2012-07-13

Prediction of Rapid Intensity Changes in Tropical Cyclones Using Associative Classification

Michael L. Jankulak
University of Miami, mike.jankulak@noaa.gov

UNIVERSITY OF MIAMI

PREDICTION OF RAPID INTENSITY CHANGES IN TROPICAL CYCLONES USING ASSOCIATIVE CLASSIFICATION

By

Michael L. Jankulak

A THESIS

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Master of Science

Coral Gables, Florida

August 2012

UNIVERSITY OF MIAMI

A thesis submitted in partial fulfillment of
the requirements for the degree of
Master of Science

PREDICTION OF RAPID INTENSITY CHANGES IN TROPICAL CYCLONES USING ASSOCIATIVE CLASSIFICATION

Michael L. Jankulak

Approved:

Mei-Ling Shyu, Ph.D.
Associate Professor of Electrical and Computer Engineering

M. Brian Blake, Ph.D.
Dean of the Graduate School

Saman Aliari Zonouz, Ph.D.
Assistant Professor of Electrical and Computer Engineering

Frank D. Marks Jr., Sc.D.
Research Meteorologist and Director
NOAA/AOML Hurricane Research Division
Miami, Florida

JANKULAK, MICHAEL

(M.S., Electrical and
Computer Engineering)
(August 2012)

Prediction of Rapid Intensity Changes in Tropical Cyclones Using Associative Classification

Abstract of a thesis at the University of Miami.

Thesis supervised by Professor Mei-Ling Shyu
No. of pages in text. (81)

Accurate forecasting of Tropical Cyclone (TC) track and intensity are vital for safeguarding the lives and property of communities in regions that are subject to TC impact. While there have been impressive advancements in TC track forecasting over the last 40 years, forecasts of TC intensity have seen virtually no improvement since 1990, chiefly because of the difficulty of predicting rapid changes in TC intensity. This study applies data mining techniques to a data set of meteorological parameters in order to construct an associative classifier that has been named AprioriGrad. This classifier is based on the association rule mining technique together with the Apriori algorithm for frequent itemset selection, but includes customizations for detecting rare events and for labeling a series of interrelated classification targets defined as yes/no thresholds on an underlying continuous measurement of 24-h TC intensity change. AprioriGrad's performance on this domain is compared to a variety of classification techniques, and implications for possible development as an operational forecasting tool or for further meteorological study are examined.

To Sim

Acknowledgments

The author would like to thank Dr. Mei-Ling Shyu for taking on a student at a very late date and making it possible to finish in a year's time, something that seemed at times to be an unattainable goal. Thanks also to Drs. Saman Aliari Zonouz and Frank Marks for much encouragement and illuminating discussions. Thanks to Drs. Jim Hendee and Michelle Wood for their personal and professional support without which this project could never have reached completion. A special thanks to Dr. Peter Ortner who provided a much-needed rescue at a point when the entire enterprise seemed to be lost. Above all, my profound gratitude to Dr. Sim Aberson who provided the kernel of the idea for this project and maintained an unflagging belief in its eventual success even when circumstances suggested this belief was sorely misguided. This work is dedicated to you with my love.

MICHAEL JANKULAK

University of Miami

August 2012

Contents

List of Figures viii

List of Tables ix

CHAPTER 1 Introduction 1

1.1 SHIPS and the Prediction of Tropical Cyclone Intensity Changes 1

1.2 Association Rule Mining and the Apriori Algorithm 2

1.3 Building an Associative Classifier for Predicting Rare Events 3

CHAPTER 2 Literature Review 6

2.1 SHIPS, Linear Regression, and LGEM 6

2.2 A Brief Survey of Classification Techniques 7

2.3 Other Techniques Applied to TC Intensity Forecasting 10

2.4 Association Rule Mining and TC Intensity Forecasting 11

CHAPTER 3 The SHIPS Data Set 14

3.1 Introduction to the SHIPS Data Files 14

3.2 Raw Attributes 16

3.2.1 SHIPS Attributes with Past, Present and Future Values 17

3.2.2 SHIPS Attributes with Present and Future Values 17

3.2.3 Other SHIPS Attributes 20

3.3 Calculated Attributes 22

3.3.1 Calculated Input Attributes 22
3.3.2 Calculated Target Class Attributes 23

CHAPTER 4 Data Preprocessing **25**

4.1 Producing an ARFF Data File 26
4.1.1 Data Consistency Checks and Corrections 26
4.1.2 Adding Calculated Attributes (Input Attributes) 29
4.1.3 Adding Calculated Attributes (Target Class Attributes) 31
4.1.4 Formatting as ARFF 33
4.2 From ARFF to ARM 34
4.2.1 Filtering the Best Track Attributes 35
4.2.2 The Optional Stratification Step 36
4.2.3 Missing Target Class Attributes and Serial Correlations 37
4.2.4 Choosing a Target Time 38
4.2.5 Cross-Validation and Discretization 39
4.2.6 Final Reduction of Attribute Pool 41

CHAPTER 5 Associative Classification **44**

5.1 Association Rule Mining with Apriori 44
5.1.1 Apriori Basics 44
5.1.2 Apriori Customizations 46
5.2 Associative Classification 48

CHAPTER 6 Results **53**

6.1 WEKA Algorithms Used for Baseline Comparison 53
6.2 Metrics Used for Comparison 55
6.3 Performance Comparison: AprioriGrad vs. WEKA Algorithms 57
6.3.1 Tabular and Graphical Presentation of Results 57

6.3.2 Discussion of Results 57
6.4 Detailed Results from AprioriGrad 66

CHAPTER 7 Conclusions **73**

7.1 Engineering Relevance 73
7.2 Use as a Forecast Tool 74
7.3 Meteorological Relevance, Future Work 75

Bibliography **77**

List of Figures

6.1 Individual rapid intensification results from all classifiers 60
6.2 Individual rapid weakening results from all classifiers 61
6.3 Ranking of average classifier performance by metric 63
6.4 Averaged classifier results from all classifiers 64
6.5 Percentage of cases classified as positive by AprioriGrad 71

List of Tables

3.1 SHIPS attributes with past, present and future values 18

3.2 SHIPS attributes with present and future values 18

3.3 SHIPS attributes from the HEAD record type 20

3.4 SHIPS attributes related to GOES data values 21

3.5 Other SHIPS attributes 22

3.6 Calculated input attributes 23

3.7 Calculated target class (rapid intensification/weakening) attributes 24

4.1 Ranges of intensities/pressures in the SHIPS data set 27

4.2 Consistency of SHIPS record types between neighboring instances 29

4.3 Counts of instances by initial intensity in the SHIPS Atlantic file 37

4.4 Average number of attributes remaining after filtering 42

4.5 Counts of rare events by target class attribute 43

6.1 WEKA classifiers used for baseline comparison 54

6.2 The confusion matrix 55

6.3 Rapid Intensification: AprioriGrad vs. selected WEKA classifiers 58

6.4 Rapid Weakening: AprioriGrad vs. selected WEKA classifiers 59

6.5 ApriorGrad confusion matrices 67

6.6 Frequently-featured SHIPS record types (summary) 69

6.7 Frequently-featured SHIPS record types (detail, intensification) 69

6.8 Frequently-featured SHIPS record types (detail, weakening) 70

Chapter 1

Introduction

1.1 SHIPS and the Prediction of Tropical Cyclone Intensity Changes

Access to accurate tropical cyclone (TC) forecasting tools is of vital important to hurricane forecasters and communities impacted by tropical systems. The more precise the forecast, both in terms of the *track* of a TC and its future *intensity*, the better a community can be forewarned to prepare in terms of deploying property protections and evacuating vulnerable populations. There is an obvious cost to underwarning, which may lead to avoidable injury or destruction of property, but there is also a cost to overwarning, which may lead to costly preparations or disruptive evacuations that could have been avoided.

Forecasts of TC track have enjoyed significant improvements since 1970 [1]. For example, National Hurricane Center (NHC) track forecasts from 1970-1998 improved at an average annual rate of 1.0% for 24-h forecasts, 1.7% for 48-h forecasts, and 1.9% for 72-h forecasts [2].

TC intensity forecasts, however, are not in general as accurate as track forecasts, and there has been virtually no improvement in intensity forecast accuracy since 1990 [1]. The main cause of this difference in performance between track and intensity forecasting is the phenomenon termed rapid intensification (RI). This occurs when a TC's intensity, as measured by its maximum sustained surface winds, undergoes rapid (i.e., within 24 h) in-

creases. This kind of nonlinearity in intensity levels, as well as the reverse phenomenon of rapid weakening (RW), is not well predicted by the available intensity forecast models [1]. In an attempt to overcome this limitation of intensity forecasting, meteorologists have focused on the phenomenon of rapid TC intensity changes in the hopes of enhancing the understanding of RI and RW in particular and thereby improving the forecast of TC intensity changes in general.

Toward the furtherance of this goal, a model known as the Statistical Hurricane Intensity Prediction Scheme (SHIPS) [3] has been developed using a multiple linear regression technique. The authors of SHIPS have collected a data set of all TC cases from the Atlantic, East Pacific and Central Pacific basins dating back to 1982. This data set is populated with climatological, persistence and synoptic predictors, measurements known or believed to be related to TC intensity changes, and this data set is publicly available [4].

1.2 Association Rule Mining and the Apriori Algorithm

The goal of the present study is to apply a data mining technique known as association rule mining (ARM) to the problem of forecasting rapid intensity changes of TCs. There are two main challenges of applying ARM in this domain: for one, the data set includes a large number of attributes and instances, which often leads to exponentially growing requirements of computer memory or processing time when searching for the best association rules. The other challenge is that this is a domain where rapid intensification is very rare and rapid weakening even more so, which means that an ARM algorithm must be customized to handle rare events.

Nevertheless, ARM is an ideal approach to this topic because it does not require detailed knowledge of the physical meanings or likely usefulness of prediction parameters in the data set. An engineer may easily apply this technique to this meteorological domain, and the yield will be a completely unbiased determination of which predictors are most

significant in the domain. This opens up the possibility of the discovery of unexpected physical relationships that a meteorologist might not otherwise consider.

The process of association rule mining may be broken down into its two constituent steps. The first step involves identifying common associations among data set attributes. In the terminology of ARM these common associations are referred to as *frequent itemsets*[1]. The second step involves generating candidate association rules from these frequent itemsets and choosing the "best" of these rules based on some optimization criteria. For the first of these steps, the present study begins with the Apriori algorithm for frequent itemset selection, as described in [5]. The generation of candidate rules from frequent itemsets is a simple matter of splitting up a frequent itemset in different ways among the rule's antecedent and consequent. One such rule induction algorithm is described in the Apriori paper [5] and in more detail elsewhere [6] [7], although the task is simplified in the present study since candidate rules are here limited to those with a single attribute, the target class attribute, in the consequent.

For simplicity of language the remainder of this study will use *Apriori-based association rule mining algorithm*, or more simply *Apriori algorithm*, to refer collectively to these two steps of frequent itemset identification followed by association rule induction. This is consistent with the terminology used by numerous publications in this field [8] [9] [10] [11] [12] [13].

1.3 Building an Associative Classifier for Predicting Rare Events

Rather than simply mine for a handful of association rules and report on their relative strengths and weaknesses, this study is framed in terms of building a classifier for RI and RW, each at four different intensity levels, for a total of eight target class attributes. When conceived as a classification problem, the data set can be randomly divided into subsets for cross-validation experiments, where the classifier is trained on one subset of cases and then

[1]The reason for this nomenclature is explained in more detail in Section 5.1.1.

tested on a disjoint subset of cases that were not used as input when building the model. This yields an objective evaluation of the model's performance as a prediction of its ability to correctly forecast a developing TC operationally.

Associative Classification is an approach that builds upon the technique of association rule mining by using the mined rules to *classify* test cases that were not considered during the building of the classifier. The starting point of the present study is an algorithm known as Classification Based on Association (CBA) [8], which is itself an enhancement of the Apriori-based association rule mining algorithm [5]. The CBA algorithm adapts Apriori to identify only those rules with a selected attribute (i.e., the target class attribute) in the consequent. CBA then generates a classifier by defining an optimal ordering of association rules, a procedure for including or discarding each of those rules in turn, and one final procedure for pruning the rule set to remove rules that do not favorably impact the accuracy of the classifier.

The present study further enhances the CBA logic to limit its search not only to a selected target class attribute but more narrowly to a selected value of that selected target class attribute. This enables positive (negative) rules, which are rules which conclude that the event of interest will (will not) occur, to be mined separately. After this an optimal subset and ordering of rules is selected, along with a default class for cases that do not match any rule.

In cases where the event of interest is quite rare, the training set of positive and negative cases may be significantly imbalanced and approaches such as CBA may not be optimal. Much work has previously been done on this problem, with approaches which seek to more nearly balance positive and negative sets by discarding certain negative cases before mining [14], or which reduce the feature space to improve accuary [10] [12]. The present study follows the example of [14] in discarding certain negative cases before mining for

positive rules, but here the decision of which cases to discard and which cases to retain is opposite to that of the prior work.

Since the end result of this study is a classification algorithm, the results are compared against classification results on this same domain from a number of standard classification algorithms, including decision trees, neural networks, support vector machines, nearest-neighbor and naive Bayes classifiers.

Chapter 2

Literature Review

2.1 SHIPS, Linear Regression, and LGEM

The Statistical Hurricane Intensity Prediction Scheme (SHIPS) is a model for forecasting the intensity of tropical cyclones (TCs). In 1994 SHIPS produced 3-day forecasts in the Atlantic basin [3] and since then has been updated to run in the Eastern North Pacific Basin [15], extended to produce 5-day forecasts [16], and enhanced to account for TC interactions with land. SHIPS uses the technique of multiple linear regression with intensity change as the dependent variable and independent variables for climatological, persistence and synoptic predictors [3]. In 2009 a simplified system for TC intensity prediction was proposed, and given the name Logistic Growth Equation Model (LGEM) [17] based on its underlying methodology. LGEM was reported to have outperformed SHIPS on a 2006-2007 independent sample, and continues to be run side-by-side with SHIPS for the benefit of hurricane forecasters.

A forecast model's *skill* is defined by the degree to which it outperforms a model based on climatology and persistence alone. For forecasts of TC intensity, the baseline model is the Statistical Hurricane Intensity Forecast (SHIFOR) [18] [19]. The National Hurricane Center and the Joint Typhoon Center produce operational intensity forecasts that are in part based on SHIPS (and its west Pacific analogue, the Statistical Typhoon Intensity Prediction

Scheme, or STIPS [20]), and these forecasts have been found to have "significant skill" [21] out to 96 h in the Atlantic and to 72 h in the east and west Pacific. However, TC *intensity* forecasts are far less skillful than forecasts of TC *track*, some of which are 2 to 5 times more skillful than the intensity forecasts by 72 h. One source of intensity forecast error is the non-linear Rapid Intensification (RI) of some TCs which cannot be accurately modeled by the linear SHIPS.

The original SHIPS was based on a dependent sample of 510 cases from 49 TCs from 1982 to 1992 in the Atlantic basin. Each year a new set of cases is added to the dependent sample which is shared online [4] and which now has 9926 cases in the Atlantic and 13221 cases in the East and Central Pacific through the end of 2011 hurricane seasons. The original SHIPS considered eight climatological and seven synoptic predictors as well as six quadratic combinations of these variables. After analysis, the model retained ten linear and one quadratic predictor. Since 1994 some predictors have been dropped and new ones have been added until the current data set contains 66 predictors, each of which may have a value for up to 23 time points beginning 12 hours before the case's initial time and progressing in 6-h time steps until 120 h after the initial time. These, then, are the elements that make up the SHIPS data set.

2.2 A Brief Survey of Classification Techniques

The goal of the present study is to construct an associative classifier that is uniquely adapted to the domain of forecasting TC intensity changes. The natural question, therefore, is how the performance of this associative classifier may compare to other available classification techniques. The Waikato Environment for Knowledge Analysis (WEKA) [22] [23] is a useful tool for answering this question, and also serves as a starting place for building the associative classifier itself. WEKA is an extensive collection of Java implementations of machine learning algorithms. Its source code is made freely available under the GNU General Public License by the WEKA creators at the University of Waikato, New Zealand.

WEKA releases date back to 1996, when it was written largely in C, and back to 1999 in its current 100% Java form. WEKA is nicely complemented by an excellent Data Mining textbook [24] written by the primary WEKA developers. For purposes of the present study, nine "competing" classifiers have been run in experiments for comparison against the newly-developed associative classifier, and these nine classifiers consist of variations of seven kinds of techniques: support vector machines, nearest-neighbor techniques, decision trees, bagging, adaptive boosting, neural networks and a naives Bayes classifier.

Support Vector Machines (SVMs) [25] are well-suited to classification problems with large numbers of numeric attributes and a binary-valued target class. SVMs look for a hyperplane boundary in highly-dimensional space which defines an optimal separation of training instances according to their target class assignments. WEKA uses the Sequential Minimal Optimization (SMO) [26] [27] algorithm to train its SVMs, and the present study makes use of a polynomial kernel function.

The Nearest-Neighbor [28] technique (k-NN, with k referring to the number of neighbors considered for classification) is similar to SVMs in that it classifies an unlabeled instance in terms of its location in multidimensional space relative to instances whose class assignments are known. Rather than determining, for a test instance, what side of a boundary hyperplane it falls on (which is the approach of SVMs), the k-NN technique examines the class assignment(s) of the k nearest instance(s) and returns the most frequently-encountered class label found among the test instance's neighbor(s). WEKA's implementation of k-NN is named IBk, since the technique is an implementation of *Instance-Based* classification. Variations of k-NN may make use of different *distance* measures; the most commonly-selected of these is Euclidean distance, which is used in the present study, and both 1-NN and 3-NN are here tested alongside the associative classifier. The k-NN technique has previously been applied to the related problem of estimating the *current* intensity of a TC based on satellite-provided microwave imagery [29].

Decision Trees have long been a productive focus of classification research, and perhaps the most widely-known decision tree algorithm is C4.5 [30]. This algorithm takes a divide-and-conquer approach to classification, looking for individual attribute tests that best divide the training set into groups of distinct target classes (as determined by a measure of highest information gain). It then recursively divides the resulting subsets according to other attribute tests. The resulting decision tree has the advantage of being intuitively easy to understand. WEKA's implementation of C4.5 is known as J48, for its J4.8 algorithm represents the last freely-available revision of C4.5 (revision 8) before it was superseded by the commercial C5.0 product. The C4.5 algorithm has previously been applied to an investigation of how four physical parameters (sea surface temperature, atmospheric water vapor, wind shear and zonal stretching deformation) influence TC formation and intensity [31].

