

Using Long Short-Term Memory Recurrent Neural Network in Land Cover Classification on Landsat and Cropland Data Layer time series

Ziheng Sun^a, Liping Di^{a,*}, Hui Fang

^a Center for Spatial Information Science and Systems, George Mason University, Fairfax, United States

** ldi@gmu.edu; Tel: +1-703-993-6114; Fax: +1-703-993-6127; mailing address: 4087 University Dr STE 3100, Fairfax, VA, 22030, United States.*

Dr. Ziheng Sun is a research assistant professor in Center for Spatial Information Science and Systems, George Mason University.

Dr. Liping Di is a professor of Geography and Geoinformation Science, George Mason University and the director of Center for Spatial Information Science and Systems, George Mason University.

Hui Fang receives her M.Sc degree from State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University.

This manuscript contains 6328 words.

Using Long Short-Term Memory Recurrent Neural Network in Land Cover Classification on Landsat Time Series and Cropland Data Layer

Land cover maps are significant in assisting agricultural decision making. However, the existing workflow of producing land cover maps is very complicated and the result accuracy is ambiguous. This work builds a long short-term memory (LSTM) recurrent neural network (RNN) model to take advantage of the temporal pattern of crops across image time series to improve the accuracy and reduce the complexity. An end-to-end framework is proposed to train and test the model. Landsat scenes are used as Earth observations, and some field-measured data together with CDL (Cropland Data Layer) datasets are used as reference data. The network is thoroughly trained using state-of-the-art techniques of deep learning. Finally, we tested the network on multiple Landsat scenes to produce five-class and all-class land cover maps. The maps are visualized and compared with ground truth, CDL, and the results of SegNet CNN (convolutional neural network). The results show a satisfactory overall accuracy (>97% for five-class and >88% for all-class) and validate the feasibility of the proposed method. This study paves a promising way for using LSTM RNN in the classification of remote sensing image time series.

Keywords: long short-term memory; recurrent network; deep learning; over-fitting; remote sensing; land cover; image classification

Deep learning; land cover classification

1. Introduction

The long-lasting observation of hundreds of onboard sensors in the past six decades has accumulated a huge volume of satellite images (Computing 2013; Ward 2008; NASA 2016; Burnett, Weinstein, and Mitchell 2007). However, the conventional image analysis techniques only expose a tiny part of the information in this rich mine (Ma et al. 2015; Li, Dragicevic, et al. 2016). In this data-rich yet analysis-poor era, the pace of analyzing data is far behind the speed of obtaining data. Too many manual processes are the major obstacles in the path. Thus deep learning (DL), which is more intelligent and requires less human intervention, becomes more and more popular. Meanwhile, the remote sensing (RS) community has vaguely recognized that conventional classification schemes are frequently driving to dead-end when improving the result accuracy (Canty 2006; Blaschke et al. 2014; Sun et al. 2015; Computing 2013; Hussain et al. 2013). In the

world-renowned ImageNet competition, the DL based solutions often outperform the conventional methods. A recent trend shows that RS researchers are enlightened to choose DL for higher accuracy in their future research. A quite number of studies have already practiced neural networks in classifying RS images and achieved a few satisfactory results.

The success of DL relies on massive training datasets and powerful compute nodes like Graphics Processing Units (GPU). A good network requires careful engineering and considerable domain expertise in network training. Feed-forward neural network (FNN) and recurrent neural networks (RNN) are two commonly used networks. The former feeds information straight through the network, while the latter cycles the information through a loop. A representative of FNN is convolutional neural network (CNN) which is dedicated for captioning objects in images such as faces, plate numbers, and signatures. RNN can form a memory of patterns and is suitable to learn sequential data like speech and text. Today, the application scope of the two networks is overlapped. There are few hardcoded restrictions on the specific use cases of either type of networks. We intend to try RNN on analysis of RS image time series to learn the temporal pattern of agricultural land covers and produce more accurate classification results.

1.1 Problem Statement

Cropland data layer, short as CDL, is a land cover product by United State Department of Agriculture National Agricultural Statistics Service (USDA NASS). Its spatial extent covers the continental U.S. at 30 meters resolution. Landsat dataset is its major data source. It has relatively high accuracy over the other existing products as NASS massively integrated their ground truth data collected by its field offices into it. Many pixel values are based on real field reports rather than classification algorithms. CDL releases only one layer for each year and labels all the pixels with a unified crop hierarchy. As the frequency is low, it is unable to in-time reflect agricultural activities like sowing, irrigation, harvesting. Cropland changes over seasons and many farms carry out multiple cropping within a year. Although CDL hierarchy has double cropping categories (107-122, 132, see the supplemental material), the time frames of each cropping are completely

unknown. Actually, in-time result from current observations is eagerly needed in the agricultural analysis. Higher-frequency updates of CDL-like products are very helpful in agricultural decision making for sure. We want to use DL technique to enhance the data mining in the window interval between CDL releases and provide more information about the land cover changes to the decision makers.

1.2 Contributions

This paper builds an LSTM RNN model to utilize CDL time series and ground-measured data to classify Landsat images. The model aims to generate CDL-like products in a more frequent manner and supplement the missing years in CDL history. This work pre-processed Landsat, CDL time series, and ground truth to get training samples. The network is trained thoroughly with the state-of-the-art techniques from the DL community. Finally, we tested the network on multiple Landsat scenes to produce five-class and all-class land cover maps. The five classes are cropland, non-crop vegetation, developed space, water, and barren land. The all-class maps use the same hierarchy as CDL. The results are plotted on charts and evaluated by reference data. The experiment shows a satisfactory accuracy (>97% for five-class and >88% for all-class) which validates the feasibility of the proposed model.

1.3 Related Work

ANN (artificial neural network), especially deep neural networks (DNN), already has plenty of application in image recognition (LeCun, Bengio, and Hinton 2015). Audebert et al revealed the general foreseeable benefits by DL to remote sensing (Audebert et al.). They tested various DL architectures in the semantic mapping of aerial images and better performances than traditional methods are achieved. Cooner et al evaluated the effectiveness of multilayer feedforward neural networks, radial basis neural networks and Random Forests in detecting earthquake damage by the 2010 Por-au-Prince Haiti 7.0 moment magnitude event (Cooner, Shao, and Campbell 2016). Duro et al compared pixel-based and object-based image analysis approaches for classifying broad land

cover classes over agricultural landscapes using three supervised learning algorithms: decision tree (DT), random forest (RF), and support vector machine (SVM) (Duro, Franklin, and Dubé 2012). Zhao et al used multi-scale convolutional auto-encoder to extract features and train a logistic regression classifier and got better results than traditional methods (Zhao et al. 2015). Kussul et al designed a multilevel DL architecture to classify land cover and crop type from multi-temporal multisource satellite imagery (Kussul et al. 2017). Maggiori et al trained CNN to produce building maps out of high-resolution remote sensing images (Maggiori et al. 2017). Das et al proposed Deep-STEP for spatiotemporal prediction of satellite remote sensing data (Das and Ghosh 2016). They derived NDVI (normalized difference vegetation index) from thousands to millions of pixels of satellite imagery using DL. Marmanis et al used a pre-trained CNN from ImageNet challenge to extract an initial set of representations which are later transferred into a supervised CNN classifier for producing land use maps (Marmanis et al. 2016). Li et al used DL to detect and count oil palm trees in high-resolution remote sensing images (Li, Fu, et al. 2016). Ienco et al evaluated the LSTM RNN on land cover classification considering multi-temporal spatial data from a time series of satellite images (Ienco et al. 2017). Their experiments are made under both pixel-based and object-based scheme. The results show the LSTM RNN is very competitive compared to state-of-the-art classifiers and even outperform classic approaches at low represented and/or highly mixed classes. These successful cases have advised the great potential of DL in RS image recognition, which inspire us to conduct this study using LSTM RNN on satellite image time series for crop classification.

In addition, LSTM RNN has many improved versions to increase the efficiency and accuracy, such as bi-directional LSTM (Schuster and Paliwal 1997; Graves, Mohamed, and Hinton 2013), which is a great extended algorithm of LSTM RNN to overcome the limitations of a regular RNN and is found significantly more effective than the unidirectional ones. However, it doesn't always make sense for all the sequence-to-sequence problems as it relies on the knowledge of the future and a specific time frame. In our case, the study time frame is dynamic. The specific time of

agricultural activities varies every year and the size of available training images changes, which makes the last time step of the sequence uncertain.

2. Materials and Methods

2.1. Study Area and Materials

We choose the North Dakota (Fig. 1), which has a sound historical archive of Landsat and CDL products (Fig. 2), as our study area. North Dakota is a state in the northern U.S. Agriculture is its number-one economic industry (Coon and Leistritz 2010). According to NASS public reports, the agricultural products of North Dakota play a significant part in the overall yield of U.S., especially on corn, soybeans, spring wheat and durum wheat (Jantzi et al. 2017).

The Landsat program has observed the Earth for more than four decades and retrieved about six million scenes (Wikipedia 2014; USGS 2016). The renowned Landsat 5 started to deliver images from the space in 1984 and was decommissioned in 2013. Landsat 7 operated smoothly before May 2003 but generated gaps after then due to the malfunction of its scan line corrector. In 2013, a new member, Landsat 8, was launched into the orbit to continue the mission (Roy et al. 2014). Landsat satellites normally scan the entire Earth about every two weeks at 30-meter resolution.

CDL is a popular yearly product made by USDA covering the continental U.S. It is widely accepted as a general reference on crop distribution. CDL directly fused a lot of ground truth data collected by NASS field offices, resulting in a much better accuracy than the other existing products. The general per-pixel accuracy is claimed as 85% to 95% for major crop types (Boryan et al. 2011). The time coverage of Landsat and CDL in North Dakota is shown in Fig. 2. North Dakota is also the only state that has CDL from the very beginning (the oldest CDL year is 1997). The CDL program only started to provide data for the entire continental U.S. after 2008. Given 1997 is the year when Landsat and CDL began to coexist, the years from 1997 to 2017 are circled into our study time pool.

