CONVECTIVE INITIATION FORECASTS THROUGH THE USE OF MACHINE LEARNING METHODS*

Mark S. Veillette, Haig Iskenderian
Patrick M. Lamey, Laura J. Bickmeier
Massachusetts Institute of Technology, Lincoln Laboratory
Lexington, MA 02420

1. Introduction

Detection and forecasting of convective initiation (CI) is an important problem, not only for aviation related purposes, but across all forms of business and general recreation. For aviation in particular, hazards related to thunderstorms, such as lightning, hail, strong winds and wind shear, are very costly to airlines through delays and wasted fuel in the events of holding or diversions. Current nowcasting systems, such as MIT Lincoln Laboratory's Corridor Integrated Weather System (CIWS) (Wolfson et al. 2006) rely heavily on the tracking and trending of existing radar signatures of precipitation intensity and storm heights to produce forecasts, and as a result, often struggle with predicting new storm growth. Improving this capability is the subject of this work.

The use of visible and infrared satellite observations to assist in the nowcasting of CI has received a great deal of attention in the past decade. Specifically, two Geostationary Operational Environmental Satellites (GOES), GOES-13 and GOES-15, provide real time capabilities for monitoring the growth of small cumulus as they grow to the cumulonimbus scale, and as a result, are useful for identifying clouds that will likely develop into hazardous storms before the presence of radar signatures. The Satellite Convection Analysis and Tracking (SATCAST) system was developed to use visible and infrared observation to predict CI in real-time (Mecikalski et al. 2006). SATCAST calculates eight satellite-based CI indicators at each satellite pixel. Each indicator is assigned a value of 1 if it is within a pre-defined threshold, and a value of 0 if it is not. Each pixel is then assigned a 0 to 8 "score" based on the number of CI indicators that fall within the given threshold, with higher scores indicating higher confidence in CI. For more information on the use of GOES imagery and SATCAST, see Section 2.2.

In addition to satellite observations, environmental parameters, e.g. Convective Available Potential Energy (CAPE), can also help in identifying areas where the atmosphere is unstable and prone to formation of new convection. In 2011, a CI capability was added to the CIWS 0-2 hour forecast that uses SATCAST in combination with environmental, numerical weather prediction model output winds, temperature, and moisture, and additional satellite data to incorporate CI into the CIWS deterministic forecast (Iskenderian et al. 2010). This algorithm uses a Fuzzy Logic based algorithm that combines the 0 to 8 score of SATCAST indicators together with environmental factors to produce an automated 0-2 hour nowcasting capability for CI. While this algorithm provides additional forecast lead for a number of initiation events, it has some known drawbacks. Namely, the Fuzzy Logic requires extensive hand tuning which can be time consuming and expensive. The Fuzzy Logic system has also been observed

* This work was sponsored by the Federal Aviation Administration under Air Force Contract No. FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government. This research is in response to requirements and funding by the Federal Aviation Administration (FAA). The views expressed are those of the authors and do not necessarily represent the official policy or position of the FAA.

to be overly conservative in instances where initiation of storms occurs away from existing convection. The algorithm also does not use a wealth of information that is potentially contained in the full distribution of the SATCAST indicators. Finally, the algorithm is somewhat inflexible to new or additional data, as each new predictor of CI will require a new set of Fuzzy Logic parameters to be carefully tuned for optimization.

In this study, we develop a machine learning approach for nowcasting CI with the goal of improving the CIWS system. This algorithm generates fine scale (1 km) forecasts of Vertically Integrated Liquid (VIL) for new storm growth up to 1 hour lead times. Similar to the Fuzzy Logic algorithm, this uses input from GOES and information about environmental stability; however the machine learning framework will, in theory, make optimal use of all available predictors and enable the model to easily incorporate new predictors as they become available. In particular, this approach allows for numerical model output to be added to the list of predictors to improve the detection of CI. Section 2 describes the current set of predictors in more detail.

This is not the first attempt to use machine learning methods to forecast CI. For example, Random Forests have been utilized to identify regions where CI is likely based on a large number of predictor fields (Williams, et. al. 2008). However, the approach described here is different and unique in several ways. First, the forecast generated using the methodology described here is deterministic (explicit forecasts of new storm locations) as opposed to creating broader probabilistic regions of new storms. Additionally, the training methodology is modified, both in the way training data is obtained, and the machine learning methods which are used. Finally, the number of inputs to the models used here is kept relatively small, since this model is designed to be run operationally, and hence it is desired to have as few inputs as necessary to ensure consistent and reliable forecasts.

This paper is organized as follows. In Section 2, we describe the set of inputs used in the machine learning model, and divide them into three categories: environmental input, satellite input, and numerical models. Next in Section 3, the machine learning methods being investigated are described. In Section 4, the training methodology is described, and Receiver Operating Characteristics (ROC) and indicator importance results are discussed. The performance of the machine learning models is discussed in Section 5, and future work is discussed in Section 6.

2. Input Data

In this section, the data used to predict CI are discussed in more detail.

2.1 Environmental

Lincoln Laboratory has created an environmental stability mask to help identify broad regions where CI is likely to occur (Iskenderian et al. 2010). This field is created from two sources of data: NOAA's Rapid Refresh (RAP) model provides upper-air temperature, moisture and winds, and NOAA's Variational Local Analysis and Prediction System (V-LAPS, formerly known as STMAS), (Xie et al. 2005) provides 5-km analyses of dry bulb temperature and dew point temperature from surface observations every 15 minutes. These two fields are used in CIWS to create a stability mask by blending the V-LAPS dry bulb temperature and dew point temperature with RAP data in approximately the lowest 50 hPa. This process updates the lowest altitudes of the RAP with surface observations to account for several hours of RAP forecast latency. The convective available potential energy (CAPE) and departure of the dry bulb temperature from the surface convective temperature are calculated from this blended data and combined to create the surface stability mask. Areas of high CAPE and small departure from dry bulb temperature from the convective temperature are favored for CI and are highlighted in this mask.

2.2 Satellite

Satellite data has been widely studied as an indicator of CI. Examination of GOES satellite data in Roberts and Rutledge (2003) in cases of CI has shown that the cumulous cloud top cooling in the 10.7 micron infrared (IR) brightness temperatures are a strong indicator. In addition, Mecikalski and Bedka (2006) combined the 10.7 micron cooling rate in addition to other IR brightness temperatures and band differences (including the 13.3 and 6.5 micron channels) in the SATCAST system to create real-time CI interest fields derived from GOES data. This system combines three components to create satellite based indicators: a cloud mask component (Berendes et al. 2008) to classify cloud types, a cloud tracking component (Velden et al. 1997) to derive cloud motion vectors, and a third component which combines the cloud type and tracking, and various IR brightness temperatures to create a set of CI interest fields.

