*CS 390*

*“School Off” School Closing Prediction Application*

*Trung Nghia Nguyen, Ryan McCarthy, Andrew Millian*

*12/15/2018*

**1. Introduction**

Our goal is to develop an Android application—the “School Off” app—that predicts whether, given the current weather forecast, a particular school likely will close for weather reasons. The app makes its prediction based on weather data from previous times the school has closed due to inclement weather. Our project will allow users of the app to anticipate school closures and as a result, manage their schedules with greater confidence. We use data for George Mason University (“GMU”) to test the app and to show how it works. This problem is worthwhile to tackle because it allows students and faculty to make more informed decisions about the possibility of school closings and delays. Ultimately, we have succeeded in creating a functional Android application, known as School Off, that provides reasonably accurate predictions for George Mason University.

**2. Data Mining Overview**

Our program primarily uses the principles of data mining in its implementation. According to Microsoft SQL Server, “data mining is the process of sorting through large data sets to identify patterns and establish relationships to solve problems through data analysis. Data mining tools allow enterprises to predict future trends” (Rouse). Data mining involves not only trivial or obvious relationships but more subtle hidden ones that may only become perceivable with the aid of large-scale computer analysis. Such analysis is frequently used by governments, corporations and individuals to predict future trends with better accuracy and higher utility than traditional “gut feeling” models. The specific data mining technique we use is known as a neural network. A neural network consists of relatively simple processors, called neurons, that learn from inputs (Cripps). Given a set of past data and a result, the neural net will self optimize to categorize future data sets correctly (Cripps).

**3. The School Closing Problem**

Our question is to see if we can use past data regarding when schools have been closed due to weather in combination with current weather forecasts to predict whether or not the school will be closed on a particular day. We will use a neural network, similar to the one that Cripps uses to predict academic performance, to analyze the data and predict whether the school will be delayed or closed. This is similar to the Cripps problems, as both utilize numerous discrete numeric inputs to predict a definitive output, in Cripps’ case graduation rates and GPA, and in School Off’s case school delay, early dismissal, and closing (Cripps). Thus, the Cripps paper provides valuable guidance for our project.

This project is worthwhile for two main reasons. First, it is an interesting technical challenge, as it is fundamentally an estimation of the human response to weather, rather than the weather itself. Second, School Off is a practical, useful program that George Mason students can use to predict the actions of faculty, and faculty can use to ensure that their closing and delay decisions are consistent with past policy. Having foreknowledge of closing decisions is highly useful for scheduling purposes. For example, this information can help determine whether to schedule or change carpools, babysitters or meetings with faculty. Additionally, it can provide a warning to students to bring books and other study materials home from campus so that they can work from home when school will be closed. Furthermore, it allows those who schedule club meetings warning so that they can cancel or reschedule meetings before George Mason closes the campus and deprives them of their meeting place.

**4. Approach to Addressing the Question**

***4.1. Information gathering***

We use GMU’s closure, delay and early closing data that is archived on its website (George Mason University). We have combined a part of this list with randomly selected data in non-closing conditions and extreme conditions from our AccuWeather data set to populate a training set for our neural network (Houston, TX). In addition, we have a second set of sequestered data from the same sources that serves as a control group for testing (Houston, TX).

***4.2. Research timeline***

The Milestones that we have met are as follows:

* Be able to retrieve and store past weather for days when school were closed.   
  We have successfully managed to retrieve and store said data in a standard SQL database. This provides easy, standardized storage of the data, and allows it to be easily manipulated.
* Be able to retrieve and manipulate weather data for a week in advance.   
  We have accomplished this, although we usually only check two days in advance, due to degrading accuracy.
* Develop a method of determining whether the weather for a particular day will or will not result in school being cancelled.   
  Our neural network can determine with an acceptable degree of accuracy whether or not school will be closed, via testing on sequestered data. [Citation here, replace with numbers]
* Develop a simple to use UI in Android Studio.   
  This has been accomplished, and the UI works correctly. [Picture here]
* Develop a simple architecture such that our UI and data elements are protected from one another.   
  You cannot access the underlying data from the customer facing user interface. This protects our data and ensures that the customer cannot access it.