Multiple classifiers may be combined through approaches such as *bagging* or *boosting*. Bagging refers to the construction of multiple classifiers of the same or similar type based on random sampling of the training set; these classifiers are then collectively applied to test cases and allowed to "vote" on the assignment of a target class. Random Forests [32] are an implementation of bagging that uses an underlying decision tree algorithm.

Adaptive Boosting is another meta-algorithm which, like bagging, produces multiple classifiers (all based on one underlying classification technique) in order to optimize its classification performance. Unlike bagging, which randomly generates multiple classifiers all at once, adaptive boosting *iteratively* produces classifiers. At each step of the iteration, adaptive boosting adjusts weights given to each instance in the training set in an effort to correct any misclassifications encountered thus far, and the classifiers thus produced are themselves weighted according to their overall performance on the training set. WEKA's adaptive boosting implementation is known as AdaBoostM1 [33], and the present study tests it using both J48 and RandomForest as an underlying classification technique.

Artificial Neural Networks are a machine learning technique that is inspired conceptually by the example of the human brain, with interconnected neurons working together to learn new concepts and apply those concepts to classification problems. The WEKA implementation of neural networks is the Multilayer Perceptron which learns by means of a technique known as backpropagation [34]. These artificial neurons or perceptrons are organized into multiple, hidden layers which attempt to learn their individual, optimal weights adaptively from feedback provided during an iterative training process. Many previous studies have applied neural networks in some form to the problem of estimating current (or forecasting future) TC intensity. Some studies [35] [36] focused on predicting TC intensity at 12-h intervals for typhoons in the western North Pacific; one study [37] made use of neural networks to predict the maximum potential intensity of typhoons, also in the western North Pacific; another study [38] combined neural networks with genetic algorithms to predict 24-h intensity changes of TCs in the South China Sea.

The Naive Bayes classifier [39] is a relatively simple application of Bayes' Theorem to a classification problem, and makes the highly unrealistic assumption that attribute values are independent of one another. However, in practice Naive Bayes often performs quite well despite its naive assumptions, and its WEKA implementation is included among the classifiers that are compared against the performance of the associative classifier in this study.

2.3 Other Techniques Applied to TC Intensity Forecasting

At Southern Methodist University's Department of Computer Science and Engineering, researchers developed a forecast algorithm which they called Weighted Feature Learning Extensible Markov Model (WFL-EMM) [40]. This is based on their earlier work of developing EMMs [41] and in this new study they make use of a genetic algorithm to learn the feature weights. Their algorithm is tested only on cases from the 2001, 2002 and 2003 hurricane seasons and they report that WFL-EMM errors are lower than those of SHIPS

through 72 h and higher thereafter. They do not comment on the statistical significance of their results.

2.4 Association Rule Mining and TC Intensity Forecasting

Another group of researchers from George Mason University's (GMU) Department of Geography and Geoinformation Science bring the techniques of association rule mining to bear on the forecasting of TC intensity. This work resulted in a doctoral dissertation [13], portions of which were also published in substantially similar form in four journal articles, with content from the dissertation's Chapter 4 [42], Chapters 5 and 6 [43], and Chapter 7 [9] [11]. The dissertation author asserts ([13], p. 28) that this work was the first time that association rule mining was used to study TC intensity changes, and the present study finds no other published science to contradict this claim. Because this body of work is very similar in intention and approach to the present study, it is worth examining its results in more detail.

The earliest paper [42] focuses on association rule mining from a data set of satellite data. This is not directly comparable to the present study, however it provides context for the GMU work because their satellite data set only had 53 cases for study, which motivated them to apply their techniques to the much larger SHIPS data set.

This led to a study [43] where the Apriori-based association rule mining algorithm was applied to the SHIPS data set to produce rules related to weakening, stable and intensifying TCs. For purposes of their study, these terms were defined based on a TC's intensity changes over a period of 12 h, where intensification (weakening) of a TC in excess of 5 kt led to a label of intensifying (weakening), and all other TCs were labeled as stable. The point of this study was to identify individual rules with optimal values of support, confidence and lift[2]. The study determined which predictors were most useful, and how

[2]Support and confidence are defined in Section 5.1.1. Lift (also known as interest) is defined as the ratio of the confidence of a rule to the support of its consequent.

the role of those predictors changed when starting with subsets of cases of different initial intensity. It examined ways of eliminating redundant predictors in rule antecedents without reducing the rule's accuracy. The study does not appear to have produced a forecast model or classifier, and it did not evaluate its rules on subsets of cases that were withheld during the association rule mining. Given the low 5-kt threshold of weakening/intensification, the study cannot be said to have concerned *rapid* intensity changes at all.

The final two papers [9] [11] are concerned with the study of rapid intensification of TCs by at least 30 kt over a 24-h period, which is directly analogous to the present study's RI30 target (as defined in Section 3.3.2). There is no mention of intensification at any other level, or of rapid weakening. These studies identify 11 statistically significant predictors from the SHIPS data set and consider the question of how many predictors per rule is the optimal number. For each number of predictors from 1 to 10, one optimal set of conditions is identified, and the study concludes that the set of conditions with 6 predictors identifies the highest probability of RI. The study applies these conditions to the raw data and identifies seven matching cases, six of which were found to have intensified at 30 kt or more. There is no indication that this approach was tested on a set of cases that were withheld during analysis.

In addition, all of the GMU work is based on a version of the SHIPS data set from 2003. At that time, the SHIPS data set included data only at 12-h increments, but in 2005 it started including more granular 6-h data. These prior works overcome this limitation by interpolating 6-h data from the 12-h time points but this approximation is no longer necessary. Since 2003 there have been several extremely active hurricane seasons leading to an increase in the number of SHIPS cases in the Atlantic basin by 44%, or 3044 additional cases for a new total of 9926 cases. Since 2003, at least 17 new predictors have been added to the data set (although some older predictors have also been removed) including at least two types that figure prominently in the present study's results[3]. In 2009 the Geostationary

[3]VVAV and HIST make numerous appearances in Table 6.6.

Operational Environmental Satellites (GOES) predictor values were extended back to 1983, having previously been available only back to 1995. All in all the SHIPS data set has become a far more detailed research tool compared to its state in 2003.

Thus the present study is based upon a much richer source of input data compared to the GMU work, and targets a wider range of rapid intensification (and here for the first time, rapid weakening) events compared to prior work. However, the novelty of the present study rests primarily on its conceptualization of this problem as a classification or forecast problem. The Apriori-based association rule mining algorithm forms the basis of this approach, as in the GMU body of work, but here it is used in ways that are carefully tuned for producing both negative and positive rules, and a strategy is presented for applying those rules to previously unseen test cases. The resulting associative classifier is tested against the performance of numerous other classifiers on the same problem and their results are compared for each of eight target class attributes.

Chapter 3

The SHIPS Data Set

3.1 Introduction to the SHIPS Data Files

The Statistical Hurricane Intensity Prediction Scheme (SHIPS) developmental data files are available for public download from their Colorado State University web site [4]. Two files are provided at this site: one for data from the Atlantic basin, and one for data from the Eastern Pacific and Central Pacific basins. The present study uses only data from the Atlantic basin.

A SHIPS data file contains data relating to tropical cyclones (TCs). These may be tropical depressions, tropical storms or hurricanes. As of this writing data are available for TCs from 1982 to 2011 inclusive. In the Atlantic this amounts to some 9926 instances from 421 distinct TCs over this 30-year period.

Each instance in the SHIPS data file corresponds to a single TC viewed from the perspective of a particular time point in its lifetime. This time point, hereafter referred to as the initial time, is the "now" time for TC forecasting when thinking operationally. Time points from the past and present provide the context of a TC's development and are useful for predicting future intensity changes.

Data from time points later than the initial time can be divided into two categories: verification data and estimated or predicted parameters. Verification data include the TC

intensity (which is the maximum sustained surface wind speed) and minimum sea-level pressure, and these data are not available as inputs for prediction. Other future-valued parameters such as storm location are nevertheless available as inputs for prediction because operationally SHIPS uses parameter values from dynamical forecast models or National Hurricane Center (NHC) official forecasts to fill in these values. It is understood by all that the intensity predictions provided by SHIPS are only as good as the forecast track provided to SHIPS as input. Furthermore, the performance of SHIPS as an intensity predictor is judged based on the assumption that its "estimated" future-valued input parameters are perfect, and so this study follows the example of SHIPS in using post-season analysis values of these parameters as inputs to the study for training and testing purposes.

An individual TC will be described by a series of instances in the SHIPS data, beginning with an instance whose initial time corresponds to TC formation and continuing in 6-h intervals until TC dissipation or extratropical transition. For example if a TC persists for 4 days then it would be represented by 16 instances with initial times separated by 6 h. If a TC instance describes a system that dissipated in less than 120 h from the instance's initial time, then data values subsequent to its dissipation time would be set to the *missing* value. Similarly, instances which describe newly-formed TCs would have their –12-h and –6-h values set to *missing*.

One consequence of this data presentation is that although there are 9926 distinct instances that may be used for training and testing, there are as mentioned only 421 distinct TCs in the set and so on average there will be 24 instances that all relate to different time perspectives of the evolution of one tropical system. To take a recent, notable example, there are 32 instances from Hurricane Irene in the summer of 2011, and essentially the data from two neighboring instances (for example, time points 2011/08/23 1200 and 2011/08/23 1800) are largely the same except for a shifting of data values by 6 h to the left or right. This fact should be kept in mind when dividing the data into training and test sets.

The data types available from the SHIPS file are described in the following sections along with certain calculated fields, both input and target class attributes, that are added during preprocessing.

3.2 Raw Attributes

In the raw SHIPS data files alone, there are a multitude of attributes. Each instance in a SHIPS data file is essentially a 23 X 68 matrix of numbers and strings, made up of 68 record types providing 23 values each. Some records are used only for file formatting and provide no data of their own, and some contain series of data values with special meanings, but for the most part each record provides up to 23 data points of a time series that may begin 12 h before the initial time of the instance and continue at 6-h intervals up to 120 h following that initial time.

This study recognizes 1351 distinct attribute values provided in the SHIPS data files and calculates 383 additional attributes from these "raw" values in the files. The raw SHIPS attributes may be divided into three semantic categories: those records whose values run from –12 h to +120 h for up to 23 values apiece, those records whose values run only from the initial time forwards for up to 21 values apiece, and those records with values in special formats. There are 11 record types in the first category, 49 in the second category, and 67 individual attributes that may be extracted from the third category.

All attributes that come from time series records in the SHIPS file are identified by a combination of their 3- or 4-character identifier and the time point they are associated with. Thus, 'VMAX –12' is the maximum sustained surface wind speed (i.e., intensity) of a TC at a time 12 h before the instance's initial time, and 'MSLP 108' refers to the minimum sea level pressure of a TC at a time 108 h following the initial time.

3.2.1 SHIPS Attributes with Past, Present and Future Values

This category of attributes includes five record types that relate directly or indirectly to perfect knowledge of a TC's future intensity changes. These are the first five attribute types listed in Table 3.1. For an exercise in predicting TC intensity changes, these attributes would generally be withheld from the analysis or used as target class attributes (themselves, or other attributes calculated from their values).

In addition to these five there are six more attribute types listed in Table 3.1 that may have data values from 6 and 12 h prior to the initial time. As noted, it is assumed that the future values of these six attributes may be known at the initial time even operationally because they are supplied by global or regional models or official NHC forecasts.

Note that the first seven attributes described in this section are taken directly or adapted from the NHC's release of *best track* data. These are post-season analyses that may have been adjusted to smooth the storm movements and intensities in a subjective manner or make use of information in hindsight that was not available operationally.

This study will refer interchangeably to TC "intensities" and TC "maximum surface wind speeds," with both expressed in knots, or nautical miles per hour (where 1 kt $\approx$ 0.51 ms^{-1} $\approx$ 1.85 kmh^{-1} $\approx$ 1.15 mph). Also worthy of mention is that the phrase "maximum surface wind speed" should be understood as a convenient shorthand for "maximum *sustained* surface wind speed."

3.2.2 SHIPS Attributes with Present and Future Values

Most of the attributes in the SHIPS files, some 49 record types, fall into this category. These attributes may have values populated at the initial time and in the future but not in the past. They are described in Table 3.2.

Table 3.1: SHIPS attributes with past, present and future values (adapted from the SHIPS predictor description file [4]).

Type	Description	Units
VMAX	Maximum surface wind speed	kt
MSLP	Minimum sea level pressure	mb
TYPE	Storm type (0=wave, remnant low, dissipating low, 1=tropical, 2=subtropical, 3=extra-tropical)	–
DELV	Intensity change relative to initial time	kt
INCV	Intensity change relative to preceding timepoint	kt
LAT	Latitude of the storm center	$^\circ$ N
LON	Longitude of the storm center	$^\circ$ W
CSST	Climatological Sea Surface Temperature	$^\circ$C
DTL	Distance to nearest major land mass	km
RSST	Reynolds Sea Surface Temperature	$^\circ$C
PHCN	Estimated ocean heat content	kJcm^{-2}

Table 3.2: SHIPS attributes with present and future values (adapted from the SHIPS predictor description file [4]).

Type	Description	Units
U200	200 mb zonal wind speed (200–800-km average)	kt
U20C	200 mb zonal wind speed (0–500-km average)	kt
V20C	200 mb meridional wind speed (0–500-km average)	kt
E000	1000 mb θ_e (Equivalent Potential Temperature) (200–800-km average)	K
EPOS	Average θ_e difference between a parcel lifted from the surface and its environment (200–800-km average); only positive differences are included in the average	K
ENEG	Same as EPOS, but only negative differences are included	K
EPSS	Same as EPOS, but the parcel θ_e is compared with the saturated θ_e of the environment	K
ENSS	Same as ENEG, but the parcel θ_e is compared with the saturated θ_e of the environment	K
RHLO	850–700 mb relative humidity	%
RHMD	700–500 mb relative humidity	%
RHHI	500–300 mb relative humidity	%
Z850	850 mb vorticity (0–1000-km average)	s^{-1}
D200	200 mb divergence (0–1000-km average)	s^{-1}
REFC	Relative eddy momentum flux convergence (100–600-km average)	ms^{-1}day^{-1}
PEFC	Planetary eddy momentum flux convergence (100–600-km average)	ms^{-1}day^{-1}

Table 3.2: (continued)

Type	Description	Units
T000	1000 mb temperature (200–800-km average)	°C
R000	1000 mb relative humidity (200–800-km average)	%
Z000	1000 mb height deviation from the U.S. standard atmosphere	m
TLAT	Latitude of 850 mb vortex center in National Centers for Environmental Prediction (NCEP) analysis	° N
TLON	Longitude of 850 mb vortex center in NCEP analysis	° W
TWAC	0–600-km average symmetric tangential wind speed at 850 mb from NCEP analysis	ms^{-1}
TWXC	Maximum symmetric tangential wind speed at 850 mb from NCEP analysis	ms^{-1}
V000	Tangential wind speed at 1000 mb azimuthally averaged at 500 km from TLAT, TLON; if TLAT, TLON are not available then LAT, LON are used	ms^{-1}
V850	Same as V000 at 850 mb	ms^{-1}
V500	Same as V000 at 500 mb	ms^{-1}
V300	Same as V000 at 300 mb	ms^{-1}
TGRD	Magnitude of the temperature gradient between 850 and 700 mb averaged from 0 to 500 km estimated from the geostrophic thermal wind	$°Cm^{-1}$
TADV	The temperature advection between 850 and 700 mb averaged from 0 to 500 km from the geostrophic thermal wind	$°Cs^{-1}$
PENC	Azimuthally averaged surface pressure at outer edge of vortex	mb
SHDC	850–200 mb shear magnitude, with vortex removed and averaged from 0–500 km relative to 850 mb vortex center	kt
SDDC	Heading of SHDC shear vector	°
SHGC	Generalized 850–200 mb shear magnitude (takes into account all levels) with vortex removed and averaged from 0–500 km relative to 850 mb vortex center	kt
DIVC	200 mb divergence (0–1000-km average) centered at 850 mb vortex location	s^{-1}
T150	200–800 km area average 150 mb temperature	°C
T200	200–800 km area average 200 mb temperature	°C
T250	200–800 km area average 250 mb temperature	°C
SHRD	850–200 mb shear magnitude (200–800 km)	kt
SHTD	Heading of SHRD shear vector	°
SHRS	850–500 mb shear magnitude	kt
SHTS	Heading of SHTS shear vector	°
SHRG	Generalized 850–200 mb shear magnitude (takes into account all levels)	kt

Table 3.2: (continued)

Type	Description	Units
PENV	200–800-km average surface pressure	mb
VMPI	Maximum potential intensity, from Kerry Emanuel equation [44]	kt
VVAV	Average (0–15 km) vertical velocity of a parcel lifted from the surface where entrainment, the ice phase and the condensate weight are accounted for	ms^{-1}
VMFX	Same as VVAV but a density weighted vertical average	ms^{-1}
VVAC	Same as VVAV but with soundings from 0–500 km with GFS vortex removed	ms^{-1}
RD20	Ocean depth of the 20 °C isotherm from satellite altimetry data	m
RD26	Ocean depth of the 26 °C isotherm from satellite altimetry data	m
RHCN	Ocean heat content from satellite altimetry data	$kJcm^{-2}$

3.2.3 Other SHIPS Attributes

The HEAD record type appears first in each SHIPS data record. It contains certain identifying and contextual information about the TC in general and this instance in particular. These fields, described in Table 3.3, include information about the system's assigned name if there was one, its storm basin, year and order of formation in that year, and initial time.

Table 3.3: SHIPS attributes from the HEAD record type (adapted from the SHIPS predictor description file [4]).

Name	Description
CASE	Unique identification string for this instance, consisting of NAME and initial time values below
NAME	Name or designation given to the TC
INYR	Year of TC initial Time
INMN	Month of TC initial Time
INDY	Day of Month of TC initial Time
INHR	Hour of TC initial Time
BASN	TC Basin (Atlantic, East Pacific, Central Pacific)
CYNM	Cyclone Number, counting from 1 at the beginning of each year
CYYR	Cyclone Year of TC formation, may differ from INYR

The 17 attributes described in Table 3.4 relate to Geostationary Operational Environmental Satellites (GOES) data values. Each instance may have one or two sets of these 17 attributes provided (labeled IRM3 and IR00 in the SHIPS files) and their timing is defined by the offset from the initial time specified in the first field. A third SHIPS record called IRXX provides non-satellite-sourced versions of these value in some cases where the satellite data are not available.