In addition to satellite processing involved with SATCAST, the visible satellite image is smoothed with a 25 X 25 km Gaussian kernel and the smoothed image is subtracted from the original image. This results in an image known as the "peaky" interest field which highlights small features with high texture (i.e. reflectance peaks in the visible imagery) such as cumulus clouds, and de-emphasizes areas of low small scale texture, such as stratus and cirrus clouds (Iskenderian et al. 2009).

2.3 Numerical Model

Numerical model forecasts can be powerful predictors of CI since they combine numerous observations in a dynamical framework to make a weather forecast. These forecasts are incorporated into the 0-2 hour forecasts through the methods described below.

Two numerical model forecasts are currently used to assist in forecasting CI. The first model forecast considered is from the Localized Aviation Model Output Statistics (MOS) Program's (LAMP) thunderstorm probability forecast (Ghirardelli, 2005). This product provides a probability of one or more cloud to ground lightning strikes in a 2 hour period in a 20 km grid box. For this study, the most recently available 2-hour LAMP forecast, issued at least an hour prior to the issue time, is used for predicting CI. LAMP forecasts are generated using a multiple linear regression of several predictors of lightning, which include (but are not limited to) surface observations, Global Forecast System (GFS) model output and climatological variables.

In addition to LAMP, NCEP's time lagged North American Rapid Refresh Ensemble Forecast System (NARRE-TL) is another numerical model included as a predictor of CI (Zhou, 2011). The NARRE-TL provides a forecast of the probability of convective precipitation and is produced hourly. Unlike the LAMP, this product is computed from a 10-member time lagged ensemble that includes six RAP and four North American Mesoscale Forecast System (NAM) model forecasts.

3. Machine Learning Methods

The methods described in general terms below each take as input a vector containing a fixed number of "features" which have been extracted from the set of all input images available (satellite, environmental, and numerical model) and outputs a probability that CI will result in the next 0-2 hours. Three machine learning techniques were investigated, Logistic Regression, Artificial Neural Networks, and Decision Tree Ensembles[†]. The probability

[†] Support Vector machines (SVMs) were also investigated, however the length of training time required in our initial version was too long to make a real-time version of the algorithm tractable, and moreover, they did not show sufficient skill above the other methods listed to warrant further investigation or optimization.

output by these models is used in the 0-2 hour forecast to create a deterministic forecast. In the following, $\mathbf{x} = (x_1, \dots, x_n)$ denotes a vector of n numerical features derived from the set of input images. The details of the feature extraction are discussed in Section 4.

The implementations of these algorithms for testing use the C++ open source Numerical Analysis package ALGLIB (Bochkanov & Bystritsky, http://www.alglib.net/). This package was chosen because it is convenient, easy to use, and it provides solid implementations of the methods described below.

3.1 Logistic Regression

The simplest model considered here, Logistic Regression (LR), is a generalized linear model that estimates the probability of convective initiation using the logistic function evaluated at a linear combination of the input features,

$$p_{CI} = P(\mathbf{x}; \boldsymbol{\omega})$$

$$= \frac{1}{1 + \exp\left(-\omega_1 x_1 - \cdots - \omega_n x_n\right)}.$$

The weight vector $\boldsymbol{\omega} = (\omega_1, \dots, \omega_n)$ is chosen to minimize the cross-entropy error function over the training set, which is defined as $J(\boldsymbol{\omega}) = -[\sum y_i \log P(\mathbf{x}_i; \boldsymbol{\omega}) + (1 - y_i) \log\big(1 - P(\mathbf{x}_i; \boldsymbol{\omega})\big)]$, where the $\mathbf{x}_i$'s represent vectors in the training set and the y_i's represent the associated binary variables which are 1 if CI occurred, and 0 if it did not. LR can be viewed as a simplified version of an Artificial Neural Network with no hidden layers.

Logistic regression has the advantage of being simple and fast to train, which is desirable when running on a real time system. In addition, the minimization procedure detailed above has a global optimum which is always achieved without the need to define any stopping criteria, meaning no hyperparameter tuning is necessary. The downside of this algorithm are that it uses only a linear combination of the input features, and hence will not perform well if the CI outcome depends non-linearly on the set of input features.

3.2 Artificial Neural Networks

In order to account for the possibly non-linear relationships involved in CI processes, we also consider Artificial Neural Networks (ANN) for estimating CI probability. While there are many network architectures possible, the architecture used here is a single feed-forward network with one hidden layer, and two output neurons. The input layers uses a non-linear (sigmoid) transfer function, and the hidden layer uses a linear transfer function with a softmax normalization to ensure the outputs can be interpreted as probabilities. Mathematically, if H denotes the number of hidden neurons, then the ANN computes the probability of CI, p_{CI}, given a feature vector $\mathbf{x} = (x_1, \dots, x_n)$ as

$$z_i = \varphi\left(\sum_{k=1}^{n} \omega_{k,i}\, x_k\right), \qquad i = 1, \dots, H,$$

$$u_j = \sum_{i=1}^{H} \beta_{i,j}\, z_i, \quad j = 1,2,$$

$$p_{CI} = \frac{\exp\left(u_1\right)}{\exp(u_1) + \exp\left(u_2\right)}.$$

Here, $\varphi(z) = (1 + \exp(-z))^{-1}$ is the standard sigmoid function. The weight vectors $\boldsymbol{\omega}_i = (\omega_{1,i}, \dots, \omega_{n,i})$ for $i = 1, \dots, n$ and $\boldsymbol{\beta}_j = (\beta_{1,j}, \dots, \beta_{H,j})$ for $j = 1,2$ are chosen to minimize mean squared error (MSE) over the training inputs when this network is fit to the training output $y = 1$ if CI occurred, and $y = 0$ if it did not.

For training an ANN, a number of hyperparameters need to be set. These parameters include the number of hidden neurons H, as well as stopping criteria like the maximum number of iterations to perform during optimization, and a stopping tolerance for the change in MSE after each step. For this study, $H = 8$ the maximum number of iterations was 10,000, and tolerance on change in MSE was set to 0.01.

ANNs have the advantages that they can fit to highly non-linear functions and have been proven to be quite successful for many applications. However, due to the large number of network structures available and hyperparameter tuning, finding the optimal fit can be difficult when compared to simpler models such as LR. Moreover, noisy training data can lead to over fitting if the optimization step finds a local minimum in the objective function. It is important to keep these drawbacks in mind while training.

3.3 Decision Tree Ensembles

Ensemble based learning, where multiple weak classifiers (base learners) are combined to improve overall predictive performance, has been shown to be effective in several areas of supervised learning. Leo Breiman applied this idea using simple decision trees as base learners, and coined the term Random Forest to describe the resulting ensemble classifier[‡] (Breiman, 2001). In a decision tree ensemble, a number $N_{trees} \geq 1$ of decision trees are trained on subsets of the training data drawn at random from the original training set. Each tree uses only a subset of $N_{feat} \geq 1$ features out of the full list of features in the feature vector $\mathbf{x}$ at each node, and is typically grown to completion with no pruning. For a new input feature vector, each tree of the ensemble casts a vote (CI or no CI), and a final probability of CI is obtained from the proportion of positive CI votes over the ensemble.