*Unfinished Milestones Include:*

* Expansion to additional schools.  
  This was a stretch goal that we did not fulfill, largely due to time constraints and difficulty in acquiring relevant data.
* Further development of the UI for appearance, as well as function  
  We have a basic, functional UI, that looks decent. However, we have not extensively customized or improved the user interface.

***4.3. Research plan***

We have worked together using some elements from agile software development in order to maintain flexibility with unexpected problems we have come across. Specifically, we have used an iterative development process, allowing us to adjust and modify the project as new obstacles and opportunities present themselves. For example, we have further studied prediction methods to achieve an accurate result, and have decided to use a neural network similar to the one used by Cripps in the paper “Using Artificial Neural Nets to Predict Academic Performance” (Cripps). We also considered alternative solutions such as supervised learning or trying to manually discover the relationships between our inputs, but discarded these alternatives as impractical.

In order to predict the weather, we have implemented an unsupervised machine learning technique, specifically a 2 layer neural network (See Fig 1.). Rather than try to come up with our own heuristics for what constitutes a day off, we concluded that developing a neural net will be more accurate. Our net has an input layer of 8 nodes, a hidden layer of 11 nodes, and an output layer of 1 node. This takes in a set of “training data” consisting of a day with its associated weather data in addition to whether or not the school was closed. The net then adjusted itself using backpropagation in order to correctly weight what values are contributing to having a day off. In other words, when we run a vector from the training data, it will try to guess the correct output based on the current edge weights of the neural net. It will then see how wrong it is and adjust the values that contributed to the wrong answer accordingly. The network had an input vector of <temperature, humidity, wind speed, precipitation type, amount of precipitation, previous day’s precipitation type, previous day’s precipitation amount>. Our output vector is <whether or not the school was, delayed closed early or canceled>. We also used sigmoid as our step function as to handle the continuous data types. Sigmoid is a function that that makes the math for the neural network much simpler. We have made the neural network by doing a k-fold sampling, where it trained the net on 1/k of the data points we have and tested it against 1-1/k of the data k times. We then selected the net with the smallest error. We then hard-coded the edge weights into our app in addition to a forward propagate method. Fig 2. Is the matrix representation of the V values (edge weights that connect the input layer to the hidden layer), and Fig 3. Is the matrix representation of the W values (edge weights that connect the hidden layer to the output layer)

***4.4 Evaluation plan***

The main assessment of our work is essentially the accuracy of our prediction, which is almost purely based on the actual real-life weather conditions. However, we have simulated a bad weather condition by manually feeding our system data and see whether its prediction lines up with actual past results. In addition, we have checked to make sure that all the UI elements of our app properly do what they are supposed to do and work on different devices. This can be done by simulating various devices on Android Studio with different screen sizes and resolutions. The results indicate that our devices will function to an acceptable degree.

***4.5 Resources***

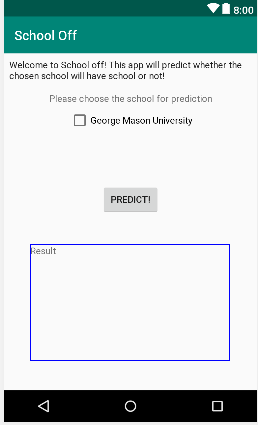
For our project we developed for Android platforms and as a result used Android Studio, which is the most popular IDE for Android and comes with a number of packages that we found useful. We also used OpenWeatherMap for our weather API. OpenWeatherMap provided the weather information we needed to make our prediction, is open source, and has been used by other popular weather apps such as Weather Feed. We also used Github as our version control platform where our source code is stored. Lastly, we located and retrieved past records of school closings for George Mason to provide the data we needed to make our prediction.

