

Holistic Brain Tumor Screening and Classification Based on DenseNet and Recurrent Neural Network

Yufan Zhou¹, Zheshuo Li¹, Hong Zhu², Changyou Chen¹, Mingchen Gao¹,
Kai Xu³, and Jinhui Xu¹

¹ Department of Computer Science and Engineering
University at Buffalo, the State University of New York

² School of Medical Information
Xuzhou Medical University

³ Affiliated Hospital of Xuzhou Medical University

Abstract. We present a holistic brain tumor screening and classification method for detecting and distinguishing multiple types of brain tumors on MR images. The challenges arise from the significant variations of location, shape, size, and contrast of these tumors. The proposed algorithms start with feature extraction from axial slices using dense convolutional neural networks; the obtained sequential features of multiple frames are then fed into a recurrent neural network for classification. Different from most other brain tumor classification algorithms, our framework is free from manual or automatic region of interests segmentation. The results reported on a public dataset and a population of 422 proprietary MRI scans diagnosed as normal, gliomas, meningiomas and metastatic brain tumors demonstrate the effectiveness and efficiency of our method.

1 Introduction

Brain tumor is one of the most fatal cancers. In the United States, an estimated 700,000 people are living with primary brain and central nervous system tumors. Nearly 80,000 new cases of primary brain tumors are diagnosed yearly, and approximately one-third are malignant [1]. Many different types of brain tumors exist. The most prevalent brain tumor types in adults are gliomas and meningiomas.

Medical imaging plays a central role in diagnosing brain tumors. There are many imaging modalities that can provide information about brain tissue non-invasively, such as Magnetic Resonance Images (MRI), Computed Tomography (CT) and Positron Emission Tomography (PET). MRI has particularly been used

Yufan Zhou, Zheshuo Li, Hong Zhu are equally contributed co-first authors; Kai Xu and Jinhui Xu are co-corresponding authors.

frequently in brain tumor detection and identification, due to its high contrast of soft tissue, high spatial resolution and free of radiation. Despite these facts, brain tumor diagnosis still remains a challenging task. Its detection heavily relies on the experience of radiologists, and diagnosing a large amount of data can be quite time-consuming and sometimes non-reproducible.

Computer-Aided Diagnosis (CAD) can provide tremendous help in brain tumor diagnosis, prognosis and surgery. A typical brain tumor CAD system consists of three main phases, tumor region of interest (ROI) segmentation, feature extraction, and classification (based on the extracted features) [4,6,5]. Brain tumor segmentation, either manual or automatic, is perhaps the most important and time-consuming phase of such a system. A great deal of effort has been devoted to this problem, e.g., releasing publicly available benchmark datasets and organizing challenges [10]. Many algorithms have been proposed to solve the brain tumor segmentation problem, such as Deep Neural Networks [7] and SVM with Conditional Random Field [3]. Classifications based on SVM and/or ANN are then followed to distinguish different types of brain tumors based on the extracted features from ROIs. An obvious limitation of such frameworks is the need of tracing ROIs, which can cause a few problems. Firstly, since brain tumors can vary dramatically in their shapes, sizes, and locations, tracing ROIs could be quite challenging and often not fully automatic. This may cause significant errors to the segmentation, and be accumulated into the following phases, thus leading to inaccurate classification. Secondly, the tumor-surrounding tissues are suggested to be discriminative between different tumor categories [5]. Thirdly, relying solely on the features of ROIs means complete ignorance of the location information of the tumors, which can affect the classification considerably.

The aforementioned problems motivate us to propose an alternative approach for brain tumor screening and classification, eliminating the segmentation phase completely. Particularly, we propose to use the holistic 3D images directly without detailed annotation at the pixel or slice levels. Our approach models the 3D holistic images as sequences of 2D slices. It first adopts an auto-encoder, based on a deep DenseNet, to extract features of each 2D image. This allows us to avoid using the original noisy and high dimensional data. After features of 2D slices extracted, it is natural to apply a Recurrent Neural Network (RNN), specifically the Long Short Term Memory (LSTM) model to handle the sequential data for the classification. We also apply a purely convolutional model for sequential data, by stacking 2D slices features together to be treated as another image data. This is inspired by a recent work of using purely convolutional auto-encoder for sequence representation learning [12].

