What is the Difference Between Data Science and Machine Learning?

Geoffrey E. Schneider

June 22, 2017

1 Introduction

Data science is the practice of extracting useful information from data. Machine learning is a set of programming techniques in which the programmer sets up a model with variable parameters and a method by which these parameters will be ”learned” from data. There. Question answered. In practice, however, when searching for a job, it isn’t so clear. Because of the large overlap between the two, a job listed as a data science job may really be more of a machine learning job, and vice versa. And perhaps you want to pick up a few new, relevant skills, but you are interested in one of the two fields, and not the other. Can we disentangle which skills are more relevant to your chosen field from the job descriptions posted for applicants?

The goal of this project is to use textual analysis to solve this issue. Code was written in the new and exciting programming language Julia. This document was produced using LaTeX and Julia (figures all come from Julia[1] or from LaTeX code produced by Julia code).

2 Data

The data for this project is 150 posted job descriptions[3]. They are divided into three datasets, the machine learning descriptions (50), the data science descriptions (50), and the control descriptions (50). These were gathered by doing searches for the relevant words (”machine learning job”/”data science job”/”job”), and then divided into sequences of words using regular expressions. In Julia this is implemented as:

```
function texttowords(text)
  matchall(
    r"(\w+)",
    lowercase(replace(text, r"([\r\n,\.\(\)!;:\?/]|\ufeff)", s" "))
    )
end
```

The sequences of words were then divided into ”bags of words” (i.e. word counts), one for each of the datasets.

```
function wordcount(wordvec)
  worddict = Dict{String,Int64}()
  for w in wordvec
    worddict[w]=get(worddict,w,0) + 1
  end
  worddict
end
```

3 WordNorm

To compare two of the datasets, we will use two variations on what we will call ”WordNorm” (for non-technical folks, you can skip from here to the tldr at the end if this section, if you want). For datasets α and β (e.g. α = data science and β = control) and word w, the ideal version of WordNorm

$$\mathrm{WordNorm}_{\alpha\beta}(w) = \frac{\alpha(w) - \beta(w)}{\alpha(w) + \beta(w)} \tag{1}$$

where $\alpha(w)$ (resp. $\beta(w)$) indicates the word count for w in α. We will write $\mathrm{WordNorm}(w)$ when the choice of α and β are clear or irrelevant. Note that

$$\mathrm{WordNorm}(w) = \begin{cases} 1 & w \in \alpha \setminus \beta \\ -1 & w \in \beta \setminus \alpha \end{cases}$$

and that 1 and -1 are the maximum and minimum values of $\mathrm{WordNorm}$. So, if w is near 1, it is much more closely associated to α, and if w is near -1, it is much more closely associated to β. In an ideal setting, with a very large amount of data (so that each word count has at least one of each word), this would work well, however in a practical setting there is an issue. For example, suppose we're interested in comparing the word ”dog” to the word ”cat”, and $\alpha(\text{”dog”}) = 5$, $\alpha(\text{”cat”}) = 1$, but $\beta(\text{”dog”}) = \beta(\text{”cat”}) = 0$. $\mathrm{WordNorm}$ does not distinguish between ”dog” and ”cat”! $\mathrm{WordNorm}(\text{”dog”}) = \mathrm{WordNorm}(\text{”cat”}) = 1$, despite the fact that it is clear that α is probably more strongly associated with ”dog” than it is with ”cat”. To deal with this problem, we will use the two modifications that follow.

When comparing a sample dataset α (e.g. the counts of words in machine learning job descriptions) to the control dataset (which we write as c) we use

$$\mathrm{WordNormCtrl}_{\alpha}(w) = \frac{\alpha(w) - (c(w) + 1)}{\alpha(w) + (c(w) + 1)} \tag{2}$$

where we've written the parentheses to emphasize the interpretation that $\mathrm{WordNormCtrl}$ is just $\mathrm{WordNorm}$ where we've added one of each word to the control dataset. Now using β in the example above as a control, we get $\mathrm{WordNormCtrl}(\text{”dog”}) = \frac{2}{3}$ and $\mathrm{WordNormCtrl}(\text{”cat”}) = 0$. For our setting $\mathrm{WordNormCtrl}$ has the nice feature that common words (”is”, ”the”, etc.) and general job words (”applicant”, ”responsibilities”) get a score close to 0, and so when we are looking at the top scores, these are automatically ignored. It loses the (anti)symmetry of $\mathrm{WordNorm}$, but this is acceptable, because the sample and control are playing different roles here.

