Data Mining for Automatic Linguistic Description of Data
Textual Weather Prediction as a Classification Problem

J. Janeiro, I. Rodriguez-Fdez, A. Ramos-Soto and A. Bugarín
CITIUS, University of Santiago de Compostela, Santiago de Compostela, Spain

Keywords: Linguistic Descriptions of Data, Natural Language Generation, Weather Forecasting, Classification.

Abstract: In this paper we present the results and performance of five different classifiers applied to the task of automatically generating textual weather forecasts from raw meteorological data. The type of forecasts this methodology can be applied to are template-based ones, which can be transformed into an intermediate language that can directly mapped to classes (or values of variables). Experimental validation and tests of statistical significance were conducted using nine datasets from three real meteorological publicly accessible websites, showing that RandomForest, IBk and PART are statistically the best classifiers for this task in terms of F-Score, with RandomForest providing slightly better results.

1 INTRODUCTION

Weather forecasting has been one of the most scientifically and technologically challenging problems around the world in the last century. To make an accurate prediction is one of the major challenges meteorologists face on a daily basis. Weather forecasts are made by collecting quantitative data about the current state of the atmosphere on a given place and using scientific understanding of atmospheric processes to project how the atmosphere will evolve on that place.

Modern weather forecasting is largely based on numerical weather predictions (NWP), which essentially are massive atmosphere simulations run on supercomputers. The output of NWP models is a set of predictions of meteorological parameters or variables (wind speed, temperature, precipitation, etc) for various spatial locations and at various points in time.

Weather forecasting organizations take NWP data and modify it according to their local knowledge and expertise. They also interpolate between the locations in the source NWP model, again using local knowledge and expertise. The result is a modified set of predicted numerical weather values, for locations of interest to their customers.

Initially, the NWP data was used by expert meteorologists to manually describe the weather forecast using texts for different places. With the increasing accuracy of predictions and the need to generate textual forecasts for a large number of locations, weather forecasting organizations require solutions which automatically build these texts.

There are several official meteorological agencies that offer weather forecast services, such as the Spanish AEMET (AEMET, 2014), American NWS (NWF, 2014) or the British Met Office (MetOffice, 2014b). Other private organizations like WeatherForecast (WeatherForecast, 2014) or Intellicast (Intellicast, 2014) offer their own forecast services. Some of them provide forecast data for specific domains, such as skying or surfing, allowing users to find the best conditions in which to perform this kind of activities. Furthermore, due to the need to provide textual forecasts to an increasing number of locations, some meteorological agencies started offering automatically generated forecast texts. For instance, in the 1990s, NLG systems such as FoG (Goldberg et al., 1994) and MultiMeteo (Coch, 1998), were used by meteorological agencies to provide this kind of information services. More recently, the Met Office with Data2Text (MetOffice, 2014a) or the Galician MeteoGalicia with GALiWeather (Ramos Soto et al., 2014) are also employing this sort of technology to address the creation of textual forecasts for increasing quantities of localized data.

Several techniques can be used for automated generation of weather forecast texts. These techniques can be divided into two broad categories: knowledge-intensive (KI) and knowledge-light (KL) approaches (Adeyanju, 2012). KI approaches require extensive consultation with domain experts during data analysis and throughout the text generation approach devel-

Janeiro J., Rodriguez-Fdez I., Ramos-Soto A. and Bugarín A..
Data Mining for Automatic Linguistic Description of Data - Textual Weather Prediction as a Classification Problem.
DOI: 10.5220/0005282905560562
In *Proceedings of the International Conference on Agents and Artificial Intelligence* (ICAART-2015), pages 556-562
ISBN: 978-989-758-074-1

opment process. On the other hand, KL approaches rely more on the use of automated methods which are mainly statistical.

The earliest KI systems generated forecast texts by inserting numeric values in standard manually-created templates. Multiple templates are created for each possible scenario and one of them is randomly selected during text generation to provide variety. Other KI systems developed linguistic models using manually-authored rules obtained from domain experts and corpus analysis.