Our contributions in this work are three-fold:

- The proposed models only need holistic label of patients other than pixel-wise/slice-wise labeling. Holistic labels are much easier to obtain in clinical routine.

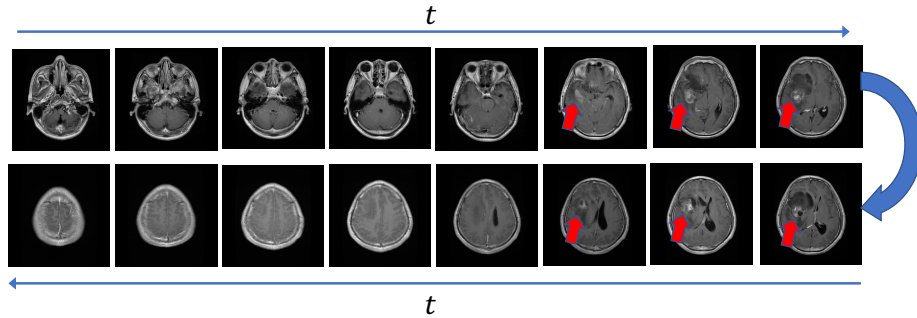


Fig. 1. An MR image sequence of a glioma patient.

- We have collected a dataset of 422 MRI scans, containing normal control images as well as three types of brain tumors (i.e., meningioma, glioma, and metastasis tumor).
- Our deep neural network implements a novel architecture, treating 3D data as sequences of 2D slices, and using RNN or CNN to learn sequence-to-label mapping, with a DenseNet based auto-encoder for feature extraction. Two proposed models **DenseNet-LSTM** and **DenseNet-DenseNet** are demonstrated with two experiments **tumor screening** and **tumor type classification** using both public and proprietary datasets.

2 Preliminaries

2.1 Brain-Tumor Image Representations

Brain tumors are usually diagnosed with MRI or CT images, where patient i is represented by a sequence of 2-D images, denoting as $\mathbf{X}_i = \{\mathbf{x}_1^{(i)}, \dots, \mathbf{x}_T^{(i)}\}$ with $\mathbf{x}_t^{(i)} \in \mathbb{R}^{\ell_1 \times \ell_2}$ being the t -th frame image. Different from existing label-exhaustive datasets where each 2-D image is associated with a label, in our dataset, each sequence of images \mathbf{X}_i is associated with a single label $y_i \in \{0, 1, \dots, P\}$, where P is the number of tumor types. As a result, our dataset is represented as $\mathcal{D} \triangleq \{(\mathbf{X}_i, y_i)\}_{i=1}^N$ with N being the total number of image sequences (including patients and normal people). Fig. 1 illustrates an example sequence of MRI images from a Glioma patient in our proprietary dataset. Note that there are only a few frames showing the existence of Glioma.

2.2 DenseNet

DenseNet [9] is a recently proposed special type of convolutional neural networks, where the current layer is connected by all its previous layers. The structure

The anonymized proprietary dataset will be shared publicly with labels later on.