Figure 1. Study area. The black lines in the map are U.S. county boundaries. The projection is NAD83/Conus Albers (EPSG code: 5070).

Figure 2. The availability of data for North Dakota since 1997

2.2. Recurrent Neural Network and Long Short-Term Memory

RNN has a feedback connection which is the most apparent difference from FNN. The underlying principle of classic RNN is very straightforward. The outputs of previous time steps will be considered as inputs in the current time step, and the results of the current time step will impact the calculation of the next time step. Thus, the historical results will have a long-term influence on the future judgment, which is similar to the definition of memory. Given $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ are the input vectors, $\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_n$ are the hidden cell output vectors and $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n$ are the result vectors, where n represents the total steps. A RNN cell is exhibited in Fig. 3 (the left one). The equations computing result vectors from input vectors are (1)-(3).

$$\mathbf{h}_t = \theta \phi(\mathbf{h}_{t-1}) + \theta_x \mathbf{x}_t \quad (1)$$

$$\mathbf{y}_t = \theta_y \phi(\mathbf{h}_t) \quad (2)$$

$$f_{\tanh_x} = \frac{e^{2x}-1}{e^{2x}+1} \quad (3)$$

where $\theta, \theta_x, \theta_y$ are weights, ϕ is the activation function (tanh in most RNNs). The self-connection weight θ is simply initiated as 1. The subsequent back-propagation will adjust all the weights in every iteration.

Figure 3. RNN and LSTM introduction

LSTM RNN is explicitly designed to avoid the long-term dependency problem (Gers and Schmidhuber 2000) and the underlying principle is a little more complicated to understand. Not all the LSTMs are uniform and almost every paper involving LSTMs uses a slightly different version (Jozefowicz, Zaremba, and Sutskever 2015). We adopt the definition from Graves (Graves,

Mohamed, and Hinton 2013) to explain its internal mechanism. As shown in Fig. 3 (the right one), a cell of LSTM RNN has three extra “gates”, which control the involvement of the past context information. The input gate is responsible for scaling input to the cell. The output gate is to scale the output from the cell. The forget gate is to scale the influence of old cell value on the current cell state. The equations for computing the gate outputs are (4)-(9):

$$\mathbf{i}_t = \sigma(\theta_{xi}\mathbf{x}_t + \theta_{hi}\mathbf{h}_{t-1} + b_i) \quad (4)$$

$$\mathbf{f}_t = \sigma(\theta_{xf}\mathbf{x}_t + \theta_{hf}\mathbf{h}_{t-1} + b_f) \quad (5)$$

$$\mathbf{o}_t = \sigma(\theta_{xo}\mathbf{x}_t + \theta_{ho}\mathbf{h}_{t-1} + b_o) \quad (6)$$

$$\mathbf{g}_t = \tanh(\theta_{xg}\mathbf{x}_t + \theta_{hg}\mathbf{h}_{t-1} + b_g) \quad (7)$$

$$\mathbf{c}_t = \mathbf{f}_t \cdot \mathbf{c}_{t-1} + \mathbf{i}_t \cdot \mathbf{g}_t \quad (8)$$

$$\mathbf{h}_t = \mathbf{o}_t \cdot \tanh(\mathbf{c}_t) \quad (9)$$

where $\mathbf{h}_{t-1}$ is the output of the last time step, $\mathbf{x}_t$ is the cell input at the current step, and $\mathbf{h}_t$ is the cell output. The t in the subscripts represents the current step number. The $\mathbf{i}, \mathbf{f}, \mathbf{o}$, and $\mathbf{g}$ respectively denote the output vectors of input gate, the forget gate, output gate and the cell itself. θ means the weights, for example, θ_{xi} is the weight between the input vector $\mathbf{x}_t$ and the input gate vector $\mathbf{i}_t$, θ_{hi} is the weight between the output vector $\mathbf{h}_{t-1}$ and the gate vector $\mathbf{i}_t$, and so forth. b represents biases. $\mathbf{c}_t$ and $\mathbf{c}_{t-1}$ are the cell outputs in the current step and the previous step. The σ represents a sigmoid function. LSTM usually limits the activation function to tanh for $\mathbf{g}_t$ and $\mathbf{h}_t$, and sigmoid for $\mathbf{i}_t, \mathbf{f}_t$, and $\mathbf{o}_t$. Sigmoid output values scope from 0 (completely get rid of this) to 1 (completely keep this), while tanh scales cell outputs into the range between -1 and 1. Other activations, like the popular RELU (rectified linear unit) (Dahl, Sainath, and Hinton 2013; Nair and Hinton 2010), makes LSTM diverge due to the existing of gates and is not suitable here (Breuel 2015). Therefore, for a LSTM RNN sigmoid and tanh are the common configuration.

Generally, the classic RNN cannot look back too far and LSTM RNN solved that problem (Hochreiter and Schmidhuber 1997). As its extraordinary performance on memory, LSTM RNN has become a very popular choice for modeling inherently dynamic process like voice and

handwriting (Graves, Mohamed, and Hinton 2013) and massively used by tech giants, e.g. Apple, Google, Microsoft, and Amazon. This work also utilizes LSTM RNN and examines its performance in the classification of RS image time series. As the pixel value changes in time series have many similar characters to the signal of speech or handwriting, high accuracy is expected to be achieved in this work.

2.3. End-to-End Framework

An end-to-end framework is designed to train LSTM RNN on Landsat and CDL time series. In the proposed model displayed in Fig. 4, each input pixel is turned into an input vector which includes seven variables corresponding to seven bands of Landsat. As shown in Table 1, the bands of different Landsat satellites are different. For Landsat 8, we use its first to seventh bands. Landsat 7 ETM+ and Landsat 5 TM have no ultra-blue so we only have six bands accordingly. The other bands are not used as they are for special purposes like the cirrus band. Each image represents a status of the cropland and the surface reflectance of the same crop in different growing stages is various, e.g., a farm field is covered by crop in the growing season but by bare soil after harvesting. In other words, each image is parameterized as a time step in LSTM RNN (Fig. 5). LSTM is supposed to remember the trained pixels on each time step and automatically judge and use the knowledge in labeling the inputted pixels in future steps. The mode in Fig. 5 is synced sequence input and output, in which the number of inputs is the same as the number of outputs. We choose this mode and discard others like many-to-one or sequence-input-and-sequence-output, because we want to generate a crop map for each time step even though the input Landsat image is not suitable (e.g., too many clouds) or the time is not good for crop classification (e.g., early-spring). Thus, every year will have a sequence of Landsat images and predicted crop maps. Generally, the crops are still in early growing stage in Spring and hard to recognize on satellite images. Therefore, the maps generated in Spring have lower accuracy than the maps generated in Summer and early Autumn (June, July, August, and September). Therefore, the first several maps in each year significantly depend on empirical knowledge learned by LSTM in training stage and will only be

used as reference to roughly estimate the crops. In the late growing season, the accuracy of the maps is supposed to increase as the patterns of crop growing become much clearer in the latter time steps of LSTM.

Three hidden layers with LSTM neurons are configured. The first layer has 500 neurons and each neuron maps a type of feature. The second layer has 250 neurons which are the high-level composition of the first-layer features. The third layer has 100 neurons which are the more abstract composition of the second-layer features. The three-tier representation tries to establish a mapping between pixels and land cover captions. As for optimization method, there are many optimizers available, like Adam, Adagrad, AdaDelta, Stochastic Gradient Descent (SGD), RMSProp, and Nesterov accelerated gradient. In this work, we choose the SGD to reduce the high cost of backpropagation as it can certainly converge to a local minimum, slowly progress towards the bottom and have little chance to encounter gradient vanishing problem (Merity, Keskar, and Socher 2017). We also tried Nesterov, Adam, and RMSProp and they give slightly quicker learning curves than SGD but easily lead to gradient vanishing problem. In our initial tests, the RMSProp produces relatively better learning curves over other optimizers before it is overfitted. The network using RMSProp was experimentally trained and its results are very similar to SGD network. By contrast, SGD seems to make LSTM remarkably easy to train. Hence, we conservatively use SGD in this study as a compromise solution. In future, we will further study how other optimizers could improve the converging speed and prediction accuracy while avoiding overfitting and divergence in this scenario. The loss function is multiclass cross entropy which is commonly used to classify a set of objects into multiple classes. Backpropagation is enabled for weight updating. The weights are initialized by Xavier's method (Glorot and Bengio 2010). Learning rate is scheduled to decay on a certain iterations so the training can switch to small steps when they are close to the optimum (Senior, Heigold, and Yang 2013).

The output layer uses SoftMax as activation function on each unit of the output vector. Each dimension corresponds to a land cover class. We made two networks for five-class and all-class

experiments respectively. In the five-class hierarchy, 0 means cropland, 1 means vegetation other than crops, 2 means developed space, 3 is water and 4 is bare land. A complete mapping from all classes in CDL hierarchy to the five classes is contained in the supplemental material. The value of each output neuron ranges from 0 to 1, representing the probability of the inputted pixel belongs to the corresponding land cover. The land cover with the highest probability will claim the pixel.

This network configuration is not the only solution to this problem. The depth could be as deep as hundreds of layers and the neurons on the hidden layers could be thousands. Actually, the cutting-edge hardware allows millions of neurons and hundreds of layers present in one network. In theory, the bigger network has more capacity to discover very implicit features, but meanwhile increases the chance of over-fitting. This network benchmarks LSTM RNN in crop classification and three hidden layers with 850 neurons are basically adequate for this study. More experiments are needed on deeper and wider networks in the future.