Similar to ANNs, decision tree ensembles can fit non-linear functions. These algorithms are fast to train, and in contrast to ANNs, do not require stopping criteria parameters to be set. As far as shortcoming of decision tree ensembles, they have been observed to be prone to over-fitting (Segal, 2004), especially with noisy training data. The hyperparameters required to be set include N_{trees} and N_{feat} , as well as the proportion of the training set to use for growing each tree. The values used in this study for these parameters were $N_{trees} = 100$, $N_{feat} = 3$, and 60% of the training set for growing each tree.

One further benefit of the decision tree ensembles is that they provide a useful measure of predictor importance. While there are multiple techniques for measuring importance, the method used here computes the Gini Importance measure, which is a measure of "impurity loss" ensemble for each input feature over all nodes of the tress in the final ensemble. If a feature possesses a large Gini Importance, then nodes split using this feature did a better job distinguishing CI events.

4. Training and Implementation

This section describes the procedure for constructing the training set used to configure the models described in Section 3, namely, how to automatically extract the features from the available satellite, environmental and numerical model data, and how to classify these points as "CI" or "Not CI".

4.1 Identification of CI Regions

In order to train the machine learning models discussed in Section 3, we start by identifying regions in which CI is going to occur in historical data. Particularly, we seek regions which transition from containing no radar signature, to a VIL observation of Level 3 and above, within a time span of less than an hour. To find such regions, radar images of VIL close in time are time aligned using a backwards advection technique so that regions of CI become apparent. Let t_0 and $t_0 + \Delta t$ denote the time

[‡] The implementation of a decision tree ensemble done in ALGLIB differs slightly from Breiman and Cutler's original Random Forest implementation, but the overall steps described here remain the same. For a more specific description, the reader is referred to the documentation of ALGLIB found on their website.

stamps of two VIL images to be used for identification of CI regions. Motion vectors derived using the CIWS cross-correlation tracking module at time $t_0 + \Delta t$ are rotated 180° and the VIL valid at $t_0 + \Delta t$ is advected backward using these vectors over Δt minutes. At this stage, storms that existed at t_0 which were simply advected to time $t_0 + \Delta t$ will be approximately aligned in space, and can be discarded using a simple masking procedure. Backward advected pixels from time $t_0 + \Delta t$ which are VIP Level 3+ and do not lie near existing VIP Level 3+ are tagged as new storm growth. Regions which contain no VIL in the radar signature in either image, or which contain storms which existed in the time t_0 image are tagged as "Not CI". Typically, a time lag of $\Delta t = 45$ minutes is used, and backwards advected storms must be more than 45 km away from existing convection to be considered as new growth. See Figure 1 for a visual summary for this procedure.

4.2 Cloud Clustering and Feature Extraction

After identifying general regions where CI has and has not occurred, the next step is to look at the neighboring clouds from the visible satellite imagery, namely, cumulous clouds which may be in the early stages of storm development, but do not yet possess a radar signature, and gather data within these clouds in order to search for patterns of CI in the data. In order to do this, we start with the SATCAST Convective Cloud Mask, which classifies pixels of the visible satellite image into categories (Berendes et al. 2008). Since we are interested in CI, we only use pixels inside this mask which fall into the Small Cumulus, Developing Cumulus, or Towering Cumulus categories. Once other cloud types (Thin and Thick Cirrus) are removed, the remaining clouds are broken down into clusters using the Mean Shift image segmentation algorithm (Comaniciu & Meer, 2002) as shown in Figure 2(c), (d) and (e).

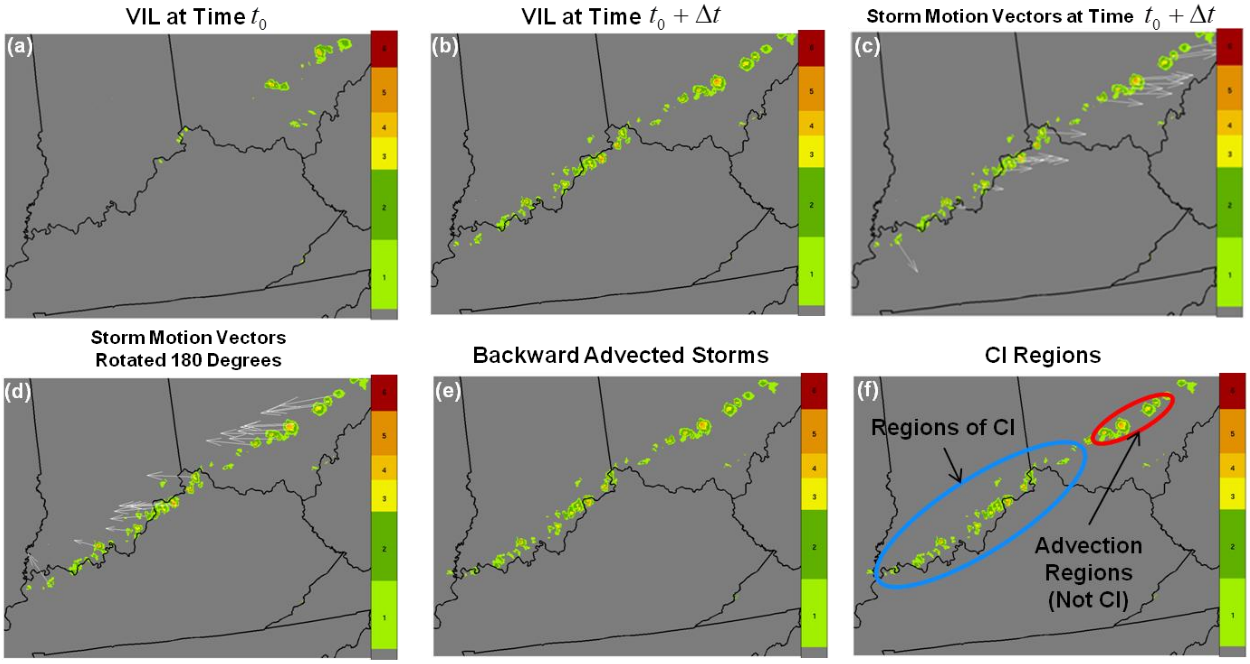
VIL at Time t_0
(a)
VIL at Time $t_0 + \Delta t$
(b)
Storm Motion Vectors at Time $t_0 + \Delta t$
(c)
Storm Motion Vectors Rotated 180 Degrees
(d)
Backward Advected Storms
(e)
CI Regions
(f)
Regions of CI
Advection Regions (Not CI)
6
5
4
3
2
1

