

Mining Aviation Safety Data: A Hybrid Approach

Eric Bloedorn
The MITRE Corporation
bloedorn@mitre.org

ABSTRACT

Data mining is broadly defined as the search for interesting patterns from large amounts of data. Techniques for performing data mining come from a wide variety of disciplines including traditional statistics, machine learning, and information retrieval. While this means that for any given application there is probably some “data mining” technique for finding interesting patterns, it also means there exists a confusing array of possible data mining tools and approaches for any given application. This problem is exacerbated when the available data contains both structured as well as unstructured (free-text) data. For example, the aviation safety data used in the reported experiments contains records which include both free text event descriptions as well as structured fields for phase-of-flight and location. Performing separate analysis on these different sources of data does not fully exploit the available information (e.g. clustering records without regard to narratives can match reports of total electrical failure with human factors problems). Unfortunately currently available tools provide little support. This paper describes one approach to combining the information available from all of these different types of data together to get a single ‘similarity’ score. The importance of picking tools appropriate to the types of data in hand is also stressed.

INTRODUCTION

As a new field, data mining has a large number of tools and associated it. While this means the field has much to offer, it makes it difficult for data owners, new to this type of analysis, to decide which tools are appropriate. One way to cope with this selection task is to let the data drive the decision. Different data types require different analysis methods.

One useful way of organizing the discussion of data types is to break these types down by structure. Structure can refer to either the structure of the entire record, or the structure of an individual field. For the most part, this discussion will assume that each record is highly structured, i.e. that each record has a fixed number of fields (attributes) and that these fields are in a known order. The problems of dealing with semi-structured data in which a fixed schema can not adequately describe the data organization are described in Seligman et al, [1998]. The following sections outline four different data types: quantitative (interval and ratio), ordinal, nominal and free-text. For each data type a definition of that type as well as suggested analysis methods is provided.

Different Analyses for Different Data

Data mining has traditionally been associated with the analysis of the flat attribute/value type of format available from relational databases, that is data in the form we are most familiar with as rows of numbers or strings with fixed length. The rows represent records or examples while the columns represent fields or attributes of that record. For example, in the NASA Aviation Safety Reporting System (ASRS) [<http://asrs.arc.nasa.gov/>] reports, a record (or row) is an incident of interest to aviation safety. This could be a description of a pilot's failure to properly yield at a hold point on a runway, or a description of how two aircraft had less than legal separation. Other fields in this record provide, for example, details on the weather, the experience of the pilots and the types of planes involved for that incident.

Quantitative Data

At one end of the structure spectrum are data whose values have not only an ordering between them (ordinal), but whose differences and ratios are meaningful. An average attribute value for a set of records, or the distance between the value of record1 and record2 are easy to define when all of the values are numeric. With numeric data, classification methods like k-nearest neighbor (k-nn), backpropagation, or multivariate regression, and clustering methods like k-means are well suited. Each of these methods makes use of the structure of numeric data to build distance functions, gradients or confidence intervals of the data to partition the space effectively. K-means, for example, uses a distance function to search for the partitioning of records such that the intra-cluster distance is minimal and the inter-cluster distance is maximum. Interestingly one common type of analysis that is not well suited to entirely numeric data is association rules or 'market basket analysis'. The large number of unique values found in numeric data is ill suited to the direct match and counting performed by association methods. This drawback is unusual, however, as most methods can be applied to entirely numeric data. The hybrid tool that we have developed uses a standard squared distance function to match numeric data.

Ordinal

When, like in ASRS reports, the data also contain non-numeric attributes that are still ordered (like phase of flight), many of these same tools can be used, as long as these values are mapped to a set of integers. While an average phase-of-flight score of "cruise.5" may not be meaningful, it is still appropriate to say that the distance between the value "cruise" and the value "takeoff" is less than the distance between "takeoff" and "landing". This difference can be used to drive classification as in k-nn or clustering using k-means.

Categorical Data

When attributes (like plane-type) present in the data are completely unordered (i.e. categorical), or have a hierarchical ordering, other tools should be used. Because of their

interest in building tools that are comprehensible to people, machine learning (ML) programs are designed to find patterns in this kind hierarchically ordered, or unordered data. Tools like AQ [Wnek et. al, 1995] for classification and CLUSTER [Michalski and Stepp, 1983] for conceptual clustering provide full support for exploiting available hierarchical information to search for patterns. Other machine learning methods like c4.5 [Quinlan, 1992] can be used in these cases, but such ordering information must be recoded as additional attributes.¹ Without distance or gradient information to guide search, these programs perform a non-exhaustive search (for example a beam search) to find patterns in a reasonable time. While statistical methods don't exploit hierarchical information directly, there are techniques like loglinear modeling and logistic regression for categorical data. Unfortunately the application of these methods to even moderately sized data sets (30 variables, 10⁴ cases) requires a Ph.D. in statistics, extensive experience and expensive hardware [DuMouchel, 1997].