Figure 4. The proposed model (per pixel)

Figure 5. Many-to-many schema to learn the temporal pattern of crop growth

Table 1. The employed bands of Landsat 5, 7 and 8 (unit: micrometers)

2.4. Training

We collected Landsat surface reflectance (SR) products in the studied period and clipped the area of North Dakota. A subarea of North Dakota is chosen for training (Fig. 6). The mask band is used to filter out the cloud, cloud shadow, bad quality pixels, etc. The CDL of the area from 2013 to 2016 is extracted and pre-processed to match with Landsat pixels. Each color in CDL represents a different crop category, e.g., yellow is corn, green is soybean and red is barley. The pixels must go through a number of processes before being considered as training samples (Fig. 7). We abandoned boundary pixels in the CDL of the corresponding year. Only the clear pixels whose eight-directional neighbors are labeled by the same category are retained. Then those pixels are validated by ground-measured datasets from North Dakota public datasets. Only the qualified pixels to the

above two rules are added into the training dataset. The affine transformation is used to transform the pixel x and y to latitude and longitude. The matching between Landsat pixels and CDL pixels is established via their geospatial coordinates. The band values will be normalized to highlight the signal variance equally. Crop captions for the pixels are mapped to the specially adopted hierarchies, e.g., five classes or all classes (the supplemental material). Finally, the training samples are encoded in multiple CSV files with eight columns (seven inputs and one class label), and each file comes from a different image. The files will be inputted to LSTM in time sequence.

Figure 6. The data series of Landsat and CDL for training. The x-axis is the input bands and the y-axis is the timeline of Julian dates (e.g., 2013143 means 23 May 2013). The false color composites are grey-scaled band images. The four classified maps at the bottom are CDLs from 2013 to 2016. The full legend for CDL is on NASS website (Boryan et al. 2011).

Figure 7. Pre-processing workflow

Training a neural network is a world-class challenge, especially when the training samples are not normally distributed. Over-fitting and under-fitting are two of the most painful things for DL practitioners. Under-fitting means the network is not well trained and the data pattern stays unrecognized. Over-fitting means that the network is over-trained and loses generality on the test dataset. A typical consequence is the accuracy of the training dataset is extremely high but very poor on the test dataset. Dropout is a technique specially designed to avoid over-fitting, but not a universal solution (Srivastava et al. 2014; Krizhevsky, Sutskever, and Hinton 2012; Dahl, Sainath, and Hinton 2013). Many studies still try to figure out a better solution. The methods we used to avoid overfitting are Bias (Schaffer 1993) and L2 regulation (Zibulevsky and Elad 2010).

We picked 13,508,899 samples from eleven Landsat scenes (6 Oct, 2 Sep, 3 Aug, 18 July, 16 June in 2016; 20 Oct, 11 Apr, 26 Mar in 2015; 14 Aug, 13 Jul, 8 Apr in 2014) to fully train the built model. The score-iteration path is plotted in Fig. 8. The score represents the value of

multiclass cross entropy (MCE) which is the loss calculated against every batch of training samples for backpropagation. The MCE bounces back and forth in different batches but is stably decreasing over epochs. After thousands of iterations, the difference of test accuracy between two consecutive iterations will become very small. If the difference is smaller than a threshold, the learning should be terminated to prevent over-fitting. The size of the network determines its tolerance to the complexity of the pattern. The completeness and distribution of training dataset have a direct impact on the steep level of the training curve.

Figure 8. Training 1,474,034 samples in five epochs.

For performance increasing the learning, rate is scheduled to decay along with the iterations (Fig. 9). Meanwhile, to accelerate training the experiment is conducted on the computation node with two Intel(R) Xeon(R) E5-2650 v3 CPUs and a Geforce GTX 1060 6GB GPU. The host operating system is Ubuntu Linux 64bit version 16.04. The training samples are learned by hundreds of epochs. DeepLearning4J API is used to program the model and NVIDIA CUDA is utilized for high-performance parallel processing.

Figure 9. The scheduled drop of learning rate

We use confusion-matrix metrics to measure the model accuracy. The usual metrics include overall accuracy (OA), producer accuracy (PA), user accuracy (UA) and Kappa coefficient (Congalton 1991). The equations are (10)-(17).

$$M_{OA} = \frac{N_{\text{pixels correctly classified}}}{N_{\text{total pixels}}} \quad (10)$$

$$M_{PA_i} = \frac{N_{\text{pixels correctly classified as } i}}{N_{\text{reference pixels of } i}} \quad (11)$$

$$M_{UA_i} = \frac{N_{\text{pixels correctly classified as } i}}{N_{\text{pixels classified as } i}} \quad (12)$$

$$M_{\text{kappa}} = \frac{p_o - p_e}{1 - p_e} \quad (13)$$

$$p_o = N_{\text{pixels correctly classified}} \quad (14)$$

$$p_e = \sum_{i=0}^n p_{.i} * p_{i.} \quad (15)$$

$$p_{.i} = N_{\text{reference pixels of class } i} \quad (16)$$

$$p_{i.} = N_{\text{pixels classified as } i} \quad (17)$$

where i represents crop number, n is the total number of crops, N is the count function, p_o is the relative observed agreement among raters, and p_e is the hypothetical probability of chance agreement. OA is the percentage of correctly classified pixels in all the pixels. PA is evaluated from the view of producers and represents the probability that the reference samples are correctly classified in the results. UA is evaluated from the view of users and reflects the probability that the classified results agree with the reference maps. Kappa coefficient measures the overall agreement between the classified map and the reference map and its value ranges from 0 to 1. 0 means complete disagreement, while 1 means the classified result and the reference data are identical. These metrics can determine if the model is correctly fit on both training and testing datasets and help avoid over-fitting or under-fitting.

3. Results

We applied the trained LSTM RNN on continuous images to produce time-series crop maps. The Landsat footprint is path 031 row 027 in WRS-2 (world reference system) (Irons, Dwyer, and Barsi 2012). The tested images include six scenes on 26 May, 30 Aug and 15 Sep in 2014, 1 Aug and 2 Sep in 2015, and 13 Apr in 2016. All the five-class results are listed in Fig. 10. The proportion of developed and barren land in the study area is small so we merge them with non-crop vegetation into a bigger class - non-crop land. Thus, the predicted final maps only contain three classes. CDL is also mapped into the three classes for comparison. Fig 10 (a), (b), and (c) are CDL in 2014, 2015 and 2016, respectively. Fig. 10 (d), (e), and (f) are the classified results. After manual supervised comparison, we concluded that the results have no obvious differences from CDL. The six maps highly agree on the distribution of water bodies. The top-left water in Fig. 10 (f) is larger than others due to flooding. Small water bodies are precisely labeled in Fig. 10 (d), (e) and (f). The north area in Fig. 10 (b) apparently contains less agricultural fields comparing to CDL in 2014 and 2016.

It is abnormal that a large scale of croplands turns into non-crop lands and switch back to croplands in next year. Fallow might be the case, but in 2015 the rainfall in North Dakota is high which makes fallow unlikely. It is more likely that CDL used a Landsat image in the rainy season and incorrectly classify those croplands into herbaceous wetlands. Correspondingly, the result in Fig. 10 (e) has a better consistency on cropland than CDL in that area.

Figure 10. The three-classes test results and CDL. (a) CDL of 2014; (b) CDL of 2015; (c) CDL of 2016; (d) test result on 26 May 2014; (e) test result on 1 Aug 2015; (f) test result on 14 Apr 2016.

We tested the all-classes LSTM on several scenes in growing season and compared the true-color composite images, CDL, and the results of SegNet (Badrinarayanan, Kendall, and Cipolla 2017) to the results (Fig. 11). SegNet inherits from the fully convolutional neural network (FCNN) (Long, Shelhamer, and Darrell 2015) and is one of the state-of-the-art technologies for semantic segmentation. It proposes a deep convolutional encoder-decoder architecture for robust semantic pixel-wise labeling. It is designed to only require forward evaluation of a fully learned function to obtain smooth label predictions, consider the larger context for pixel labeling with increasing network depth, and can visualize the effect of feature activations in the pixel label space at any depth (Badrinarayanan, Kendall, and Cipolla 2017). Thus, this work chooses SegNet and trained it with the same Landsat images and CDL mentioned in Section 2.4 and tested it in the same region. Regardless of the network and used training sample size, both the LSTM RNN and SegNet achieved good accuracies. The differences are small, scattered and distributed on the edges and intersections. In details, the RNN results are more similar to CDL than the CNN results. In both results, the corn and soybeans are confused in some places, and the edge pixels are a little irregular comparing to CDL. The LSTM RNN results reflect more reasonable changes in crop distribution over the growing season. The landscapes in the growing season of the same year keep basically the same in the RNN results. Ambiguous classified places exist mainly in the alfalfa and grass/pasture, corn and soybeans, barley and spring wheat, which look very similar to each other in some growing

stages and easy to be misclassified. Besides that, the CNN results seem struggling in separating water from spring wheat and alfalfa, which lead to an apparent under-estimation of the lake and rivers. Overall, the RNN and CNN have little differences on the primary crop types (corn, soybean) and the RNN outperformed CNN on secondary crop types (barley, spring wheat, alfalfa). Due to the encoder-decoder mechanism, the pixel clusters in CNN results are bigger and more uniform than LSTM RNN results. It makes the CNN maps neat and smooth, but meanwhile ignores pixel-wise temporal patterns in the areas with the complicated context of multiple crop types. LSTM RNN gives a pixel-wise independent prediction to reflect the crop growing patterns in each pixel area. The RNN results show that the LSTM RNN provides more specific and customized independent analysis for each pixel and contains abundant crop context information of the farms.

Figure 11. Landsat true-colour composites, CDL, and the all-class results of LSTM RNN and SegNet. The x-axis is the product category, and the y-axis is product observation date. The classification legend is at the bottom.

As CDL is a synthetic product rather than ground truth, our result maps are not necessary to mimics CDL. Instead, the absolute accuracy is estimated by metrics based on a more accurate dataset. We created six sample sets of five classes and four sample sets of all classes. The sample sets are derived from various Landsat scenes and ground measured datasets. Each sample goes through the restricted processes in Section 2.4 to ensure their correctness. Table 2 and 3 shows the metrics of five-class and all-class results respectively. In the five-class results, the overall accuracy of LSTM RNN ranges from 97.33% to 99.21%, and in the all-class results ranges from 88.85% to 91.35%. The producer accuracy (PA) and user accuracy (UA) are higher than 93% in the five-class and higher than 80% in the all-class results. The Kappa values of LSTM RNN are greater than 88%, representing very good agreements on the major areas are reached by both producers and consumers. Table 3 also contains the metrics of CNN results, whose overall accuracy (average 86.74%) is generally a little lower than the LSTM RNN results (average 89.73%). By comparison,

the SegNet has less PA, UA, and Kappa than LSTM RNN in this case, which reflects that the LSTM RNN brings improvements on accuracy to the state-of-the-art and is very suitable for crop classification on time series of Landsat scenes.