Figure 1: Visualization of the method used for identifying regions of CI. *Consecutive VIL mosaics, (a) and (b), are time aligned by rotating storm motion vectors shown in (c) 180 degrees and using these for advection for Δt minutes, (d) and (e). The backwards advected image is compared to the earlier VIL image in (a) with a masking procedure to label regions as CI regions or Non-CI regions, shown in (e).*

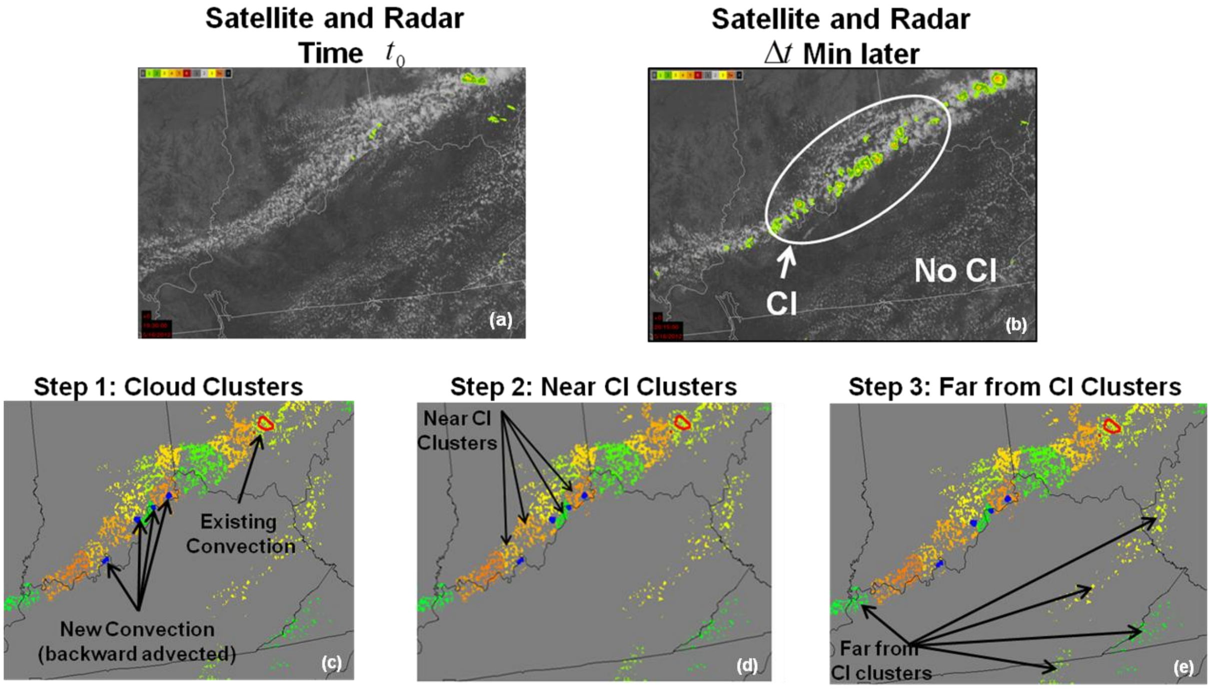

Figure 2: *Satellite imagery is broken down into cloud clusters by identify candidate cumulus clouds, and forming clusters using a mean shift segmentation. In (a) and (b), a growing line is shown. The cumulous clouds are broken into clusters, which are represented by different colors in (c), (d) and (e). The blue contours represent regions of imminent CI detected through the backwards advection process (Figure 1), and the red contour represents existing convection. The resulting clusters are then labeled "CI" or "No CI" based on their vicinity to the CI regions, as seen in (d) and (e). Clusters near existing convection are not included in the training set.*

The Mean Shift algorithm partitions large contiguous groups of clouds into clusters based on their spatial density and cloud category. This clustering step uses a spatial band width to perform clustering, which can be interpreted as the approximate average size of the cluster, though clusters will vary in size and shape. Clusters which fail to meet preset size criteria are discarded. The goal of the clustering is to automatically partition large areas of cumulus into smaller non-overlapping groups such that training data can be collected within each. The fact that resulting clusters are non-overlapping is important, because this ensures the training data does not contain duplicate data from the input images. This helps avoid over fitting when training the machine learning models.

Once clusters have been computed, environmental, satellite and numerical model fields are sampled from underneath each cluster. Initially, several statistical features are gathered, e.g. mean, standard deviation, minimum, maximum, and several percentiles of the distribution under the cluster. If a predictor field is unavailable, or does not contain valid data within a cluster, the cluster is discarded. Finally, each cluster with valid data is labeled as "Near CI" or "Far from CI" using the CI region map described in Section 4.1, and clusters which fall in regions of existing convection are discarded. This full list of features together with the CI classification is saved to a database for later analysis.

4.3 Feature Selection and Receiver Operating Characteristics

The processes described in Sections 4.1 and 4.2 are performed over several weeks of data during the summer of 2012 to construct a training set containing thousands of points, both near developing CI and far from it. At this point, the database contains a large number of features extracted from the distribution of predictor fields within each cloud cluster derived during sampling. Before training any models, some preprocessing of the data set is performed. Clusters over the ocean are discarded in order to focus the training on land based CI. In addition, clusters which fail to fall in a region of sufficient environmental instability are also discarded, since this will force the model to focus more in the more difficult situations of broad instability, and will avoid false alarms from happening in regions where the environment is not suitable for storm growth.

In order to trim down the large list of features under consideration as predictors, the importance of various features was measured using decision tree ensemble Gini Importance method described in Section 3.3. A sample of the most important features is shown in Figure 3. In addition to the important features, a "control" feature was added to the training set which consists of a vector of random numbers which are independent of the training set. This is used as a baseline to determine which features truly contribute to the results, that is, if a feature does not out-perform a random feature, it is not used in the final training. Features that are highly correlated with other features are also discarded as they add no further value to the classification and tend to bias the importance results.

From Figure 3, the satellite indicators, particularly those associated to the 10.7-μm channel and the local spread of the satellite peaks field, score high on the importance measures. Observe also that the importance of the environmental instability is not as large as one might expect, however this is due to the preprocessing step where clusters in a region with low environmental instability are removed before the model is trained. The LAMP forecast also proves to be a favorable indicator of CI, trailed by the NARRE-TL, which also adds value to the classification.