Free-Text

Free-text data has the least amount of structure. The length of each value is variable, and the strings found in that field may be drawn from a vocabulary of tens of thousands of words. In addition many of these words can be syntactically different but mean the same thing, or they can be syntactically the same and have different meanings. These characteristics make analysis of free-text very difficult.

One way of dealing with free-text is to add structure. If the items of interest can be extracted from the free-text, then new, structured attributes can be populated based on the values extracted from the text. Tools for the extraction of proper nouns from arbitrary text are now available commercially from IBM –Intelligent Miner for Text, and SRA's NetOWL. These methods allow proper nouns naming people, organizations and locations to be extracted and collected from a text document. If this variable length list of names is then fed to an association-rule tool, associations between these names can be discovered. When applied to newswire text, [Clifton, 1997] reports finding patterns such as the one found between the leader of Zaire (Mobuto Sese Seko), the rebel Laurent Kabila, and the capital city Kinshasa (p. 57).

Another way of exploiting the information available in text is to try to use the texts directly. One popular method is to treat the entire field as a single large 'bag of words'. This 'bag of words' method ignores the structure of the text and considers the field value to be a set of strings that either match or not. In this method a word-by-document index is built to reduce the time needed to perform a match between two documents (or fields from different records). Then a weighted sum of the word-by-word matches is used to provide the final overall 'match' score. Variants of this vector-space' approach [Salton, McGill 83] are widely used in search engines to rapidly provide matching documents to user-supplied queries. Classification and clustering can both be done with this underlying model providing 'distances' between documents.

¹ Hierarchical information can be very useful. We found that it improved classification accuracy in a text classification task by 72% in one case [Bloedorn, Mani and MacMillan, 1996].

This previous discussion has provided some pointers to how different kinds of data can be analyzed as part of a data-mining task. While many of the tools mentioned (e.g. backpropagation, c5.0) are effective for some combinations of data types, especially numeric and categorical, there is little support for combinations that include free-text. When presented with data that contains a mix of free-text and other data, either the text is ignored, or two separate analyses are performed. The goal of this research was to find a method that better utilized the information found in both free-text and structured fields simultaneously.

DESCRIPTION OF METHOD

The hybrid approach we took for this project was motivated by a need we found when discussing data mining with airline safety officers. One task that they are repeatedly called on to perform is to find records of incidents that are similar to those that just recently occurred. If the new event is found to be similar to some past records, it may be part of a larger, more serious pattern. When this is the case, past actions taken to prevent this incident may have to be reviewed and updated. If, on the other hand, the incident is anomalous, it may be noted and closed, or simply announced to the relevant departments as a warning. Surprisingly this determination of record similarity was not well supported by the tools available to the safety officer. Officers could perform queries on both the structured and free-text fields, but these only respond with exact matches. Similarity of match between records was not supported.

To provide safety officers with a tool that found similar records from mixed free-text and structured data we took a hybrid approach. In this hybrid approach, a match between two records is evaluated as the weighted match between each of the available fields within those records. When doing this match we use methods that are appropriate for the type of attribute. More precisely this means

$$\text{Sim}(\text{record}_i, \text{record}_j) = w_1 * \text{match}(a_{1i}, a_{1j}) + w_2 * \text{match}(a_{2i}, a_{2j}) + \dots + w_n * \text{match}(a_{ni}, a_{nj})$$

Our system will support three different types of matching: 1) strict Boolean, 2) ordinal, or 3) vector-based. In strict matching, which is appropriate for nominal typed attribute such as location, the match function takes the following form:

$$\text{Match}(a_{ni}, a_{nj}) = \begin{cases} 1 & \text{if } a_{1i} = a_{1j} \\ \text{else} & = 0 \end{cases}$$

