

On the Feasibility of Automatically Describing n -dimensional Objects

Pablo Ariel Duboue (pablo.duboue@gmail.com)
Les Laboratoires Foulab – 999 du College, Montreal, Québec

Abstract

This paper introduces the problem of generating descriptions of n -dimensional spatial data by decomposing it via model-based clustering. I apply the approach to the error function of supervised classification algorithms, a practical problem that uses Natural Language Generation for understanding the behaviour of a trained classifier. I demonstrate my system on a dataset taken from CoNLL shared tasks.

Thoughtland

- Generation of textual descriptions for n -dimensional data.
 - Early stage research
- Contributions
 - Introducing the problem
 - Describing a potential application and source of interesting n -dimensional objects
 - * The error function for a machine learning algorithm for particular training data
 - Preliminary work on a traditional NLG system
 - * McKeown's [McKeown, 1985] schemata and Gatt and Reiter's [Gatt and Reiter, 2009] SimpleNLG.
 - * Non-trivial NLG application easy to improve (good for classroom use)
 - Modular design: easy to add new machine learning libraries, clustering approaches, feature extraction and verbalization backends.
 - Free Software: GPLv3+ and it is written in Scala (easy extension in both Java and Scala + access to many ML libraries in Java)
- <http://thoughtland.duboue.net>

Input

a small data set from the UCI ML repo, the Auto-Mpg Data:

<http://archive.ics.uci.edu/ml/machinelearning-databases/auto-mpg/>

```
@relation auto_mpg
@attribute mpg numeric
@attribute cylinders numeric
@attribute displacement numeric
@attribute horsepower numeric
@attribute weight numeric
@attribute acceleration numeric
@attribute modelyear numeric
@attribute origin numeric

@data
18.0,8,307.0,130.0,3504.,12.0,70,1
14.0,8,455.0,225.0,3086.,10.0,70,1
24.0,4,113.0,95.00,2372.,15.0,70,3
22.0,6,198.0,95.00,2833.,15.5,70,1
27.0,4,97.00,88.00,2130.,14.5,70,3
26.0,4,97.00,46.00,1835.,20.5,70,2

... +400 more rows
```

Output

- MLP, 2 hidden layers (3, 2 units), acc. 65%, Thoughtland generates:

There are four components and eight dimensions. Components One, Two and Three are small. Components One, Two and Three are very dense. *Components Four, Three and One are all far from each other.* The rest are all at a good distance from each other.

- MLP, 1 hidden layer (8 units), acc. 65.7%, Thoughtland generates:

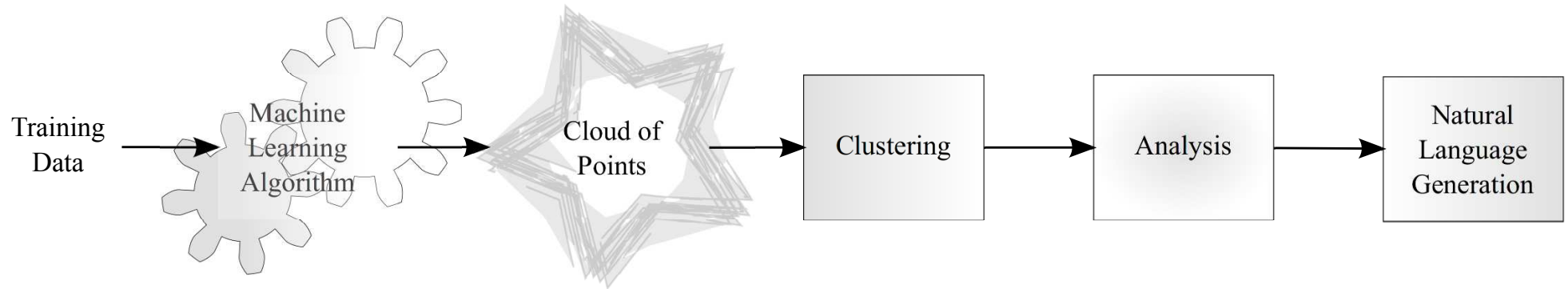
There are four components and eight dimensions. Components One, Two and Three are small. Components One, Two and Three are very dense. *Components Four and Three are far from each other.* The rest are all at a good distance from each other.