When comparing two sample datasets, we can use

$$\text{WordNormComp}_{\alpha\beta}(w) = \frac{\alpha(w) - \beta(w)}{\alpha(w) + \beta(w) + 1}. \tag{3}$$

This is equivalent to adding half a count of each word to both datasets. In a sense, we are saying that we expect that if $\alpha(w) = 0$, the "real value" of $\alpha(w)$ is between 0 and 1, but we need a larger sample to determine the value, so we guess that it is 0.5, but then to be fair to the other words where $\alpha(w) > 0$, we also add 0.5 to their counts. Here the (anti)symmetry is restored, as it should be since both datasets play interchangeable roles.

Finally, to score a full document, we could try to naively take the average score of its words, but this would be a bad approach because there is a skewed level of importance for WordNorm scores close to ± 1. To understand this issue, consider what happens when we average a word with a score of 0.99 with a word with a score of 0.00 (using WordNormComp). The score of 0.99 might represent 298 examples in the first dataset and 1 in the second, whereas a score of 0.00 might represent 1 example in the first and 1 in the second. The average is ≈ 0.50, which is the same as the average of two words with scores of 0.50 each representing 4 examples in the first dataset and 1 in the second. Clearly, this is a problem for the naive average. To fix this issue, we can transform the range of WordNormComp (from $(-1, 1)$ to $(-\infty, \infty)$) by taking the inverse hyperbolic tangent, averaging these transformed scores, and then inverting the transformation, taking the hyperbolic tangent of the average. Written in mathematical notation this is

$$\widehat{\text{TextScore}}(\textsf{text}) = \tanh(\underset{w \in \textsf{text}}{\text{Mean}}(\tilde{w})) \tag{4}$$

where $\tilde{w} = \text{arctanh}(\text{WordNormComp}(w))$. TextScore should be interpreted relative to the score on a control dataset, so we introduce a modified version of the above score

$$\text{TextScore}(\textsf{text}) = \tanh(\underset{w \in \textsf{text}}{\text{Mean}}(\tilde{w} - \text{arctanh}(\widehat{\text{TextScore}}(\textsf{control})))) \tag{5}$$

[*tldr: There are a few variations of* WordNorm*, but in all cases, it assigns a value between 1 and -1 to each word. 1 means the word is much more closely associated to the dataset, and -1 means the word is much more closely associated to the second. When one of these is the control, it will always be the second.* TextScore *is a kind of average* WordNorm *over an entire document.*]

4 Results and Discussion

In Figure 1 we show the results of applying WordNormCtrl to data science (vs. control) and to machine learning (vs. control). We see that the two give broadly similar scores validating the assertion that there is much overlap between the two. Languages such as R, Python and Spark, as well as the distributed system framework Hadoop appear in the top 10 for both.

For "visualization" we see a higher data science score, confirming the stronger importance of communicating findings to allow companies to make informed business decisions, whereas the higher machine learning score for "ai" indicates the importance of building automated systems as a product in themselves. The word "academy" appears at first to be strongly associated with data science and strongly dissassociated with machine learning, however, it is an aberration which should be ignored, as it is on this list only because it appeared a very large number of times in one particular data science job description, and not at all in the control and machine learning data.