Table 2. Metrics of the five-class results

Table 3. Metrics of the all-class results by LSTM RNN (R) and SegNet (S)

4. Discussion

4.1 Why using RNN?

Convolutional neural network, short as CNN, is one of the most popular networks for image semantic segmentation (Maggiori et al. 2017). In contrary, RNN is typically used in speech and text processing. However, using RNN in crop classification has several reasons. First, the crop growth is staged and time-sensitive. We take advantage of RNN to discover the temporal variation pattern of crops during the growing season. RNN can detect the coherence among consecutive pixels over time steps and recognize the characteristics of crops in different growing stages. The overall pattern of samples will be more exposed using LSTM RNN where each sample has a long influence on the future judgment. We can increase the sample pool to involve more Landsat scenes without completely overwriting the network memory of old samples (CNN has this problem. Training on a new sample set will under-fit the old one). Thus, this study chooses RNN over CNN and achieved very good results. It indicates the playground of RNN should not be limited and it may harvest exceptional outcome in unconventional scenarios. In addition, recurrent convolutional neural network (RCNN) has been proposed and already put on the table [38]. In future, the performance of CNN, RNN, and RCNN need be further examined on Landsat scenes.

4.2 Why all-class results less accurate than five-class results?

One reason is we are short of validation samples of crop ground truth. CDL only provides a reference which is not one hundred percent correct. Even only a few wrong samples involved in the training will lead to the errors that are augmented and propagated during the propagation. In other

words, neural network tolerance cannot fully offset the consequences of original errors in the samples. If the network picks the wrong answer based on error samples, it will always prefer the wrong answer in the subsequent learning. Thus, training samples should be as accurate as possible at all costs, and the training samples of five-class are more accurate (either corn or soybean are crop), which increases all-class results' accuracy.

In addition, the Landsat bands limit the distinguishing capability they provide. For example, corn has long and narrow leaves, and soybean has round and wide leaves. Corn leaves are not as dense as soybean. However, in its growing season (245=Sep 3), the two crops have very identical spectral features. We select 199,459 pixels of corn, soybeans, grass, and water from the Landsat 2015245 image. Their band values are plotted in Fig. 12 which shows the curves of corn and soybean are overlapped and hard to distinguish. Increasing the dimensions of the band space with more bands or data sources may help enlarge the differences.

Figure 12. Spectral characteristics of soybean, corn, water and grass (based on 199,459 samples of 3 Sep 2015). The x-axis is the band number and the y-axis is the reflectance value. The markers on data points are random letters, which can help highlight the macro band value differences between crops.

4.3 Broad benefits

The traditional methods like SVM, Random Forest, and CART decision tree are non-parametric models. It means their complexity increases along with the size of the training set. It is very expensive to train them with big data. Meanwhile, the trained models can hardly be applied to other strange images. The old models are disposable products which are huge wastes of producers' and users' time and efforts. On the contrary, the proposed LSTM RNN model can be used across images. It can remember the trained samples and generalize their features into a universal pattern for the crops, which is adaptable to strange images. RS experts can feed the model with a huge number of samples from different images. The memory mechanism of LSTM RNN will help distinguish plenty of representation features to improve the classification results. The flexible

training and recycling usage will make the proposed model a very meaningful tool in the current landscape of RS community and show a promising future on fully automatic image classification (Sun et al. 2016; Sun et al. 2014). However, as the crop patterns in growing season vary in different places according to the various weather and environment conditions like soil salinity and moisture, it is still a long way to fully train the LSTM RNN to become universally applicable on all the crops over the U.S. Thus, at present this study only deals with North Dakota, and it is not recommended to directly use the trained network in another state. One solution is to reuse the trained LSTM RNN by fine-tuning it on new Landsat scenes. For example, the network trained in this study is especially for the P31R27 cell (North Dakota) in Landsat WRS2 grid, and if some people need use it for the P32R31 cell (Nebraska), they can directly train the old network with the Landsat images of P32R31, which is supposed to sum the memory of the two grids and gain capabilities to recognize more crop patterns. Continuously training LSTM RNN with the Landsat images of all the other agricultural states is an essential part of our next step of work.

5. Conclusions

This paper builds an LSTM RNN model to utilize CDL time series and ground-measured data to classify Landsat images. The model aims to generate CDL-like products with higher accuracy in an easier manner. We pre-processed Landsat, CDL time series, and ground truth, and used the validated samples in training. The network is trained thoroughly with the state-of-the-art techniques from the DL community. We tested the trained model on multiple Landsat scenes to produce five-class and all-class crop maps. The model results are plotted on charts and compared with CDL and ground truth. The evaluation results show a very satisfactory overall accuracy (>97% for five-class and >88% for all-class) and validate the feasibility of the model in the land cover classification of image time series.

In the future, we will compare LSTM RNN with CNN and RCNN to determine their application range. Landsat 5 and 7 TM/ETM+ bands will also be involved to mend the lack of CDL in some states before 2008. We will further study how other optimizers could improve the

converging speed and prediction accuracy while avoiding overfitting and divergence, and experiment with deeper and wider networks for better network capability. In addition, continuously training LSTM RNN with the Landsat images of larger scale (e.g. the entire U.S.) will be an important part of our next step work.

Acknowledgments: We sincerely thank the authors and copyright holders of all the datasets and software used to accomplish this research, and the editor and anonymous reviewers for their careful reading of our manuscript and thoughtful comments. This study was partially supported in part by NSF EarthCube under grants #1740693.

References

- Audebert, Nicolas, Alexandre Boulch, Hicham Randrianarivo, Bertrand Le Saux, Marin Ferecatu, Sébastien Lefevre, and Renaud Marlet. 2017. Deep Learning for Urban Remote Sensing. Paper presented at the Joint Urban Remote Sensing Event (JURSE).
- Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. 2017. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." *IEEE transactions on pattern analysis and machine intelligence* 39 (12):2481-95.
- Blaschke, Thomas, Geoffrey J. Hay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, et al. 2014. "Geographic Object-Based Image Analysis – Towards a new paradigm." *ISPRS Journal of Photogrammetry and Remote Sensing* 87:180-91. doi: <http://dx.doi.org/10.1016/j.isprsjprs.2013.09.014>.
- Boryan, Claire, Zhengwei Yang, Rick Mueller, and Mike Craig. 2011. "Monitoring US agriculture: the US department of agriculture, national agricultural statistics service, cropland data layer program." *Geocarto International* 26 (5):341-58.
- Breuel, Thomas M. 2015. "Benchmarking of LSTM networks." *arXiv preprint arXiv:1508.02774*.
- Burnett, Michael, Beth Weinstein, and Andrew Mitchell. 2007. ECHO—enabling interoperability with NASA earth science data and services. Paper presented at the Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International.
- Canty, Morton J. 2006. *Image analysis, classification and change detection in remote sensing: with algorithms for ENVI/IDL*. CRC Press.
- Computing, Hyspeed. "Big Data and Remote Sensing – Where does all this imagery fit into the picture?", Accessed 2016.5.16. <https://hyspeedblog.wordpress.com/2013/03/22/big-data-and-remote-sensing-where-does-all-this-imagery-fit-into-the-picture/>.
- Congalton, Russell G. 1991. "A review of assessing the accuracy of classifications of remotely sensed data." *Remote Sensing of Environment* 37 (1):35-46. doi: [http://dx.doi.org/10.1016/0034-4257\(91\)90048-B](http://dx.doi.org/10.1016/0034-4257(91)90048-B).
- Coon, Randal C, and F Larry Leistritz. 2010. "The role of agriculture in the North Dakota economy."
- Cooner, Austin J, Yang Shao, and James B Campbell. 2016. "Detection of Urban Damage Using Remote Sensing and Machine Learning Algorithms: Revisiting the 2010 Haiti Earthquake." *Remote Sensing* 8 (10):868.
- Dahl, George E, Tara N Sainath, and Geoffrey E Hinton. 2013. Improving deep neural networks for LVCSR using rectified linear units and dropout. Paper presented at the Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on.
- Das, Monidipa, and Soumya K Ghosh. 2016. "Deep-STEP: A Deep Learning Approach for Spatiotemporal Prediction of Remote Sensing Data." *IEEE Geoscience and Remote Sensing Letters* 13 (12):1984-8.
- Duro, Dennis C., Steven E. Franklin, and Monique G. Dubé. 2012. "A comparison of pixel-based and object-based image analysis with selected machine learning algorithms for the classification of agricultural landscapes using SPOT-5 HRG imagery." *Remote Sensing of Environment* 118:259-72. doi: <http://dx.doi.org/10.1016/j.rse.2011.11.020>.
- Gers, Felix A, and Jürgen Schmidhuber. 2000. Recurrent nets that time and count. Paper presented at the ijcn.
- Glorot, Xavier, and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. Paper presented at the Aistats.
- Graves, Alex, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. Paper presented at the Acoustics, speech and signal processing (icassp), 2013 ieee international conference on.

- Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. "Long short-term memory." *Neural computation* 9 (8):1735-80.
- Hussain, Masroor, Dongmei Chen, Angela Cheng, Hui Wei, and David Stanley. 2013. "Change detection from remotely sensed images: From pixel-based to object-based approaches." *ISPRS Journal of Photogrammetry and Remote Sensing* 80:91-106. doi: <http://dx.doi.org/10.1016/j.isprsjprs.2013.03.006>.
- Ienco, Dino, Raffaele Gaetano, Claire Dupaquier, and Pierre Maurel. 2017. "Land Cover Classification via Multi-temporal Spatial Data by Recurrent Neural Networks." *arXiv preprint arXiv:1704.04055*.
- Irons, James R, John L Dwyer, and Julia A Barsi. 2012. "The next Landsat satellite: The Landsat data continuity mission." *Remote Sensing of Environment* 122:11-21.
- Jantzi, Darin, Kara Hagemeister, Brenda Krupich, Kenneth F. Grafton, Cris Boerboom, Doug Goehring, Sonny Perdue, and Hubert Hamer. 2017. "North Dakota Agricultural Statistics 2017." In. Ag Statistics: USDA NASS.
- Jozefowicz, Rafal, Wojciech Zaremba, and Ilya Sutskever. 2015. An empirical exploration of recurrent network architectures. Paper presented at the International Conference on Machine Learning.
- Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. Paper presented at the Advances in neural information processing systems.
- Kussul, Nataliia, Mykola Lavreniuk, Sergii Skakun, and Andrii Shelestov. 2017. "Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data." *IEEE Geoscience and Remote Sensing Letters*.
- LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. "Deep learning." *Nature* 521 (7553):436-44.
- Li, Songnian, Suzana Dragicevic, Francesc Antón Castro, Monika Sester, Stephan Winter, Arzu Coltekin, Christopher Pettit, et al. 2016. "Geospatial big data handling theory and methods: A review and research challenges." *ISPRS Journal of Photogrammetry and Remote Sensing* 115:119-33. doi: <http://dx.doi.org/10.1016/j.isprsjprs.2015.10.012>.
- Li, Weijia, Haohuan Fu, Le Yu, and Arthur Cracknell. 2016. "Deep Learning Based Oil Palm Tree Detection and Counting for High-Resolution Remote Sensing Images." *Remote Sensing* 9 (1):22.
- Long, Jonathan, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.
- Ma, Yan, Haiping Wu, Lizhe Wang, Bormin Huang, Rajiv Ranjan, Albert Zomaya, and Wei Jie. 2015. "Remote sensing big data computing: Challenges and opportunities." *Future Generation Computer Systems* 51:47-60. doi: <http://dx.doi.org/10.1016/j.future.2014.10.029>.
- Maggiori, Emmanuel, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez. 2017. "Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification." *IEEE Transactions on Geoscience and Remote Sensing* 55 (2):645-57.
- Marmanis, Dimitrios, Mihai Datcu, Thomas Esch, and Uwe Stilla. 2016. "Deep learning earth observation classification using ImageNet pretrained networks." *IEEE Geoscience and Remote Sensing Letters* 13 (1):105-9.
- Merity, Stephen, Nitish Shirish Keskar, and Richard Socher. 2017. "Regularizing and optimizing LSTM language models." *arXiv preprint arXiv:1708.02182*.
- Nair, Vinod, and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. Paper presented at the Proceedings of the 27th international conference on machine learning (ICML-10).
- NASA. "An Overview of EOSDIS." Accessed 2016.5.16. <https://earthdata.nasa.gov/about>.
- Roy, David P, MA Wulder, TR Loveland, CE Woodcock, RG Allen, MC Anderson, D Helder, JR Irons, DM Johnson, and R Kennedy. 2014. "Landsat-8: Science and product vision for terrestrial global change research." *Remote Sensing of Environment* 145:154-72.
- Schaffer, Cullen. 1993. "Overfitting avoidance as bias." *Machine learning* 10 (2):153-78.
- Schuster, Mike, and Kuldeep K Paliwal. 1997. "Bidirectional recurrent neural networks." *IEEE Transactions on Signal Processing* 45 (11):2673-81.
- Senior, Andrew, Georg Heigold, and Ke Yang. 2013. An empirical study of learning rates in deep neural networks for speech recognition. Paper presented at the Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on.
- Srivastava, Nitish, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research* 15 (1):1929-58.
- Sun, Z., C. Peng, M. Deng, A. Chen, P. Yue, H. Fang, and L. Di. 2014. "Automation of Customized and Near-Real-Time Vegetation Condition Index Generation Through Cyberinfrastructure-Based Geoprocessing Workflows." *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of* 7 (11):4512-22. doi: 10.1109/jstars.2014.2377248.
- Sun, Ziheng, Hui Fang, Meixia Deng, Aijun Chen, Peng Yue, and Liping Di. 2015. "Regular Shape Similarity Index: A Novel Index for Accurate Extraction of Regular Objects from Remote Sensing Images." *Geoscience and Remote Sensing, IEEE Transactions on* 53 (7):3737-48. doi: 10.1109/TGRS.2014.2382566.
- Sun, Ziheng, Hui Fang, Liping Di, and Peng Yue. 2016. "Realizing parameterless automatic classification of remote sensing imagery using ontology engineering and cyberinfrastructure techniques." *Computers & Geosciences* 94:56-67.
- USGS. 2018. "Landsat Project Statistics." Accessed 2018.7.3. http://landsat.usgs.gov/Landsat_Project_Statistics.php.

- Ward, K. 2008. NASA Earth Observations (NEO): Data imagery for education and visualization. Paper presented at the AGU Fall Meeting Abstracts.
- Wikipedia. "Landsat Program." Accessed 2014.9.21. http://en.wikipedia.org/wiki/Landsat_program.
- Zhao, Wenzhi, Zhou Guo, Jun Yue, Xiuyuan Zhang, and Liqun Luo. 2015. "On combining multiscale deep learning features for the classification of hyperspectral remote sensing imagery." *International Journal of Remote Sensing* 36 (13):3368-79.
- Zibulevsky, Michael, and Michael Elad. 2010. "L1-L2 optimization in signal and image processing." *IEEE Signal Processing Magazine* 27 (3):76-88.

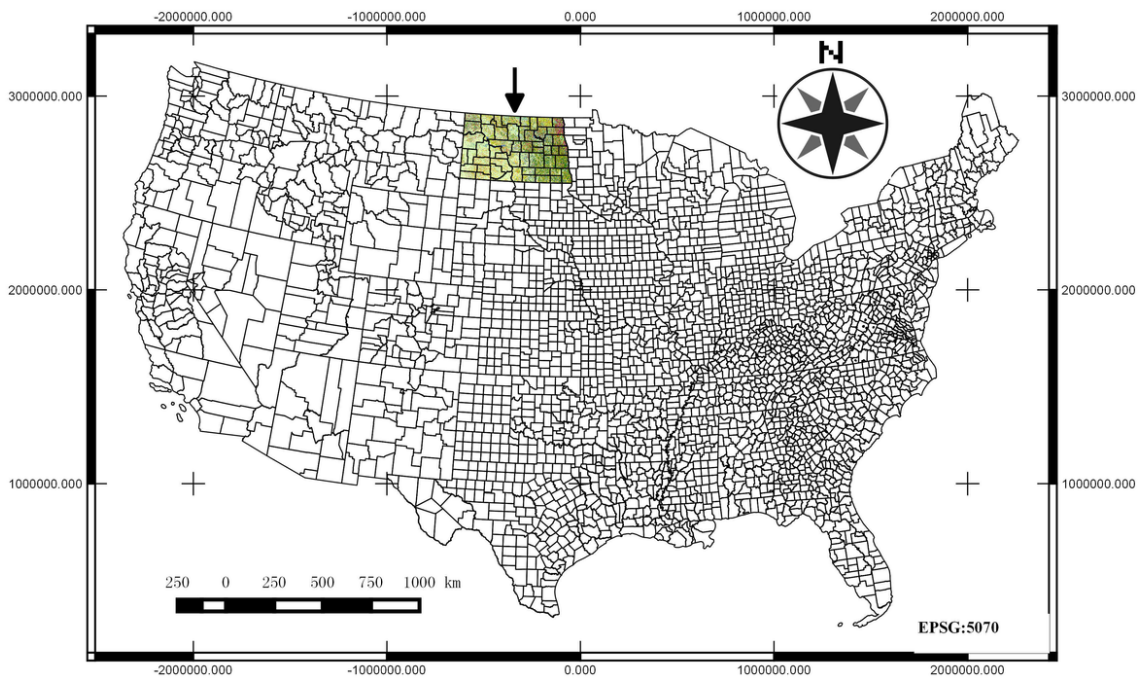


Figure 1. Study area. The black lines in the map are U.S. county boundaries. The projection is NAD83/Conus Albers (EPSG code: 5070).