The KL approach to generate forecast texts typically employs machine learning techniques. Trainable systems are built using models based on statistical methods such as probabilistic context-free grammars and phrase based machine translation. The advantage is that systems are built in less time and with less human effort as compared to the KI approach.

In this paper we consider forecast services with a KI approach and use these templated textual forecasts to obtain linguistic predictions presented as a classification problem to generate natural language (NLG) descriptions. The paper is organized as follows: in section 2 we present the problem and the different types of automatic textual forecasts. In section 3 we provide the steps needed to solve it using classification techniques. In section 4 we explain the five different classification techniques tested, the results obtained for each one and a statistical comparison between them and, finally, we present the most relevant conclusions of this approach.

2 LINGUISTIC WEATHER PREDICTIONS AS A NLG PROBLEM

The generation of natural language text uses the NWP data and additional expert information to generate textual weather forecasts that are issued to the public. There are two main approaches for generating textual forecasts automatically (Van Deemter et al., 2005):

- Template-based systems are natural language generating systems that map their non-linguistic input directly to the linguistic surface structure. This linguistic structure may contain gaps that must be filled with linguistic structures that do not contain gaps. For example, a template such as ”[amount] rain at [time]”, where the gaps represented by [amount] and [time], can be filled with information from the data.
- Standard NLG systems, by contrast, use less direct mapping between input and surface form. These systems could start from the same input semantic representation subjecting it to a number of consecutive transformations until a surface structure results. Various NLG submodules would operate on it, jointly transforming the representation into an intermediate representation where lexical items and style of reference have been determined while linguistic morphology is still absent. This intermediate representation may in turn be transformed into a proper sentence in one of the available output languages.

The typical stages of natural language generation systems (Reiter et al., 2000), are:

- Content determination: Deciding what information to mention in the text.
- Document structuring: Overall organization of the information to convey.
- Aggregation: Merging of similar sentences to improve readability and naturalness.
- Lexical choice: Mapping words to concepts.
- Referring expression generation: Creating referring expressions that identify objects and regions.
- Realization: Creating the actual text, which should be correct according to the rules of syntax, morphology, and orthography.

The texts generated by these two approaches usually have a similar structure, from which we can extract the main information and apply data mining techniques to the raw data to generate the same forecasts. To achieve this, we applied classification algorithms that learn the textual forecasts using data samples. In the next example, using the temperature data values, we can learn the forecast text for the weekly temperature:

- Full forecast: *“Mostly dry. Warm. Mainly fresh winds.”*
- Daily Temperature values (°C): 21, 22, 20, 19, 20, 18, 19
- Learned textual temperature value: “Warm”

3 LINGUISTIC PREDICTIONS AS A CLASSIFICATION PROBLEM

From the two approaches for automatically generate textual forecasts explained before, we selected the template based forecasts since they are more abundant and they have a more regular structure that allows us to extract the relevant information from the text. To test the classification of these forecasts we need

to transform the textual forecast into a class, extracting the relevant information and building descriptive phrases. We selected three different datasets from the web that offer NWP data and a descriptive, template-based textual forecast. Then we transformed these textual forecasts into classes and used them along with the raw meteorological data to perform the classification.

3.1 Weather-forecast Dataset

Weather-Forecast (WeatherForecast, 2014) uses the Global Forecast System from the National Oceanic and Atmospheric Administration (NOAA) to get their raw forecast data and use their own computers to generate the actual forecasts. Their textual forecasts include information about precipitation, temperature and wind, as shown in the example that follows:

Mostly dry. Warm (max 29 °C *on Tue afternoon, min* 23 °C *on Wed night). Wind will be generally light.*

From this forecast service we can extract three datasets (one for each of the variables considered), as indicated in table 1. The selected samples from this service come from different locations worldwide. Some examples of the classes considered are:

- Precipitation: *"mostly dry", "light rain", "some drizzle", "moderate rain"*.
- Temperature: *"warm", "very mild", "freeze-thaw conditions"*.
- Wind: *"generally light", "increasing light to fresh winds", "mainly fresh ", "decreasing fresh to calm"*.

