the improvements, mentioned above, resulted in no further improvement a few more steps in the direction of non-increasing performance were taken to test this hypothesis. The resulting structure of the search steps/space that was followed can be found in table V.

TABLE II
HYPER-PARAMETERS OF 1 LSTM LAYER NETWORK

Parameter	Value
LSTM number of units	500
Batch size	32
Learning rate	0.005
Maximum epochs for training	300
Optimizer	RMSprop
Loss function for training	binary cross entropy
Bias vector initialization	HE normalization
Weight vector initialization	HE normalization
Recurrent kernel vector initialization	HE normalization
Activation function	softmax

TABLE III
RNN TRAINING CROSS VALIDATION SCHEME

Rep.	Training data	Validation data	Test data
1	S2*, S3*, S5*, S6*, S7*, S8*	Sv	S1, S4
2	S1*, S3*, S4*, S5*, S6*, S8*	Sv	S2, S7
3	S1*, S2*, S3*, S4*, S5*, S7*	Sv	S6, S8
4	S1*, S2*, S4*, S6*, S7*, S8*	Sv	S3, S5

S1 to S8 represent the sets that are assigned into 4 repetitions
* These sets contain the same samples as the original S1-8 sets but with some samples randomly removed and assigned to the validation set Sv

TABLE IV
TRAINING CALLBACKS

Callback priority	conditions
1	*binary accuracy* > 0.95
2	*binary cross-entropy* < 0.1
3	*binary cross-entropy* < 0.05
4	*binary cross-entropy* < 0.01
5	*binary cross-entropy* < 0.005
6	*binary cross-entropy* < 0.001

TABLE V
STEPS TAKEN IN SEARCH FOR BEST PERFORMANCE

Steps	Accuracy	run time [s]
Units: 1	0.7669	1719.26
Units: 2	0.7796	1360.97
Units: 5	**0.7932**	1013.58
Units: 10	**0.7933**	851.90
Units: 20	0.7917	695.63
Units: 50	0.7746	552.97
Units: 100	0.7750	587.97
Units: 200	0.7909	595.19
Units: 500	0.7845	619.33
Units: 1000	0.7775	695.48
Layers: 1	-	10
Layers: 2	-	10
Layers: 3	-	10
Layers: 5	-	10
Layers: 8	-	10
Layers: 10	-	10
Merging*: average	-	10
Merging*: maximum	-	10
Merging*: concatenation*	-	10

* Merging refers to the Bidirectional LSTM layer merging method

C. 2D image data combination

III. RESULTS

A. MSI data classification

The results of the first baseline method, applied by modifying the T. Boskamp implementation are shown in figure 2. The accuracy at 100 features, using PCA as feature extraction and LDA as classifier, reach an accuracy of 82.17%. In the T. Boskamp paper, the actual resulting accuracy of the same parameters, task ADSQ, core level, 100 feature, reaches a value of about 82.50%. The implemented PCA approach deviates by an average 6.1% from the CSP implementation. The reason for this is because CSP represents a more accurate biological tissue representation of features than PCA at which later the LDA classifier can perform better.

The results of the implementation of the Isotope CNN, are as described in the J.Behrmann paper, [9]. The balanced

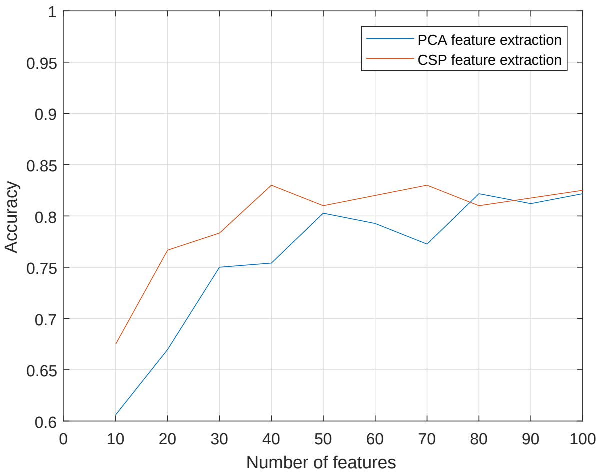

Fig. 2. Baseline accuracy performance of conventional machine learning techniques for cancer classification using MSI data

accuracy for the ADSQ task, at core level, is found to be 87.2% which is as defined in the respective paper.

The results of the RNN implementation, while searching for a good RNN architecture model, are as explained earlier in table V.

From this table it is clearly visible that the following parameters result to the best accuracy performance: (1) Units: (2) Layers: (3) LSTM . When combined together the resulting balanced accuracy is: .

B. 2D Imaging and MSI data combination

IV. Discussions

V. Conclusions

References

[1] G. J. Mendis, J. Wei, and A. Madanayake, "Deep learning cognitive radar for micro UAS detection and classification," in *2017 Cognitive Communications for Aerospace Applications Workshop, CCAA 2017*, 2017.