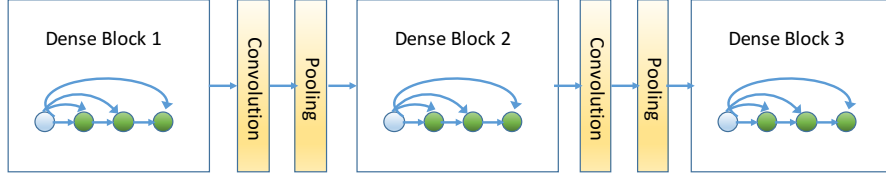


Fig. 2. A Deep DenseNet with 3 dense blocks. In each dense block, the input for a particular layer is the concatenation of all outputs from its previous layers; the output is obtained by convolving the input with some kernels to be learned.

has some advantages over existing structures such as alleviating the vanishing-gradient problem, strengthening feature propagation, encouraging feature reuse, and reducing the number of parameters. A deep DenseNet is defined as a set of DenseNets (called dense blocks) connected sequentially, with additional convolutional and pooling operations between consecutive dense blocks. By such a construction, we can build a deep neural network flexible enough to represent complicated transformations. An example of the deep DenseNet is illustrated in Fig. 2.

2.3 Recurrent Neural Network (RNN)

RNN is a powerful framework to model sequence-to-sequence data. In our brain tumor application, the input sequence corresponds to features of the MRI images, which are extracted with a DenseNet described above; the output sequence degenerates to a single label, indicating whether the input sequence is diagnosed as tumor or not. Specifically, consider an input sequence $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_T\}$, where \mathbf{x}_t is the input data vector at

time t . The corresponding hidden state vector \mathbf{h}_t at each time t is recursively calculated by applying a transition function $\mathbf{h}_t = \mathcal{H}(\mathbf{h}_{t-1}, \mathbf{x}_t)$ (specified below). Finally, the output y is calculated by mapping the final state \mathbf{h}_T to the label space. Fig. 3 illustrates the RNN structure in our setting.

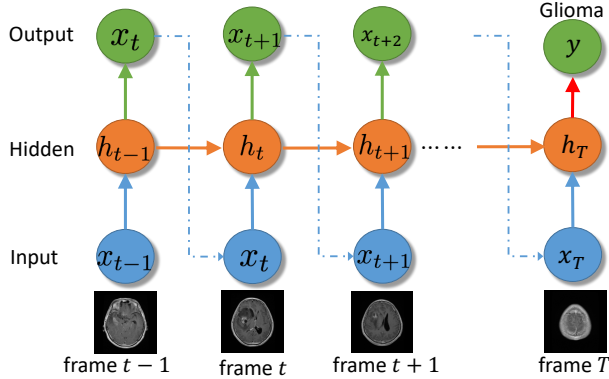


Fig. 3. The RNN structure.

Long Short-Term Memory (LSTM) Vanilla RNN defines \mathcal{H} as a linear transformation followed by an activation function. This simple structure is unable to model long-term dependency from the input, as is the case in our application. Instead, we adopt the more powerful LSTM transition function by introducing a memory cell that is able to preserve the state over long periods [8]. Specifically, each LSTM unit contains a cell \mathbf{c}_t at time t , which can be viewed as a memory unit. Reading or writing the cell is controlled through sigmoid gates: input gate \mathbf{i}_t , forget gate \mathbf{f}_t , and output gate \mathbf{o}_t . Consequently, the hidden units \mathbf{h}_t are updated as:

$$\begin{aligned}\mathbf{i}_t &= \sigma(\mathbf{W}_i \mathbf{x}_t + \mathbf{U}_i \mathbf{h}_{t-1} + \mathbf{b}_i), & \mathbf{f}_t &= \sigma(\mathbf{W}_f \mathbf{x}_t + \mathbf{U}_f \mathbf{h}_{t-1} + \mathbf{b}_f), \\ \mathbf{o}_t &= \sigma(\mathbf{W}_o \mathbf{x}_t + \mathbf{U}_o \mathbf{h}_{t-1} + \mathbf{b}_o), & \tilde{\mathbf{c}}_t &= \tanh(\mathbf{W}_c \mathbf{x}_t + \mathbf{U}_c \mathbf{h}_{t-1} + \mathbf{b}_c), \\ \mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, & \mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t)\end{aligned}$$