3.2 National Weather Service Dataset

The National Weather Service (NWF, 2014) is a component of the National Oceanic and Atmospheric Administration (NOAA). They provide weather, water, and climate data, forecasts and warnings for the U.S. territory. Their textual forecasts include information about precipitation, cloud coverage and wind:

A chance of showers, mainly before 11pm. Mostly cloudy, with a low around 60. West wind 3 to 5 mph.

From this forecast we can extract three datasets as indicated in table 2. The selected samples from this service come from different locations on the United States of America. Some class examples considered from this dataset are:

- Precipitation: *"chance showers", "showers likely", "scattered showers and thunderstorms", "slight chance showers then slight chance showers and thunderstorms"*.
- Cloud coverage: *"mostly cloudy", "partly sunny", "mostly clear", "sunny and hot"*.
- Wind: *"west", "calm becoming west", "northwest becoming calm", "west becoming northeast"*.

3.3 Intellicast Dataset

Intellicast (Intellicast, 2014) delivers site-specific forecasts for 60,000 sites in the U.S. and around the globe including detailed local forecasts to hurricane tracks and severe weather warnings to international conditions. Their textual forecasts include information about cloud coverage, precipitation, temperature and wind, for example:

Partly cloudy skies. Hot. High 93F. Winds WSW at 10 to 20 mph.

From this forecast service we can extract three datasets, one of them includes both cloud coverage and precipitation information as indicated in table 3. The selected samples from this service come from different locations worldwide. Some examples of the classes considered are:

- Mixed cloud coverage and precipitation: *"partly cloudy", "sunshine and clouds", "partly cloudy with thunderstorms", "mix of clouds and sun with the chance of isolated thunderstorm"*.
- Temperature: *"hot", "warm", "hot and humid", "very hot"*.
- Wind: *"WSW", "light and variable", "S decreasing"*.

4 EXPERIMENTAL SETUP

We evaluated the performance of five different classification techniques for the three datasets introduced previously using the data mining software "Weka" (Hall et al., 2009). We selected these ones to test different types of supervised learning techniques which, in general, provide comprehensible visual models. Other techniques such as Artificial Neural Networks were not considered due to its black box structure.

4.1 Classification Methods

The classification techniques applied are:

- J48 (Quinlan, 1993) is an open source Java implementation of the C4.5 algorithm. C4.5 builds decision trees from a set of training data. At each node of the tree, C4.5 chooses the attribute of the data that most effectively splits its set of samples

Table 1: Datasets from Weather-Forecast (WF datasets).

Precipitation			
Type	Classification	Origin	Real world
Features	22	(Real / Integer / Nominal)	(0/22/0)
Instances	83754	Classes	17
Missing values?			No

Temperature			
Type	Classification	Origin	Real world
Features	23	(Real / Integer / Nominal)	(1/22/0)
Instances	83754	Classes	5
Missing values?			No

Wind			
Type	Classification	Origin	Real world
Features	15	(Real / Integer / Nominal)	(1/13/0)
Instances	83754	Classes	42
Missing values?			No

Table 2: Datasets from National Weather Service (NWS datasets).

Precipitation			
Type	Classification	Origin	Real world
Features	27	(Real / Integer / Nominal)	(13/13/1)
Instances	93896	Classes	122
Missing values?			No

Cloud coverage			
Type	Classification	Origin	Real world
Features	40	(Real / Integer / Nominal)	(1/38/1)
Instances	93896	Classes	26
Missing values?			No

Wind			
Type	Classification	Origin	Real world
Features	38	(Real / Integer / Nominal)	(2/36/0)
Instances	93896	Classes	74
Missing values?			No

Table 3: Datasets from Intellicast (ICAST datasets).