Table 3.4: SHIPS attributes related to Geostationary Operational Environmental Satellites (GOES) data values (adapted from the SHIPS predictor description file [4]).

Type	Description	Units
IRTM	Time of the GOES image, relative to the initial time	hr
IRT0	Average GOES Channel 4 Brightness Temperature (BT) (0–200-km average)	°C
IRD0	Standard Deviation of GOES BT (0–200-km average)	°C
IRT1	Average GOES BT (100–300-km average)	°C
IRD0	Standard Deviation of GOES BT (100–300-km average)	°C
IRP1	Percent area of GOES BT < –10 °C (50–200-km average)	%
IRP2	Percent area of GOES BT < –20 °C (50–200-km average)	%
IRP3	Percent area of GOES BT < –30 °C (50–200-km average)	%
IRP4	Percent area of GOES BT < –40 °C (50–200-km average)	%
IRP5	Percent area of GOES BT < –50 °C (50–200-km average)	%
IRP6	Percent area of GOES BT < –60 °C (50–200-km average)	%
IRX1	Maximum BT from 0 to 30 km radius	°C
IRXA	Average BT from 0 to 30 km radius	°C
IRXR	Radius of maximum BT	km
IRN2	Minimum BT from 20 to 120 km radius	°C
IRNA	Average BT from 20 to 120 km radius	°C
IRNR	Radius of minimum BT	km

Two more record types in the SHIPS files provide a handful of attributes that have not yet been mentioned. The first of these, the steering layer pressure, is a single data value per instance that relates only to the initial time. The other record type, HIST, provides the only TC history that may go back further than 12 h before the initial time. These 21 values count the number of 6-h time points in the TC's entire history for which the system's intensity (VMAX) was above a certain measure, from 20 kt through 120 kt in 5-kt increments.

By their very nature these counts are non-increasing as their intensity levels increase (for example, 'HIST 45' must be greater than or equal to 'HIST 50'). These last raw SHIPS attributes are described in Table 3.5.

Table 3.5: Other SHIPS attributes: PSLV and HIST (adapted from the SHIPS predictor description file [4]).

Type	Description	Units
PSLV	Steering Layer Pressure, or pressure of the center of mass of the layer where storm motion best matches environmental flow	mb
HIST 20	Number of 6-h time points to date when the TC intensity has been above 20 kt	–
HIST 25	Number of 6-h time points to date when the TC intensity has been above 25 kt	–
⋮	⋮	⋮
HIST 120	Number of 6-h time points to date when the TC intensity has been above 120 kt	–

3.3 Calculated Attributes

3.3.1 Calculated Input Attributes

Additional input attributes have been added to this analysis, following the lead of [16]. These attributes are calculated from raw SHIPS attributes using simple formulas (Table 3.6). JDAY and JDTE relate to the instance's initial time and each contributes one attribute to the analysis. The storm displacement attributes (CSM, USM and VSM) are calculated as great-circle distances between successive pairs of LAT and LON attributes. As such, they are populated from –6 h to 120 h but not at –12 h because the data record does not contain the TC location information at –18 h that would be required for this calculation. Four more attributes (POT, POT2, LSHR and VSHR) are populated from 0 h through 120 h because they are based on inputs that are not populated at times before the initial time. A final attribute (VINC) may be fully populated from –12 h through 120 h if the TC was active

throughout the entire range of times. Details of each of these calculations are given in Section 4.1.2.

Table 3.6: Calculated input attributes.

Type	Description	Units
JDAY	Absolute value of the Julian Date minus the Peak Date[4] of the season	–
JDTE	Gaussian function of the Julian Date minus the Peak Date[4] of the season	–
CSM	Storm displacement in previous 6 h	km
USM	Zonal component of storm displacement in previous 6 h	km
VSM	Meridional component of storm displacement in previous 6 h	km
POT	Intensification Potential, equal to VMPI minus VMAX	kt
POT2	The square of POT	kt^2
LSHR	Equal to SHRD times the sine of LAT	kt
VSHR	Equal to VMAX times SHRD	kt^2
VINC	Equal to VMAX times INCV	kt^2

3.3.2 Calculated Target Class Attributes

This last set of Rapid Intensification/Weakening attributes is calculated from present and future values of TC intensity (VMAX). These are the main target class attributes of the present study. By their nature, these attributes define successive subsets of instances since it is trivially true that an instance which intensifies by (for example) 40 kt in the coming 24 h will also intensity by at least 35 kt, 30 kt and 25 kt, and the same is true for rapid weakening. The calculation of these attributes is discussed in more detail in Section 4.1.3.

[4]The Peak Date is day 253 in the Atlantic basin, and day 238 in the east Pacific basin [16].

Table 3.7: Calculated target class (rapid intensification/weakening) attributes.

Type	Description	Units
RINT	Signed difference in TC intensity between 24 h and 0 h	kt
RI25	True if TC intensity at 24 h exceeds that at 0 h by 25 kt	–
RI30	True if TC intensity at 24 h exceeds that at 0 h by 30 kt	–
RI35	True if TC intensity at 24 h exceeds that at 0 h by 35 kt	–
RI40	True if TC intensity at 24 h exceeds that at 0 h by 40 kt	–
RW25	True if TC intensity at 24 h falls short of that at 0 h by 25 kt	–
RW30	True if TC intensity at 24 h falls short of that at 0 h by 30 kt	–
RW35	True if TC intensity at 24 h falls short of that at 0 h by 35 kt	–
RW40	True if TC intensity at 24 h falls short of that at 0 h by 40 kt	–

Chapter 4

Data Preprocessing

As described in Chapter 3, the SHIPS developmental data are made available in their own distinctive data format. Each instance in the data files is essentially a 23 X 68 matrix of values that can be unpacked into 1351 distinct attributes. The present study concerns only the data from the Atlantic hurricane basin, so the raw data set consists of 9926 instances described by these 1351 attributes. This chapter describes the process of analyzing and preparing the SHIPS data files for association rule mining and subsequent classification.

This process begins with the examination of the SHIPS data file on an instance by instance basis to understand its internal logic and to verify that it obeys its own rules. In some cases, data inconsistencies must be corrected. Next, certain calculations are performed to create new attributes that are simple mathematical combinations of the raw attributes. The complete data set is then written back to disk in the Attribute-Relation File Format (ARFF) [45] for easier manipulation by the Waikato Environment for Knowledge Analysis (WEKA)[5].

Once the data are in ARFF, the data set must be filtered and edited in order to run tests with this study's associative classifier as well as other comparison classifiers from the WEKA collection of algorithms. Useful and appropriate subsets of instances and attributes are identified and retained, and numeric attributes are discretized as necessary. Any filter

[5]WEKA was introduced and described in more detail in Section 2.2.

that makes use of target class information must be applied using only data from the training set and must not be contaminated with knowledge of class assignments in the test set.

Care must be taken to eliminate any "spoiler" attributes from the data set prior to the analysis. Spoiler attributes are those attributes that would be unavailable operationally and which may provide direct or indirect information about the target class attributes. For example, any of the RI or RW target class attributes could be calculated by subtracting 'VMAX 0' from 'VMAX 24'. 'VMAX 0', the TC's intensity at 0 h, is operationally observable whereas 'VMAX 6', 'VMAX 12' and so on through 'VMAX 120' are all spoiler attributes.

When the ARFF data files are ready for analysis, they will be mined for class association rules using a customized Apriori-based association rule mining algorithm (Section 5.1) and those rules may then be applied for classification (Section 5.2).

4.1 Producing an ARFF Data File

4.1.1 Data Consistency Checks and Corrections

A variety of data consistency checks are performed when first reading in the SHIPS data file. Each instance is verified to begin with a HEAD record and end with a LAST record, and all SHIPS record types (68 types including HEAD and LAST) are verified to be present. The HEAD record repeats certain data fields from other records at the initial time – Maximum Surface Wind Speed, Minimum Sea-Level Pressure, Latitude and Longitude – and these are verified to be consistent and the duplicates then discarded. Year values are confirmed to be within data set limits, as are latitudes and longitudes.

TC intensities are measured in terms of the maximum sustained surface wind speeds (in knots). In NHC's *best track* post-season analyses of TC track and intensity data, intensities are nearly always rounded to the nearest 5 kt. A limited number of cases from the 1988 seasons (Hurricane Helene and Tropical Storm Keith in the Atlantic, with 38 and 14 cases

respectively, and Hurricane Iva in the East Pacific, with 22 cases) were not rounded in the best track in this manner, but these cases were rounded in preprocessing for this study.

In addition to VMAX, which is the maximum surface wind speed and the primary measure of a cyclone's intensity, the data set provides two calculated intensity *differences*, DELV and INCV. The first of these, DELV, is the difference between VMAX at the time in question and VMAX at the initial time (with the result that 'DELV 0' is always 0 kt by definition). The second of these attributes, INCV, gives a measure of the change of VMAX over the previous six hours. This means that all DELV and INCV parameters may be trivially calculated from other parameters in the data set (with the exception of 'INCV –12,' since it would require knowledge of 'VMAX –18'). However, one important note is that both DELV and INCV are set to *missing* whenever the TC is impacted by land in the timeframe covered by the variable, which means that these missing values can be used as indicators to identify times when a TC is interacting with land. All of these relationships between VMAX, DELV and INCV are verified in the consistency checks of the preprocessing routines. Table 4.1 shows the ranges of values for these attributes (as well as MSLP, or the Maximum Surface-Level Pressure) found in the SHIPS data files.

Table 4.1: Ranges of intensities/pressures in the SHIPS data set.

Type	Atlantic			East Pacific		
VMAX	10	to	160	10	to	160
DELV	–115	to	135	–130	to	135
INCV	–35	to	55	–40	to	45
MSLP	882	to	1022	902	to	1017

In addition to the intra-instance consistency checks described thus far, SHIPS instances are also compared one to the next for consistency. All instances relating to a single TC are presented contiguously, ordered chronologically, with the cyclones sorted by their order of naming within each season (with some exceptions for unnamed storms sorted to the beginning or end of the season). Since each individual instance may describe a TC's evolution

from 12 h before to 120 h after the instance's initial time, there is considerable overlap with neighboring instances whose initial times are separated by 6 h. The *series* of instances relating to one TC describe the evolution of that cyclone and they can be expected to follow a certain logic related to the beginning and end of the series, the continuity of the series, and data consistency between adjacent instances in the series.

The SHIPS data files use "9999" as a *missing* value, which is translated to "?" when producing the WEKA-compatible ARFF files. Every instance may contain values 12 h or 6 h before its initial time, but these values may be set to "9999" if the instance describes a recently formed TC. Similarly, each instance may contain values relating to times up to 120 h after its initial time but the later values may be set to "9999" if the TC dissipates (or is no longer tropical) prior to reaching the 120-h mark. Thus it is possible to define a start-of-series instance which should have no past-valued data and an end-of-series instance which should have no future-valued data. With the exception of four TCs in the Central Pacific basin which end one instance sooner than expected, all TCs in the SHIPS data sets begin with such a start-of-series instance and end with an end-of-series instance. Since the SHIPS data set only provides data for tropical or subtropical systems, it is also possible for a series of instances to be interrupted temporarily if its TC goes extratropical or dissipates but later reforms as a (sub)tropical system.

Given these definitions of series beginning, ending and interruption, it is possible to compare in a detailed way the attribute values from one instance to the next. According to such a comparison, the attribute types can be separated into categories of those which remain consistent from case to case and those which do not, necessarily. A third category is composed of those attribute types which are uniquely associated with each instance's initial time or some fixed time relative to the initial time, and also DELV, which by definition is calculated relative to each instance's 'VMAX 0' value. These categories of SHIPS record types are detailed in Table 4.2.

Table 4.2: Consistency of SHIPS record types between neighboring instances.

Record Types	Instance-to-Instance Behavior
D200, DTL, E000, ENEG, ENSS, EPOS, EPSS, INCV, LAT, LON, MSLP, PEFC, PENV, R000, REFC, RHHI, RHLO, RHMD, SHRD, SHRG, SHRS, SHTD, SHTS, T000, T150, T200, T250, TYPE, U200, VMAX, Z000, Z850	All values are consistent from one instance to the next relating to the same TC. REFC is *mostly* consistent, with only 19 inconsistencies (15 in the Atlantic, 4 in the East Pacific).
CSST, DIVC, PENC, PHCN, RD20, RD26, RHCN, RSST, SDDC, SHDC, SHGC, TADV, TGRD, TLAT, TLON, TWAC, TWXC, U20C, V000, V20C, V300, V500, V850, VMFX, VMPI, VVAC, VVAV	No particular consistency between neighboring instances.
DELV, HEAD, HIST, IR00, IRM3, IRXX, LAST, PSLV, TIME	Consistency checks are not meaningful.

The differences between those parameters with inter-instance consistency and those without are likely due to their different origins. Some, like those reproduced or calculated from the NHC best track analyses, are derived from point-by-point (time-by-time) assessments of the TC's evolution. For example, the value of 'VMAX 24' from a TC's first instance or 'VMAX 18' from that same TC's second instance would come from the same place, that cyclone's re-analyzed intensity 24 h into its lifetime. Other attributes, however, may be forward-looking values derived operationally from models valid at each instance's initial time. For example, the model estimate of 'CSST 24' from the first instance of a TC may be somewhat different from the output of the model running six hours later which produced the value of 'CSST 18' reported in the cyclone's second instance.

4.1.2 Adding Calculated Attributes (Input Attributes)

As discussed in Section 3.3.1, certain mathematical combinations of the SHIPS attributes are added to the data set during pre-processing to more clearly capture the physical realities that may be important for predicting large changes in TC intensity. For example, rapid intensification may be more closely correlated with a measure of how close the initial time is to the statistical peak of the hurricane season than to any individual timing parameters

(year, month, day, hour) that are present in the input set. Another example is that while a cyclone may be less likely to rapidly intensify if its intensity (VMAX) is quite large, an even better predictor might be the difference between its intensity and its estimated maximum potential intensity (VMPI). All of the calculations discussed in this section have been part of the SHIPS linear regression analysis and are calculated based on descriptions in [16].

The first two calculated parameters, JDAY and JDTE, express the nearness in time of the Julian Day (J_d, or day of year) of the instance's initial time and the statistical peak of the hurricane season (P_d, which is day 253 in the Atlantic basin and day 238 in the East Pacific basin). The first attribute, JDAY, is simply the absolute value of this time difference in days (Equation 4.1). The second attribute, JDTE, is a Gaussian function of this time difference intended to avoid the drawback of JDAY, which overly penalizes very early and very late storms. This Gaussian JDTE varies from zero to one; it is close to zero for TCs distant in time from P_d and equal to one for TCs whose initial time falls exactly on P_d (Equation 4.2). A time-scale factor of $R_d = 25$ days is adopted following the example of [16].

$$JDAY = |J_d - P_d| \tag{4.1}$$

$$JDTE = e^{-[(J_d - P_d)/R_d]^2} \tag{4.2}$$

The next three calculated attributes are intended to differentiate between quickly-moving storms and slowly-moving storms, both in absolute terms (CSM) and in terms of zonal (USM) and meridional (VSM) motion components. These attributes are all measures of a TC's displacement which are calculated according to the great-circle distances between the TC's current position (according to its LAT and LON attributes) and its position 6 h earlier. They are expressed in kilometers, based on an average Earth radius of 6371 km. The values of these three attributes at time –12 h are set to 9999 because calculation of these values would require knowledge of the TC's position at –18 h. Calculation of the

zonal displacement is based on the midpoint of latitude between positions and calculation of the meridional displacement is independent of longitude.

The next two calculated attributes are expressions of a TC's potential for intensification. The first attribute, POT (Equation 4.3), is the signed difference between the TC's estimated maximum potential intensity and its current intensity, while the second attribute, POT2 (Equation 4.4), is essentially the square of POT. Note, however, that certain instances may in fact have a current intensity that exceeds its estimated maximum potential intensity, so POT2 is calculated in a manner intended to produce a signed result with more negative values corresponding to TCs which further exceed their estimated maximum potential intensities. These two attributes are undefined at –12 h and –6 h because VMPI is not provided for those times.

$$POT = VMPI - VMAX \tag{4.3}$$

$$POT2 = (VMPI - VMAX) \cdot |VMPI - VMAX| \tag{4.4}$$

The final three calculated attributes are mathematical combinations of TC intensity (VMAX), 6-h intensity change (INCV), shear (SHRD), and latitude (LAT). They are VINC, LSHR and VSHR, and are as detailed in Equations 4.5, 4.6 and 4.7. The calculation of VINC depends on inputs that are available at all times including –12 h and –6 h, and the other two are calculable beginning at 0 h.

$$VINC = VMAX \cdot INCV \tag{4.5}$$

$$LSHR = SHRD \cdot \sin(LAT) \tag{4.6}$$

$$VSHR = VMAX \cdot SHRD \tag{4.7}$$

4.1.3 Adding Calculated Attributes (Target Class Attributes)

The attributes targeted for prediction in this study are determined based on a TC's change in intensity over a 24-h period. The first, RINT, is the difference between the TC's current intensity (VMAX) and its intensity 24 h later. For example, 'RINT 48' would be equal

to ‘VMAX 72’ minus ‘VMAX 48’. These attributes, RINT and the eight Boolean-valued attributes calculated from it (described below), may be populated starting at 0 h and continuing no later than 96 h (since calculating their values at 102 h and later would require knowledge of VMAX at 126 h and later).

The present study concerns itself only with 24-h periods of rapid intensity changes beginning at the initial time. Following the example of SHIPS as detailed in [46], there are three conditions required to be met for an instance to be included in this analysis, and these conditions are applied from the time 12 h before the initial time to 24 h afterwards. The conditions are: (1) the TC must exist and have non-missing VMAX; (2) the TYPE of TC must be tropical (and not subtropical or extratropical); and (3) the TC must not have had any interactions with land throughout the 36-h period, as determined by its non-missing INCV values (as discussed in Section 4.1.1). A trivial consequence of these conditions is that the first two instances of every TC are discarded since they will have missing VMAX values at –12 h, and the last three instances will similarly be discarded since they will be missing VMAX at 24 h. The combined effect of these three conditions is to reduce the number of instances under consideration from 9926 to 5956, or to 60% of the data set’s original size, although at this point of the analysis no instances are removed and instead the target class attributes are merely set to the missing value.