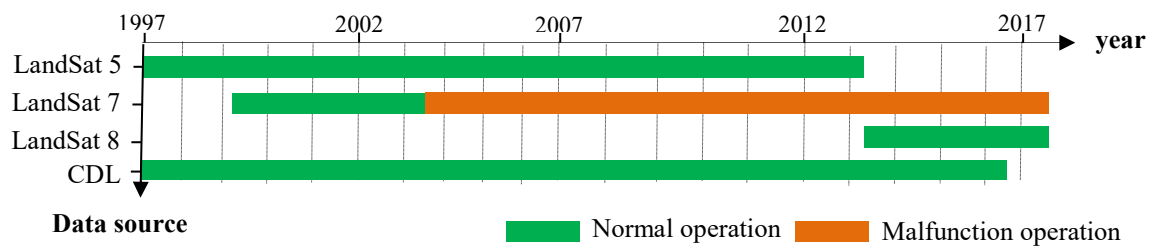


Figure 2. The availability of data for North Dakota since 1997

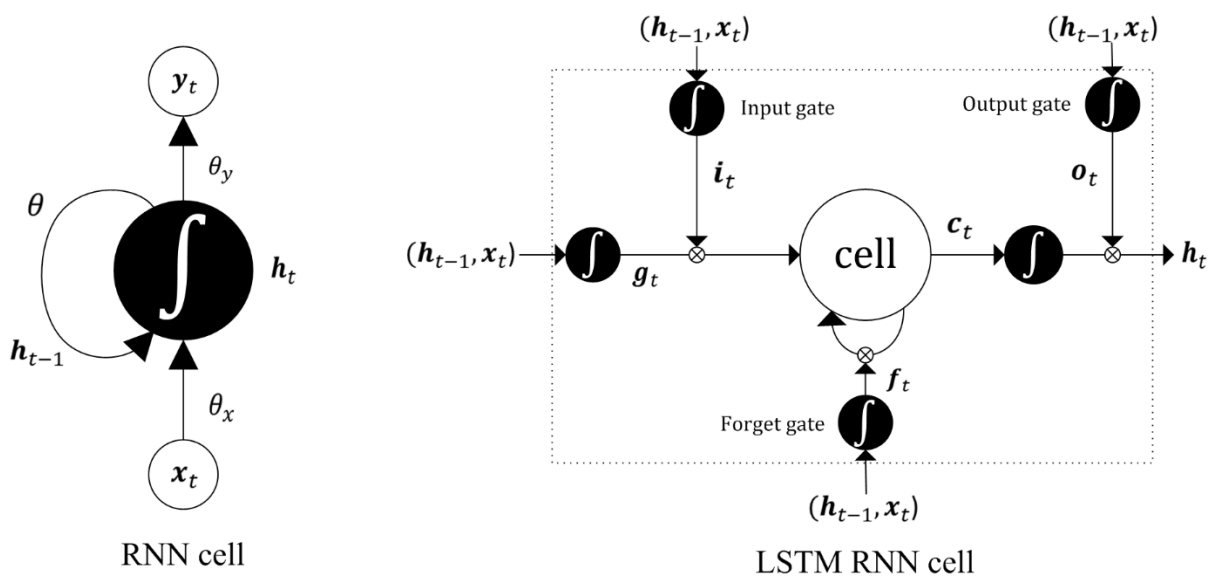


Figure 3. RNN and LSTM introduction

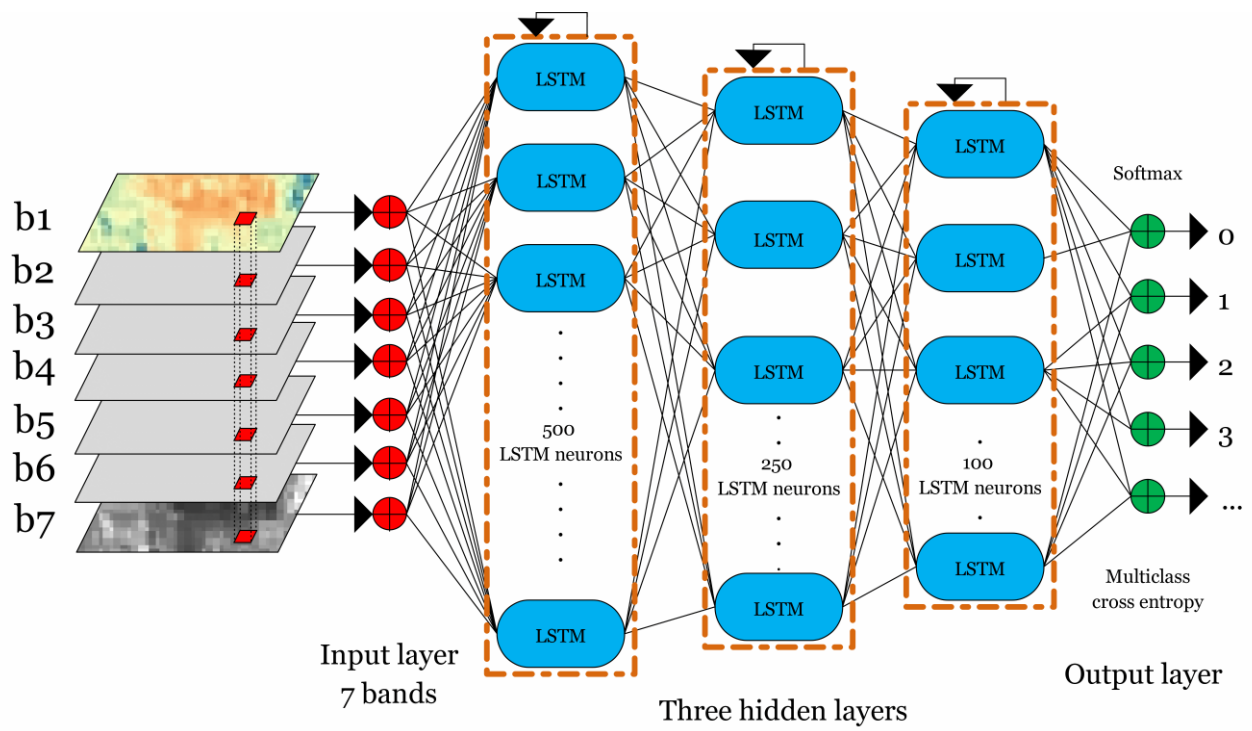


Figure 4. The proposed model (per pixel)

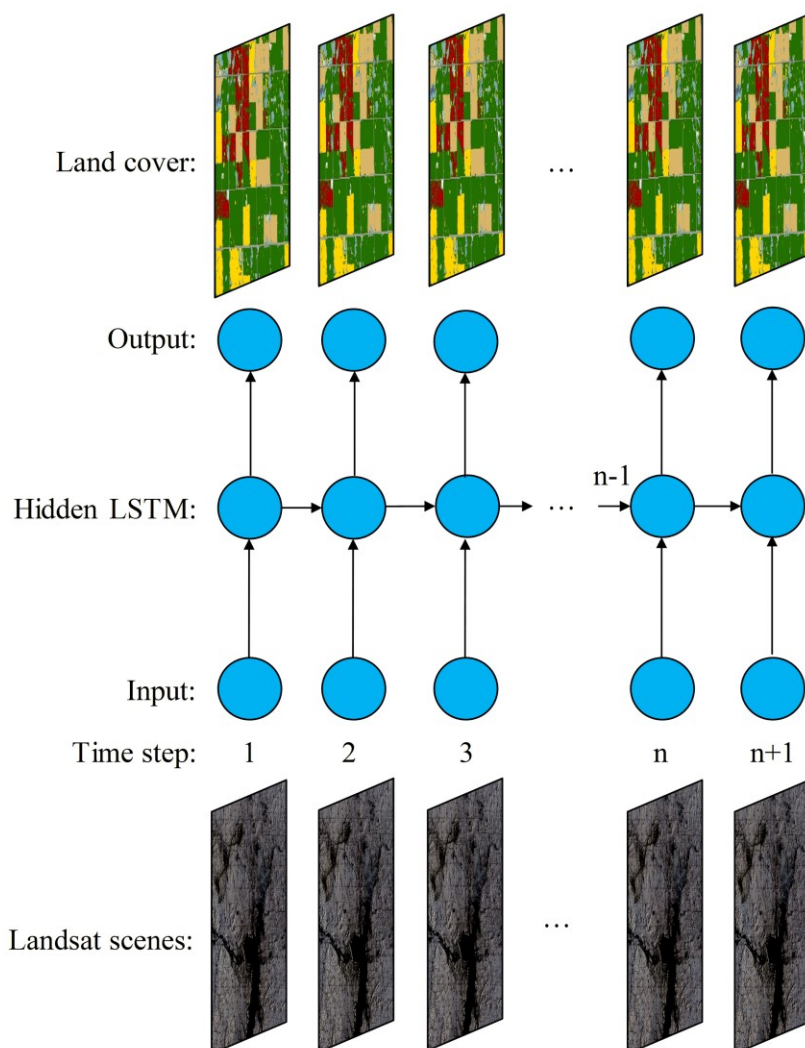


Figure 5. Many-to-many schema to learn the temporal pattern of crop growth

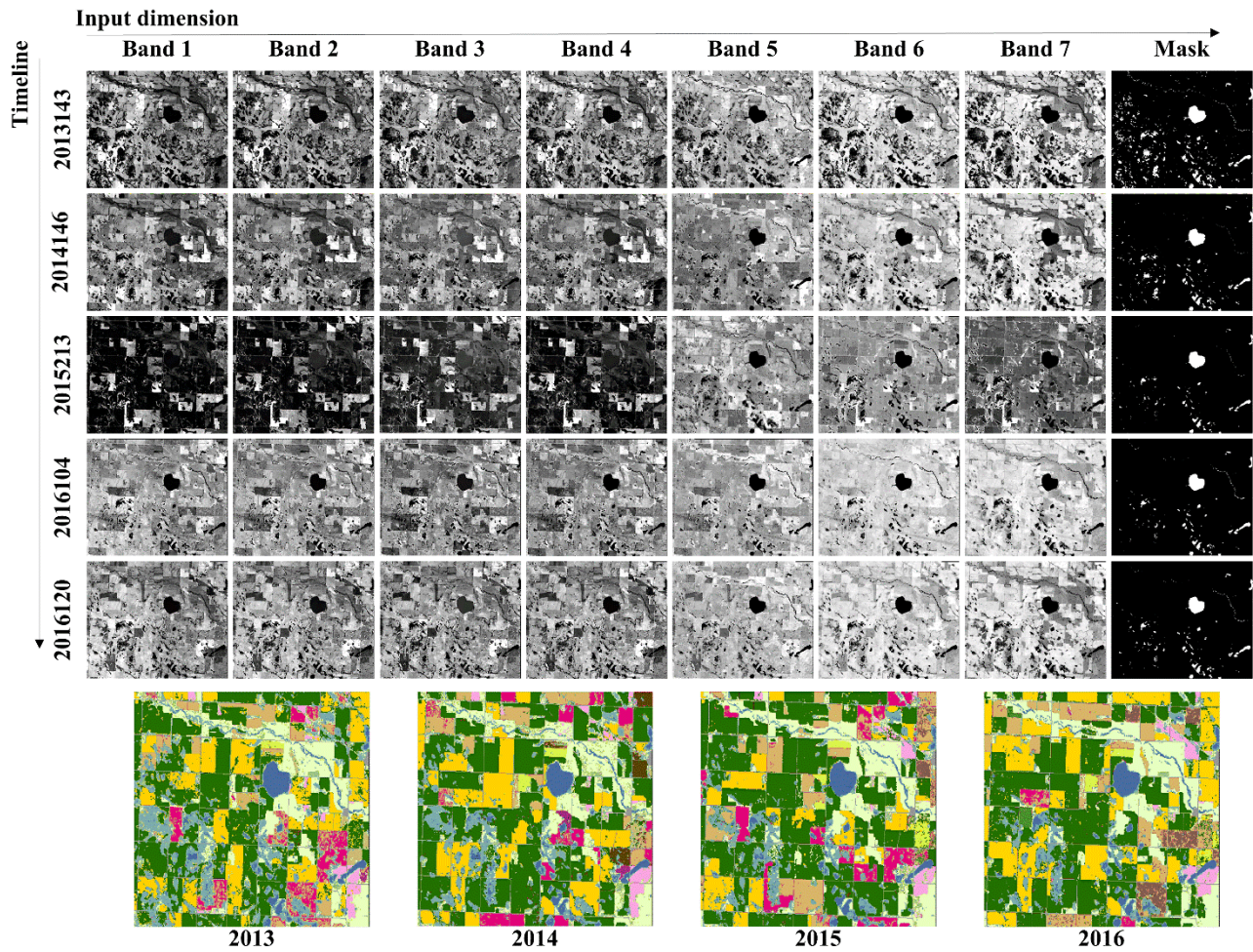


Figure 6. The data series of Landsat and CDL for training. The x-axis is Landsat band and y-axis is timeline of Julian dates (e.g., 2013143 means 23 May 2013). The false colour composites are grey-scaled band images. The four classified maps at the bottom are CDLs from 2013 to 2016. The legend for CDL is on NASS website.