***4.6. Results***

We were able to confirm that data is available for George Mason University, our main target school. We have also completed our implementation of the neural network, as well as the user interface of the Android app. Furthermore, we have combined the two parts of our program into a single, functioning whole. From the k/fold testing, we have found our neural net to have a mean error of 0.005031323 when testing against a set of 20 sequestered data points. In addition, we used our app for the past 9 days to see if it would correctly predict any delays or cancellations. We found that our app correctly predicted that school would be open every day with 100% accuracy.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Day | 12/4 | 12/5 | 12/6 | 12/7 | 12/8 | 12/9 | 12/10 | 12/11 | 12/12 |
| Prediction | Open | Open | Open | Open | Open | Open | Open | Open | Open |
| Actual | Open | Open | Open | Open | Open | Open | Open | Open | Open |

Front-end example:



***4.7 Discussion***

We have moved forward roughly on schedule, and have finished our initial basic guidelines for functionality. One issue to note is that in the available data sets the accumulation of precipitation (snow) is not given as a numerical measurement, but rather as words such as “light rain” or “heavy snowfall.” We have converted the available precipitation data into quantitative values to feed into our neural network. We evaluated our result by the number of stages that we have completed. [Under Construction]

**5. Conclusion**

1. Conclusions:  
   In conclusion, we have succeeded in accomplishing our goals. We have created a LISP neural network to train on our data and implemented the results in our android application, Degree Works. Furthermore, Degree Works is capable of downloading weather data from the Open Weather API and using it in its program.
2. Summary of Contributions  
   All members of our project contributed greatly to our project, and no one failed to provide meaningful assistance. Andrew Millian was the lead writer on our report and presentation. Additionally, he provided debugging support for our program, giving valuable insight. Trung was able to come up with the initial project idea, as well research the topic and available data. Trung is also in charge of designing the Android App UI. Ryan was also highly helpful during the project, as he was in charge of creating the neural net in LISP.
3. Future Research  
   There are several promising avenues for future research and implementation. The first avenue is expanding School Off to more schools, gathering and training data for other educational institutions. However, any future researchers will have to be careful to ensure that any data they gather is complementary, as the amount of snow required to cause George Mason to close, for example, will be quite different from what the University of Alaska can tolerate before closing. Additionally, this method can be used to predict the closing of other institutions, such as the federal government, public parks or outdoor attractions such as theme parks. Finally, a future researcher could reverse the problem, and instead of attempting to predict closings, they could use a similar program to determine closings impartially. This would bring predictability and stability to the closing process, instead of essentially relying on the whims of the school administration.

**6. References**

Cripps, Al. “Using Artificial Neural Nets to Predict Academic Performance.” Proceedings of the 1996 ACM Symposium on Applied Computing - SAC '96, 1996, doi:10.1145/331119.331137.

“George Mason University.” Degree Works for Students | Office of the University Registrar | George Mason University, George Mason University, registrar.gmu.edu/facultystaff/scheduling/class-cancellations/.

“Houston, TX.” Local Weather from AccuWeather.com - Superior Accuracy™, www.accuweather.com/.

Rouse, Margaret. “What Is Data Mining? - Definition from WhatIs.com.” SearchSQLServer, Mar. 2017, searchsqlserver.techtarget.com/definition/data-mining.