When the data are ordered, the system requires information from the user concerning the size and ordering of the domain. This matching is appropriate for any ordinal or interval type of data from numeric (e.g. Number_hours_flown) to string-based (Phase_of_flight). Given |Domain a|, the match function is

$$\text{Match}(a_{ni}, a_{nj}) = 1 - (a_{ni} - a_{nj}) / |\text{Domain } a_n|$$

When the data are textual, vector space matching is used. There are a number of different weighting schemes that could be supported, but by default the tool uses a tf-idf (term frequency inverse document frequency) method. In this approach, a vector with length equal to the size of the vocabulary is built for each field. The value at position x represents the ratio of the number of times that word appears in the document (term frequency or tf), and the number of times that word appears in the collection (document frequency or df). Geometrically speaking the overall document match is the distance in this large dimensional vector space between these two vectors, or the sum of the products over the square root of the sum of the squares.

$$\text{Match}(a_{ni}, a_{nj}) = \sum_{x=1toV} \frac{\text{weight}_{nix} * \text{weight}_{njx}}{\sqrt{(\text{weight}_{nix})^2 * (\text{weight}_{njx})^2}}$$

where:

V = size of vocabulary, weight_{nix} = (weight of word x in field n of record i) and the default weighting method is $\text{tf.idf} = (\text{term frequency}_{ix} / \text{document frequency}_x)$

The tool currently supports stemming, three different weighting schemes, the use of a stop word list, and the use of a thesaurus file for matching synonymous words. In stemming, words that are the same except for different endings (morphological variants e.g. engineered, and engineering) all map to the same base term (in this case engineer). Stop word lists are used to filter out words that are unlikely to add any additional meaning to the text. Examples of stop words are “and”, and “the”.

EVALUATION

While we are preparing for an evaluation by safety officers directly, we performed an initial evaluation on the hybrid matching method just described. This method has been coded in a tool called ‘Findsimilar’. In this evaluation three sets of records were extracted from the NASA ASRS collection. In order to get roughly equal classes, slightly different time ranges were used to extract each group. Group1 and group3 reports are taken from January to August of 1997. Group2 reports come from April to July 1997. All groups contain 15 fields; two of these fields are free-text, “narrative” and “synopsis,” and the other 13 are nominal type (binary match) attributes. Group1 contained 51 incident records describing the category “runway transgressions/unauthorized landings”. Group2 contained 49 incident reports from the category “runway transgressions/other” and group3 contained 43 “VFR (visual flight rules) in IMC (instrument meteorological conditions)”. The expectation is that the similarity between records within a group (intra-similarity) will be higher than the similarity reported between groups. In addition, we expect to see improved matching when using all of the fields over matching that uses just the fixed or just the words alone.

Findsimilar was run using three different sets of attributes. Table 1 shows the average similarity between records when both fixed and free-text fields are used to compute the

match. Table 2 shows the results when only the two free-text fields are used, and Table 3 uses only the fixed fields to calculate a match. As expected, the within-group similarity score is highest (shown in bold) in all runs. For the text-only match (table 2) the match between group1 and group2 (11.9) is higher than the match between group1 and group3 (8.7) or the match between group2 and group3 (7.2). While the between-group match between group1 and group3 (table 1 and table 2) is slightly higher than the between-group match of group1 and group2, the between group match of group2 and group 3 is consistently the smallest.

	Group1	Group2	Group3
Group1	27.1	23.7	24.2
Group2	23.7	24.5	20.1
Group3	24.2	20.1	29.2

Table 1. Average similarity scores calculated from both fixed and free-text fields

	Group1	Group2	Group3
Group1	14.5	11.9	8.7
Group2	11.9	22.3	7.2
Group3	8.7	7.2	10.7

Table 2. Average similarity scores calculated from just free-text fields

	Group1	Group2	Group3
Group1	30.1	26.9	27.5
Group2	26.9	27.7	23.1
Group3	27.5	23.1	33.2

Table 3. Average similarity scores calculated from just fixed fields

The top matches to the twenty-ninth record of group 3 (G3.29) provide a good example of the different results obtained for the different matching methods. When matching used both the text and fixed fields the top records was G3.34. When only fixed field matching was used the top record was G1.51 and when just the text was used the top match was G3.33. Here are the synopsis fields from each of these records

G3.29 (target record): PVT PLT RUNS INTO WX ON HIS XCOUNTRY TO HOUSTON. GETS LOST, USES A MAYDAY CALL FOR AN EMER DIVERSION TO DWH, NEAR HOUSTON. PLT HAD ALLOWED A 3 HR ELAPSED TIME PERIOD FROM WX BRIEFING TO TKOF.

G3.34 (top pick when using both text and fixed fields): C150 PLT FAILS TO CHK WX, GETS TRAPPED IN IMC ENRTE AND HAS TO CALL ATC FOR HELP. HE WASNOT IFR CERTIFIED. ATC VECTORED HIM TO A VFR ALTERNATE WHERE HE WAS CLRD FOR A VISUAL APCH.