Having calculated the RINT values as described, from 0 h through 96 h, it is a simple matter to calculate the other eight Boolean-valued target class attributes. Note first that the RINT values are themselves rounded to the nearest 5 kt, since they are calculated as the differences between two intensity measures that are rounded to the nearest 5 kt. There are four target class attributes related to Rapid Intensification (RI) and four related to Rapid Weakening (RW); each of the four is set according to the degree of intensification or weakening in the instance, whether it is greater than or equal to 25, 30, 35 or 40 kt. These target class attributes are named as described in Table 3.7.

It is important to note that there exist certain logical relationships among these parameters by virtue of the manner in which they are defined. If an instance under consideration describes a TC that will intensify by at least 40 kt over the next 24 h, then it is trivially true that it will intensify by at least 35, 30 and 25 kt as well. This applies also to the negative cases (if a TC does *not* intensify by at least 25 kt in 24 h, it can neither have intensified by greater amounts) and to the rapid weakening cases as well. In fact, all of the implications in Equations 4.8 through 4.11 are true.

$$RI40 = 1 \quad \Rightarrow \quad RI35 = 1 \quad \Rightarrow \quad RI30 = 1 \quad \Rightarrow \quad RI25 = 1 \tag{4.8}$$

$$RI25 = 0 \quad \Rightarrow \quad RI30 = 0 \quad \Rightarrow \quad RI35 = 0 \quad \Rightarrow \quad RI40 = 0 \tag{4.9}$$

$$RW40 = 1 \quad \Rightarrow \quad RW35 = 1 \quad \Rightarrow \quad RW30 = 1 \quad \Rightarrow \quad RW25 = 1 \tag{4.10}$$

$$RW25 = 0 \quad \Rightarrow \quad RW30 = 0 \quad \Rightarrow \quad RW35 = 0 \quad \Rightarrow \quad RW40 = 0 \tag{4.11}$$

4.1.4 Formatting as ARFF

Thus far the SHIPS data files have been unpacked from their format, analyzed for internal consistency, and augmented by certain calculated attributes. The end result of this process is an Attribute-Relation File Format (ARFF) data file suitable for input to WEKA. This is a simple format where data values are separated by commas with each instance written out to one line in the file. The top of the file contains a section which defines the names and types of the data fields.

ARFF recognizes several types of attributes, among which are nominal, numeric and string[6]. A nominal attribute can take any one of a finite number of discrete values, all of which must be enumerated in the header. Examples of nominal enumerations might include "{black, green, orange}," "{cold, warm, hot}" or "{1, 2, 3}." Note that WEKA recognizes no *ordering* of nominal attributes so it cannot tell that "hot" and "cold" are quite different from one another but "warm" is closer to both of them, or that "1" and "2" are exactly as far

[6] ARFF also includes support for date and relational attribute types that are not relevant to this study.

apart from one another as "2" and "3." WEKA numeric attributes can accept any numeric value, integer or real, and need not be enumerated in the header. Some classifiers (including the associative classifier described herewith) cannot make use of numeric attributes so they must be *discretized* (see Section 4.2.5 for a detailed discussion) before processing can continue.

The present study uses two string-valued attributes for identifying the specific instance as well as the TC series to which it belongs. A total of 209 other attributes are defined as nominal at this stage. These include the basin (AL for Atlantic, EP for East Pacific and CP for Central Pacific), cyclone number (counting from 1 up to 31 TCs per year), and all TYPE attributes, each of which has only four possible values (wave/low, tropical, subtropical and extratropical). Eight of the calculated Rapid Intensification/Weakening attributes are also nominal, given that their possible values are 1 and 0. The remaining 1455 attributes are numeric. The value "9999" which in the SHIPS files indicates a missing value is replaced with "?" to indicate an ARFF missing value.

Having read, analyzed, corrected, augmented and written the SHIPS data files, the next step will be to start working with these data, to reduce the numbers of both instances and attributes under consideration to manageable levels, and to divide the data into training and test sets for evaluating the performance of the associative classifier relative to other algorithms.

4.2 From ARFF to ARM

At this stage of processing, there is an ARFF input file and an algorithm that can read ARFF files. However, input files with too many attributes and/or instances can cause the Apriori-based association rule mining algorithm to quickly exhaust reasonable limits of computer memory and run time. Furthermore, the targets of prediction in this study are very rare events that will not appear in the frequent itemsets found by Apriori.

The first of these issues is addressed by this second phase of data preprocessing, which begins with the ARFF input file and ends with training and test files suitable for input to an Association Rule Mining (ARM) algorithm. The problem of mining for rare events is addressed by structural changes to the Apriori-based algorithm and will be described in Section 5.1.

The steps described in this section are ordered from most general to most specific. The first step as described in Section 4.2.1 can be applied to the entire data set regardless of whether the target of prediction is a numeric estimate of TC intensity or the degree of rapid intensification or weakening, and regardless of whether the target is the immediate evolution of the TC or its intensity at some point in the more distant future. As these steps grow more specific, their output will depend more and more specifically on the targeted time of prediction, the specific prediction attribute chosen, and on which repetition of several cross-validation runs is being performed.

4.2.1 Filtering the Best Track Attributes

As described in Section 3.2.1, seven of the SHIPS attributes are taken from the NHC *best track* post-season analyses. These are the three TC intensity attributes (VMAX, DELV and INCV), the minimum sea-level pressure (MSLP), storm type (TYPE), and storm position (LAT and LON). The point of SHIPS (and the present study) is to answer the question: given perfect knowledge of a TC's future track, how well can its future intensity (and chance of rapid intensification or weakening) be predicted? Therefore it is acceptable for the study to have access to future values of LAT and LON even if these cannot possibly be known at prediction time. It is also acceptable to make use of current and past values for all other parameters on the assumption that these are observable in real time. However, all future-valued best track parameters other than LAT and LON must be removed from the analysis. Additionally, any future-valued attributes that were calculated from these

removed attributes must themselves be removed at this time. This affects POT, POT2, VINC and VSHR attributes.

At this time, all RINT and RI/RW target class attributes defined at –12 h and –6 h are removed, since by definition they are never populated. The nominal 'CYCLONE NUM' attribute is also removed at this time, for a total of 199 attributes removed, leaving 1535 attributes and 9926 instances in the analysis.

4.2.2 The Optional Stratification Step

Some experiments with association rule mining and the SHIPS data set, for instance [43], take the initial step of dividing instances into smaller groups based on the initial intensity of the represented cyclone. The reasoning is that the initial intensity is known operationally and that a TC may evolve more similarly to a TC of similar initial intensity than to one significantly stronger or weaker. There are many ways to approach this stratification of instances but most involve separating tropical depressions and tropical storms out from hurricanes of different ratings (categories 1 through 5) on the Saffir-Simpson scale [47]. Care must be taken with the granularity of separation of the stronger storms because there are fewer high-intensity TCs and the groupings should not become too unequal.

The present study investigated the effect of dividing its input into four categories: tropical depressions, tropical storms, hurricanes and major hurricanes. Instances are assigned to one of these four categories based on their intensity at the initial time ('VMAX 0'); hurricanes and major hurricanes are distinguished according to their Saffir-Simpson categories, with categories 1 and 2 considered to be hurricanes and categories 3, 4 and 5 considered to be major hurricanes. Some statistics about the instance counts and percentages of these categories for the SHIPS Atlantic data set are given in Table 4.3, including the counts/percentages of instances remaining after discarding (1) instances with missing RI/RW values and (2) instances with initial times other than 0 h (see discussion of the serial correlation problem in Section 4.2.3, below). However, experimentation did not sug-

gest any clear advantage to this stratification step when building an associative classifier. Therefore, this kind of stratification was not done in the present study.

Table 4.3: Counts of instances by initial intensity in the SHIPS Atlantic file. Also shown are counts/percentages of instances whose RI/RW attributes are populated, and counts/percentages of those instances with initial times at 0 h.

Label	Intensity	Total Count	RI/RW Populated		0-h Initial Time	
			Count	Percent	Count	Percent
Tropical Depression	0–34 kt	2699	883	33%	222	8%
Tropical Storm	35–63 kt	4315	2738	63%	658	15%
Hurricane	64–95 kt	2168	1715	79%	428	20%
Major Hurricane	96+ kt	744	620	83%	160	22%
Total	–	9926	5956	60%	1468	15%

4.2.3 Missing Target Class Attributes and Serial Correlations

Section 4.1.3 discussed the conditions that an instance must meet in order to be included in this analysis: continued TC existence, remaining tropical in nature, and no impact from interactions with land. If one or more of these conditions was not met then the analysis sets the target class attributes to *missing*. At the present stage of pre-processing these instances are removed from the analysis.

It has also been noted that SHIPS data instances are not independent of one another, but that "neighboring" instances (two instances from the same TC series separated only by 6 h) are substantially similar. If special care is not taken when dividing cases into training and test sets, then a classifier might be given unfair or unreasonable insight into some of the test instances through exposure to their close neighbors in the training set. A simple workaround for this problem, assuming an overabundance of test cases, is to ensure that all instances whether in the training set or the test set are separated by at least 24 h, by discarding all instances except those whose initial time occurs at 0 h of the date in question (see Appendix B of [48] for a discussion of the Serial Correlation problem and this approach to overcoming it). This next step, therefore, discards approximately 75% of the data set, namely those cases whose initial times occur at 6 h, 12 h or 18 h. A

separate experiment was conducted where instances from all initial times were retained but all instances from a single year were grouped together in either the training set or the test set. This experiment's results (not shown) were not significantly better or worse than those achieved using only the 0-h instances.

At the conclusion of this step, there has been no change to the number or type of attributes but the number of (Atlantic) instances has been reduced from 9926 to 5956 (60% of the original count) by discarding those instances with missing target class attributes, and subsequently to 1468 instances (15% of the original count or 25% of the count with populated target class attributes) by removing all instances except those with initial times occurring at 0 h. Table 4.3 shows more detailed instance counts broken out by initial TC intensity groups.

4.2.4 Choosing a Target Time

The next step in pre-processing is to choose the target time. Arguably the most interesting question is whether or not a TC will rapidly intensify or weaken in the coming 24 h. However, it is also possible to ask, from the vantage point of the initial time, whether a TC will undergo rapid intensity changes during some later 24-h time period in the future. For example, one might target the 'RI25 48' attribute and ask whether a TC can be expected to intensify by 25 kt in the period between 48 h and 72 h from the initial time. The remainder of this study will concern itself only with the 0 h intensity change variables (i.e., the question of immediate intensification or weakening), but all procedures are constructed to be flexible enough to focus on any available target time.

Once a target time has been chosen, a major purge of attributes becomes possible, because it is assumed that data about the TC's disposition after the period in question are not helpful to the analysis. For example, when predicting whether a TC will intensify by at least 25 kt between 0 h and 24 h, the parameter values at 30 h and beyond are not likely

to be relevant. Also at this stage, the RI/RW attributes relating to any time other than the target time are discarded, whether before or after the target time.

The result of this processing step on the Atlantic SHIPS data set when targeting any one of the RI/RW attributes at 0 h is a reduction in the number of attributes from 1535 to 411, with the total number of instances holding steady at 1468.

4.2.5 Cross-Validation and Discretization

The Apriori-based algorithm used for mining association rules from the SHIPS data set requires that its input attributes all be nominal. Since most of the attributes are still numeric at this point, the next required step is discretization of those numeric variables. The approach chosen for discretization in this study is a *supervised* filter, which means that it makes use of target class information. This kind of algorithm must not be applied to the data set until it has been divided into training and test sets, because the target class attribute assignments in the test set must not influence the development of the classifier in any way.

Therefore the time has come to divide the input file into training and test sets. The present study performs five-fold cross-validation using the weka.filters.supervised.instance.-StratifiedRemoveFolds[7] method. This means that the input set is divided randomly into five subsets of approximately equal size and approximately equal distribution of positive and negative cases, and then the algorithm is tested five times with each of the five subsets being held back once as a test set while the other four subsets are combined to form the training set. A new random seed is provided to this method for each run of the experiment, with that seed being a number between 0 and 65535 that is chosen using the irand() function from the Perl Math::Random::Secure module.

Each experiment is run ten times with a different random seed each time. With eight different target class attributes, ten experimental runs, and five-fold cross-validation, this means that 400 classifiers are built and tested during this experiment.

[7] Author: Eibe Frank (eibe@cs.waikato.ac.nz); Version: Revision 5492.

Discretization is done separately for negative and positive rule mining. This approach was selected in case the supervised discretization algorithm selected threshold values for negative rule mining that are suboptimal when considering the reduced-size inputs for positive rule mining. However, it turned out that the algorithm chosen for supervised discretization during negative rule mining was doing a poor job during positive rule mining, because it was turning too many variables into single-valued nominal attributes which were useless for mining. Therefore a different discretizer was chosen for positive rule mining.

Discretization for negative rule mining is carried out by means of the weka.filters.supervised.attribute.Discretize[8] method. This approach makes use of class assignments (in the training set only) using the Minimum Description Length (MDL) Principle described in [49]. As an example of the effect of discretization, in one training set chosen at random the values of attribute "VMAX 0" were grouped into "less than or equal to 67.5 kt" and "greater than 67.5 kt" nominal values.

Discretization for positive rule mining, on the other hand, uses a slightly different (and *unsupervised*) WEKA method called weka.filters.unsupervised.attribute.Discretize[9]. This approach divides all numeric attributes into two groups of approximately equal numbers, without making any use of class assignment information.

At the end of this step, there has been no further reduction in the total numbers of instances or attributes. However, the input files have been copied and divided into five sets of training and test sets and their numeric attributes in the training sets have been discretized leaving only nominal and string attributes. Note that the string attributes have been retained to this point in pre-processing as identifiers in the test sets so that individual results of particular interest may be easily traced back to their origin.

[8] Authors: Len Trigg (trigg@cs.waikato.ac.nz), Eibe Frank (eibe@cs.waikato.ac.nz); Version: Revision 6564.

[9] Authors: Len Trigg (trigg@cs.waikato.ac.nz), Eibe Frank (eibe@cs.waikato.ac.nz); Version: Revision 6567.

4.2.6 Final Reduction of Attribute Pool

The final step before association rule mining is to strip away attributes that may contain "spoiler" knowledge of the targeted attribute or those that are believed to add little benefit to the outcome.

First, one of the eight Rapid Intensification/Weakening attributes is selected as a classification target and then, because of the strong logical relationships among these attributes, the others (include the numeric RINT attribute) are stripped out of the input set. At the same time, the two remaining string-valued attributes are removed. This results in the removal of ten attributes. This step is applied to the training sets but also to a copy of the full data sets, in order to produce complete data files (one per targeted attribute) suitable for input to the other WEKA classifiers whose results are compared against this associative classifier.

Next, the weka.filters.unsupervised.attribute.RemoveUseless[10] method is applied to the training sets. This has the effect of stripping out constant-valued attributes (such as 'DELV 0', which is defined as the difference between 'VMAX 0' and itself and is therefore always equal to zero). It will also remove attributes that were discretized into a single bin, rendering them constant-valued as well. Depending on the effect of the chosen discretization algorithm, this may remove a large number of formerly-numeric attributes, presumably because the discretizer was not able to identify subranges of attributes with significant predictive value for the chosen target class attribute.

The final pre-processing step involves applying the weka.filters.supervised.attribute.-AttributeSelection[11] filter using the weka.attributeSelection.CfsSubsetEval[12] subset evaluator (as described in [50]) with the weka.attributeSelection.BestFirst[13] search method. The purpose of this step is to take a large group of attributes and extract an optimally predictive

[10] Author: Richard Kirkby (rkirkby@cs.waikato.ac.nz); Version: Revision 7468
[11] Author: Mark Hall (mhall@cs.waikato.ac.nz); Version: Revision 5987.
[12] Author: Mark Hall (mhall@cs.waikato.ac.nz); Version Revision 6132.
[13] Authors: Mark Hall (mhall@cs.waikato.ac.nz), Martin Guetlein; Version: Revision 1.29

subset of those attributes, to reduce the classifier's demands on time and memory. CfsSubsetEval was chosen as a subset evaluator because it not only searches for attributes that are highly correlated with the class attribute but it also avoids the redundant inclusion of multiple attributes that are highly correlated with one another.

The number of instances remaining at the end of all preprocessing steps is 1468, although these are divided into five different sets of training and test files for each full experimental run. The number of attributes remaining depends on how the random divisions into training and test sets were conducted, which in turn determines the effect of the Discretize, RemoveUseless and CfsSubsetEval methods. Given ten full experimental runs of five cross-validation splits apiece, the average number of attributes remaining after these steps is listed in Table 4.4.

Table 4.4: Average number of attributes remaining after Discretize, RemoveUseless and CfsSubsetEval methods are applied.

Mining Type	Target	Attributes
Negative	All RI	63.1
	All RW	13.9
Positive	RI25	18.0
	RI30	16.2
	RI35	19.9
	RI40	21.1
	RW25	20.3
	RW30	20.7
	RW35	18.5
	RW40	27.1

Since this is the final version of the data set before association rule mining begins, a closer examination of the targets would be illuminating, in order to define exactly how rare these events are. Table 4.5 lists the counts, out of the 1468 instances in the population at this point, of how many cases are positive for each of the target class attributes. It also lists percentages of positive instances for the entire population, and percentages of the higher-level targets that are positive relative to populations of the lower-level targets.

Table 4.5: Counts and percentages of rare events by target class attribute.

Target	Count	Percentage of...			
		Total	RI/RW25	RI/RW30	RI/RW35
RI25	164	11.1%	100.0%		
RI30	103	7.0%	62.8%	100.0%	
RI35	54	3.7%	32.9%	52.4%	100.0%
RI40	35	2.4%	21.3%	34.0%	64.8%
RW25	75	5.1%	100.0%		
RW30	46	2.9%	56.0%	100.0%	
RW35	18	1.2%	24.0%	42.9%	100.0%
RW40	6	0.4%	8.0%	14.3%	33.3%

Chapter 5

Associative Classification

An associative classifier is composed of two parts: a class-targeted form of Association Rule Mining (ARM) to produce Class Association Rules (CARs), and a strategy to build a classifier by selecting and ordering these rules and specifying a default class for those cases that aren't matched by any rule. These two parts will be discussed individually in Sections 5.1 and 5.2.