where $\sigma(\cdot)$ denotes the logistic sigmoid function, and \odot represents the element-wise matrix multiplication operator. $\mathbf{W}_{\{i,f,o,c\}}$, $\mathbf{U}_{\{i,f,o,c\}}$ and $\mathbf{b}_{i,f,o,c}$ are the weights of the LSTM to be learned. Having obtained the hidden unit for the last time step T , we map \mathbf{h}_T to y by simply using a linear transformation followed by a softmax-layer, *i.e.*, $p(y = k | \mathbf{h}_T) = \text{Softmax}_k(\mathbf{W}_y \mathbf{h}_T + \mathbf{b}_y)$, where $\text{Softmax}_k(\mathbf{a}) \triangleq \frac{\exp(\mathbf{a}_k)}{\sum_i \exp(\mathbf{a}_i)}$, and \mathbf{W}_y and \mathbf{b}_y are the parameters to be learned.

3 Labeling-free Brain-Tumor Classification

We describe our model based on the above building blocks. Different from existing methods for tumor classification using a standard alone CNN, we propose two models to predict image sequences directly, completely eliminating the time consuming procedure of labeling each frame independently, thus free of labeling.

3.1 DenseNet-LSTM model

There are mainly two challenges in our task: *i)* Directly using CNN to tackle image sequences is inappropriate as CNN is originally designed for static data. Fortunately, LSTM provides us a natural way to deal with sequence data. As a result, we adopt LSTM for image-sequence classification. *ii)* Directly feeding original image sequences to an RNN works poorly because the original images are usually noisy and high-dimensional.

To alleviate this problem, we propose an auto-encoder structure based on the deep *DenseNet* to extract features of the original images. The features from the auto-encoder are then fed to an RNN for classification. Specifically, in an auto-encoder, one trains an encoder and a decoder together, to reconstruct the output the same as input. To train the auto-encoder given brain-tumor images $(\mathbf{x}_t^{(i)})_{i,t}$, the objective is to minimize the reconstruction error: $\mathcal{F} = \sum_i \sum_t \left\| \mathbf{x}_t^{(i)} - \text{DEC}(\text{ENC}(\mathbf{x}_t^{(i)})) \right\|^2$, where $\|\cdot\|$ is the standard Frobenius norm;

$\text{ENC}(\cdot)$ and $\text{DEC}(\cdot)$ denote the encoder and decoder implemented by two deep DenseNets, respectively. After training the auto-encoder, the extracted features for all the images are then used as the input data to train an RNN classifier for holistic brain-tumor classification. We adopt the standard cross-entropy loss function to train the RNN. The whole framework is illustrated in Fig. 4. We denote this model as DenseNet-LSTM.

3.2 DenseNet-DenseNet model

An alternative way to RNN for sequence classification discovered recently is to replace the RNN with a CNN [12]. We stack the features of a tumor-sequence returned from the auto-encoder as a 2-D tensor, and treat it as input data to a second deep DenseNet for classification. In this way, the inter-frame correlations is translated into column-wise correlations in a single 2-D tensor, which can be effectively modeled by the convolutional operator in a DenseNet. We denote this model as DenseNet-DenseNet.

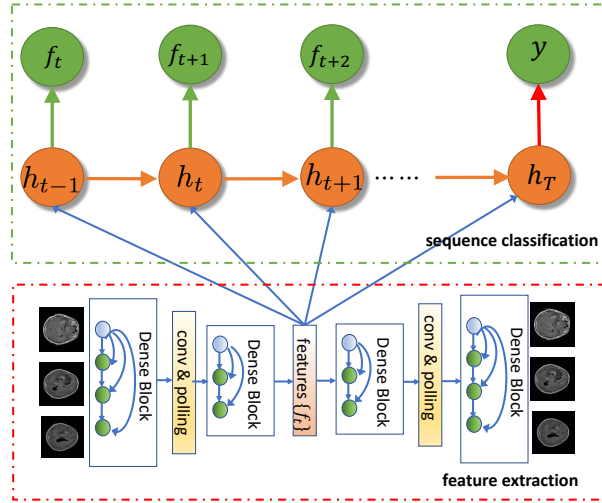


Fig. 4. The proposed DenseNet-LSTM model for labeling-free brain tumor classification.