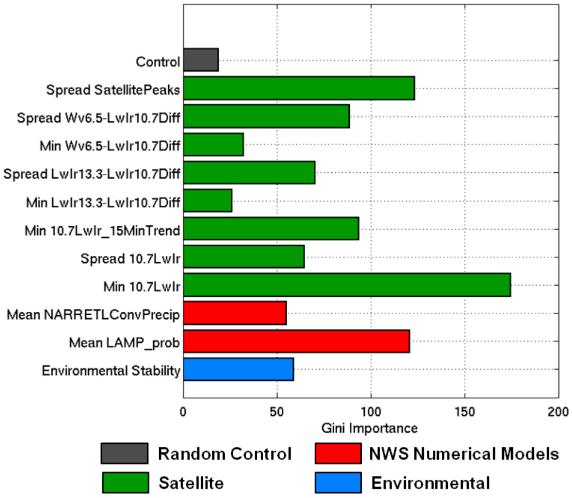

Figure 3: Feature importance as measured by the Decision Tree Ensemble. *The colors represent the different data categories, satellite, numerical models and environmental. The control variable (shown in grey) is a random feature which is used as a baseline importance. The importance measure used is Gini importance, which quantifies the contribution of each predictor for classifying CI.*

After the preprocessing and feature selection, the indicators shown in Figure 3 are used to train the models discussed in Section 3. The final training set used for the results below consisted of approximately 3000 near CI clusters, and approximately 12,000 far from CI clusters, which were collected randomly over the summer of 2012. A small subset of this training set was withheld for testing, and the models were trained. A ROC curve, showing the performance of these models on the testing set is shown in Figure 4.

Of the three methods, the decision tree ensemble appears to perform the best, however not by a significant amount. Overall performance can be measured by the area under the ROC curve (AUC), averaged over a 10-fold cross validation by withholding different subsets of the training data for testing. The results of this cross validation are 0.841 for Decision Tree Ensembles, 0.821 for Artificial

Neural Networks and 0.809 for Logistic Regression. The differences in the outputs of the cross validation are significant to the 5% significance level.

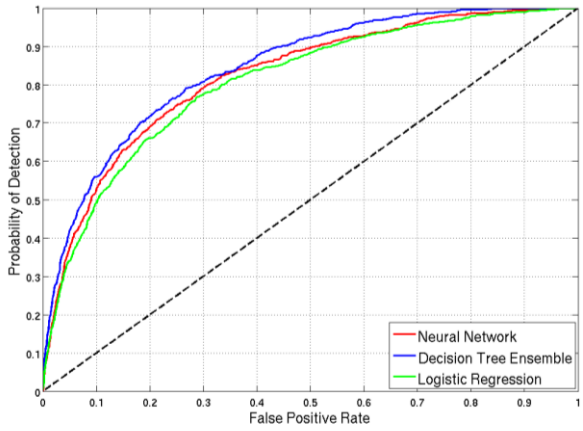

Figure 4: ROC performance of the three models described in Section 3 for predicting CI. *The models were trained and applied to a separate testing set withheld during training. The decision tree ensemble shows the best performance, followed by neural networks and logistic regression.*

4.4 Real-Time System Implementation

The models listed in Section 3 are each configured to run on the Lincoln Laboratory real-time research system and are used to create a deterministic CI forecast. New forecasts are generated every 5 minutes. The inputs listed in Section 2 are ingested in real time, and are time aligned to the current forecast time stamp if necessary. Satellite products are discarded if they become more than 15 minutes old. Numerical model products are ingested hourly, and are discarded if they become more than an hour old. Currently, only GOES-E is used in this study and as such, CI is added to the forecast only east of the Rockies.

Similar to training, only areas with developing cumulus clouds (small, developing or towering) are processed, which ignores a large proportion of the domain and provides a real time computational speed-up. Additionally, pixels which fail to meet a preset environmental stability threshold are ignored. Around each pixel, a circular kernel is used to gather all neighboring pixels containing cumulous clouds, and the same features used to train the models are extracted from all predictor fields. The size of the kernel was chosen to match the average size of the clusters formed in the training process. The vector of extracted features is passed to the trained machine learning model which generates a probability of CI at each grid point. If this probability exceeds a preset threshold, that pixel is activated in a CI Interest field which is used to grow new convection in the 0-2 hour forecast. The details of how new storms are added to the forecast in the CIWS system are not described here (Iskenderian, et al. 2010). We found it important to choose a conservative threshold range because even a small to moderate false alarm rate can be detrimental in a deterministic forecast for an aviation application.

5. Results

This Section provides some results of the various machine learning algorithms described previously. Some case studies are presented, and then statistics gathered over a larger sample of days is discussed.

Figures 5 and 6 show examples of a 1 hour forecast with the new CI module, the 2011 version of CIWS with the fuzzy logic module, and the radar measurements of VIL for verification. Figure 5c shows the results of a 1 hour forecast over the southern US where the Decision Tree Ensemble model detected storm growth over Arkansas and Mississippi before any radar signature was available, and shows a noticeable improvement over the previous version of CIWS (Figure 5d), which was unable to detect the new storm growth far from the existing VIL Level 3+ growth. Despite the improvement, the difficulties in forecasting convective initiation are also apparent in this case, namely the phase errors observed in this forecast caused partially by the lack of reliable storm motion vectors at forecast time, and difficulty in time aligning satellite images with radar images. These factors should be weighed when assessing accuracy of these forecasts. That being said, the storms introduced into the forecasts give the character and approximate location of the severe weather, which we feel provides significant benefit for CIWS users.

Figure 6 shows a high aviation impact case over the Northeast where the Decision Tree Ensemble detected a large line of storms growing over Pennsylvania, New York, Connecticut and Massachusetts. The improvement over the previous version of CIWS is clear, especially in the storm growth in western and northeastern Pennsylvania. The CI module could admittedly have been more aggressive in this case; however the addition of new storms would likely have given some warning and alleviated some of the burden on aviation planning.