5.1 Association Rule Mining with Apriori

5.1.1 Apriori Basics

The Apriori algorithm [5] in its simplest form does not target any one attribute or class above any other. It simply looks for frequent associations among attribute values, and evaluates which combinations of attribute values have the best predictive value for determining the values of other attributes. Apriori's usefulness is in part due to its simplicity – it is easy to understand, explain and implement.

Apriori's design and terminology are most easily understood with reference to the example of a database of shopping cart purchases. Such a database would have one attribute for every possible *item* that could be purchased, and every instance would assign 1 or 0 to each attribute depending on whether the corresponding item was included in the instance, or purchase. Apriori's purpose is to investigate which items are most frequently purchased

together. These *frequent itemsets* can then be used to identify which sets of items have the best predictive value for determining whether (and which) additional items are likely to be included in the same purchase.

The two most important metrics for an Apriori-based ARM algorithm are *support* and *confidence*. The support of an itemset is equal to the proportion of the total number of instances which include that itemset. The confidence of a rule measures the proportion of instances, out of the subset that matches the rule's antecedent (its left-hand side, or "if" clause), that additionally matches the rule's consequent (its right-hand side, or "then" clause). In other words, the confidence of a rule expresses the proportion of instances matched by the rule for which the rule is found to be true.

Apriori begins by finding frequent itemsets whose support is greater than or equal to a minimum support value, which is a parameter for the algorithm. It makes use of the fact that an itemset's support cannot exceed the support of any of its subsets of items, including singleton itemsets. Apriori therefore starts out by looking for singleton itemsets that have the required minimum support, and then looks for larger itemsets by creating candidate itemsets from combinations of smaller ones and testing their support levels until such time as no larger itemsets can be found that have the required support. Association rules can be built from an itemset by dividing its items into two subsets, one for the rule's antecedent and one for its consequent. The algorithm examines all possible rules generated in this fashion and measures their confidence against a minimum confidence value, which is another parameter to the algorithm. The final output of the Apriori-based ARM algorithm is a collection of rules that satisfy these constraints of minimum itemset support and minimum rule confidence as defined by the algorithm's parameters.

Apriori-based algorithms as originally conceived do not produce rules with a targeted attribute (the *class* attribute) in the consequent. This goal is served by a modified version of the algorithm which is known as the Classification Based on Associations (CBA) Rule

Generator, or CBA-RG [8]. This version of Apriori limits its search to itemsets containing the targeted attribute and limits its rules to those whose consequents consist only of that targeted attribute.

5.1.2 Apriori Customizations

The present study uses as its baseline the WEKA implementation of CBA-RG in the method called weka.associations.Apriori[14]. Certain customizations were made to this method and to the related weka.associations.LabeledItemSet[15] method.

The most significant customization made to these WEKA Apriori methods is the introduction of a new parameter for restricting the method's resulting Class Association Rules (CARs) by the value of the targeted attribute. Simply put, this allows the method to be used to produce negative-only or positive-only rules. This is important for the present study because the positive and negative classes are extremely unbalanced, and without this additional parameter most attempts to use this method would produce only negative rules. As written, the original WEKA Apriori method begins looking for itemsets with a 95% minimum support and iteratively reduces the required support level by 5% until requirements for the minimum confidence and the number of rules can be met. Recognizing, however, that it is impossible to achieve a higher support for an itemset that includes the positive-valued target class attribute than the proportion of positive cases in the set, the customized algorithm instead begins with a minimum support requirement equal to the count of 95% of *positive* cases, and iteratively reduces the minimum support requirement by the count of 5% of *positive* cases until the requirements can be met.

The twin goals of this study are to make no assumptions about which data set attributes may be useful for predicting rapid TC intensity changes (i.e., keeping as many attributes in consideration as possible) while at the same time producing a result in a reasonable amount

[14] Authors: Eibe Frank (eibe@cs.waikato.ac.nz), Mark Hall (mhall@cs.waikato.ac.nz), Stefan Mutter (mutter@cs.waikato.ac.nz); Version: Revision 7476.

[15] Author: Stefan Mutter (mutter@cs.waikato.ac.nz); Version: Revision 1.5.

of time. Therefore this study defines a new *complexity* parameter intended to cut off the analysis when the minimum support requirement has reached a low enough level that the number of itemsets under consideration has grown to the point where the algorithm can be expected to run unreasonably long or exhaust computer memory resources or both. This complexity limit is implemented as a power of two and by trial and error its default has been chosen to be 2^{16}. If at any point in the analysis the number of itemsets under consideration exceeds this limit, then the minimum support level currently under consideration is judged to be *too low* and it is raised to its previously-analyzed level. Given infinite time and memory resources it might have been possible to find rules with higher confidence levels at the lower level of support but the present study chooses this way of deciding that a minimum support level is unacceptably low while at the same time retaining as many attributes in the analysis as possible. In practice this complexity limit is triggered about 47% of the time for negative rules and about 11% of the time for positive rules. The main reason that this happens more frequently with negative rules is the larger numbers of attributes and instances input to negative rule mining (38 attributes, 1174 instances on average) compared to the pared-down inputs for positive rule mining (20 attributes, 305 instances on average).

A third parameter added to the customized Apriori method is that of a minimum size for the itemsets used to produce the final rules. Following the example of [14], the present study has chosen to require that all rules be generated from itemsets containing at least six items, which would correspond to no fewer than five attributes in the antecedent and one (i.e., the target class attribute) in the consequent. The expectation is that larger itemsets may produce better rules since the risk of misclassification of the rule on unseen cases is reduced.

The basic operation of the customized algorithm is therefore as follows: positive rules and negative rules are mined separately in the manner described in Section 5.2. The algorithm begins with the aggressive requirement of a 99% minimum confidence for its rules

(this is higher than the default 90% minimum coded into the original WEKA method). It starts looking with a minimum support of 95% of the count of positive cases. If at any point the algorithm finds the required number of rules (the original WEKA default of 10 rules has been adjusted upwards to 200 rules) with 99% confidence then the algorithm terminates. Otherwise, it will lower the minimum support level by 5% of the count of positive cases and try again. This continues until such time as the minimum support level falls below its lower limit (which defaults to 30% of the count of positive cases) or the number of itemsets under consideration grows higher than the limit dictated by the maximum complexity argument. If either of these two conditions occurs then then algorithm backs off the previous value of minimal support and terminates. If at the point of early termination the minimum confidence can be relaxed from 99% to a level that will allow the algorithm to output the requested number of rules then this is done.

The end result is the mining of the required number of rules at as high a complexity level as possible with the minimum support level relaxed as far as necessary while still allowing full analysis of all itemsets at that support level and remaining above a user-supplied lower bound for the minimum support level.

5.2 Associative Classification

Once the WEKA Apriori algorithm has been customized to mine not just for Class Association Rules but for CARs targeted to a specific class value (positive or negative), it becomes a step in a larger process of training the associative classifier. This process begins with mining for negative rules for the lowest intensity-change targets, RI25 or RW25 (intensification or weakening at 25 kt), with the highest-possible confidence levels, followed by the pruning of this rule set.

Pruning of a rule set, whether positive or negative, is done as follows: the set of rules is first ordered from highest to lowest confidence level, and the rules are applied in turn to the same training set that was used to produce them. After applying each rule the performance

metrics (see Section 6.2) are calculated on the cumulative results so far. Once all rules have been applied it is determined at what point the best performance was achieved on the training set, and any rules beyond this point are discarded.

The goal of mining for negative rules is not an comprehensive and accurate separation of positive and negative cases; rather, at this stage the goal is to throw away the most easily identifiable negative cases in order to construct a less unbalanced training set to use when mining for positive rules. Any genuinely negative training instance matched by the pruned set of training rules is discarded at this point. Generally there are few if any positive instances that are falsely matched by the negative rules at this stage but if they exist, they are not removed from the training set that will be used to mine for positive rules. The filtered training set is then used to mine for positive rules, which are also pruned based on their performance on this reduced training set.

These pruned sets of negative and positive rules are applied in the same manner to a test set of instances that was held in reserve while training the classifier. First, the negative rules are applied and any test instance that matches any negative rule is assigned a label of negative. Next, the pruned set of positive rules is applied and again, any test instance (which has not already been assigned a negative label) that matches any positive rule is assigned a label of positive. Finally, any test instance that has not matched any rule, positive or negative, is assigned a default class of negative.

The primary advantage of this associative classifier over the standard classification algorithms available from WEKA is that its design is informed by the underlying relationships among the target class attributes (as listed in Table 3.7). Every one of the eight target class attributes under consideration is defined to be a threshold on a continuum of intensity change. A tropical cyclone is defined to have an initial intensity and is determined to have a certain intensity 24 h later. The difference between those two intensities (termed RINT in Table 3.7) may be zero, or may describe a weakening or intensifying TC, and these in-

tensity changes may be gradual or rapid. However, it is obvious that not all positive and negative instances are created equally. An instance that describes a weakening or a rapidly weakening TC is obviously "more negative" for the RI25 target than is an instance that was found to have strengthened by only 20 kt. An instance that was assigned a negative label by an RI25 classifier cannot consistently be assigned a positive label by RI30, RI35 or RI40 classifiers. It is in making use of this awareness, that all RI/RW target class attributes are merely thresholds of a single RINT variable, that the associative classifier can achieve results that are generally superior to and less biased than any other available classification algorithm.

The WEKA classifiers described in Chapter 6 must consider each target class attribute independently of any other. However, this associative classifier reuses the results of classification at the lower intensity-change levels by mining for negative rules only twice, once for weakening and once for intensifying, each at the 25-kt levels. For this reason the associative classifier has been termed AprioriGrad, since it takes a *graduated* approach to TC intensity-change classification.

Taking intensification as an example, the classifier first mines for negative and positive rules targeting RI25 as described above. Then, the full (unpruned) set of RI25 negative rules is re-evaluated on the same training set except that its degree of pruning is chosen based on its performance on the RI30 target class attribute. This means that the RI25, RI30, RI35 and RI40 negative rules are all the same except because of target-specific pruning they may retain more or fewer of the original 200 negative rules. Once a pruned set of RI30 negative rules has been selected, the mining of positive rules and the classification of test instances proceeds as described above. A similarly abbreviated training process is used to produce negative and positive rules for RI35 and RI40, and a similar process for weakening begins with training a classifier on the RW25 target before proceeding with RW30, RW35 and RW40.

The question of classifier bias is an important one. A classifier that is biased towards recall (probability of detection) is more likely to *overforecast*, that is, to return a higher number of false positives while missing fewer true events. A classifier that is biased towards precision (success ratio), on the other hand, will *underforecast*, or miss a higher number of positive events entirely while identifying with greater certainty those cases that it labels as positive. Particularly when working with rare (positive) events, an engineer would be inclined to favor a classifier that is biased towards recall, since in such cases it is trivially easy to achieve a high proportion of correct classifications merely by labeling all cases as negative. The more interesting engineering question is how to recall the most number of cases possible without an unacceptably high false alarm rate. A forecaster, on the other hand, is more likely to value a classifier that is biased towards precision. This is because an overabundance of false alarms produces a credibility problem and the classifier results are less likely to be trusted.

For purposes of this study the intention was to produce as evenly-biased a classifier as possible. This was accomplished by first identifying the most-easily classifiable negative cases and discarding them, thereby reducing the false alarm rate on the positive rules. The possibility of falsely discarding a positive case is minimized at this stage by requiring an extremely high confidence factor, even if such rules can only be produced by relaxing the minimum support to very low levels. When mining for negative rules, AprioriGrad looks for 200 rules with a minimum support of 30% and a minimum itemset size of 6. All negative rules resulting from this analysis have at least 95% confidence.

When mining for positive rules, the customized Apriori-based algorithm can be applied to a more evenly-balanced training set. It begins from scratch with the raw numeric inputs from the SHIPS data set, not the discretized versions of the data used in mining for negative rules. This is because the negative training sets are discretized with a supervised (class-informed) algorithm that chooses split points based on optimal separation of instances into

positive and negative classes. With so many negative instances now removed from the training sets, these split points may no longer be optimal. Therefore, for positive rule mining an unsupervised binarization algorithm is used to divide all numeric attributes into two discrete values with the split points chosen to separate the instances into approximately equal numbers. After that, the positive rules are mined in almost the same manner as were the negative rules – looking for 200 rules with minimum itemset size of 6 – except that with the more evenly balanced training sets the minimum support requirement is tightened so that itemset support must be no lower than 60%. This produces positive rules with a much wider range of confidence levels (ranging from 13% to 100%) but higher support values.

The results of the AprioriGrad classifier are presented and compared to a selection of common classification algorithms in Chapter 6.

Chapter 6

Results

6.1 WEKA Algorithms Used for Baseline Comparison

In order to evaluate the performance of the new AprioriGrad associative classifier, the same SHIPS data set was tested with a selection of classification algorithms selected from the collection of machine learning techniques found in WEKA [22] [23]. These classification algorithms, which were described in detail in Section 2.2, are listed by their WEKA method names in Table 6.1 and were chosen to be representative of a broad range of classification methods. This includes representation from such approaches as support vector machines (Sequential Minimal Optimization, or SMO), nearest-neighbor techniques (1-NN and 3-NN), decision trees (C4.5, whose WEKA implementation is known as J48), bagging (RandomForest), adaptive boosting (AdaBoostM1, which is here applied to the J48 and RandomForest classification algorithms), neural networks (MultilayerPerceptron) and a Naive Bayes classifier. Apart from the 3-NN algorithm, which had its *number of neigh-*

bors argument changed to 3 from the default value of 1, all algorithms were run using their default values as defined in their WEKA implementations.

Table 6.1: WEKA classifiers used for baseline comparison.

Label	Method and Arguments
SMO	weka.classifiers.functions.SMO[16] -C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 -K "weka.classifiers.functions.supportVector.PolyKernel[17] -C 250007 -E 1.0"
1-NN	weka.classifiers.lazy.IBk[18] -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch[19] -A "weka.core.EuclideanDistance[20] -R first-last""
3-NN	weka.classifiers.lazy.IBk[18] -K 3 -W 0 -A "weka.core.neighboursearch.LinearNNSearch[19] -A "weka.core.EuclideanDistance[20] -R first-last""
Boost-Forest	weka.classifiers.meta.AdaBoostM1[21] -P 100 -S 1 -I 10 -W weka.classifiers.trees.RandomForest[22] -- -I 10 -K 0 -S 1 -num-slots 1
Boost-J48	weka.classifiers.meta.AdaBoostM1[21] -P 100 -S 1 -I 10 -W weka.classifiers.trees.J48[23] -- -C 0.25 -M 2
Forest	weka.classifiers.trees.RandomForest[22] -I 10 -K 0 -S 1 -num-slots 1
J48	weka.classifiers.trees.J48[23] -C 0.25 -M 2
Perceptron	weka.classifiers.functions.MultilayerPerceptron[24] -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a
NaiveBayes	weka.classifiers.bayes.NaiveBayes[25]

[16] Authors: Eibe Frank (eibe@cs.waikato.ac.nz), Shane Legg (shane@intelligenesis.net), Stuart Inglis (stuart@reeltwo.com); Version: Revision 6024.

[17] Authors: Eibe Frank (eibe@cs.waikato.ac.nz), Shane Legg (shane@intelligenesis.net), Stuart Inglis (stuart@reeltwo.com); Version: Revision 5807.

[18] Authors: Stuart Inglis (singlis@cs.waikato.ac.nz), Len Trigg (trigg@cs.waikato.ac.nz), Eibe Frank (eibe@cs.waikato.ac.nz); Version: Revision 6572.

[19] Author: Ashraf M. Kibriya (amk14@cs.waikato.ac.nz); Version: Revision 5953.

[20] Authors: Gabi Schmidberger (gabi@cs.waikato.ac.nz), Ashraf M. Kibriya (amk14@cs.waikato.ac.nz), FracPete (fracpete@waikato.ac.nz); Version: Revision 5953.

[21] Authors: Eibe Frank (eibe@cs.waikato.ac.nz), Len Trigg (trigg@cs.waikato.ac.nz); Version: Revision 5928.

[22] Author: Richard Kirkby (rkirkby@cs.waikato.ac.nz); Version: Revision 7372.

[23] Author: Eibe Frank (eibe@cs.waikato.ac.nz); Version: Revision 6088.

[24] Author: Malcolm Ware (mfw4@cs.waikato.ac.nz); Version: Revision 6559.

[25] Authors: Len Trigg (trigg@cs.waikato.ac.nz), Eibe Frank (eibe@cs.waikato.ac.nz); Version: Revision 5928.

6.2 Metrics Used for Comparison

The metrics used for evaluating classification results in the present study are all based on the *confusion matrix* (Table 6.2), which reports counts of instances based on their actual target class vs. the label they were assigned by a classifier. True Positives (TP) and True Negatives (TN) are those instances that were accurately labeled by the classifier as positive or negative; False Positives (FP) are instances that shouldn't have been labeled positive but were; and False Negatives (FN) are positives instances that were missed by the classifier. All of the evaluation metrics are calculated from these four instance counts[26].

Table 6.2: The confusion matrix.

	is positive in reality	is negative in reality
labeled as positive	TP	FP
labeled as negative	FN	TN

It should be noted that in cases with very rare positives such RW35 and RW40, small changes in TP counts can lead to large variations in the calculated metrics. As an example of this, there are six positive RW40 instances out of 1468 in the pre-processed SHIPS data set. If only one of these six cases were labeled positive by a classifier, this would give it a 17% recall metric whereas if just one more case were labeled positive it would double the score to 33%. This means that the statistics for the very rare targets can show wide variability. This effect is minimized by running ten trials (each one consisting of five-fold cross-validation based on different random splits of the data) of each classifier and averaging the resulting statistics.

