# Predicting Metacritic Film Reviews Using Linked Open Data and Semantic Technologies

Meyer A. Bossert

Cray Inc., Seattle Washington, USA
bossert@cray.com

**Abstract.** Aristotle was quoted as saying that “the whole is more than the sum of its parts”. Using Linked Open Data, we are finally able to test and quantify Aristotle’s theory[1]. By using the flexible data representation of RDF as well as the graph-oriented nature of SPARQL, we attempt to answer the questions: What makes a movie good? And more specifically, what makes a critic think that a movie is good? We take a novel approach to predictive analytics that is implemented entirely in SPARQL rather than using more traditional statistical machine learning platforms (e.g. Rapidminer, etc.).

## 1 Introduction

The Linked Data Mining Challenge (LDMC)[2] proposes that we use Linked Open Data (LOD) sets in order to predict Metacritic reviews. Data in the Resource Description Framework (RDF) format provides many advantages; Principal among these are the flexible nature of data representation, especially when multiple properties are associated with a given entity, but in an inconsistent way (i.e. not every film’s DBpedia entry contains the same properties). Further, RDF is better suited for analyzing relationships between entities, especially when the potential relationships are not known entirely. Certainly, one can assume that any given film will have actors, a production company, a genre, and director. When one moves beyond those simple connections, one can then consider information that is multiple hops away from the film. This opens the door to much deeper understanding of a given topic (in this case, films) and also allows us to develop a much more robust and flexible mechanism for predicting the general impression that critics have of that film. In this paper, we document our analytic approach, explore the resulting output, and finally lay down a roadmap for future research and improvements.

Due to the number of pages for this publication being limited to 4 pages in LNCS format, we are unable to include any of the SPARQL queries, full results, or ETL scripts in the body of the document. All materials other than the source datasets consisting of the DBpedia 2014

[1] If not obvious, this statement is made with a little humor in mind. At no point does this work attempt to seriously tackle Aristotle’s work.

[2] http://knowalod2015.informatik.uni-mannheim.de/en/linked-data-mining-challenge/

Copyright © 2015 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors.

dump[3] and the Freebase 2014 RDF dump[4] have been stored on Github[5] and made accessible to the general public under the MIT license.

# 2 Analytic Approach

In order to predict Metacritic reviews based on the training and testing data provided, we note the basic steps needed to accomplish our goal. Some pre-processing is required as is some general data cleansing prior to tackling the problem of generating predictions.

## 2.1 Pre-processing

In order to link the provided training and testing data to the DBpedia dataset (local copy of 2014 dump), we develop a simple Perl script to convert each row of data into RDF format (N-Triples). Also, an important disambiguation step is necessary because the provided DBpedia URI's are not directly connected to all the available data about any given film. We take the additional step of running SPARQL queries that identify URI's within the provided datasets that contain little information (e.g. fewer than 75 properties), have a disambiguation link, and/or a redirect link. When one of those linked URI's matches the release year and also contains significantly more properties than the initially provided one, we swap them out.

## 2.2 Data Cleansing

Using the Cray Urika GD[6] graph appliance, which implements the Apache Jena quad store on top of a global shared memory platform containing 2TB of shared memory, we load the entire DBpedia and Freebase datasets as well as the training and testing dataset. The entire dataset consists of approximately 3.446 billion triples[7].

As an initial exploratory step, we profile the predicates and properties associated with the films found in the testing and training datasets.

Based on the results of the profiling queries, we ignore properties and predicates that are obviously irrelevant or have the potential to produce erroneous predictions (e.g. having an RDF type of owl:Thing is not useful in predicting how a film will be reviewed). In addition, we find that additional weight should be given to films that received various forms of recognition (e.g. Academy Awards, various film critic's association awards, etc.) In order to implement the data cleanup, we perform several INSERT and DELETE queries to flag both predicates and objects associated with films as "keep" or "drop". This step is for convenience and is performed in lieu of a long and complex series of FILTER operations in the actual queries used to generate the

[3] http://wiki.dbpedia.org/Downloads2014

[4] http://commondatastorage.googleapis.com/freebase-public/rdf/freebase-rdf-latest.gz?

[5] https://github.com/mabossert/LDMC_2015

[6] http://www.cray.com/products/analytics/urika-gd

[7] Not all triples in the dataset were used or analyzed

Copyright © 2015 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors.

film score predictions. Functionally, this step is not required as the same results could be obtained through implementation of the FILTER clause.