(difference is *highlighted*)

- MLP, 1 hidden layer (1 unit), acc. 58%, Thoughtland generates:

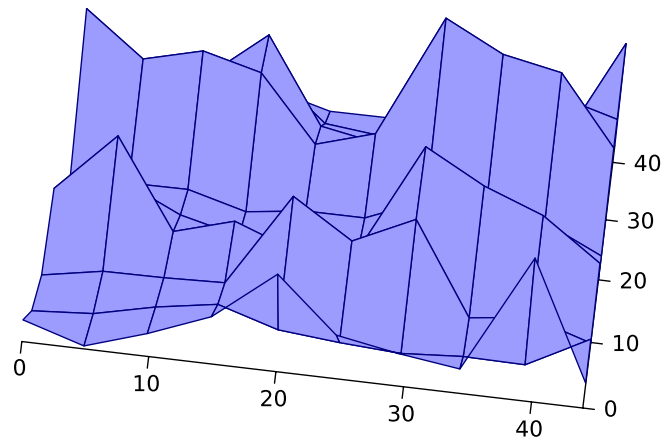
There are five components and eight dimensions. Components One, Two and Three are small and Component Four is giant. Components One, Two and Three are very dense. Components One and Four are at a good distance from each other. Components Two and Three are also at a good distance from each other. Components Two and Five are also at a good distance from each other. The rest are all far from each other.

Architecture



Machine Learning

- The error function is computed as the error for each point in the input data.
- For a numeric target class and training instance (\vec{x}, y) , $e = \|f(\vec{x}) - y\|$
 - f is trained on the folds that do not contain (\vec{x}, y) (cross-validation)
- For a nominal target class, the error is 1.0 if the class is different from the target or 0 if it the same.



Clustering

- The cloud of error points is clustered using a mixture of Dirichlet models [McCullagh and Yang, 2008].
 - As implemented by Apache Mahout [Owen et al., 2011]
 - This clustering approach has a geometrical representation in the form of n -balls (n -dimensional spheres)
- Some input features present a natural geometric groupings which will opaque the error function.
 - The error coordinate is re-scaled using the radius of an n -ball that encompasses all the input features


Analysis

- For each n -ball: determining the overall size, density, distances to the other n -balls.
- Put the numbers into perspective with respect to the n -ball encompassing the whole cloud of points
- This stage is at its infancy
 - Future work: analyze the pairs of n -balls in terms of rotations and how many dimensions are actually being used by the sets of n -balls
- Example: DENSITY
 - number of points in an n -ball given its volume:
 - * density $> 10 \times$ main density \rightarrow very dense
 - * density $>$ main density \rightarrow dense
 - * density $< \frac{\text{main density}}{2} \rightarrow$ sparse

TYPE	COMPONENT(S)	VALUE
Size	0	Big
Distance	0,1	Big
Distance	0,2	Big
Distance	0,3	Medium
Distance	0,4	Big
Distance	0,5	Big
Size	1	Big
Distance	1,2	Big
Distance	1,3	Big
Distance	1,4	Big
Distance	1,5	Big
Size	2	Very Big
Density	2	Small
Distance	2,3	Big
Distance	2,4	Big
Distance	2,5	Big
Size	3	Small
Density	3	Very Big
Distance	3,4	Big
Distance	3,5	Big
Size	4	Small
Density	4	Very Big
Distance	4,5	Big
Size	5	Small
Density	5	Very Big

Demo

<blink>Ask to see the demo</blink>



Thoughtland

*I spoke not of a physical Dimension,
but of a Thoughtland whence, in theory,
a Figure could look down upon Flatland
and see simultaneously the insides of
all things*

Submit a Weka ARFF file for analysis

Algorithm to use:

Tue Mar 12 20:59:54 EDT 2013 There are five components and eight dimensions. Component four, component two and component three are small and component one is giant. Component three, component four and component two are very dense. The components one and two are far from each other. The components one and four are far from each other. The components one and five are far from each