Precision (defined in Equation 6.1), also known as *success ratio*, is a measure of a classifier's accuracy when labeling an instance as positive. Precision is degraded by large numbers of false alarms (FPs) and not at all impacted by missed positives (FNs). Recall (defined in Equation 6.2), also known as the *probability of detection (POD)*, is a measure

[26]Note that [51] is an excellent reference for verification metrics and graphics.

of how completely a classifier identifies positive instances in the test set. Recall is degraded by large numbers of missed positives (FNs) and not at all impacted by false alarms (FPs).

$$Precision = \frac{TP}{TP+FP} \tag{6.1}$$

$$Recall = \frac{TP}{TP+FN} \tag{6.2}$$

Precision and recall must often be balanced against one another, since increasing precision can lead to reduced recall and vice versa. Therefore, the best measures of overall performance of a classifier (such as the Critical Success Index or the F-measure, defined below) take both precision and recall into account. However, these combined scores can be misleading if classification results are heavily biased toward either precision or recall, since this can yield fairly high scores while still producing results that may not be acceptable to an end-user. It will be seen that the NaiveBayes classifier in this study is a good example of this, since it achieves relatively high F-measure results by scoring highly on recall while its precision scores are very low. The bias score (defined in Equation 6.3) highlights this kind of shortcoming. A classifier with even bias would score a perfect 1, a classifier biased toward precision would score in the range $0-1$ and a classifier biased towards recall would score above 1. Reciprocal bias scores might be said to be equivalently biased, that is, a classifier with a bias score of 0.1 could be said to be "as biased" towards precision as a classifier with a bias score of 10 is biased towards recall.

$$Bias = \frac{TP+FP}{TP+FN} \tag{6.3}$$

Two measures of overall classification performance are reported. One is the Critical Success Index (CSI), which is also known as the *threat score*. CSI, defined in Equation 6.4, is similar to precision and recall except it is penalized for high numbers of both false alarms and missed positives in its denominator. The F-measure, defined in Equation 6.5, is simply the harmonic mean of the precision and recall scores. Both measures range from 0 (worst)

to 1 (best), and sorting results by either measure results in the same relative ordering of classifiers.

$$CSI = \frac{TP}{TP + FP + FN} \tag{6.4}$$

$$F\text{-}measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6.5}$$

6.3 Performance Comparison: AprioriGrad vs. WEKA Algorithms

6.3.1 Tabular and Graphical Presentation of Results

Performance metrics are presented in Tables 6.3 and 6.4 for AprioriGrad and all of the WEKA algorithms. One very useful way of presenting all of these metrics at once is with the Categorical Performance Diagram [51] [52]. This is a scatter diagram of recall (Y-axis) vs. precision (X-axis). Due to the mathematical relationships among the performance metrics, it is also possible to show lines of constant bias and CSI[27] on these diagrams. Straight lines meeting at the origin are lines of constant bias, with even bias (considered optimal for purposes of this study) found along the diagonal line $y = x$. These lines are labeled where they meet the right-hand and upper edges of the diagram. Curved lines are lines of constant CSI and are labeled within the diagram. The best classifiers would fall closest to the upper right of the diagram. Eight Categorical Performance Diagrams are shown, four for the Rapid Intensification targets (Figure 6.1) and four for the Rapid Weakening targets (Figure 6.2).

6.3.2 Discussion of Results

The results in these tables and figures show five performance metrics for each of eight classification targets for each of ten classifiers. This section examines the question of which classifier is "best."

[27] The critical success index, or threat score, can be thought of as a stand-in for the F-measure metric.

Table 6.3: Rapid Intensification: relative performance of AprioriGrad and selected WEKA classifiers. Algorithm names are as described in the text, and values are precision (Prec), recall (Rec), Bias, Critical Success Index (CSI) and F-measure (F-ms).

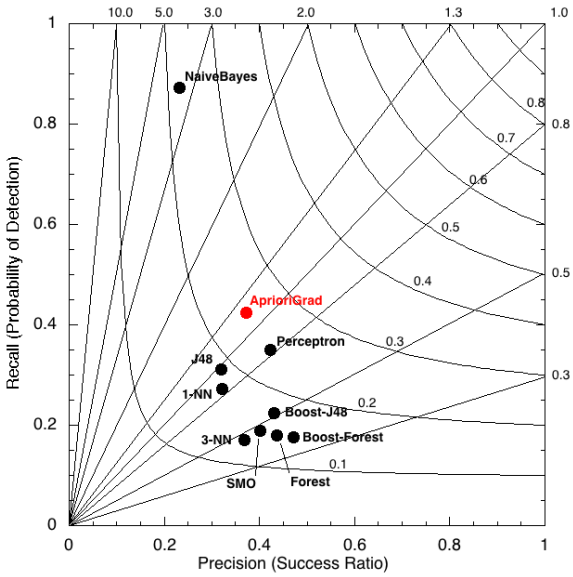
	RI25					RI30				
Algorithm	Prec	Rec	Bias	CSI	F-ms	Prec	Rec	Bias	CSI	F-ms
SMO	0.40	0.19	0.47	0.15	0.26	0.04	0.00	0.02	0.00	0.00
1-NN	0.32	0.27	0.85	0.17	0.29	0.16	0.13	0.84	0.08	0.14
3-NN	0.37	0.17	0.47	0.13	0.23	0.22	0.08	0.35	0.06	0.11
Boost-Forest	0.47	0.18	0.37	0.15	0.26	0.26	0.03	0.10	0.02	0.05
Boost-J48	0.43	0.22	0.52	0.17	0.30	0.27	0.08	0.29	0.06	0.12
Forest	0.44	0.18	0.41	0.15	0.26	0.20	0.03	0.17	0.03	0.06
J48	0.32	0.31	0.98	0.19	0.31	0.16	0.16	0.96	0.09	0.16
Perceptron	0.42	0.35	0.83	0.24	0.38	0.21	0.15	0.70	0.10	0.18
NaiveBayes	0.23	0.87	3.77	0.22	0.37	0.14	0.84	5.91	0.14	0.24
AprioriGrad	0.37	0.42	1.14	0.25	0.40	0.20	0.21	1.04	0.11	0.20
	RI35					RI40				
Algorithm	Prec	Rec	Bias	CSI	F-ms	Prec	Rec	Bias	CSI	F-ms
SMO	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.02	0.00	0.00
1-NN	0.09	0.07	0.77	0.04	0.08	0.05	0.05	0.95	0.03	0.05
3-NN	0.20	0.06	0.30	0.05	0.09	0.21	0.07	0.33	0.06	0.11
Boost-Forest	0.15	0.01	0.07	0.01	0.02	0.10	0.00	0.03	0.00	0.01
Boost-J48	0.17	0.03	0.15	0.02	0.05	0.21	0.02	0.10	0.02	0.04
Forest	0.07	0.01	0.08	0.01	0.01	0.05	0.00	0.05	0.00	0.01
J48	0.14	0.13	0.90	0.07	0.13	0.08	0.07	0.90	0.04	0.07
Perceptron	0.15	0.09	0.58	0.06	0.11	0.14	0.07	0.49	0.05	0.09
NaiveBayes	0.09	0.84	8.87	0.09	0.17	0.07	0.80	11.20	0.07	0.13
AprioriGrad	0.15	0.16	1.13	0.08	0.15	0.17	0.18	1.04	0.09	0.17

Table 6.4: Rapid Weakening: relative performance of AprioriGrad and selected WEKA classifiers. Algorithm names are as described in the text, and values are precision (Prec), recall (Rec), Bias, Critical Success Index (CSI) and F-measure (F-ms).

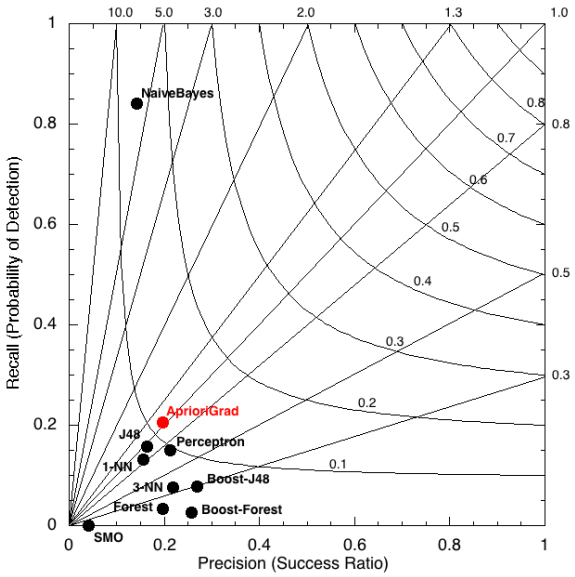
	RW25					RW30				
Algorithm	Prec	Rec	Bias	CSI	F-ms	Prec	Rec	Bias	CSI	F-ms
SMO	0.34	0.11	0.31	0.09	0.16	0.40	0.14	0.34	0.11	0.20
1-NN	0.13	0.08	0.60	0.05	0.09	0.13	0.10	0.76	0.06	0.11
3-NN	0.27	0.03	0.13	0.03	0.06	0.35	0.04	0.11	0.04	0.07
Boost-Forest	0.36	0.04	0.12	0.04	0.07	0.52	0.03	0.05	0.03	0.05
Boost-J48	0.37	0.09	0.25	0.08	0.15	0.30	0.04	0.13	0.04	0.07
Forest	0.38	0.05	0.14	0.05	0.09	0.44	0.02	0.04	0.02	0.04
J48	0.21	0.21	0.98	0.12	0.21	0.23	0.20	0.86	0.12	0.22
Perceptron	0.33	0.18	0.55	0.13	0.23	0.31	0.16	0.52	0.12	0.21
NaiveBayes	0.14	0.64	4.63	0.13	0.23	0.10	0.69	6.64	0.10	0.18
AprioriGrad	0.24	0.23	0.97	0.13	0.23	0.21	0.19	0.91	0.11	0.20
	RW35					RW40				
Algorithm	Prec	Rec	Bias	CSI	F-ms	Prec	Rec	Bias	CSI	F-ms
SMO	0.05	0.01	0.24	0.01	0.02	0.00	0.00	0.35	0.00	0.00
1-NN	0.00	0.00	1.07	0.00	0.00	0.00	0.00	1.65	0.00	0.00
3-NN	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.12	0.00	0.00
Boost-Forest	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.03	0.00	0.00
Boost-J48	0.05	0.01	0.12	0.00	0.01	0.27	0.10	0.37	0.08	0.15
Forest	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.07	0.00	0.00
J48	0.12	0.09	0.80	0.06	0.10	0.10	0.03	0.33	0.03	0.05
Perceptron	0.03	0.01	0.36	0.01	0.02	0.00	0.00	0.35	0.00	0.00
NaiveBayes	0.05	0.55	11.88	0.04	0.09	0.06	0.22	3.62	0.05	0.09
AprioriGrad	0.08	0.07	0.87	0.04	0.08	0.12	0.08	0.70	0.05	0.10

Figure 6.1: Categorical Performance Diagrams (after [52]) showing the individual rapid intensification results from all classifiers.

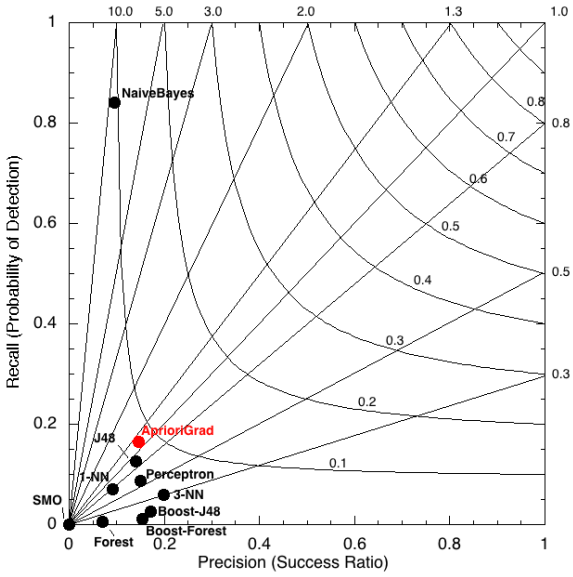
(a) RI25



(b) RI30



(c) RI35



(d) RI40

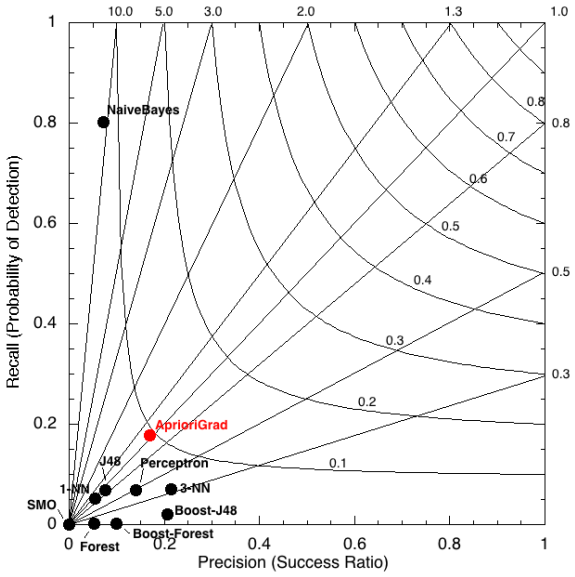
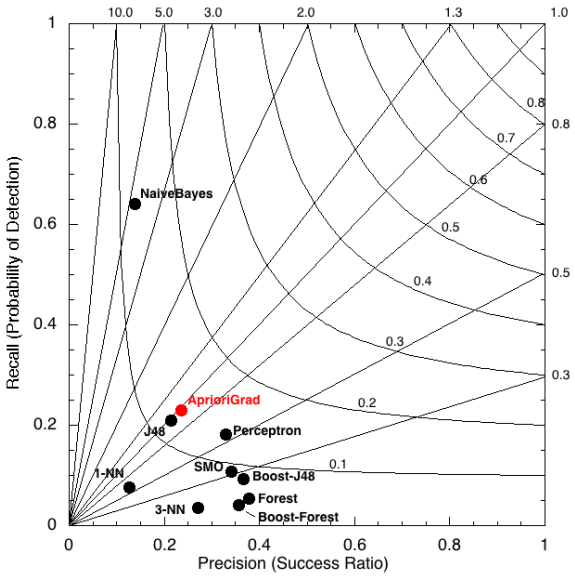
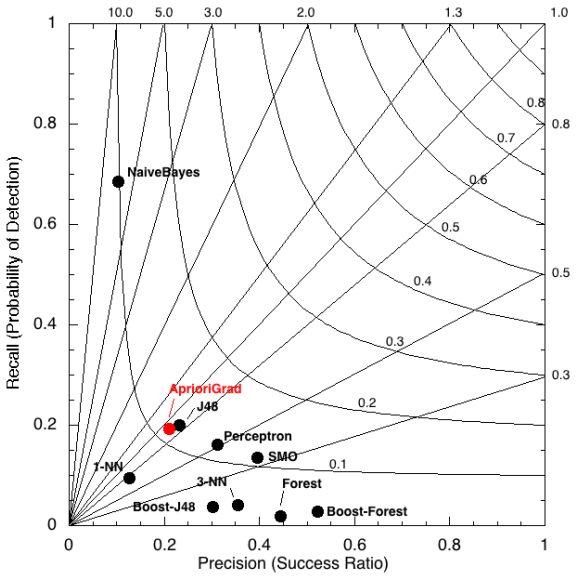


Figure 6.2: Categorical Performance Diagrams (after [52]) showing the individual rapid weakening results from all classifiers.

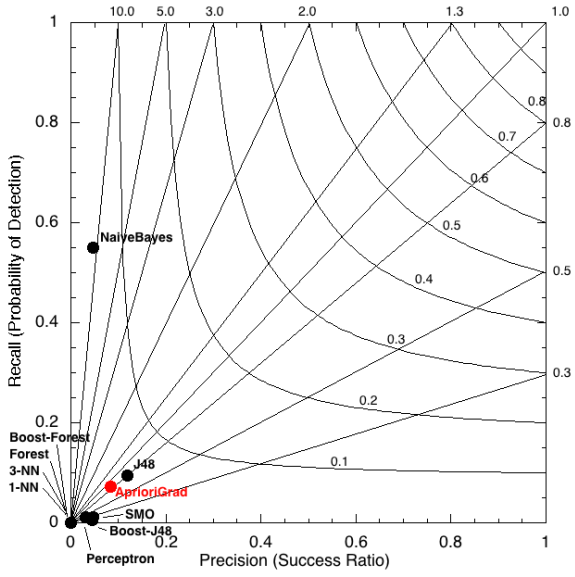
(a) RW25



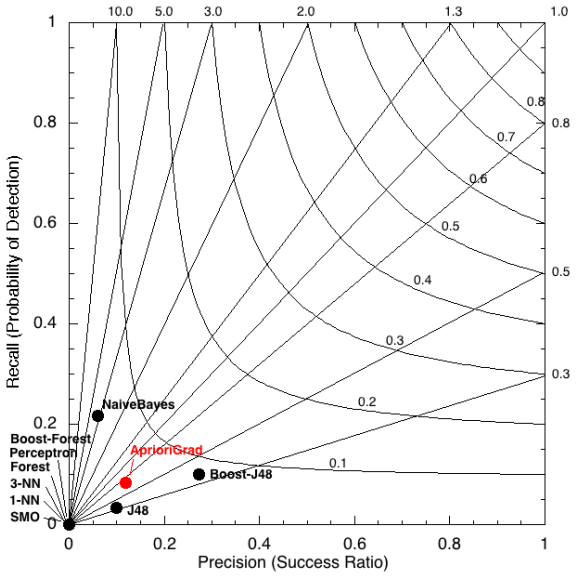
(b) RW30



(c) RW35



(d) RW40



First, consider the classifiers' performance overall, with metrics averaged across all eight classification targets. If considering only precision or recall *individually*, AprioriGrad does not necessarily stand out among the classifiers. For precision (Figure 6.3a), AprioriGrad is bested by (in order of higher precision to lower) Boost-J48, Boost-Forest, 3-NN, Perceptron, and Forest. For recall (Figure 6.3b) AprioriGrad does better, for it is beaten only by NaiveBayes, which achieves its very high recall scores at the expense of poor precision scores. Two other classifiers, J48 and Perceptron, return reasonably good recall scores though not as good as AprioriGrad.

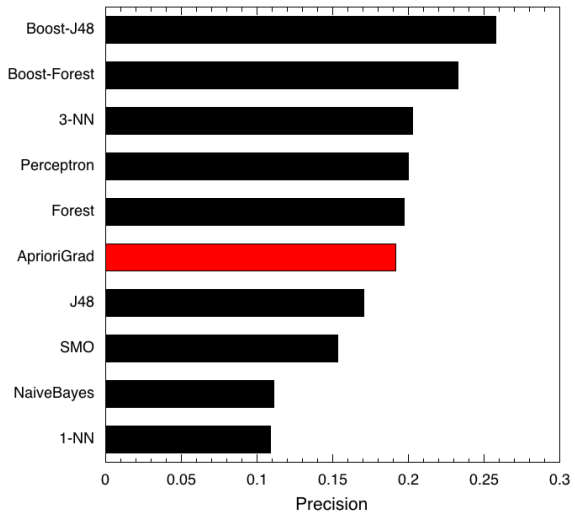
When considering bias and F-measure (or equivalently, CSI), AprioriGrad fares much better. AprioriGrad is the most evenly biased (Figure 6.3c) of all tested classifiers, with J48 and 1-NN coming in second and third, and Perceptron a more distant fourth. Note that for purposes of ranking the classifiers' performance, the reciprocal is taken for bias scores higher than 1.0 in order to measure each classifier's "nearness" to an even bias in a consistent manner. Also, AprioriGrad has on average the highest F-measures (Figure 6.3d) of all classifiers, with NaiveBayes coming in second (having sacrificed precision for higher recall scores) and J48 and Perceptron relatively close to one another in third place. Overall, in terms of average performance across all eight classification targets, AprioriGrad must be considered the most successful approach.