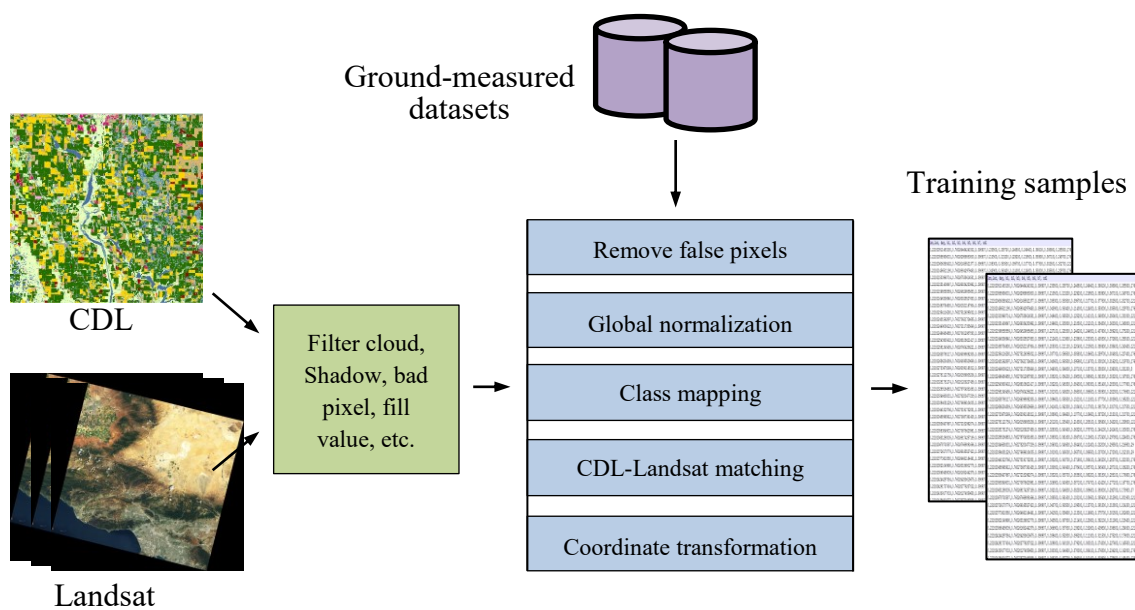


Figure 7. Pre-processing workflow

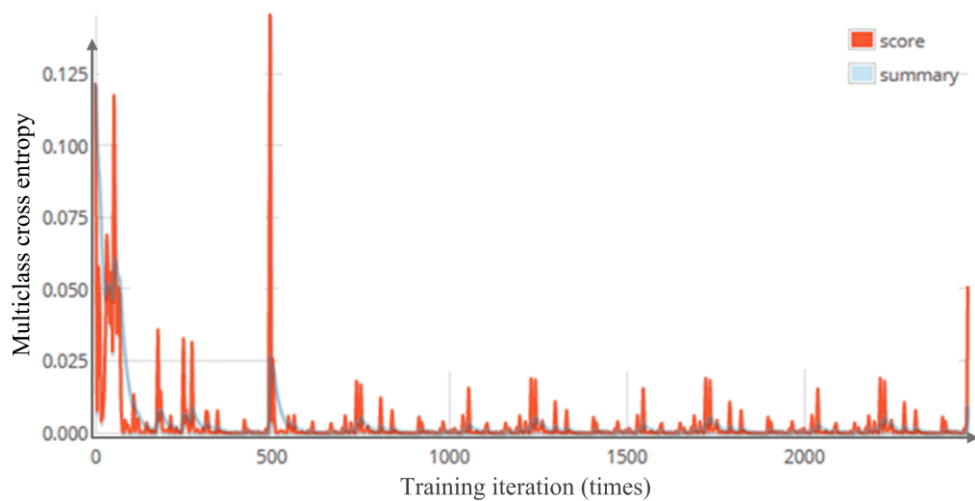


Figure 8. Training 1,474,034 samples in five epochs.

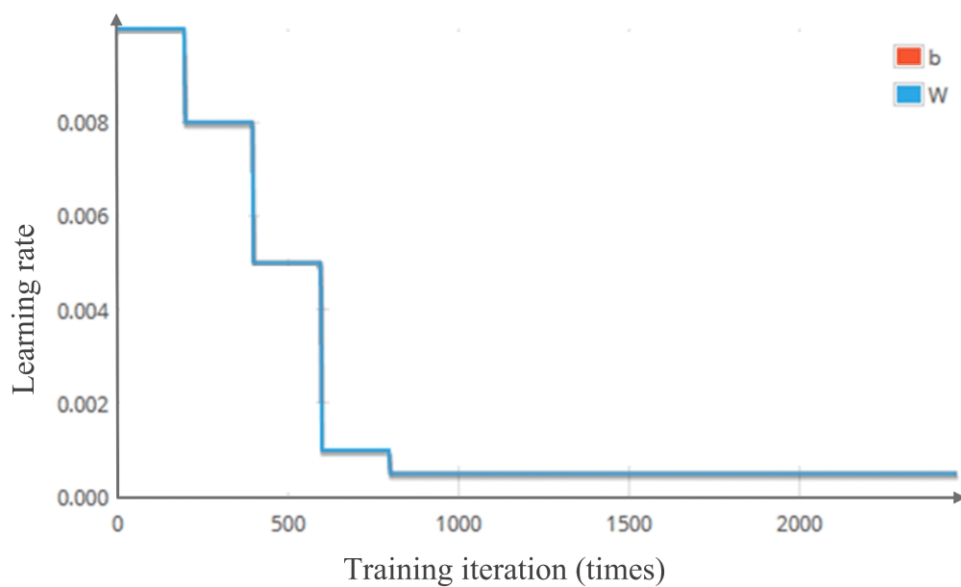
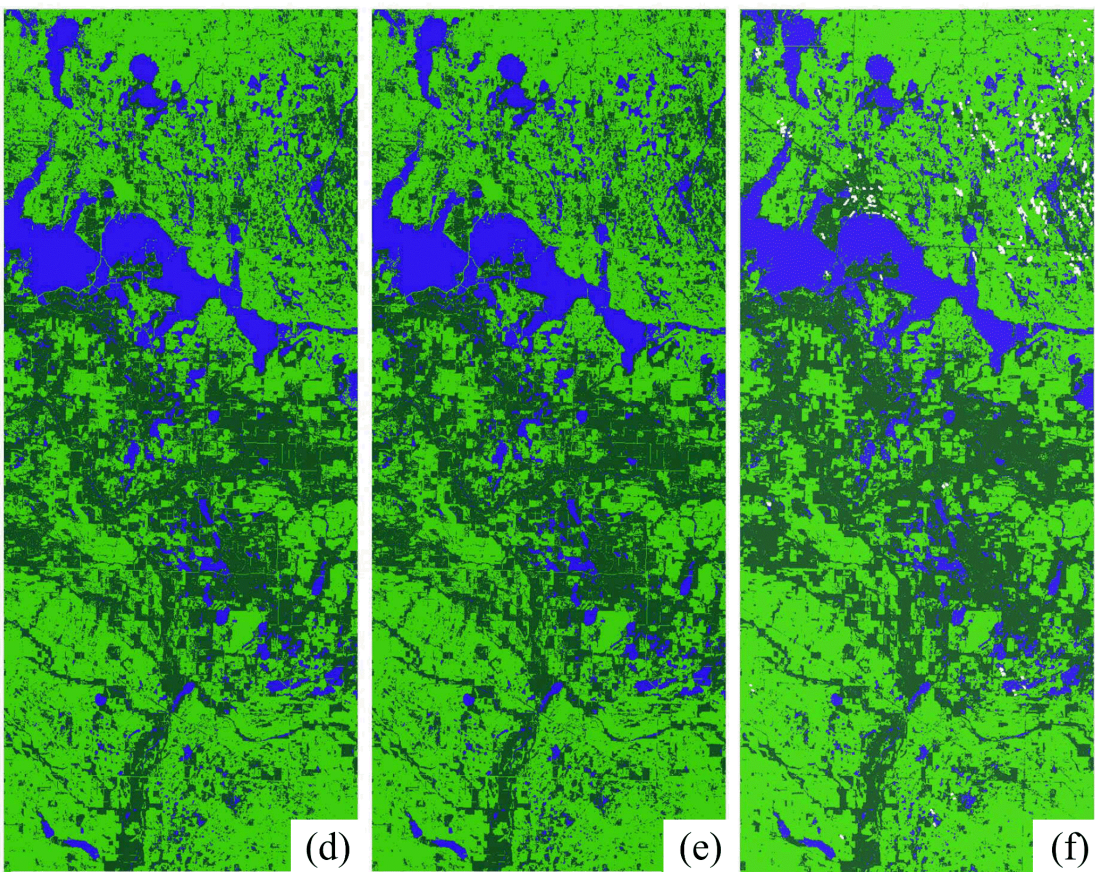
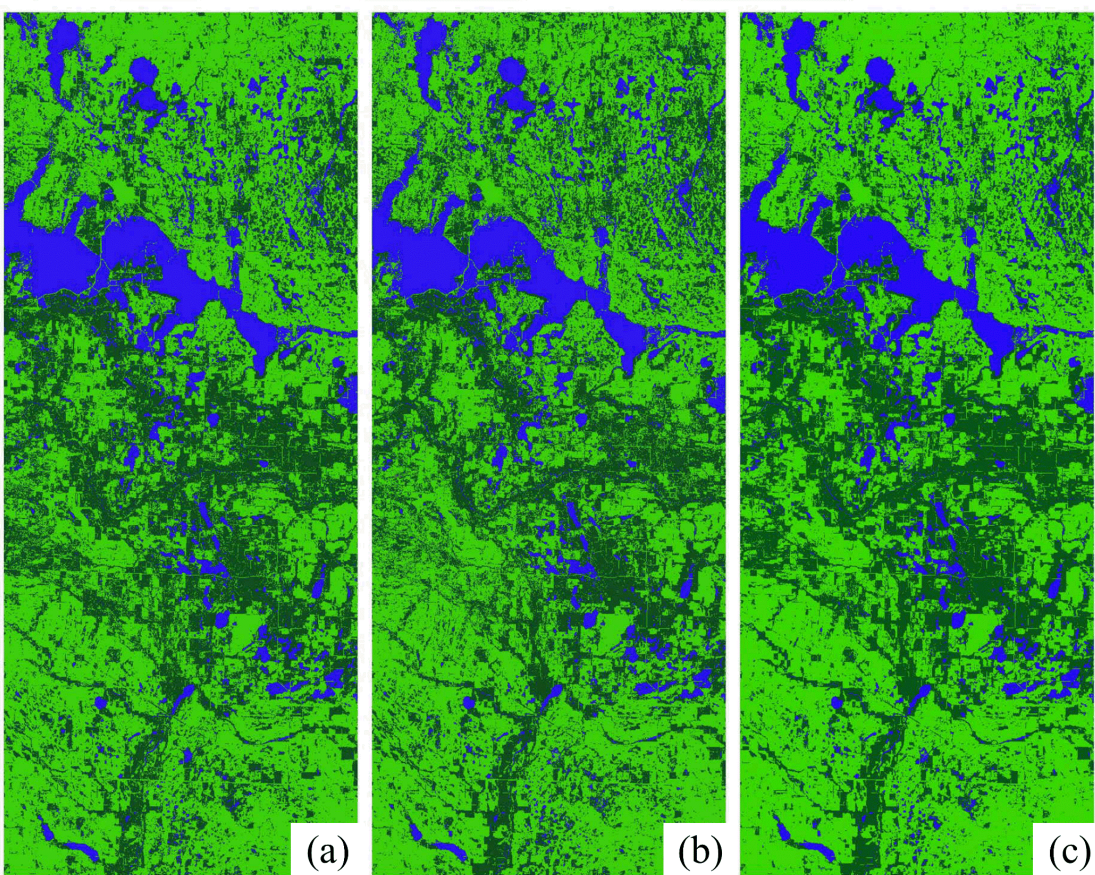