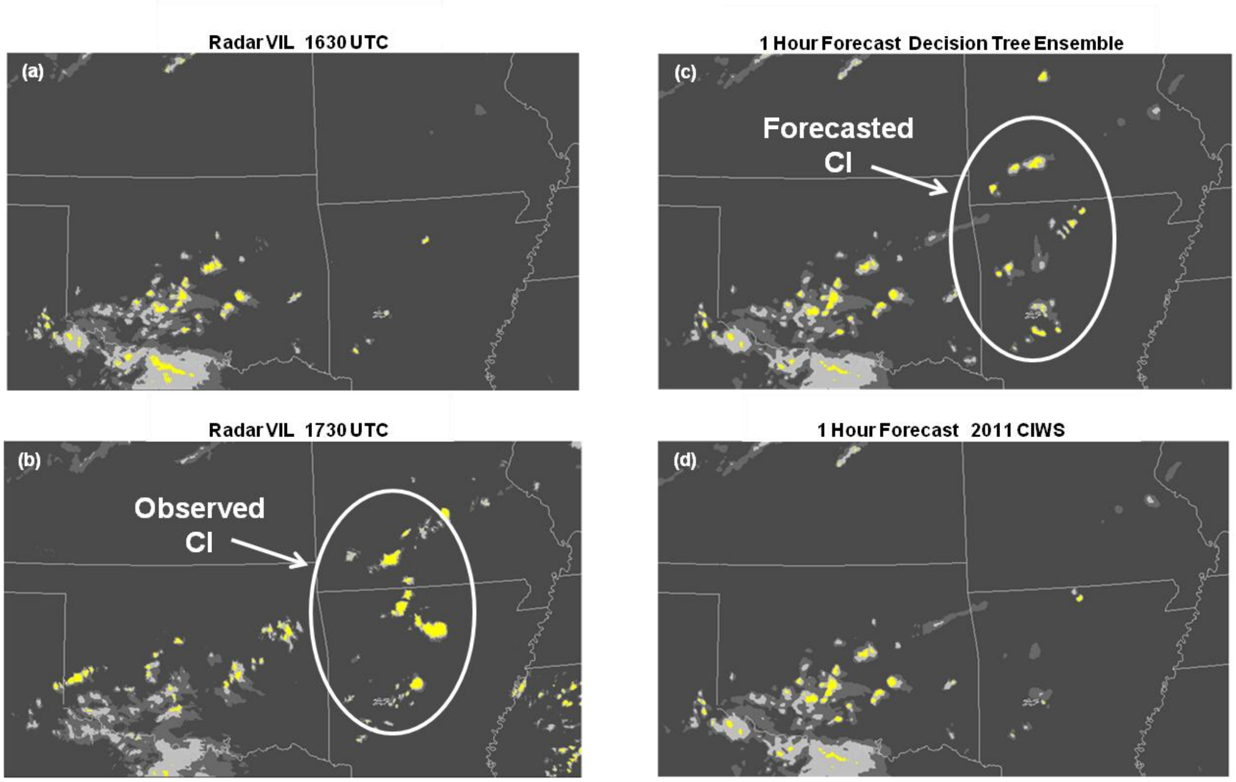

Figure 5: Example of CI added to the 0-2 hour forecast using the Random Forest. *The left column shows radar VIL at 1630 UTC August 16th 2012(a) and 1 hour later (b). The right column shows a 1 hour forecast with the Decision Tree Ensemble (c) and the CIWS forecast without any machine learning model (d) valid at 1730 UTC. The Decision Tree Ensemble correctly predicted CI (circled) throughout southern Arkansas and northern Louisiana, however was unable to capture the air mass storms that develop over northwestern Mississippi.*

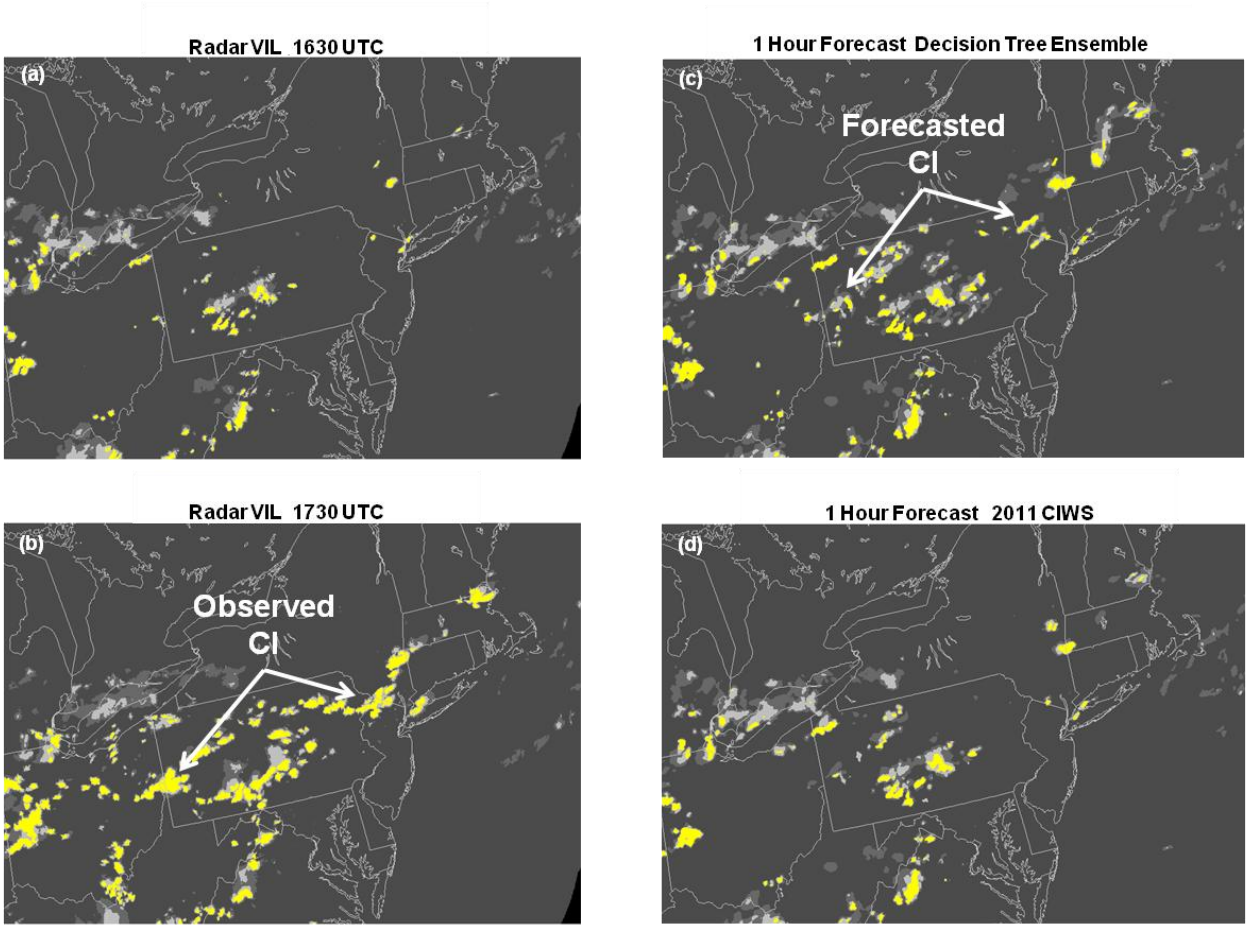

Figure 6: Another example of the CI module over the Northeast on a high aviation impact case. *The left column shows radar VIL at 1630 UTC on July 18th 2012 (a) and 1 hour later (b). The right column shows a one hour forecast with the Decision Tree Ensemble (c) and without any machine learning model (d). In this case, the Decision Tree Ensemble model produced a better forecast the storms about to grow over western and northeastern Pennsylvania.*

To obtain statistics over a larger number of cases, seven days during the summer of 2012 were chosen to test the new algorithm. These days were chosen because they exhibited substantial new storm development, and were cases where the 2011 version of CIWS showed a low bias. To measure improvement in the forecast, CSI and Bias are measured over the convective portion of the day and averaged over all days for each valid time. Forecasts were scored only in areas of VIP Level 3 and above, because these are the areas of aviation impact. Due to the phase errors and time alignment issues observed in the case studies, prior to computing the CSI, forecast and truth images are mapped to a 21 km domain.