<blink>Ask to see the demo</blink>

Content Planner

- Implemented on top of McKeown's [McKeown, 1985] Document Structuring Schemata
 - Using my recent implementation Open-Schema: <http://openschema.sf.net>
- Two schemata
 - Components are presented in order
 - Attributes are presented in order (pictured next)
 - The system presents the user the shorter description

```
schema by-attribute(whole: c-full-cloud)
; first sentence, overall numbers
pred-intro(cloud|whole)
aggregation-boundary
star
  pred-size()
aggregation-boundary
star
  pred-density()
aggregation-boundary
star
  pred-distance()
```

```
predicate pred-density
variables
  req def component : c-n-ball
  req attribute : c-density
properties
  component == attribute.component
output
  pred has-attribute
  pred0 component
  pred1 attribute
  pred2 magnitude
```

Sentence Planner

- Thoughtland weaker component
 - Contributions welcomed!
- Some basic aggregation rules
- All components with the same property are put together to make complex sentences
 - That works well for size and density
- To verbalize distances, we group the different pairs by distance value and then look for cliques
 - Bron-Kerbosch clique-finding algorithm [Bron and Kerbosch, 1973]
 - We also determine the most common distance and verbalize it as a defeasible rule [Knott et al., 1997]
- An experimental lexical chooser using breeds of dogs to signify sizes and chemical elements to signify densities is also available

Case Study

CoNLL Shared Task for the year 2000 [Sang and Buchholz, 2000].

Splitting a sentence into syntactically related segments of words:

(NP He) (VP reckons) (NP the current account deficit) (VP will narrow)
(PP to) (NP only # 1.8 billion) (PP in) (NP September) .

Training: each word POS and its Beginning/Inside/Outside chunk info:

He	PRP	B-NP
reckons	VBZ	B-VP
the	DT	B-NP
current	JJ	I-NP
account	NN	I-NP
deficit	NN	I-NP
will	MD	B-VP
narrow	VB	I-VP

THREE DIMENSIONS

Naive Bayes

C4.5

Accuracy 88.9%

Accuracy 89.8%

There are five components and three dimensions. Component One is big and components Two, Three and Four are small. Component Four is dense and components Two and Three are very dense. Components Three and Five are at a good distance from each other. The rest are all far from each other.

There are six components and three dimensions. Component One is big, components Two, Three and Four are small and component Five is giant. Component Five is sparse and components Two, Three and Four are very dense. Components One and Two are at a good distance from each other. The rest are all far from each other.

FOUR DIMENSIONS

Accuracy 90.4%

Accuracy 91.4%

There are six components and four dimensions. Components One, Two and Three are big and components Four and Five are small. Component Three is dense, component One is sparse and components Four and Five are very dense. Components Two and Three are at a good distance from each other. The rest are all far from each other.

There are six components and four dimensions. Components One, Two and Three are big and components Four and Five are small. Component One is dense, component Three is sparse and components Four and Five are very dense. Components Three and Four are at a good distance from each other. Components Six and Four are also at a good distance from each other. The rest are all far from each other.

Extending and Modifying Thoughtland

- Machine Learning Algorithms

- Implement `CloudExtractor`: `TrainingData` x `algorithm_name` x `algorithm_params`
→ `CloudPoints`
- Existing: `net.duboue.thoughtland.cloud.weka.WekaErrorCloudExtractor`

- Adding new Clustering Algorithms

- Implement `Clusterer`: `CloudPoints` x `number_iterations` → `Components`

- Adding new Component Analyzers

- Implement `ComponentAnalyzer`: `Components` → `Analysis`

- Adding new Generators

- Implement `Generator`: `Analysis` → `GeneratedText`

Related Work

- NLG, long interest in describing
 - 3D scenes [Blocher et al., 1992],
 - Spatial/GIS data [Carolis and Lisi, 2002],
 - Or just numerical data [Reiter et al., 2008]
- Explaining machine learning decisions, ExOpaque [Guo and Selman, 2007]
- Graphical visualization
 - Dimensionality reduction and projection [Kaski and Peltonen, 2011]
 - However (from Janert [Janert, 2010]):

As soon as we are dealing with more than two variables simultaneously, things become much more complicated –in particular, graphical methods quickly become impractical.
- Machine Learning Integrated Development Environments (ML IDEs) [Kapoor et al., 2012, Patel et al., 2010]

Future Directions

- Enrich the analysis with positional information
 - Find planes on which a majority of the n -balls lie so as to describe their location relative to them.
- Hierarchical decomposition in up to five to seven n -balls
 - Cognitively acceptable [Miller, 1956]
- Generating comparisons (following [Milosavljevic, 1999])
- Objective-based generation (following [Dethlefs and Cuayáhuatl, 2011])
- Evaluation
 - Start with simple cases such as overfitting or feature leaks
 - See if the descriptions help humans detect such cases faster

Acknowledgments

The author would like to thank the anonymous reviewers as well as Annie Ying, Or Biran, Samira Ebrahimi Kahou and David Racca for valuable feedback and insights.