G1.51 (top pick when matching used only fixed fields): <EMPTY>

G3.33 (top pick when using just text fields for matching): PVT PLT IN A C172 GETS LOST AFTER RUNNING INTO BAD WX AT NIGHT. NO FLT PLAN HAD BEEN FILED. PLT LOW TIME, NOT IFR RATED. WAS ON HIS WAY HOME TO ZZV FROM FL. HE HAD FAILED TO UPDATED ANY ENRTE WX.

The match between G3.29 and G3.34 is very good. Both records describe problems with bad weather, a call for help and an emergency diversion. The record G1.51 is from the runway excursions, unauthorized landings group, not weather problems group of the target so this is not a good match. There is no narrative at all for g1.51 so it is difficult to know more about what happened in this record. The reason for the high match between G1.51 and G3.29 is based on matches on the fixed attributes “acft_type”, “engine_type”, “flt_condition”, “lighting”, “num_engines”, and “quarter_day”. G3.33, the top pick when using just text is also a good match to g3.29. Like G3.29, G3.33 also talks about getting lost in bad weather and the use of a “May Day” call. One difference between G3.33 and g3.34 though, is that g3.34 is a record of an incident that occurred in the daylight (like g3.29) while g3.33 occurred at night. The fixed field “lighting” captured this information and allowed g3.34 to have a higher match score than g3.33. This example shows how information from both text and fixed fields can be exploited to provide a better overall match between incident reports than using just text or just fixed fields alone.

CONCLUSION

This paper introduced a hybrid approach to finding similar aviation safety incident reports. In this approach free-text matches between fields are combined with matched between fixed fields in order to obtain a more complete total match between any two records. The initial empirical results with this hybrid approach were shown to be promising for a collection of NASA ASRS reports, but further evaluation is required. In addition to a description of this new approach, this paper suggested that the type of data should be used to drive the types of tools used for data mining.

The author wishes to thank the other members of the data mining in aviation safety group working on this project at MITRE: John Fauntleroy, Kevin Forbes, Earl Harris and Zohreh Nazeri.

BIBLIOGRAPHY

Bloedorn, E., Mani, I., and MacMillan, T.R., “Representational Issues in Machine Learning of User Profiles”, Proceedings of the Thirteenth National Conference on Artificial Intelligence, AAAI Press, Portland, OR, p. 433-438, 1996.

Clifton, C., Rosenthal, A., and Ullman, J.D., "Knowledge Discovery in Text", First Federal Data Mining Symposium, AFCEA, Washington, DC, December 16-17, 1997.

DuMouchel, W., “Statistical Methods for Categorical Response Data”, Tutorial notes from the Third International Conference on Knowledge Discovery and Data Mining, Newport Beach, CA, 1997.

Michalski, R.S., and Stepp, R., “Learning from Observation: Conceptual Clustering, “Chapter in the book, Machine Learning: An Artificial Intelligence Approach, R.S. Michalski, J.G. Carbonell and T.M. Mitchell (Eds.), TIOGA Publishing Co., Palo Alto, P. 331-363, 1983.

Quinlan, J., “C4.5: Programs for Machine Learning”, Morgan Kaufmann, San Mateo, CA, 1992.

Seligman, L., K. Smith, I. Mani, and B. Gates, “Databases for Semistructured Data: How Useful are They? (position paper)” Proceedings of the International Workshop on Knowledge Representation and Databases (KRDB-98) at ACM-SIGMOD, Seattle, WA, May 1998.

Salton, G., M.J., McGill, “Introduction to Modern Information Retrieval, McGraw-Hill, New York, 1983.

Wnek, J., Kaufman, K., Bloedorn, E., and Michalski, R.S., “Selective Inductive Learning Method Aq15c: The Method and User’s Guide”, Reports of the Machine Learning and Inference Laboratory, MLI95-4, George Mason University, Fairfax, VA.

ABOUT THE AUTHOR

Eric Bloedorn is a lead staff member of the Artificial Intelligence Technical Center at the MITRE Corporation. His interests include machine learning and its application to text. Dr. Bloedorn received his B.A. degree in Physics from Lawrence University in 1989, and his M.Sc. and Ph.D. from George Mason University in 1992 and 1996 respectively. Dr. Bloedorn is a member of the American Association for Artificial Intelligence and the Association for Computing Machinery.

Dr. Bloedorn can be reached as follows:

email: bloedorn@mitre.org, phone: (703)883-5274,
fax: (703)883-1379