Basketball Article Recommender for Bleacher Report

Stephen Albro
Master of Business Analytics
Massachusetts Institute of Technology
Cambridge, MA 02139
salbro@mit.edu

Cyrille Combettes
Master of Business Analytics
Massachusetts Institute of Technology
Cambridge, MA 02139
cyrille@mit.edu

1 Motivation

As avid basketball fans, we have enjoyed reading a variety of NBA articles during our busy year in the MBAn program to keep up-to-date on the league. Bleacher Report, a primary digital destination for sports article readers, contains well written articles, and we thought about returning the favor.

In an age of cut-throat digital competition, it is vital for websites to retain visitors. The hope for a Bleacher Report article reader is two-fold. First, that the visitor jumps from article to article on the website, rather than returning to the search engine results page. Second, that the visitor falls in love with the content and coverage of Bleacher Report's articles, and develops a loyalty to the website.

We feel that having a micro targeted article recommendation system would provide Bleacher Report with an analytics-based edge over its competitors. In this paper, we develop a prototype for an article recommendation system using state of the art natural language processing techniques for both cleaning and analyzing Bleacher Report articles. Specifically, we aim to cluster over 3000 Bleacher Report articles by testing three different approaches for topic modeling: Word2Vec, Latent Dirichlet Allocation (LDA), and Non-negative Matrix Factorization (NMF).

2 Data collection and preprocessing

We queried Bleacher Report NBA articles from last October 17, the start day of the 2017-2018 NBA season. Our entire query consisted of joining words like *basketball* and *NBA* with all 30 teams and the top 25 active players: we received 3,314 URLs in return. We then wrote a script to request the webpage for each URL and scrape its content, picking the paragraphs out of the HTML.

In the real world, data is never clean, and this is magnified in the case of textual data. There were a number of preprocessing steps we had to do before our articles were ready for analysis. In our cleaning efforts, we made decisions based on the purpose of our task: to cluster documents for better recommendation power. In particular, a bag-of-words approach to cleaning was preferred over one that preserved punctuation and grammar.

3 Key results

Recurrent neural networks can be used as generative models. We will train such a network on the text of many sports articles (via ESPN API) using the Python libraries Keras and/or Tensorflow. For this project we will be using the long short-term memory (LSTM) model which attempts to add memory into the architecture of a neural network. LSTMs can take a while to train, so we will seek to use GPUs in a cloud computing environment, either through AWS or MIT.

15.071C Advanced Analytics Edge: project abstract. May 9, 2018. Massachusetts Institute of Technology, MA.