The parameters used in these simulations, which include the settings for model training, and the probabilistic threshold used to initiate CI in the forecast, were set based on the ROC results in Section 4.3. As a result, the Decision Tree Ensemble ended up having a lower (more aggressive) probabilistic threshold than the ANN or LR, and this is evident in the bias results discussed below. Prior to running simulations, it is difficult to know if the parameters which performed best in the ROC setting would continue to perform well when used to generate a deterministic forecast since other factors such as phase error and growth and decay rates which impact the CSI score are not captured by the machine learning models. In the future, more simulations using different settings need to be run in order to optimize these choices.

Figure 7 shows the comparison of CSI scores for the three machine learning models considered here normalized to the CSI of the 2011 version

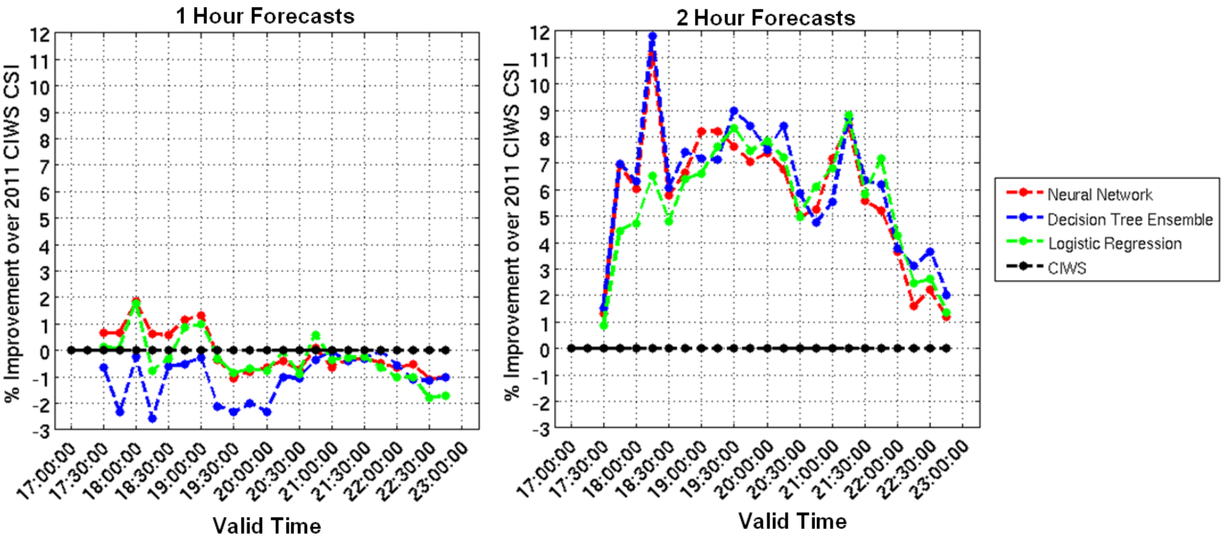

Figure 7: Average CSI by time of day for the three different machine learning models. *Forecasts and VIL mosaics used for validation are first mapped to a 21 km grid and the CSI is computed for VIP Level 3+. For 1 hour forecasts, a small improvement is seen for the earlier hours considered (17-20 UTC). For 2 hour forecasts, each model shows improvement over the 2011 version of CIWS for forecasts made in the most convective initiation portion of the day (18-22 UTC).*

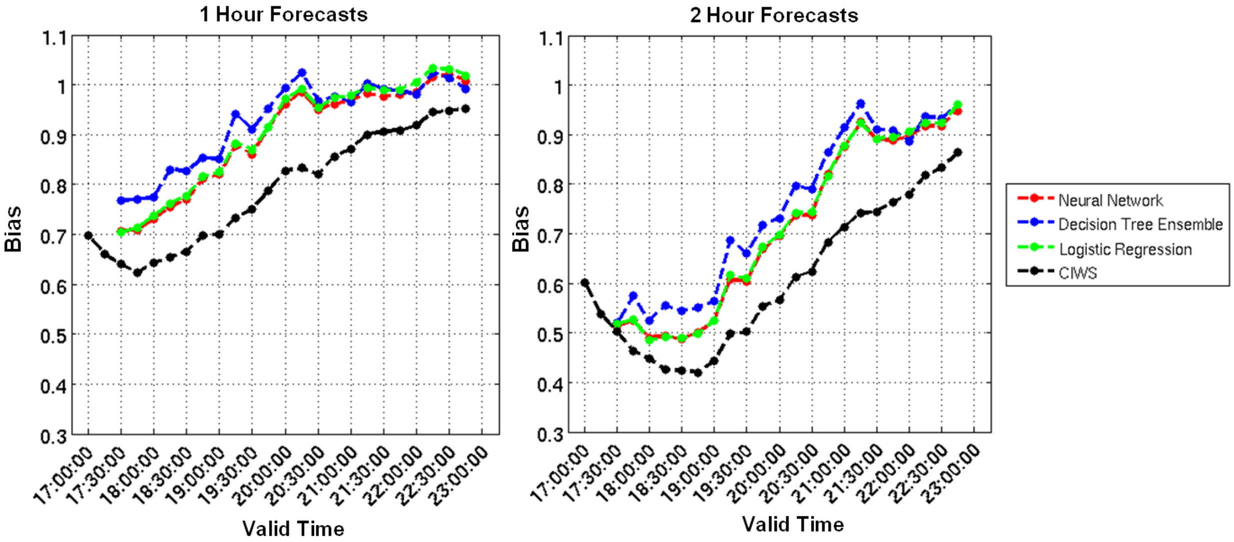

Figure 8: Forecast Bias of VIP Level 3+ versus time of day. *The 1 and 2 hour forecast bias was computed on a 1 km grid for the Forecasts made with all three CI models trained in this study. For both 1 and 2 hour forecasts, each model adds additional storms and improves the bias towards 1. The Decision Tree Ensemble added the most in these forecasts and shows the highest bias of the models considered.*

of CIWS. These scores are for 1 and 2 hour VIL forecasts. The real benefit of the models can be seen in the 2 hour forecast scores while for 1 hour forecasts, the CSI shows variable results, with a slight improvement seen in the ANN and LR models in the earlier portion (17-20 UTC) and slightly degraded performance later on. The Decision Tree Ensemble, which had the most aggressive settings, did not perform as well for 1 hour forecasts. For 2 hours, it is clear that each of the three methods show improvement over CIWS over the convective portion of the day (18 – 22 UTC). All three models perform well in this case, though the Decision Tree Ensemble performed best on average for the two hour forecasts, followed by ANN and LR close behind.