Figure 9. The scheduled drop of learning rate

20 0 20 40 60 80 km



 Crop land  Non-crop land  Water



Figure 10. The three-classes test results and CDL. (a) CDL of 2014; (b) CDL of 2015; (c) CDL of 2016; (d) test result on 26 May 2014; (e) test result on 1 Aug 2015; (f) test result on 14 Apr 2016.



Figure 11. Landsat true-colour composites, CDL, and the all-class results of LSTM RNN and SegNet. The x axis is product category, and the y axis is product observation date. The classification legend is at the bottom.

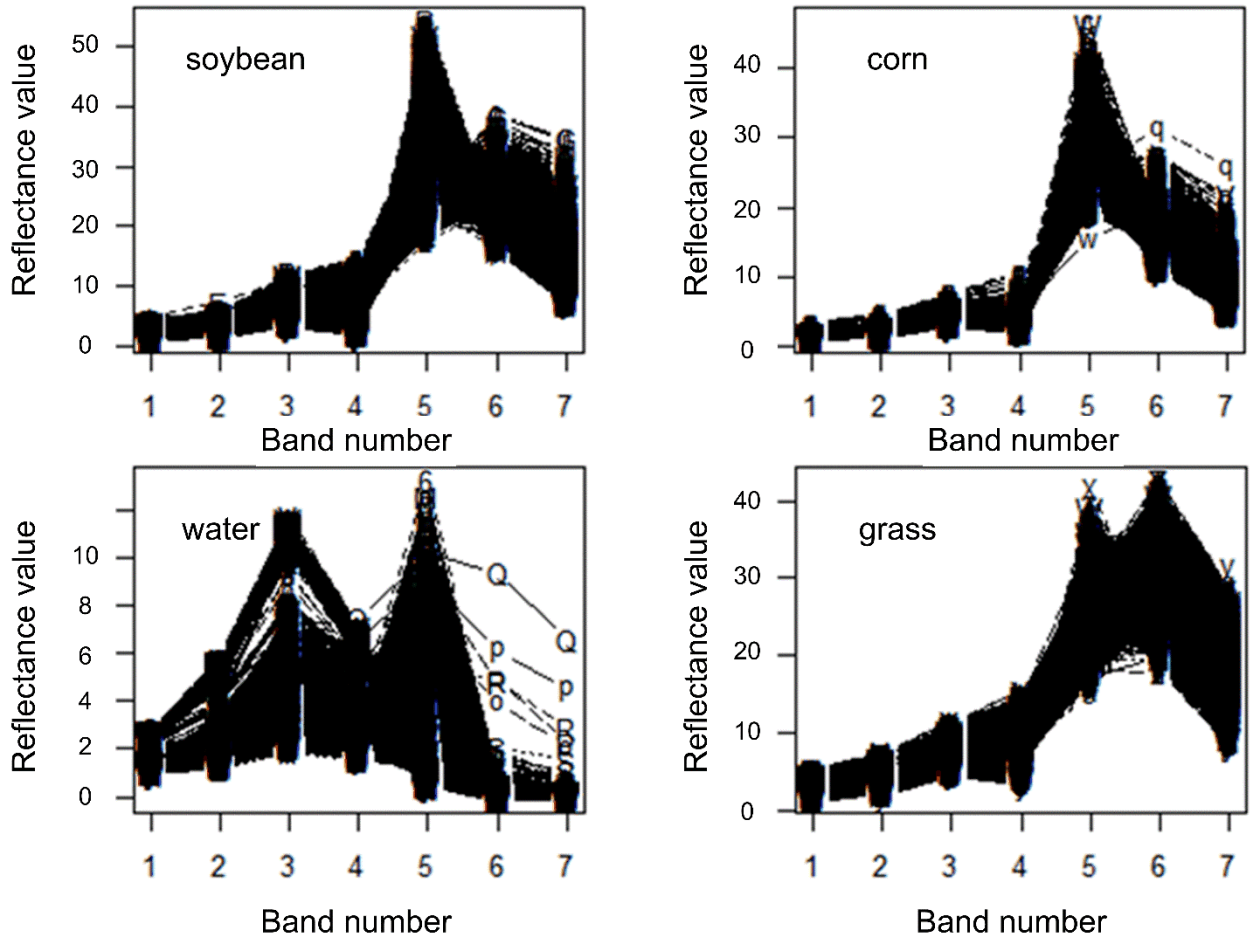


Figure 12. Spectral characteristics of soybean, corn, water and grass (based on 199,459 samples of 3 Sep 2015). The x-axis is band number and y-axis is the reflectance value. The markers on data points are random letters, which can help highlight the macro band value differences of crops.

Table 1. The employed bands of Landsat 5, 7 and 8 (WL: wavelength)

Landsat 5 TM			Landsat 7 ETM+			Landsat 8 OLI		
No.	Name	WL(μm)	No.	Name	WL(μm)	No.	Name	WL(μm)
1	Blue	0.45-0.52	1	Blue	0.45-0.52	1	Ultra blue	0.43-0.45
2	Green	0.52-0.60	2	Green	0.52-0.60	2	Blue	0.45-0.51
3	Red	0.63-0.69	3	Red	0.63-0.69	3	Green	0.53-0.59
4	NIR	0.76-0.90	4	NIR	0.77-0.90	4	Red	0.64-0.67
5	SWIR1	1.55-1.75	5	SWIR1	1.55-1.75	5	NIR	0.85-0.88
7	SWIR2	2.08-2.35	7	SWIR2	2.09-2.35	6	SWIR 1	1.57-1.65
						7	SWIR 2	2.11-2.29

Table 2. Metrics of the five-class results (Avg is short for average)

Test dataset	Pixel amount	Accuracy	UA (Avg)	PA (Avg)	Kappa
S1	736,558	0.9921	0.9649	0.9874	0.9917
S2	737,017	0.9909	0.9645	0.9767	0.9904
S3	1,358,173	0.9878	0.9615	0.9540	0.9872
S4	1,595,654	0.9845	0.9518	0.9389	0.9837
S5	1,616,563	0.9822	0.9459	0.9356	0.9812
S6	1,752,372	0.9733	0.9083	0.9294	0.9717

Table 3. Metrics of the all-class results (R: LSTM RNN; S: SegNet)

Test dataset	Accuracy	UA (Avg)	PA (Avg)	Kappa
2 Sep 2015 (R)	0.9135	0.8229	0.8615	0.9111
2 Sep 2015 (S)	0.8714	0.7377	0.8236	0.8643
1 Aug 2015 (R)	0.8885	0.8055	0.8065	0.8854
1 Aug 2015 (S)	0.8608	0.7720	0.8133	0.8509
15 Sep 2014 (R)	0.8922	0.8030	0.8142	0.8891
15 Sep 2014 (S)	0.8875	0.7973	0.8042	0.8847
30 Aug 2014 (R)	0.8953	0.8058	0.8096	0.8924
30 Aug 2014 (S)	0.8502	0.6527	0.7664	0.8386