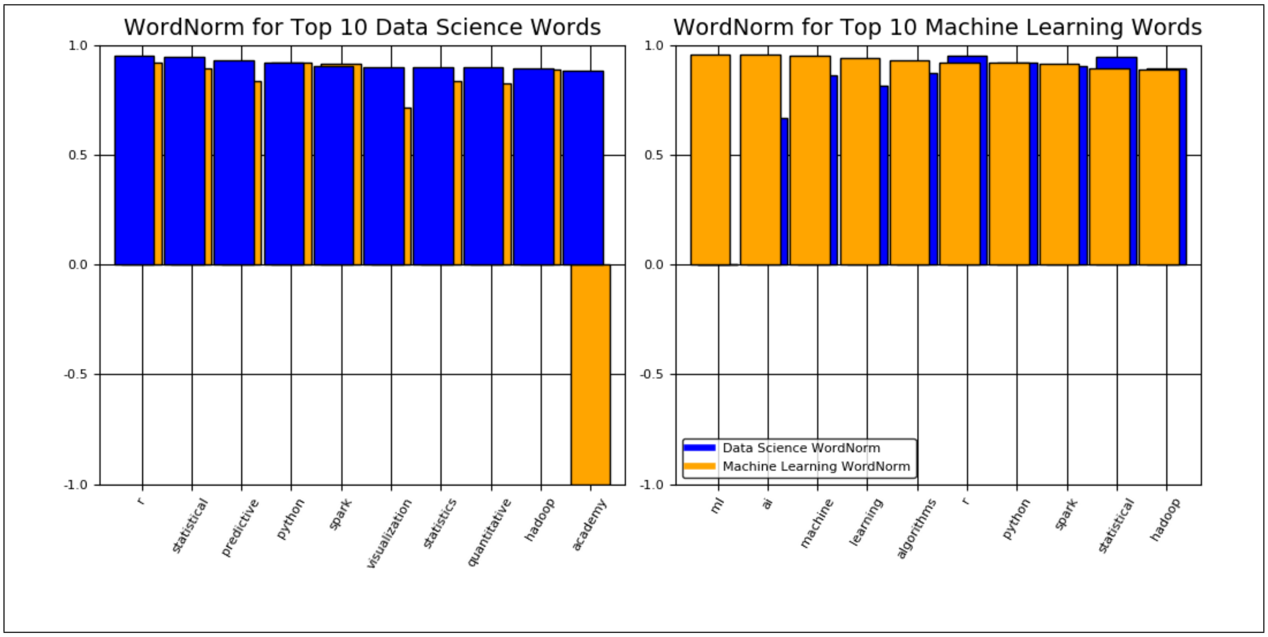

Figure 1: The words with the highest $\mathrm{WordNormCtrl}$ scores for both data science and machine learning.

Comparing the data science and machine learning data to the control tells us a lot about what is important to each of them, but due to their large overlap, it does not allow us to *distinguish* them. To do this, we compare them directly using $\mathrm{WordNormComp}$. Figure 2 shows the result of applying $\mathrm{WordNormComp}_{ml,ds}$ to directly compare machine learning to data science.

Machine learning frameworks TensorFlow and Caffe are both represented with a stronger association with machine learning whereas words like "consumers" and "audiences" are associated with data science. We also see the names of companies that are associated with machine learning or data science. These company names may be due to the inclusion of jobs advertised by those companies in the dataset, though this is not entirely clear in the case of software companies, where the inclusion may be due to requirements that applicants have knowledge of their software.

Using the $\mathrm{TextScore}$, we can score any text in terms of whether it is more closely associated with data science or machine learning[1]. Note that because this is a kind of average, we should expect scores to be closer to 0 than on individual words. On an additional data science job description, the $\mathrm{TextScore}$ was -0.084. On an additional machine learning job description the $\mathrm{TextScore}$ was 0.103. The first two pairs of texts in Figure 3 show the first 50 words of these document, words colored according to their $\mathrm{WordNorm}$[2].

We can even try this on documents that are not directly descriptions of a particular job. For example, the Insight Data Science Fellows Program has a white paper with a section titled "What is a Data Scientist?"[2]. The third pair of texts Figure 3 shows the first 50 words of this document,

[1]On the concatenation of the documents that made up the control dataset, the $\widehat{\mathrm{TextScore}}$ is -0.143, and so all $\mathrm{TextScore}$ scores following are normalized by this value.

[2]Normalized by the $\widehat{\mathrm{TextScore}}$ of the control.