Mixed cloud coverage and precipitation			
Type	Classification	Origin	Real world
Features	66	(Real / Integer / Nominal)	(4/48/14)
Instances	127118	Classes	631
Missing values?			No

Temperature			
Type	Classification	Origin	Real world
Features	53	(Real / Integer / Nominal)	(3/50/0)
Instances	127118	Classes	9
Missing values?			No

Wind			
Type	Classification	Origin	Real world
Features	26	(Real / Integer / Nominal)	(1/13/12)
Instances	127118	Classes	24
Missing values?			No

into subsets enriched in one class or the other. The splitting criterion is the normalized information gain (difference in entropy). The attribute with the highest normalized information gain is chosen to make the decision. The C4.5 algorithm then recurs on the smaller sublists.

- RandomForest (Breiman, 2001) is an ensemble learning method for classification (and regression) that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes output by individual trees. The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set, bagging repeatedly selects a bootstrap sample of the training set and fits trees to these samples. After training, predictions for unseen samples can be made by averaging the predictions from all the individual regression trees or by taking the majority vote in the case of decision trees.
- IBk (Aha et al., 1991) is a K-Nearest Neighbours classifier (k-NN). In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.
- BayesNet (Bouckaert, 2004) uses a Bayesian network for classification. A Bayesian network is a directed acyclic graph that encodes a joint probability distribution over a set of random variables. It is defined by the pair B =$\{G, \theta\}$ where G is the structure of the Bayesian network, θ is the vector of parameters that quantifies the probabilistic model and B represents a joint distribution PB(X), factored over the structure of the network. The goal of a Bayesian network classifier is to correctly predict the label for class given a vector of attributes. It models the joint distribution and converts it to a conditional distribution. The prediction for a class can be obtained by applying an estimator to the conditional distribution.
- PART (Frank and Witten, 1998) combines C4.5 and RIPPER to avoid their respective problems since it does not need to perform global optimization to produce accurate rule sets and this added simplicity is its main advantage. It adopts the divide-and-conquer strategy in that it builds a rule, remove the instances it covers, and continue creating rules recursively for the remaining instances until none are left. It differs from the standard approach in the way each rule is created. In essence, to make a single rule a pruned decision tree is built for the current set of instances, the leaf with the largest coverage is made into a rule and the tree is discarded. This avoid hasty generalization by only generalizing once the implications are known.

4.2 Classification results

We performed experimentation with a 10 fold cross-validation using the previously indicated classifiers on the datasets considered. The results obtained in each case are shown in tables 4, 5, and 6 for the F-Score, since it provides better accuracy measure (Sokolova et al., 2006), computed as a weighted average of the precision and recall, and the root-mean-square error (RMSE).

With the WF datasets (Table 4) we got results with a score over 0.96 in all cases for the best classifier, that was RandomForest.

With NWS (Table 5) we got similar results, close to 1 for the precipitation and cloud coverage datasets with four of the classifiers, but the results are poorer for the wind dataset. In this case IBk is the best classifier except for the Cloud Coverage dataset where is slightly improved by RandomForest, PART and J48.

With the ICAST datasets (Table 6) the results are good for the temperature and wind datasets and worse for the cloud coverage and precipitation mainly because the high amount of different classes. RandomForest is, once again, the best classifier only improved by IBk in one dataset.

4.3 Statistical Comparison

In order to provide quantitative evidences for supporting the results presented in tables 4, 5, and 6, we used the STAC platform (STAC, 2014) to perform the tests of statistical significance on the previously presented experimentation results, with the aim of determining if statistical differences existed among the performances achieved by the five classifiers. Since the samples do not follow a normal distribution, a nonparametric test had to be used, more specifically the Iman-Davenport test (Iman and Davenport, 1980) with a significance level of 0.05. The test results are presented in table 7, showing RandomForest as the best classifier followed by IBk, as it was expected analysing the previous results.