It is also possible to plot these overall precision and recall scores, averaged across all eight target variable results, in their own Categorical Performance Diagram (Figure 6.4). This provides a simple way of looking at all of the performance metrics at once, where at a glance it can be seen that AprioriGrad comes in sixth for precision (X-axis), second for recall (Y-axis), first for bias (nearness to $y = x$ diagonal), and first for CSI (threat score, as denoted by curved contour lines).

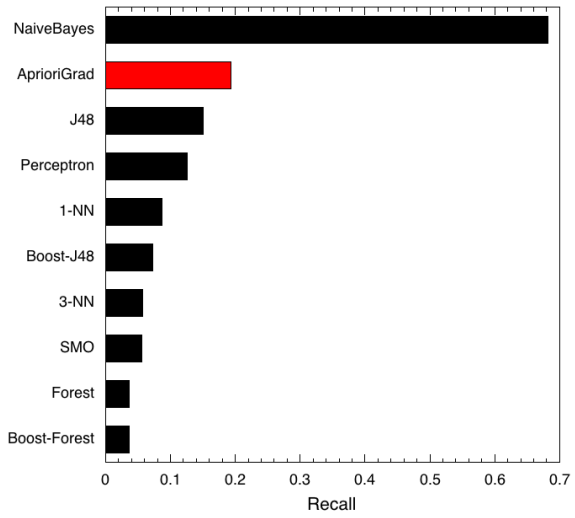
Next, consider the classification targets one by one, based on a subjective examination of their Categorical Performance Diagrams as well as a numeric comparison of their

Figure 6.3: Ranking of average classifier performance by precision, recall, bias and F-measure metrics.

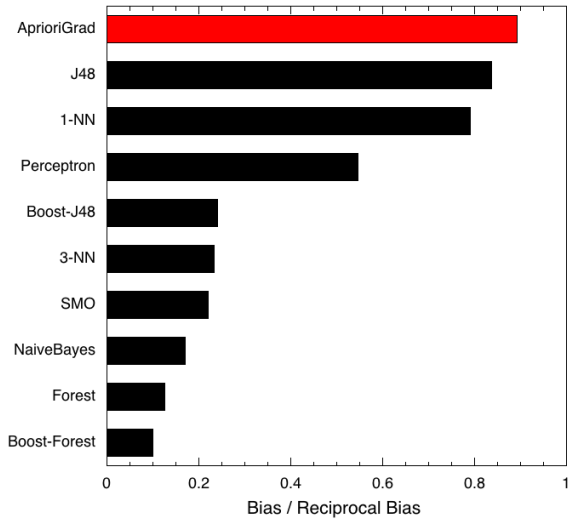
(a) Precision



(b) Recall



(c) Bias / Reciprocal Bias



(d) F-Measure

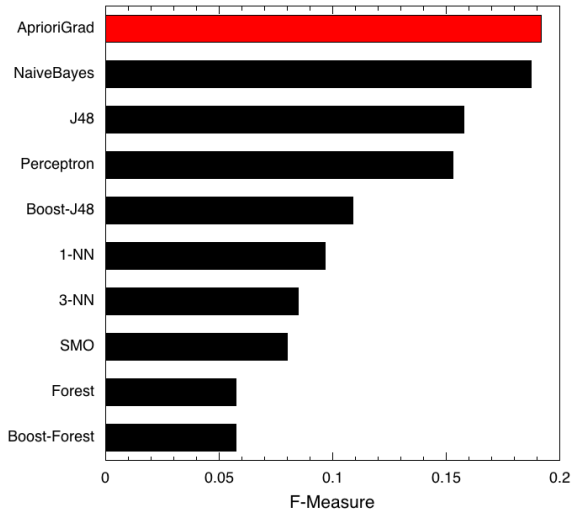
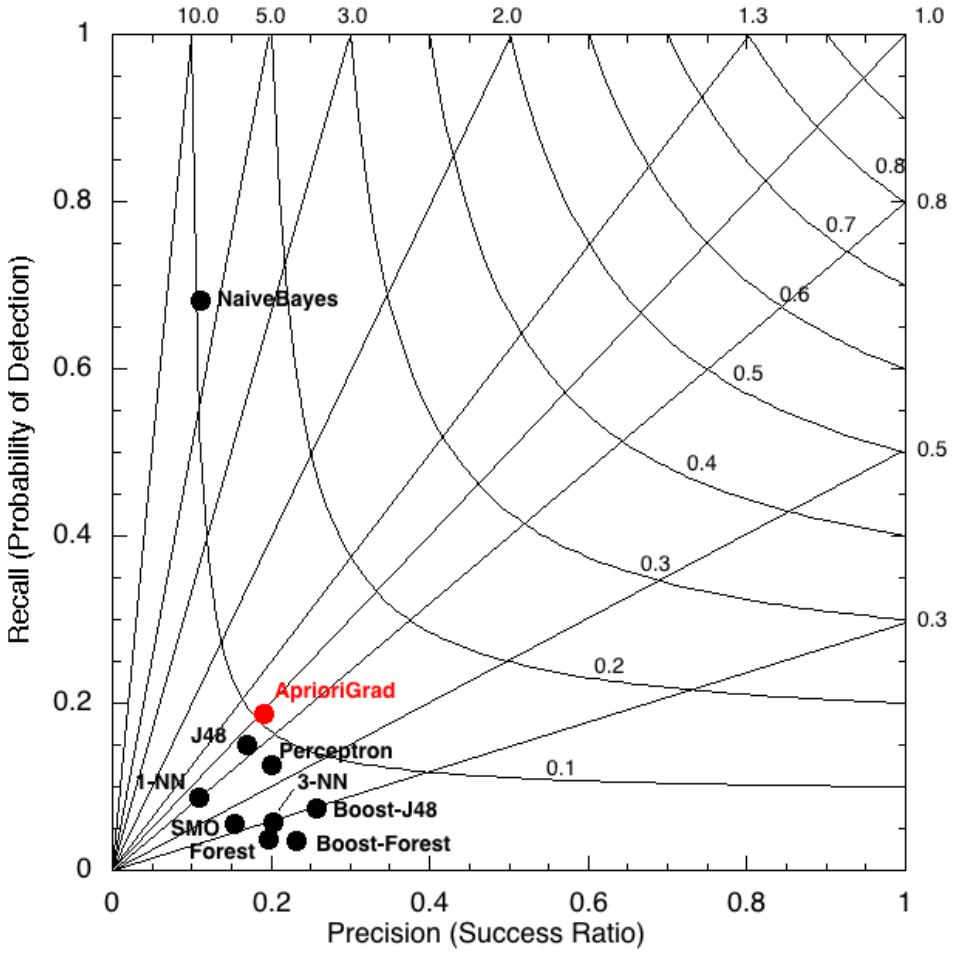


Figure 6.4: A Categorical Performance Diagram (after [52]) showing precision and recall scores that were averaged from the eight individual target results.



F-measure scores. For all four Rapid Intensification targets, AprioriGrad looks best in the diagrams, in that it is closest to the central diagonal line (even bias) and upwards and to the right of any of its close neighbors. Based on the F-measure scores alone, AprioriGrad leads in RI25 and RI40, and comes in second to NaiveBayes in RI30 and RI35. NaiveBayes is not considered to be a reasonable alternative to AprioriGrad in this context, however, because of its extremely high bias and poor precision.

For the four Rapid Weakening targets AprioriGrad turns in a more mixed performance. It is clearly the best for the RW25 target, though J48 comes in a close second and Perceptron (biased towards precision) and NaiveBayes (biased towards recall) also score well. For RW30, however, AprioriGrad is overtaken by J48, which like AprioriGrad is evenly biased, and by the numbers it also loses to Perceptron and SMO (which are more heavily biased towards precision). For RW35, AprioriGrad falls even further behind J48 while both classifiers remain evenly biased, and also loses to NaiveBayes on their F-measure scores. For RW40, AprioriGrad regains some lost ground because, while its F-measure score is less than that of Boost-J48, the latter is heavily biased toward precision and AprioriGrad is the best of the evenly-biased classifiers.

Individually, then, AprioriGrad arguably performs best on six of the eight targets, and takes second place to J48 on the other two. Given that AprioriGrad's average performance metrics are best, this suggests that it is the best overall classifier for this domain, although there is room for further improvement with some of the Rapid Weakening targets.

Two other points of interest are striking from these results: AprioriGrad's performance (like the other classifiers) generally degrades as the classification targets become more rare, at the higher degrees of rapid TC intensity change. This makes intuitive sense, since the classification problems become more difficult as the target classes become rarer. However, it is striking that AprioriGrad results on RI40 and RW40, the most rare of classification targets, are better than those on RI35 and RW35, the next most rare of classification tar-

gets. This is very likely an artifact of calculating percentages based on very small instance counts. For the rarest of classification targets, the correct classification of even one or two more positive cases can have a large positive effect on the performance metrics. It is likely that AprioriGrad performs equally well on RI35 and RI40, and on RW35 and RW40, and that the rarer classification results appear somewhat inflated by the calculations involving very small instance counts.

A second point of interest is noting that the Rapid Weakening results are generally much poorer than the Rapid Intensification results. Even the RW25 results recall only about a quarter of positive instances and, of the instances labeled positive, only about a quarter are correctly labeled. This does not seem to be a strong result on first glance. However, there are several reasons for this poorer performance on the Rapid Weakening domain. First of all, RW is much rarer in the SHIPS data set and therefore harder to predict. Meteorologists are now beginning to study Rapid Weakening but thus far it has not been given as much attention as Rapid Intensification, in part because the consequences of failing to anticipate a RW event are less severe than failing to forecast RI accurately. The data set underlying this study, it must also be recalled, was not collected with RW prediction in mind and therefore may not contain all of the available information that might be helpful for predicting RW.

6.4 Detailed Results from AprioriGrad

This section presents some more detailed results that are specific to AprioriGrad. Table 6.5 contains the complete confusion matrices for all eight classification targets, with precision, recall and F-measure scores copied from Tables 6.3 and 6.4 as a convenience. Instance counts in this table are not necessarily whole numbers because they are averaged results from ten experimental runs of the algorithm, each run using different random splits of the data set for its five-fold cross-validations.

Table 6.6 lists the SHIPS record types that were most frequently selected by Apriori-Grad, broken out by negative and positive rules, and by intensification and weakening

Table 6.5: AprioriGrad confusion matrices for all eight classification targets.

RI25

TP	69.6	117.7	FP
FN	94.4	1186.3	TN

precision: 0.37
recall: 0.42
f-measure: 0.40

RW25

TP	17.3	55.8	FP
FN	57.7	1337.2	TN

precision: 0.24
recall: 0.23
f-measure: 0.23

RI30

TP	21.2	86.2	FP
FN	81.8	1278.8	TN

precision: 0.20
recall: 0.21
f-measure: 0.20

RW30

TP	8.1	30.3	FP
FN	33.9	1395.7	TN

precision: 0.21
recall: 0.19
f-measure: 0.20

RI35

TP	8.9	52.1	FP
FN	45.1	1361.9	TN

precision: 0.15
recall: 0.16
f-measure: 0.15

RW35

TP	1.3	14.3	FP
FN	16.7	1435.7	TN

precision: 0.08
recall: 0.07
f-measure: 0.08

RI40

TP	6.2	30.3	FP
FN	28.8	1402.7	TN

precision: 0.17
recall: 0.18
f-measure: 0.17

RW40

TP	0.5	3.7	FP
FN	5.5	1458.3	TN

precision: 0.12
recall: 0.08
f-measure: 0.10

targets. The negative rules were largely the same across all four RI targets and across all four RW targets, since they differed only in the degree of pruning applied to the original 200 negative rules that were mined. However, the positive rules differed significantly from target to target and so Tables 6.7 and 6.8 show a more detailed analysis, per RI and RW targets respectively, of the SHIPS record types most frequently selected for positive rules.

This analysis was done by extracting all of the individual conditions from the 50 rule sets per classification target (ten repeated experiments with each one producing rule sets for five cross-validation random splits of the data set) and counting how many times each record type appears. The table shows by arrow whether the condition was required to be "high" (a value greater than some selected split point) or "low." Horizontal lines indicate neither high nor low, i.e., those cases where discretization produced more than two categories and the rule condition referred to one of the ranges in the middle of the domain. Also shown in these tables are the average number of rules per rule set from which these frequency counts were calculated, which indicates the strength of the result. The negative rules were far more successful at returning the full 200 rules requested and the pruning process usually did not remove many of these rules, leading to average rule counts of 185.7 (intensification) and 166.9 (weakening). The positive rules however, particularly those for weakening, are a much smaller set of rules overall and therefore their attribute selections are somewhat more idiosyncratic.

The majority of these attribute tests make intuitive meteorological sense, which means for the most part that an unsupervised selection of attribute subsets has found meaningful meteorological relationships. Examples of such meaningful relationships include the finding that weakening is less likely if the TC does not have a long history of higher intensities, or that intensification is more likely when shear is low, or that weakening is more likely later on in the season. However, there are a few oddities where attribute tests appear to be non-intuitive or contradictory, such as the testing for low HIST attributes in both the RI and

RW positive rules, or requiring a low POT (potential for intensification) for RI35. These attribute selections are in all likelihood given the appearance of selection frequency in part because of the relatively small number of positive rules that are mined and left unpruned.

Table 6.6: Frequently-featured SHIPS record types (summary).

Negative						Positive					
RI-All 185.7 rules			RW-All 166.9 rules			RI-All 38.9 rules			RW-All 13.7 rules		
VVAV	⇓	17%	VINC	=	16%	HIST	⇓	41%	V000	⇓	20%
EPSS	⇓	16%	HIST	⇓	13%	DELV	⇑	6%	V300	⇓	11%
VVAC	⇓	8%	V20C	⇓	12%	IR00	⇓	4%	V500	⇓	8%
U200	⇑	7%	VSHR	⇓	10%	TADV	⇓	4%	HIST	⇓	8%
VMFX	⇓	6%	VMAX	⇓	9%	INCV	⇑	3%	JDAY	⇑	4%
RHMD	⇓	6%	POT	⇑	6%	IRM3	⇓	3%	INCV	⇑	4%
SDDC	⇓	5%	IR00	⇓	6%	VINC	⇑	2%	SHRS	⇑	4%
SHDC	⇑	4%	VINC	⇓	5%	E000	⇓	2%	DIVC	⇑	3%
SHTD	⇓	3%	REFC	⇑	5%	SHGC	⇓	2%	USM	⇑	3%
Other		28%	Other		17%	Other		33%	Other		35%

Table 6.7: Frequently-featured SHIPS record types (detail, positive, intensification).

RI25 94.9 rules			RI30 40.9 rules			RI35 16.2 rules			RI40 3.7 rules		
HIST	⇓	50%	HIST	⇓	29%	JDAY	⇑	10%	HIST	⇓	17%
DELV	⇑	7%	T150	⇑	6%	V000	⇓	8%	TADV	⇓	16%
INCV	⇑	5%	Z850	⇓	6%	USM	⇑	8%	U20C	⇓	6%
E000	⇓	4%	DELV	⇑	5%	INCV	⇑	7%	SHRS	⇓	5%
TADV	⇓	4%	R000	⇓	4%	R000	⇑	6%	IR00	⇓	5%
IR00	⇓	3%	IRM3	⇓	3%	POT	⇓	6%	SHRG	⇓	5%
VINC	⇑	3%	IR00	⇓	3%	JDTE	⇓	6%	IRM3	⇓	5%
PENC	⇑	3%	D200	⇓	3%	DIVC	⇑	6%	SHGC	⇓	4%
ENSS	⇑	3%	RHMD	⇓	3%	DELV	⇑	5%	RSST	⇑	4%
Other		18%	Other		38%	Other		39%	Other		32%

Another question of possible interest to anyone who might want to use this technology as a forecasting tool is whether the algorithm's misclassified cases, either the false alarms or the missed events, are "near misses" or not. For example, if the classifier falsely labels a case as positive for RI25, is it more likely to have intensified at 20 or 15 kt, or are there a large number of false positives corresponding to cases that are intensifying very little or

Table 6.8: Frequently-featured SHIPS record types (detail, positive, weakening).

RW25 31.6 rules			RW30 13.1 rules			RW35 9.2 rules			RW40 0.8 rules		
V000	⇓	24%	V000	⇓	21%	JDAY	⇑	10%	JDAY	⇑	18%
V300	⇓	14%	V300	⇓	11%	V000	⇓	8%	HIST	⇑	13%
V500	⇓	13%	DIVC	⇑	9%	USM	⇑	8%	USM	⇑	12%
HIST	⇓	12%	USM	⇑	6%	INCV	⇑	7%	VMAX	⇑	11%
SHGC	⇑	4%	VMFX	⇓	6%	R000	⇑	6%	V000	⇓	9%
SHRS	⇑	4%	TGRD	⇑	5%	POT	⇓	6%	R000	⇑	5%
SHRG	⇑	4%	SHRS	⇑	5%	JDTE	⇓	6%	RHMD	⇑	4%
INCV	⇑	4%	JDAY	⇑	4%	DIVC	⇑	6%	RSST[28]	⇓	2%
SHDC	⇑	2%	PENC	⇓	4%	DELV	⇑	5%	DIVC	⇑	2%
Other		19%	Other		30%	Other		39%	Other		25%

even weakening? A corresponding example relating to missed events might be whether the positive RI25 cases that were missed by the algorithm are more likely to have been borderline 25-kt intensifiers, or if in fact cases with much greater intensification (weakening) are being missed even at the lowest RI (RW) classification stages. The nature of this classification problem makes it possible to ask not merely whether classification was done correctly but *by how far* classification is off when it misses.