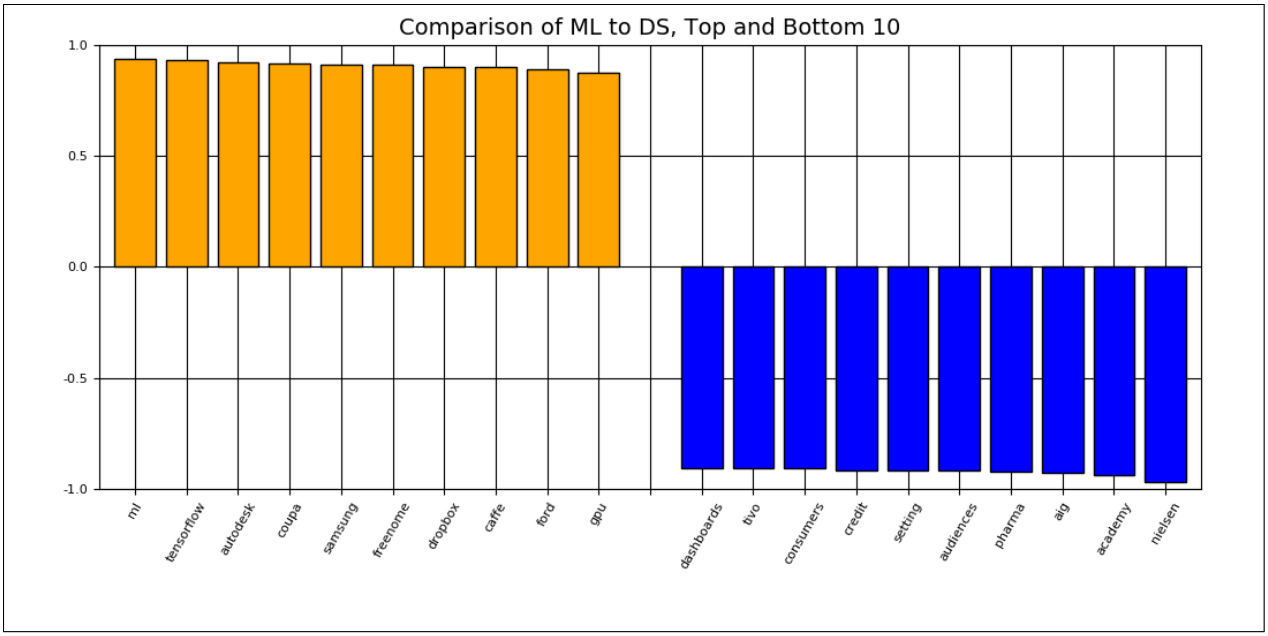

Figure 2: The result of comparing machine learning to data science with $\mathrm{WordNormComp}$. Positive scores indicate stronger association with machine learning, and negative scores indicate stronger association with data science.

each colored according to their $\mathrm{WordNormComp}$ score. The overall $\mathrm{TextScore}$ for this document is slightly less than 0, at -0.005.

Finally, if we compute the $\mathrm{TextScore}$ for the concatenated documents that made up the control dataset, we get -0.143, perhaps indicating that the difference between data science jobs and the average job advertised online is smaller than the difference between machine learning jobs and the same average.

our clients seeks an experianced leader who will sit at the intersection of analytics data management it software infrastructure in the center for data science and analytics is an innovative corporate analytics group within the business they are a rapidly growing entrepreneurial department which aims to design create and offer

our clients seeks experianced leader who sit the intersection analytics data management it software infrastructure the center data science analytics innovative corporate analytics group within the business they growing entrepreneurial department which aims design offer

machine learning researcher this is a new role designed for a new and unexplored environment designing machine learning and deep learning algorithms you will have petabytes of data state of the art hardware such as massive distributed clusters and gpus and professional support to develop ways to design complex and

machine learning researcher this new role designed new unexplored designing machine learning deep learning algorithms have petabytes data state the art hardware such massive distributed clusters gpus develop ways design complex

the amount of data produced across the globe has been increasing exponentially and will continue to grow at an accelerating rate for the foreseeable future at companies across all industries servers are overflowing with usage logs message streams transaction records sensor data business operations records and mobile device data effectively

the amount data produced across the globe has been increasing exponentially continue grow accelerating rate the foreseeable future companies across all industries servers overflowing usage logs message streams records sensor data business operations records mobile device data

Figure 3: The first 50 words of three different texts: a data science job description, a machine learning job description and ”What is a Data Scientist?”. First shown in black, then colored according to its (normalized) WordNormComp score. Data science associated words are blue, machine learning associated words are in orange, and lighter means closer to 0.

References

[1] Jeff Bezanson, Stefan Karpinski, Viral B. Shah, and Alan Edelman. Julia: A fast dynamic language for technical computing. *CoRR*, abs/1209.5145, 2012.

[2] Insight data science white paper. `xyz.insightdatascience.com/Insight_Data_Science_White_Paper.pdf`. Accessed: 21-06-2017.

[3] Linkedin. `www.linkedin.com`.