4 Experiments

We test our proposed framework on two datasets, one public dataset and one proprietary dataset (collected by our collaborators in their hospital). We have two experiments to evaluate the proposed models: Tumor screening and tumor type classification. Tumor screening is for testing the accuracy of our approach on deciding (or screening) whether a 2D sequence image contains a tumor. Tumor type classification is to classify tumors into multiple types.

Our implementation is based on TensorFlow. To alleviate overfitting, we adopt the weight-decay regularization and dropout in the training. The auto-encoder part needs to be trained only once. It takes around 5 hours for 10,000 slices from 500 MRI sequences. The second part takes about half an hour for LSTM or one hour for DenseNet. The models were trained on a Nvidia Titan Xp GPU. For all

the experiments, we randomly partition the dataset into a training dataset (72%), a test dataset (14%) and a validation dataset (14%). We repeat this process for six times and report the mean and variance of the accuracies. Fig. 7 shows some examples of learning curves. More detailed experiment settings are given in the supplementary material.

Public Dataset The public dataset [5] includes 3064 (2D) slices of brain MRI from 233 patients, containing 708 meningiomas, 1426 gliomas, and 930 pituitary tumors. The tumors were manually delineated by experienced radiologists. Since our approach does not rely on segmentation, we utilize only the holistic label of each slice to indicate the tumor type. Since this dataset does not have the sequence images needed by our model, we convert each 2D image (slice) into a sequence of 20 slices by either duplicating it 19 times (for DenseNet-DenseNet) or adding 19 zero matrices (for DenseNet-LSTM). Our purpose of using this dataset is for both validating the robustness of the proposed framework and achieving the state-of-the-art performance, though our model is not designed for handling such 2D datasets.

Proprietary Dataset We have collected a dataset of 422 MRI scans diagnosed as normal (75), glioma (150), meningiomas (67) and metastatic brain tumors (130). For each patient, T1, T2 and Flair MR images are available. Examples of the three tumor types are depicted in Fig. 5, which shows high variations of tumors in terms of locations, shapes and sizes.

Experimental Setup In the DenseNet-based auto-encoder, for the encoder, it is a deep DenseNet with 4 dense blocks. In each block, there are 5 convolutional layers with kernel sizes of 3×3 and 1×1 . We adopt the same configurations for the decoder. For other parameters of the DenseNet, we adopt the default setting as in [9]. The dimension of the latent space for RNN is set to 128.

Minibatch size is set to 32. We use a validation set to select the learning rates from $\{1e-1, 1e-2, 1e-3, 1e-4, 1e-5\}$; the dropout rates for the input-hidden layer and each convolutional layer in the DenseNet from $\{0, 0.05, 0.1, 0.15, 0.2\}$, and the weight-decay rates from $\{1e-2, 1e-3, 1e-4, 1e-5\}$.

Tumor Screening The public dataset is not suitable for this task since it only contains images with tumors. We evaluated three models for tumor screening on the proprietary dataset: DenseNet-RNN (with vanilla RNN as a sequence classifier), DenseNet-LSTM and DenseNet-DenseNet. Their accuracies are $87.15\% \pm 3.79\%$, $91.09\% \pm 3.62\%$, $92.66\% \pm 2.73\%$ respectively. DenseNet-DenseNet presents the best performance for the proprietary dataset.

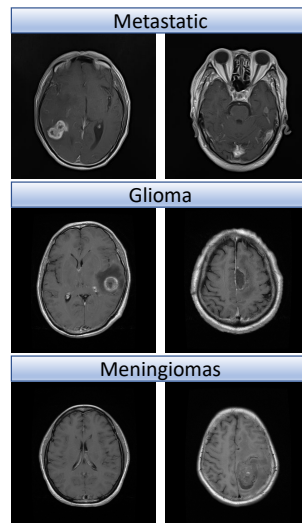


Fig. 5. Examples of the three types of brain tumors.