Additionally, we needed to verify that RandomForest statistically outperforms all the others classifiers. To confirm it, we applied a Finner test (Finner, 1993) with an alpha value of 0.05 to the classifica-

Table 4: Classification results for the WF datasets.

Weather-Forecast	Precipitation		Temperature		Wind	
	F-Score	RMSE	F-Score	RMSE	F-Score	RMSE
J48	0.965127	0.0396	0.999474	0.0121	0.944170	0.0344
RandomForest	0.968698	0.0361	0.999540	0.0112	0.962207	0.0265
IBk	0.963753	0.0396	0.997274	0.0279	0.959188	0.0302
BayesNet	0.949996	0.0477	0.974612	0.0899	0.809128	0.0639
PART	0.965273	0.0382	0.999248	0.0144	0.948923	0.0332

Table 5: Classification results for the NWS datasets.

National Weather Service	Precipitation		Cloud Coverage		Wind	
	F-Score	RMSE	F-Score	RMSE	F-Score	RMSE
J48	0.974298	0.019	0.993365	0.0216	0.785243	0.0221
RandomForest	0.978431	0.0166	0.994685	0.0196	0.820437	0.018
IBk	0.979700	0.0169	0.982945	0.0362	0.832648	0.0209
BayesNet	0.715449	0.0665	0.869578	0.1125	0.490977	0.0283
PART	0.973348	0.019	0.993160	0.0223	0.782653	0.0225

Table 6: Classification results for the ICAST datasets.

Intellicast	C. Cov. / Precipitation		Temperature		Wind	
	F-Score	RMSE	F-Score	RMSE	F-Score	RMSE
J48	0.883397	0.0178	0.987056	0.0505	0.965454	0.0507
RandomForest	0.906389	0.0146	0.989658	0.0403	0.977720	0.0391
IBk	0.907474	0.0164	0.988727	0.0458	0.963848	0.0537
BayesNet	0.551672	0.0348	0.780627	0.2556	0.758892	0.1308
PART	0.885185	0.0179	0.987364	0.0495	0.965783	0.051

Table 7: Iman-Davenport Test results.

Ranking	Algorithms
1.444	RandomForest
2.667	IBk
2.778	PART
3.111	J48
5.000	BayesNet
p-value < 0.001	

Table 8: Finner Test results.

Control Method	Control Method VS	Adjusted p-value	Result
RandomForest	BayesNet	0.000	H0 is rejected
-	J48	0.050	H0 is rejected
-	PART	0.097	H0 is accepted
-	IBk	0.101	H0 is accepted

tion results. The null hypothesis (H0): "There is no difference between classifier A and classifier B" was accepted against IBk and PART and, in consequence, we cannot conclude that RandomForest is statistically better than IBk and PART as shown in table 8. For the other two classifiers the null hypothesis (H0) was rejected meaning that RandomForest is statistically better than both of them.

5 CONCLUSIONS

In this document we have presented a summary of the performance of five classifiers over nine different datasets from websites that provide textual meteorological forecasts. We tried to find which one of them obtains better classification results using the raw meteorological data to generate these textual forecasts.

In general terms all of the classifiers, with the exception of BayesNet, achieve good results. In terms of its F-Score results, RandomForest outperforms the rest, achieving the best F-Score in most of the cases (6/9), followed by IBk (3/9). According to this criteria we can sort these five classifiers by their performance: RandomForest, IBk, PART, J48 and BayesNet.

In order to verify and provide quantitative evidences for supporting these results, tests of statistical significance were performed to determine if statistical differences existed among the performances achieved by the five classifiers. The test results confirmed that RandomForest, IBk and PART are statistically the best classifiers, although RandomForest achieved slightly better results. On the other hand, J48 and BayesNet present significant performance differences, and therefore they do not show to be so valid from an experimental viewpoint.