In Figure 8, the Bias of the three models is shown for 1 and 2 hour forecasts. The case days chosen for the scoring all exhibited CI, which explains the low bias seen in CIWS over the course of the day. All three models improved the low bias, though there is still more that can be done. The more aggressive setting in the Decision Tree Ensemble is evident since these forecasts show the highest bias. Each of the models can be made more aggressive by lowering the probabilistic threshold; however lowering this threshold by too much increases the frequency of false alarms. When comparing the bias to the 1 hour CSI in Figure 7, it seems the higher bias did not always result in an improved forecast, while for 2 hour forecasts, the additional storm growth provided benefit to the forecast. We believe this can be attributed to the new storm growth being too fast in the 0-1 hour forecast range. By two hours, many of these predicted new storms have had a longer time to grow, and hence storms added into the forecast increase the CSI. We will work to improve the storm growth model in the future.

6. Conclusions and Future Work

In this study, multiple forms of data were fused in a machine learning framework to improve the nowcasting of convective initiation in 0-2 hour forecasts. Data included in this model are satellite, including the Satellite Convection Analysis and Tracking (SATCAST) fields, numerical models, including the Local Aviation Model Output Statistics (MOS) Program (LAMP) and North American Rapid Refresh Ensemble Forecast System (NARRE-TL) thunderstorm probability forecasts, and an environmental stability product developed by Lincoln Laboratory to show regions conducive to new storm growth in the near future. Training data are gathered over several weeks for both near and far from CI events. Using this data, different machine learning models, including Logistic Regression, Artificial Neural Networks, and Decision Tree Ensembles are trained. The results from the three models are compared to each other, as well as to the 2011 CIWS system to measure improvement in the nowcasting system.

Of the three models trained, the Decision Tree Ensemble performed the best over a seven day sample for 2 hour forecasts, while a less aggressive Artificial Neural Network performed well for 1 hour forecasts. For 2 hour forecasts, all three models considered provided improvement over 2011 CIWS forecasts, as measured by CSI and Bias. Additional model tuning and feature engineering is still underway, however these results show that the machine learning methodology can be an effective technique for easily adding new predictors and improving deterministic nowcasts.

Moving forward, we seek to improve the machine learning methodologies in several ways. First the model will be retrained regularly with recent data so that the model can adapt to changing meteorological conditions. Second, additional fields are being investigated for inclusion in the model. These fields include the Short Range Ensemble Forecast (SREF) (Du et al. 2003), and tracking and trending information from the High Resolution Rapid Refresh (HRRR) model. The HRRR has the potential to be a particularly useful predictor of CI because of the fine scale resolution (3 km). Finally, expansion of the CI model to the western US using GOES-W is also underway.

7. References

Berendes, T. A., Mecikalski, J. R., MacKenzie Jr, W. M., Bedka, K. M., & Nair, U. S. (2008). Convective cloud identification and classification in daytime satellite imagery using standard deviation limited adaptive clustering. *Journal of Geophysical Research*, *113*(D20), D20207.

Bochkanov, S., Bystritsky, V. AlgLib http://www.alglib.net

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Comaniciu, Dorin, and Peter Meer. (2002) "Mean shift: A robust approach toward feature space analysis." Pattern Analysis and Machine Intelligence, IEEE Transactions on 24.5, 603-619.

Du, J., DiMego, G., Tracton, M. S., & Zhou, B. (2003). NCEP Short Range Ensemble Forecasting (SREF) system: multi-IC, multi-model and multi-physics approach. *Research Activities in Atmospheric and Oceanic Modelling (edited by J. Cote), Report*, *33*, 5-09.

Ghirardelli, J. E. (2005, August). An overview of the redeveloped Localized Aviation MOS Program (LAMP) for short-range forecasting. In *Preprints, 21st Conf. on Weather Analysis and Forecasting/17th Conf. on Numerical Weather Prediction, Washington, DC, Amer. Meteor. Soc., 13B5.* (Available from http://ams.confex.com/ams/pdfpapers/95038.pdf).

Iskenderian, H., Ivaldi, C.F., Wolfson, M., Mecikalski, J. R., Bedka, K. M., Sieglaff, J., Feltz, W., Dworak, R. & MacKenzie, W. M. (2010, January). Satellite data applications for nowcasting of convective initiation. In *14th Conference on Aviation, Range, and Aerospace Meteorology*. (Available from http://www.ll.mit.edu/mission/aviation/publications/publicatiofiles/mspapers/Iskenderian_2010_ARAM_WW-23498.pdf)

Mecikalski, J. R., & Bedka, K. M. (2006). Forecasting convective initiation by monitoring the evolution of moving cumulus in daytime GOES imagery. *Monthly Weather Review*, *134*(1), 49-78.

Roberts, R. D., & Rutledge, S. (2003). Nowcasting storm initiation and growth using GOES-8 and WSR-88D data. Weather and Forecasting, 18(4), 562-584.

Segal, Mark R. (2004). Machine Learning Benchmarks and Random Forest Regression. Center for Bioinformatics & Molecular Biostatistics.

Velden, C. S., Hayden, C. M., Nieman, S. J., Menzel, W. P., Wanzong, S., & Goerss, J. S. (1997). Upper-tropospheric winds derived from geostationary satellite water vapor observations. *Bulletin of the American Meteorological Society*, *78*(2), 173-196.

Williams, J. K., Ahijevych, D. A., Kessinger, C. J., Saxen, T. R., Steiner, M., & Dettling, S. (2008, January). A machine learning approach to finding weather regimes and skillful predictor combinations for short-term storm forecasting. In *13th Aviation, Range and Aerospace Meteorology Conference*. (Available from http://nldr.library.ucar.edu/repository/assets/osgc/OSGC-000-000-003-270.pdf)

Wolfson, M. M., & Clark, D. A. (2006). Advanced aviation weather forecasts. Lincoln Laboratory Journal, 16(1), 31. (Available from http://www.ll.mit.edu/publications/journal/journalarchives16-1.html)

Xie, Y., Koch, S. E., McGinley, J. A., Albers, S., & Wang, N. (2005, August). A sequential variational analysis approach for mesoscale data assimilation. In *Preprints, 21st Conf. on Weather Analysis and Forecasting/17th Conf. on Numerical Weather Prediction, Washington, DC, Amer. Meteor. Soc. B* (Vol. 15). (Available from https://ams.confex.com/ams/WAFNWP34BC/techprogram/paper_93468.htm)

Zhou, B., Du, J., Manikin, G., & DiMego, G. (2011, August). Introduction to NCEP's time lagged north american rapid refresh ensemble forecast system (NARRE-TL). *In 15th Conference* *on Aviation, Range, and Aerospace Meteorology.* (Available from https://ams.confex.com/ams/14Meso15ARAM/techprogram/paper_190788.htm)