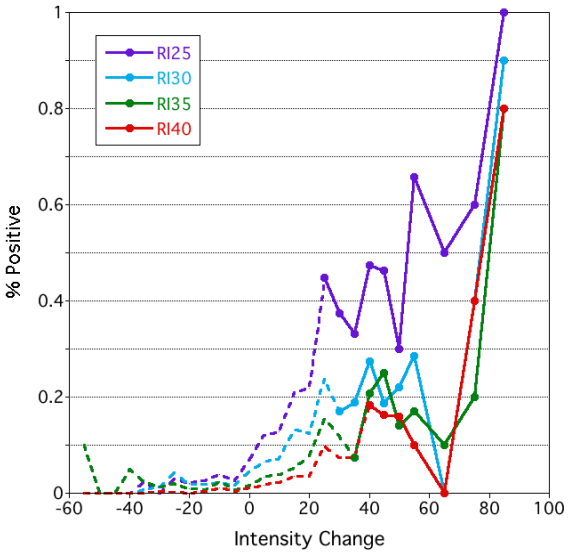
One way of answering this question is to examine all of the cases labeled as positive for each of the eight targets and divide them into bins based on the degree to which they actually intensified or weakened. One might hope to see a peak near the classifier's threshold point, for example at 25 kt for RI25, that falls off quickly as intensification levels grow smaller. Instead, one generally observes a peak near the center of the range of intensities (which in this study's sample range from weakening at 55 kt to intensifying at 85 kt), with falloff to either side. This is because the vast majority of cases fall in the middle of this range and the outlier events which are the targets of prediction are considerably more rare. Therefore a more useful measure of the nearness of missed cases is given by calculating, for all cases at each 5-kt bin of intensifying or weakening, what *percentage* of cases is

[28]RSST Analysis Age parameter, selected a total of 5 times during 50 trials.

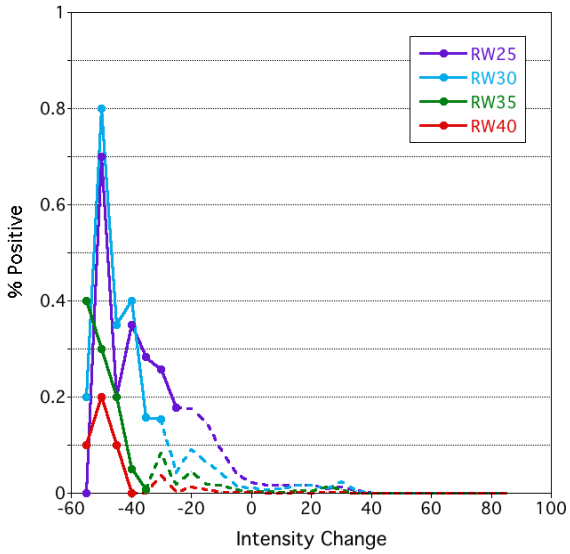
labeled positive for each of the eight targets. These calculations are shown for the four intensification targets in Figure 6.5a and for the four weakening targets in Figure 6.5b. In both of these figures, dotted lines indicate cases where the classifier should have returned a negative class and solid lines (with markers) indicate cases where the classifier should have returned a positive class.

Figure 6.5: Percentage of cases by intensification/weakening level that were classified as positive by AprioriGrad for each of the eight targets.

(a) Intensification targets



(b) Weakening targets



From these two figures it is clear that there is the expected falloff of false alarms as the real intensity changes grow less extreme. For RI, the false alarm rate is never more than 24%, and is only this high within one or two intensification bins of the threshold point. For RW, the false alarm rate is never more than 18%. The opposite question of whether missed events are likely to have been closer to the threshold values is a great deal more difficult to answer because of the very small total numbers of cases at these levels, particularly for RW. The final three data points on the intensification side, at 65, 75 and 85 kt, correspond to only one case apiece, and in particular the 65-kt case appears to have been missed in almost all of the experimental runs. Similarly on the weakening side there was only one case apiece for

−55 and −50 kt and two cases apiece for −45 and −40 kt. Still the RI targets in general and RI25 in particular appear to show a favorable upward trend as intensification grows more extreme, which suggests that this trend would be more pronounced if more test cases at these extreme levels of intensification or weakening were available.

Chapter 7

Conclusions

This study described an approach to customize the Apriori-based association rules algorithm for specific application to the SHIPS data set to produce rules that could be combined into an associative classifier for predicting rapid changes in the intensity of tropical cyclones. It combines the disciplines of computer engineering and tropical meteorology, and the results have implications for both of these fields, as well as for the prospect of crafting a forecast tool for operational use.

7.1 Engineering Relevance

The most important engineering question is whether this approach described can be reused in other domains, or to what extent its SHIPS customizations could limit its portability. There are two defining characteristics of the SHIPS data set that informed the basic design of this approach: the rarity of its positive events, and the definition of its multiple positive/negative targets based on a common, underlying, continuous-domain parameter. This parameter is of course the signed difference in intensity of a TC at its initial time as compared to 24 h later. In this domain, a target class attribute's binary values are simple positive and negative, but behind the scenes every case can be said to be more positive/negative (i.e., further from the threshold value) or less so. The algorithm is able to capitalize on its knowledge of how positive/negative the training cases are, as well as the interrelationships

among the target class attributes, since labeling one case as positive or negative for one variable will limit the labels which may be consistently given to that case's other target class attributes.

Therefore it may be said that other domains with similar characteristics, both the rarity of positive events and the threshold nature of their construction, would be logical choices for applying the same approaches described in this study. As it happens, many meteorological or oceanographic events of interest might fall into this category, such as extreme rainfall events, or extremities of sea temperatures both high and low, which happen to have implications for coral health.

7.2 Use as a Forecast Tool

One possible application of this work would be to produce a forecast tool that could be used operationally by hurricane forecasters as another piece of evidence to weigh when composing an intensity forecast for a tropical cyclone. This is more of a product development question, since its design would need to take into consideration the needs and wants of the intended user community. Specifically, there is a design choice to be made around the issue of bias: would a forecaster prefer a tool that errs on the side of caution, where the classifier favors recall (probability of detection) and is less likely to miss real events at the expense of a higher false alarm rate? Alternatively, would a forecaster prefer a tool that favors precision (success ratio), yielding a classifier whose positive pronouncements are more likely to be accurate even at the expense of missing a greater number of real events? A third alternative would be a tool along the lines described in this study, one that is evenly biased, and would favor neither overforecasting nor underforecasting of a rapid intensity-change event. Informal discussions with hurricane forecasters suggest that a forecast tool of even bias or one that is biased towards precision would be preferred, because an overforecasting tool with a high false-alarm rate would be less likely to be trusted.

Another question regarding the possible use of this approach as an operational forecasting tool would be whether its performance as measured in this study is good enough to be of any use. A layman might look at the best-performing classifier from this study, which forecasts RI25 with roughly 40% accuracy and 40% recall rate, and conclude that "less than half" of real events recalled and "less than half" of predicted events verified as positive does not describe an intuitively reliable tool. However, in the context of the overall predictability of this domain (which is low), the extreme importance of anticipating true events (particularly when speaking of rapid intensification), and the relative dearth of intensity forecasting tools based on diverse technologies, a forecast tool of this level of proficiency may in fact find a warm welcome in the TC forecasting community.

Operational use would differ slightly from experimental use in that the classifier would be trained using all available cases, 1982-2011, and re-trained every year once the past season's cases were made available. This training process would yield two sets of negative rules, one for intensification and one for weakening, as well as eight sets of positive rules, with one for each individual target class attribute. With the bulk of processing finished beforehand, real-time processing of a single case would take only seconds on a simple laptop computer.

7.3 Meteorological Relevance, Future Work

One of the most important goals of the present study was to approach the problem of the prediction of TC rapid intensity change with no meteorological preconceptions, and to examine the predictors selected by the AprioriGrad algorithm for possible meteorological significance. One might expect that in very broad terms the selected predictors would simply recover some very basic physical truths: that high levels of shear might prevent intensification, or a short-term trend of intensification may indicate further intensification to come, or that a long history at very high intensities might indicate that weakening will soon

follow. In the eyes of a meteorologist the selected predictors[29], on the face of things, either do or do not make obvious intuitive sense. For those that do not make sense, they may fall out from a certain randomness of selection (particularly when dealing with a small numbers of positive cases), or they may potentially be signs of previously-unexamined nuances in the physical processes that govern intensity change of TCs. Such a meteorological analysis is beyond the scope of this engineering study but further collaborations between the engineering and meteorological communities are planned to follow with more investigations along the lines described here.

[29]Tables 6.6, 6.7 and 6.8 are an excellent starting place for investigations of this kind.

Bibliography

[1] E. N. Rappaport, J. L. Franklin, L. A. Avila, S. R. Baig, J. L. Beven, E. S. Blake, C. A. Burr, J.-G. Jiing, C. A. Juckins, R. D. Knabb, C. W. Landsea, M. Mainelli, M. Mayfield, C. J. McAdie, R. J. Pasch, C. Sisko, S. R. Stewart, and A. N. Tribble, "Advances and challenges at the national hurricane center," *Weather and Forecasting*, vol. 24, no. 2, pp. 395–419, 2012/06/15 2009. [Online]. Available: http://dx.doi.org/10.1175/2008WAF2222128.1

[2] J. L. Franklin, C. J. McAdie, and M. B. Lawrence, "Trends in track forecasting for tropical cyclones threatening the united states, 1970-2001," *BULLETIN OF THE AMERICAN METEOROLOGICAL SOCIETY*, vol. 84, no. 9, pp. 1197–1203, Sep 2003.

[3] M. DeMaria and J. Kaplan, "A statistical hurricane intensity prediction scheme (ships) for the atlantic basin," *Weather and Forecasting*, vol. 9, no. 2, pp. 209–220, 1994.

[4] Statistical tropical cyclone intensity forecast technique development. [Online]. Available: http://rammb.cira.colostate.edu/research/tropical_cyclones/ships/

[5] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference on Very Large Data Bases*, vol. 1215, 1994, pp. 487–499.

[6] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *SIGMOD Rec.*, vol. 22, no. 2, pp. 207–216, Jun. 1993. [Online]. Available: http://doi.acm.org/10.1145/170036.170072

[7] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Research Report Report RJ 9839*. San Jose, CA, USA: IBM Almaden Research Center, June 1994.

[8] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," in *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, 1998.

[9] R. Yang, J. Tang, and M. Kafatos, "Improved associated conditions in rapid intensifications of tropical cyclones," *Geophysical Research Letters*, vol. 34, 2007.

[10] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Effective feature space reduction with imbalanced data for semantic concept detection," in *Proceedings of the 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (sutc 2008)*, ser. SUTC '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 262–269. [Online]. Available: http://dx.doi.org/10.1109/SUTC.2008.66

[11] R. Yang, D. Sun, and J. Tang, "A "sufficient" condition combination for rapid intensifications of tropical cyclones," *GEOPHYSICAL RESEARCH LETTERS*, vol. 35, no. 20, p. L20802, Oct 2008.

[12] L. Lin and M.-L. Shyu, "Mining high-level features from video using associations and correlations," in *Proceedings of the 2009 IEEE International Conference on Semantic Computing*, ser. ICSC '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 137–144. [Online]. Available: http://dx.doi.org/10.1109/ICSC.2009.59

[13] J. Tang, "Analysis of north atlantic tropical cyclone intensify change using data mining," Ph.D. dissertation, George Mason University, Fairfax, VA, USA, 2010, aAI3411108.

[14] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Video semantic concept discovery using multimodal-based association classification," in *Multimedia and Expo, 2007 IEEE International Conference on*, july 2007, pp. 859 –862.

[15] M. DeMaria and J. Kaplan, "An updated statistical hurricane intensity prediction scheme (ships) for the atlantic and eastern north pacific basins," *Weather and Forecasting*, vol. 14, no. 3, pp. 326–337, 1999.

[16] M. DeMaria, M. Mainelli, L. K. Shay, J. A. Knaff, and J. Kaplan, "Further improvements to the statistical hurricane intensity prediction scheme (ships)," *Weather and Forecasting*, vol. 20, no. 4, pp. 531–543, 2005.

[17] M. DeMaria, "A simplified dynamical system for tropical cyclone intensity prediction," *Monthly Weather Review*, vol. 137, no. 1, pp. 68–82, 2009.

[18] B. Jarvinen and C. Neumann, *Statistical forecasts of tropical cyclone intensity for the North Atlantic basin*. NOAA Tech. Memo NWS NHC-10, 22 pp., 1979.

[19] J. A. Knaff, M. DeMaria, C. R. Sampson, and J. M. Gross, "Statistical, 5-day tropical cyclone intensity forecasts derived from climatology and persistence," *Weather and Forecasting*, vol. 18, no. 1, pp. 80–92, 2003.

[20] J. A. Knaff, C. R. Sampson, and M. DeMaria, "An operational statistical typhoon intensity prediction scheme for the western north pacific," *Weather and Forecasting*, vol. 20, no. 4, pp. 688–699, 2005.

[21] M. DeMaria, J. A. Knaff, and C. Sampson, "Evaluation of long-term trends in tropical cyclone intensity forecasts," *Meteorology and Atmospheric Physics*, vol. 97, no. 1-4, pp. 19–28, Aug 2007.

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009. [Online]. Available: http://doi.acm.org/10.1145/1656274.1656278

[23] Weka 3: Data mining software in java. [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/

[24] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers, January 2011.

[25] V. N. Varpnik, *The nature of statistical learning theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.

[26] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in kernel methods: Support Vector Machines*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA, USA: MIT Press, December 1998, pp. 185–208. [Online]. Available: http://dl.acm.org/citation.cfm?id=299094.299105

[27] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to platt's smo algorithm for svm classifier design," *Neural Comput.*, vol. 13, no. 3, pp. 637–649, Mar. 2001. [Online]. Available: http://dx.doi.org/10.1162/089976601300014493

[28] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Mach. Learn.*, vol. 6, no. 1, pp. 37–66, Jan. 1991. [Online]. Available: http://dx.doi.org/10.1023/A:1022689900470

[29] R. L. Bankert, M. Hadjimichael, A. P. Kuciauskas, K. L. Richardson, J. Turk, and J. D. Hawkins, "Automating the estimation of various meteorological parameters using satellite data and machine learning techniques," *Geoscience and Remote Sensing Symposium, 2002. IGARSS '02. 2002 IEEE International*, vol. 2, pp. 708–710 vol.2, 2002.

[30] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[31] W. Li, C. Yang, and D. Sun, "Mining geophysical parameters through decision-tree analysis to determine correlation with tropical cyclone development," *Comput. Geosci.*, vol. 35, pp. 309–316, February 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1497634.1497854

[32] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001. [Online]. Available: http://dx.doi.org/10.1023/A:1010933404324

[33] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *International Conference on Machine Learning*. MORGAN KAUFMANN PUBLISHERS, INC., 1996, pp. 148–156.

[34] D. E. Rumelhart, J. L. McClelland, and T. P. R. Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MS, USA: The MIT Press, 1986, vol. 1.

[35] J. J. Baik and H. S. Hwang, "Tropical cyclone intensity prediction using regression method and neural network," *Journal of the Meteorological Society of Japan*, vol. 76, no. 5, pp. 711–717, 1998.

[36] J. J. Baik and J. S. Paek, "A neural network model for predicting typhoon intensity," *Journal of the Meteorological Society of Japan*, vol. 78, no. 6, pp. 857–869, 2000.

[37] J. N. K. Liu and B. Feng, "A neural network regression model for tropical cyclone forecast," *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 7, pp. 4122–4128 Vol. 7, 18-21 Aug. 2005.

[38] L. Jin, C. Yao, and X. Y. Huang, "A nonlinear artificial intelligence ensemble prediction model for typhoon intensity," *Monthly Weather Review*, vol. 136, no. 12, pp. 4541–4554, 2008.

[39] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, ser. UAI'95. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995, pp. 338–345. [Online]. Available: http://dl.acm.org/citation.cfm?id=2074158.2074196

[40] Y. Su, S. Chelluboina, M. Hahsler, and M. H. Dunham, "A new data mining model for hurricane intensity prediction," *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*, pp. 98–105, 13-13 Dec. 2010.

[41] M. H. Dunham, Y. Meng, and J. Huang, "Extensible markov model," *Data Mining, 2004. ICDM '04. Fourth IEEE International Conference on*, pp. 371–374, 1-4 Nov. 2004.

[42] J. Tang, R. Yang, and M. Kafatos, "Data mining for tropical cyclone intensity prediction," in *Sixth Conference on Coastal Atmospheric and Oceanic Prediction and Processes*, 2005.

[43] R. Yang, J. Tang, and D. Sun, "Association rule data mining applications for atlantic tropical cyclone intensity changes," *Weather and Forecasting*, vol. 26, no. 3, pp. 337–353, 2012/04/22 2011. [Online]. Available: http://dx.doi.org/10.1175/WAF-D-10-05029.1

[44] K. A. EMANUEL, "The maximum intensity of hurricanes," *Journal of the Atmospheric Sciences*, vol. 45, no. 7, pp. 1143–1155, APR 1 1988.

[45] Attribute-relation file format (arff). [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/arff.html

[46] J. Kaplan and M. DeMaria, "Large-scale characteristics of rapidly intensifying tropical cyclones in the north atlantic basin," *Weather and Forecasting*, vol. 18, no. 6, pp. 1093–1108, 2003.

[47] R. H. Simpson, "The hurricane disaster-potential scale," *Weatherwise*, vol. 27, no. 4, pp. 169–186, 1974.

[48] S. D. Aberson and M. DeMaria, "Verification of a nested barotropic hurricane track forecast model (vicbar)," *Monthly Weather Review*, vol. 122, no. 12, pp. 2804–2815, 2012/04/14 1994. [Online]. Available: http://dx.doi.org/10.1175/1520-0493(1994)122<2804:VOANBH>2.0.CO;2

[49] U. M. Fayyad and K. B. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," Jet Propulsion Laboratory, Tech. Rep., 1993.

[50] M. A. Hall, "Correlation-based feature subset selection for machine learning," Ph.D. dissertation, University of Waikato, 1998.

[51] Forecast verification: Issues, methods and faq. [Online]. Available: http://www.cawcr.gov.au/projects/verification/

[52] P. J. Roebber, "Visualizing Multiple Measures of Forecast Quality," *Weather and Forecasting, vol. 24, issue 2, p. 601*, vol. 24, p. 601, 2009.