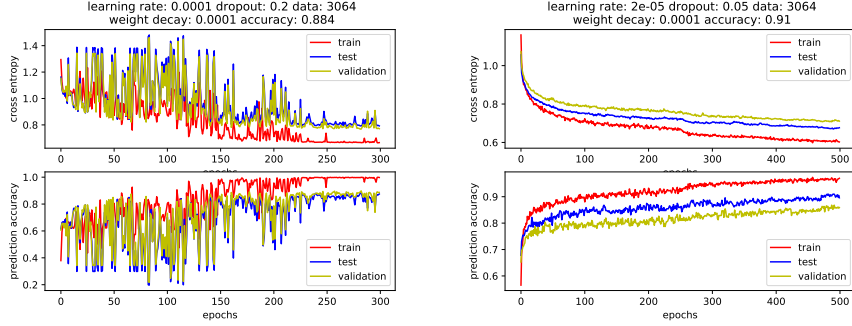


Fig. 6. Learning curves on public dataset. Left: tumor type classification with DenseNet-DenseNet. Right: tumor type classification with DenseNet-LSTM.

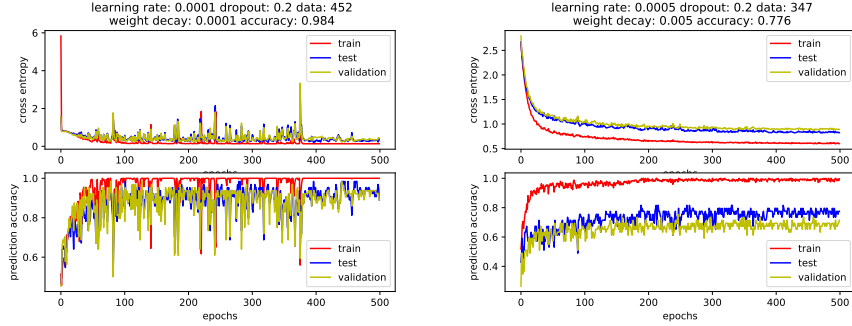


Fig. 7. Learning curves on proprietary dataset. Left: tumor screening with DenseNet-DenseNet. Right: tumor type classification with DenseNet-LSTM.

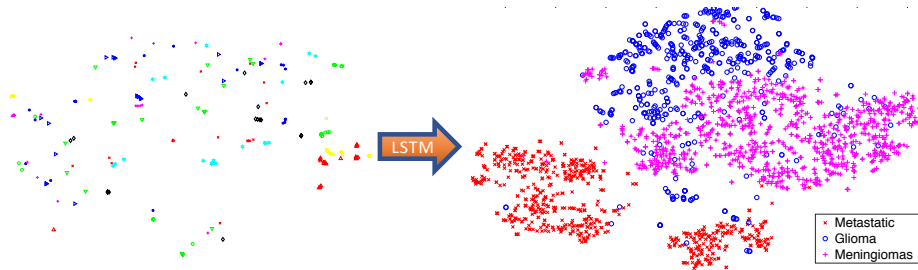
Tumor Type Classification For the public dataset, DenseNet-LSTM outperforms all the previous work on this dataset. The baseline methods [5] reports an accuracy of 91.28% for its best model based on a complicated feature engineering and extra data information (from pixel-wise labeling). A recent model based on capsule networks [2] achieves 86.56% accuracy. Furthermore, our models are much more robust and practically useful because they are designed to handle 3D sequence images and is labeling free.

Our proprietary data is significantly more difficult to learn than the public one. Our DenseNet-LSTM is the best among different variations. DenseNet-LSTM is also tested on one versus one tumor type classification, resulting in three groups of experiments. Table 1 summarized the results. Fig. 7 and Fig. 6 shows the learning curves of our models on proprietary and public dataset, respectively.