References

- [Blocher et al., 1992] Blocher, A., Stopp, E., and Weis, T. (1992). ANTLIMA-1: Ein System zur Generierung von Bildvorstellungen ausgehend von Propositionen. Technical Report 50, University of Saarbrücken, Sonderforschungsbereich 314, Informatik.
- [Bron and Kerbosch, 1973] Bron, C. and Kerbosch, J. (1973). Finding all cliques of an undirected graph (algorithm 457). *Commun. ACM*, 16(9):575–576.
- [Carolis and Lisi, 2002] Carolis, B. D. and Lisi, F. A. (2002). *Foundations of Intelligent Systems*, chapter A NLG-based presentation method for supporting KDD end-users, pages 535–543. Springer.
- [Dethlefs and Cuayáhuitl, 2011] Dethlefs, N. and Cuayáhuitl, H. (2011). Hierarchical reinforcement learning and hidden markov models for task-oriented natural language generation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 654–659. Association for Computational Linguistics.
- [Gatt and Reiter, 2009] Gatt, A. and Reiter, E. (2009). SimpleNLG: A realisation engine for practical applications. In *Proceedings of 12th European Workshop on Natural Language Generation (ENLG2009)*. ACL.
- [Guo and Selman, 2007] Guo, Y. and Selman, B. (2007). Exopaque: A framework to explain opaque machine learning models using inductive logic programming. In *ICTAI (2)*, pages 226–229. IEEE Computer Society.
- [Janert, 2010] Janert, P. K. (2010). *Data Analysis with Open Source Tools*. O’Reilly.
- [Kapoor et al., 2012] Kapoor, A., Lee, B., Tan, D., and Horvitz, E. (2012). Performance and preferences: Interactive refinement of machine learning procedures. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*.
- [Kaski and Peltonen, 2011] Kaski, S. and Peltonen, J. (2011). Dimensionality reduction for data visualization [applications corner]. *Signal Processing Magazine, IEEE*, 28(2):100–104.
- [Knott et al., 1997] Knott, A., O’Donnell, M., Oberlander, J., and Mellish, C. (1997). Defeasible rules in content selection and text structuring. In *Proceedings of the Sixth European Workshop on Natural Language Generation*, pages 50–60, Duisburg, Germany.
- [McCullagh and Yang, 2008] McCullagh, P. and Yang, J. (2008). How many clusters? *Bayesian Analysis*, 3(1):101–120.
- [McKeown, 1985] McKeown, K. R. (1985). *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Cambridge University Press, Cambridge, England.
- [Miller, 1956] Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The psychological review*, 63:81–97.
- [Milosavljevic, 1999] Milosavljevic, M. (1999). *Maximising the Coherence of Descriptions via Comparison*. PhD thesis, Macquarie University, Sydney, Australia.
- [Owen et al., 2011] Owen, S., Anil, R., Dunning, T., and Friedman, E. (2011). *Mahout in Action*. Manning Publications Co., Manning Publications Co. 20 Baldwin Road PO Box 261 Shelter Island, NY 11964, first edition.
- [Patel et al., 2010] Patel, K., Bancroft, N., Drucker, S. M., Fogarty, J., Ko, A. J., and Landay, J. (2010). Gestalt: integrated support for implementation and analysis in machine learning. In *Proceedings of the 23rd annual ACM symposium on User Interface software and technology*, pages 37–46. ACM.
- [Reiter et al., 2008] Reiter, E., Gatt, A., Portet, F., and van der Meulen, M. (2008). The importance of narrative and other lessons from an evaluation of an nlg system that summarises clinical data. In *Proceedings of the Fifth International Natural Language Generation Conference, INLG ’08*, pages 147–156, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [Sang and Buchholz, 2000] Sang, T. K. and Buchholz, S. (2000). Introduction to the conll-2000 shared task: Chunking. In *Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning*, pages 13–14.