In this work we have conceived a feasible model for providing rapid and accurate linguistic predictions in an intermediate language composed of linguistic labels. As future work we aim to extend this analysis to test with more powerful classification methods such as SVM or Artificial Neural Networks and develop a NLG-oriented approach which generate textual forecasts from the intermediate language obtained by the classifiers.

ACKNOWLEDGEMENTS

This work was supported by the Spanish Ministry of Economy and Competitiveness under grant TIN2011-29827-C02-02. I. Rodriguez-Fdez is supported by the Spanish Ministry of Education, under the FPU Fellowships Plan. A. Ramos-Soto is supported by the Spanish Ministry for Economy and Competitiveness (FPI Fellowship Program). This work was also supported in part by the European Regional Development Fund (ERDF/FEDER) under grants CN2012/151 and GRC2014/030 of the Galician Ministry of Education.

REFERENCES

Adeyanju, I. (2012). Generating weather forecast texts with case based reasoning. *International Journal of Computer Applications*, 45.

AEMET (2014). Spanish meteorological agency website. *http://www.aemet.es/*, Retrieved: 2014-10-08.

Aha, D. W., Kibler, D., and Albert, M. K. (1991). Instance-based learning algorithms. *Machine learning*, 6(1):37–66.

Bouckaert, R. R. (2004). *Bayesian network classifiers in weka*. Department of Computer Science, University of Waikato.

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

Coch, J. (1998). Multimeteo: multilingual production of weather forecasts. *ELRA Newsletter*, 3(2).

Finner, H. (1993). On a monotonicity problem in step-down multiple test procedures. *Journal of the American Statistical Association*, 88(423):920–923.

Frank, E. and Witten, I. H. (1998). Generating accurate rule sets without global optimization. In Shavlik, J., editor, *Fifteenth International Conference on Machine Learning*, pages 144–151. Morgan Kaufmann.

Goldberg, E., Driedger, N., and Kittredge, R. I. (1994). Using natural-language processing to produce weather forecasts. *IEEE Expert*, 9(2):45–53.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Iman, R. L. and Davenport, J. M. (1980). Approximations of the critical region of the fbietkan statistic. *Communications in Statistics-Theory and Methods*, 9(6):571–595.

Intellicast (2014). Intellicast website. *http://www.intellicast.com/*, Retrieved: 2014-10-08.

MetOffice (2014a). British meteorological office data2text website. *http://www.metoffice.gov.uk/public/weather/forecast-data2text*, Retrieved: 2014-10-08.

MetOffice (2014b). British meteorological office website. *http://www.metoffice.gov.uk/*, Retrieved: 2014-10-08.

NWF (2014). National weather forecast website. *http://www.weather.gov/*, Retrieved: 2014-10-08.

Quinlan, J. R. (1993). *C4. 5: programs for machine learning*, volume 1. Morgan kaufmann.

Ramos Soto, A., Bugarin, A., Barro, S., and Taboada, J. (2014). Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data. *IEEE Transactions on Fuzzy Systems*, Early Access.

Reiter, E., Dale, R., and Feng, Z. (2000). *Building natural language generation systems*, volume 33. MIT Press.

Sokolova, M., Japkowicz, N., and Szpakowicz, S. (2006). Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In *AI 2006: Advances in Artificial Intelligence*, pages 1015–1021. Springer.

STAC (2014). Stac: Web platform for algorithms comparison through statistical tests. *http://tec.citius.usc.es/stac/*, Retrieved: 2014-10-08.

Van Deemter, K., Krahmer, E., and Theune, M. (2005). Real versus template-based natural language generation: A false opposition? *Computational Linguistics*, 31(1):15–24.

WeatherForecast (2014). Weather-forecast website. *http://www.weather-forecast.com/*, Retrieved: 2014-10-08.