Patient embeddings with DenseNet and LSTM features: To illustrate how our proposed framework achieves high discrimination ability, we embed the features from the DenseNet auto-encoder and the LSTM classifier onto a 2-D space, respectively. Note that the features from the auto-encoder do not consider

Table 1. Summary of experimental results on tumor type classification.

Tumor type classification accuracies (three types)					
Models	DenseNet-RNN	DenseNet-LSTM	DenseNet-DenseNet	[5]	[2]
Public	84.61% \pm 1.87%	92.13% \pm 1.59%	86.68% \pm 1.54%	91.28%	86.56%
Proprietary	60.00% \pm 5.70%	71.10% \pm 3.82%	64.95% \pm 5.16%	-	-
Tumor type classification accuracies (two types) with DenseNet-LSTM					
	Glioma vs Meningiomas	Glioma vs Metastatic	Meningiomas vs Metastatic		
Proprietary	80.83% \pm 6.65%	80.00% \pm 8.44%	82.50% \pm 4.18%		

**Fig. 8.** Patient embeddings with DenseNet output (left) and LSTM output (right). Frame-wise patient embeddings (only shows a small number of patients for ease of visibility) in the feature extraction stage (left) are not well-separable; whereas they are almost well-separable after learning with LSTM.

the label information; thus the patients are not expected to be separable from the normal people. Fig. 8 illustrates the corresponding feature embeddings using tSNE [11]. We can see that while patients are not separable in the auto-encoder-feature space, they are highly separable in the feature space learned by LSTM.

5 Conclusion

In this paper, we presented an alternative approach for screening and classifying the brain tumors using holistic 3D MR images. Our approach is capable of utilizing 3D sequence images and does not need the pixel-wise or slice-wise labeling. Experiments on public and proprietary datasets indicate that our approach is effective and highly efficient. As future work, we plan to 1) expand our proprietary dataset for more types of brain tumors, and 2) provide model interpretability based on weakly-supervised pathology localization.

Acknowledgement

This work was supported in part by NSF through grants IIS-1422591, CCF-1422324, and CCF-1716400 and by NSFC through grants 81771904 and 61828205.

It was also supported in part by start-up funds (for Drs. Mingchen Gao and Changyou Chen) from the Department of Computer Science and Engineering, University at Buffalo, the State University of New York.

References

1. Brian tumor statistics. <http://www.abta.org/about-us/news/brain-tumor-statistics/>. Accessed: 2018-02-16.
2. P. Afshar, A. Mohammadi, and K. N. Plataniotis. Brain tumor type classification via capsule networks. *arXiv preprint arXiv:1802.10200*, 2018.
3. S. Bauer, L.-P. Nolte, and M. Reyes. Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 354–361. Springer, 2011.
4. S. Bauer, R. Wiest, L.-P. Nolte, and M. Reyes. A survey of mri-based medical image analysis for brain tumor studies. *Physics in Medicine & Biology*, 58(13):R97, 2013.
5. J. Cheng, W. Huang, S. Cao, R. Yang, W. Yang, Z. Yun, Z. Wang, and Q. Feng. Enhanced performance of brain tumor classification via tumor region augmentation and partition. *PloS one*, 10(10):e0140381, 2015.
6. E.-S. A. El-Dahshan, H. M. Mohsen, K. Revett, and A.-B. M. Salem. Computer-aided diagnosis of human brain tumor through MRI: A survey and a new algorithm. *Expert systems with Applications*, 41(11):5526–5545, 2014.
7. M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle. Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35:18–31, 2017.
8. S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
9. G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269. IEEE, 2017.
10. B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2015.
11. L. van der Maaten and G. Hinton. Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
12. Y. Zhang, D. Shen, G. Wang, Z. Gan, R. Henao, and L. Carin. Deconvolutional paragraph representation learning. In *Advances in Neural Information Processing Systems*, pages 4169–4179, 2017.