[2] J. H. Cole, R. P. Poudel, D. Tsagkrasoulis, M. W. Caan, C. Steves, T. D. Spector, and G. Montana, "Predicting brain age with deep learning from raw imaging data results in a reliable and heritable biomarker," *NeuroImage*, vol. 163, pp. 115–124, 2017. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1612/1612.02572.pdf

[3] American Cancer Society, "Cancer facts & figures 2018," pp. 3–8, 2018. [Online]. Available: https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2018/cancer-facts-and-figures-2018.pdf

[4] R. K. Brouwer, "A method for training recurrent neural networks for classification by building basins of attraction," *Neural Networks*, vol. 8, no. 4, pp. 597–603, 1995. [Online]. Available: https://ac.els-cdn.com/089360809400102R/1-s2.0-089360809400102R-main.pdf?{_}tid=18d7d265-d605-4c2f-bae1-36668cc25371{&}acdnat=1523796355{_}ecd17365ef92670181408e4e0e40d9b1

[5] T. Boskamp, D. Lachmund, J. Oetjen, Y. Cordero Hernandez, D. Trede, P. Maass, R. Casadonte, J. Kriegsmann, A. Warth, H. Dienemann, W. Weichert, and M. Kriegsmann, "A new classification method for MALDI imaging mass spectrometry data acquired on formalin-fixed paraffin-embedded tissue samples," *Biochimica et Biophysica Acta - Proteins and Proteomics*, vol. 1865, no. 7, pp. 916–926, 2017. [Online]. Available: https://ac.els-cdn.com/S1570963916302308/1-s2.0-S1570963916302308-main.pdf?{_}tid=061910df-082d-4f6d-8523-b3328116c077{&}acdnat=1523787850{_}c7bad088ef8de2c41b74024fb558147d

[6] M. Kriegsmann, R. Casadonte, J. Kriegsmann, H. Dienemann, P. Schirmacher, J. H. Kobarg, K. Schwamborn, A. Stenzinger, A. Warth, and W. Weichert, "Reliable entity subtyping in Non-small cell Lung Cancer by MALDI Imaging Mass Spectrometry on Formalin-fixed Paraffin-embedded Tissue Specimens." *Molecular & cellular proteomics : MCP*, p. mcp.M115.057513, 2016. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5054336/pdf/zjw3081.pdf

[7] S. A. Thomas, A. M. Race, R. T. Steven, I. S. Gilmore, and J. Bunch, "Dimensionality Reduction of Mass Spectrometry Imaging Data using Autoencoders," *IEEE Symposium Series on Computational Intelligence*, 2016. [Online]. Available: http://vigir.missouri.edu/{~}gdesouza/Research/Conference{_}CDs/IEEE{_}SSCI{_}2016/pdf/SSCI16{_}paper{_}558.pdf

[8] "Data-driven identification of prognostic tumor subpopulations using spatially mapped t-SNE of mass spectrometry imaging data," *Proceedings of the National Academy of Sciences*, vol. 113, no. 43, pp. 12 244–12 249, 2016. [Online]. Available: http://www.pnas.org/content/pnas/early/2016/10/07/1510227113.full.pdf?with-ds=yeshttp://www.pnas.org/lookup/doi/10.1073/pnas.1510227113

[9] J. Behrmann, C. Etmann, T. Boskamp, R. Casadonte, J. Kriegsmann, and P. Maass, "Deep Learning for Tumor Classification in Imaging Mass Spectrometry," 2017. [Online]. Available: https://seafile.zfn.uni-bremen.de/d/85c915784e/http://arxiv.org/abs/1705.01015

[10] P. Inglese, J. S. McKenzie, A. Mroz, J. Kinross, K. Veselkov, E. Holmes, Z. Takats, J. K. Nicholson, and R. C. Glen, "Deep learning and 3D-DESI imaging reveal the hidden metabolic heterogeneity of cancer," *Chem. Sci.*, vol. 8, no. 5, pp. 3500–3511, 2017. [Online]. Available: http://xlink.rsc.org/?DOI=C6SC03738K

[11] J. Zhang, J. Liu, Y. Luo, Q. Fu, J. Bi, S. Qiu, Y. Cao, and X. Ding, "Chemical substance classification using long short-term memory recurrent neural network," in *17th IEEE International Conference on Communication Technology (ICCT 2017)*. IEEE, October 2017. [Online]. Available: http://epubs.surrey.ac.uk/842529/

[12] G. Hadjeres, F. Pachet, and F. Nielsen, "DeepBach: a Steerable Model for Bach Chorales Generation," 2016. [Online]. Available: https://arxiv.org/pdf/1612.01010.pdfhttp://arxiv.org/abs/1612.01010

[13] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent Neural Network Regularization," *Iclr*, no. 2013, pp. 1–8, 2014. [Online]. Available: https://arxiv.org/pdf/1409.2329.pdfhttp://arxiv.org/abs/1409.2329

[14] N. H. Tran, X. Zhang, L. Xin, B. Shan, and M. Li, "De novo peptide sequencing by deep learning," *Proceedings of the National Academy of Sciences*, vol. 114, no. 31, pp. 8247–8252, jul 2017. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/28720701