**7. Appendices**

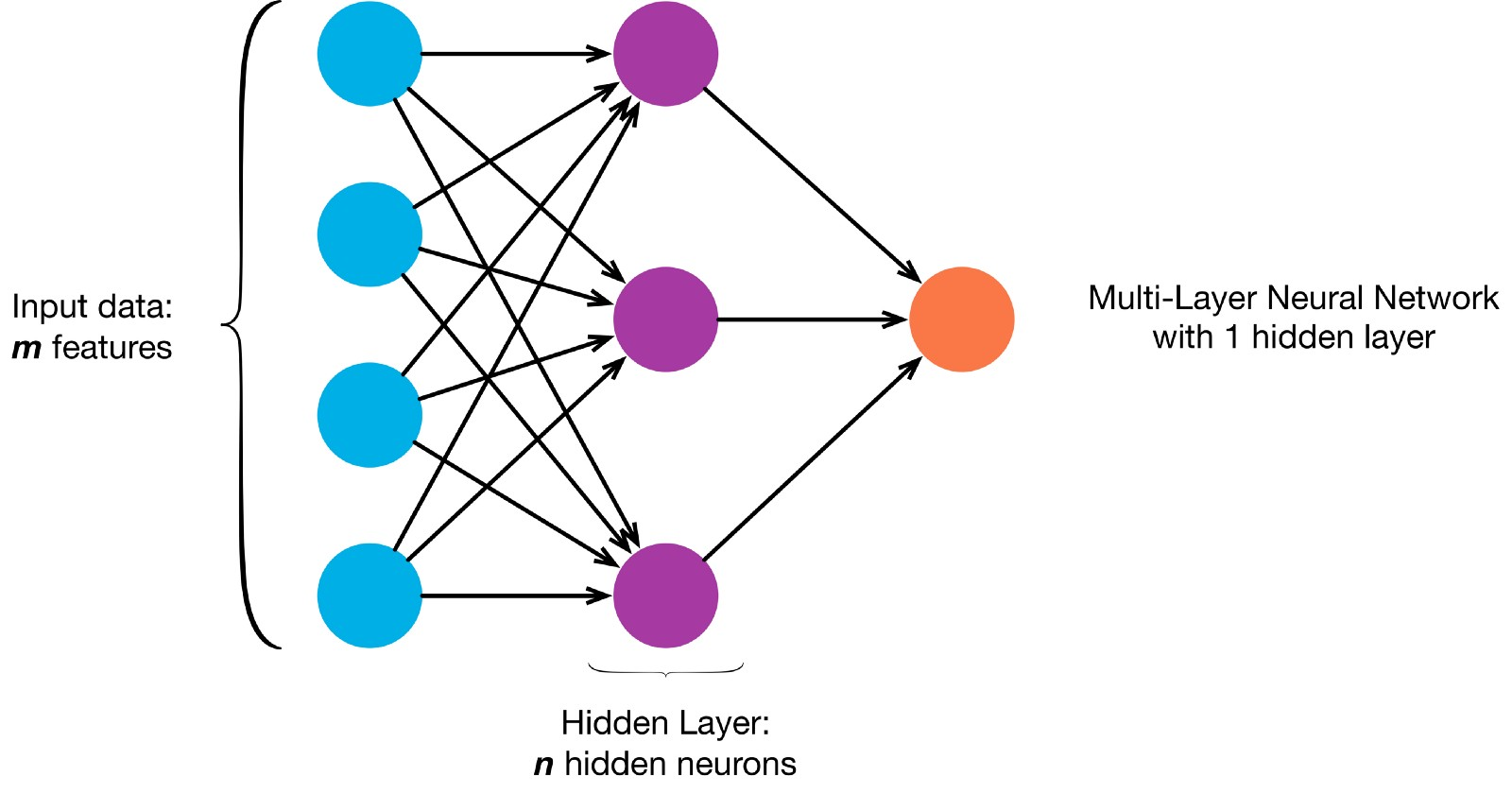
Fig 1. **Neural Net** 

Fig 2. **V**

(4.7013874 -4.7174378 -1.0369709 -4.956034 2.5164626 0.5604324)

(-1.5886842 -0.38823006 -1.631562 -1.2673955 0.53925777 0.6615484) (-1.9173675 -0.34544966 -1.5111042 -1.4408562 -0.8266896 -0.003480704) (-1.3548119 0.17843686 -1.452835 -1.6177709 -0.7073143 0.47258297) (-1.8377072 -0.25232753 -2.2538448 2.1703732 -1.424829 -2.0900054) (0.23770707 0.62467766 1.3459951 6.8916216 4.0529995 0.7274381) (-5.260584 0.6071666 -3.2904344 8.359922 -5.615922 -2.371472)

(2.5081337 -11.754191 0.8108118 3.8488455 -3.0932854 1.6069659)

(-3.087058 -0.098526515 -2.3476894 1.5305188 3.125813 0.123444475) (-0.49285176 3.219892 1.057497 -11.691372 -11.871463 -3.7923734) (-2.0161963 -0.7169333 -2.0390017 1.02494 0.96613497 -0.40423346)

Fig 3. **W**

(-5.7572065 -0.14543024 -0.26810464 0.33025634 -1.6753031 2.559705 -7.2792063 10.256568 1.90281 -7.403379 -0.15245272)