In order to simplify the SPARQL queries, we transpose all of the desired properties associated with film entities found in the Freebase dataset to their equivalent entities found in the DBpedia dataset (e.g. as identified by the owl:sameAs relationship).

### 2.3 Predictive Algorithm Development

In the abstract, we alluded to Aristotle's famous quote "The whole is greater than the sum of its parts". Our theory is that if we know, for each attribute associated with a film, on average how many times that attribute is associated with a "good" or "bad" film, then we can surmise with some degree of certainty that the average score as determined by taking the average of all desired[8] attributes will be a good indicator of the likelihood of a film receiving positive or negative reviews as quantified by the overall Metacritic score. Finally, we acknowledge that there are certainly some properties that should be considered with a higher weight than others. In particular, we make the assumption that films that receive awards of any type are likely to considered "good". Though most films that receive awards are well received by critics, not all are, thus we account for this observation by weighting awards differently based on if films in the training dataset were rated at "good" or "bad" (e.g. weight of 5).

### 2.4 Algorithm tuning

In order to tune the algorithm against the training and testing datasets, we run the algorithm against the training dataset using the award weight and the score decision breakpoint. We find that the optimal breakpoint is a score of 55 percent or higher and we find that the optimal weight for the awards-related properties is 5. Finally, careful inspection of the resulting scores showed a need to add additional weight to objects that were particularly polarized with respect to the ratio of good to bad films associated with the object in question. After several manual iterations, we settle on cutting off the weighting at those higher than 75% for good and lower than 35% for bad. The weights attributed to each set are 0.2 and -0.7, respectively. The final calculation is expressed in the following snippet from the SPARQL query with the weight being the sum of the previously determined multiple (i.e. based on certain awards) and the weight determined for the polarized objects (i.e. higher than 75% and lower than 35%):

```
(AVG(?percentGood * (?multiple + ?topBottom)) AS ?percent)
```

These experimental results are confirmed by our submission being evaluated at 92.25% accurate[9]. So, with our lighthearted poke at Aristotle, we can now say that roughly 7.75% is the delta between the sum of the parts and the whole.

[8] As determined in the data cleansing step identified earlier with "keep" and "drop" values.

[9] The un-tuned algorithm generated an accuracy of approximately 84%

Copyright © 2015 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors.

## 3 Interesting observations

During the course of our analysis, we noted that there are some counterintuitive to our existing anecdotal knowledge of films. First, we initially noted that several films that had been featured at various film festivals had been generally well received by the public, but not necessarily the critics. Our assumption had been that, in general, films featured at a film festival (e.g. Cannes, etc.) would disproportionately be well reviewed by critics, however, our experiments showed that by weighting film festivals, accuracy was drastically reduced.

Another interesting observation was that in our initial experiments, the types of properties that were considered were restricted to just a few obvious choices (e.g. actors, cinematography, musical score, screen play author, etc.) resulted in relatively poor accuracy. Once most property types were included, counterintuitively, accuracy was drastically increased. We hypothesize that it is actually the rich variety of links between films as expressed in DBpedia and Freebase that enable more accurate predictions due to the deep linking between films.

Documentaries deserve a special mention. Regardless of the film, those that were identified as documentaries received overwhelmingly high praise from film critics.

## 4 Future work

In future iterations of this predictive algorithm, we will attempt to overcome the problems of sparse data by incorporating community detection algorithms as part of the pre-processing.

We will also incorporate more datasets (e.g. IMDB, Rotten Tomatoes, etc.) as well as Natural Language Processing (NLP) of actual critic review text (e.g. entity extraction and sentiment analysis) in order to derive deeper understanding of each film beyond its constituent parts.

## 5 Conclusion

In our experiments, we find that with great accuracy, we can predict Metacritic reviews of any given film that has data found in LOD. The aspect of this work that was most interesting to us was that we could break from traditional statistical machine learning approaches to get to an acceptable result. Using nothing more than SPARQL queries, any person can implement this same technique; democratizing data analysis and access is, of course, one of the main principles behind LOD projects. Further, we note that the same algorithm, with minor adjustments, can be applied to many other types of entities and could have significant impacts in areas like social media and news predictive analytics, for instance.

## References:

1. W3C. SPARQL 1.1 Query Language. http://www.w3.org/TR/sparql11-query/
2. Google, Freebase Data Dumps, https://developers.google.com/freebase/data
3. University of Manheim, http://knowalod2015.informatik.uni-mannheim.de/en/linked-data-mining-challenge/

Copyright © 2015 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors.