LinkedStars - Celebrities Linkage Network

Vineet Gadodia[1], Tarneet Singh Sidana[1]

[1] University of Southern California
Computer Science Department
Los Angeles, CA 90089
{gadodia, sidana}@usc.edu

Abstract. Are you interested in how celebrities are related with each other? How they have being portrayed in the recent news? LinkedStars helps to get an overview: It extracts and visualizes relationships between given celebrities in RDF data and makes these relationships look mesmerizing. This paper presents an approach for the interactive discovery of relationships between selected celebrities via the Semantic Web. For example, one might like two actors while watching a movie and want to know how they both are connected to each other. Information about actors are scattered on the web and it is not easily accessible to the user in a visually interactive way. LinkedStars is the application they are looking for where they can visualize all the connections graphically between the two celebrities. This lures the user in exploring interesting relationships between celebrities of their choice and to know how close they are to each other. LinkedStars helps the user to conveniently get answers to complex queries regarding relationships between the actors.

Keywords: Linkage Network graph, Celebrity, Information Integration, Link-O-meter.

1 Introduction

Most of the people nowadays are interested in knowing the hidden links between the celebrities. There are plenty of websites that either provides personal information, professional details or recent news about individual but hardly anyone provides connections or links between them. The aim of this project is to give a platform for general users to find family/friends links, co-worked movies, latest news trends and other relevant hidden information between their favorite celebrities. In addition to this, they will be able to visualize the comparison between them based on certain statistics such as gross earnings, range of movies they have done etc. This would instigate further interest to investigate and explore.

2 Motivation

Details about the individual celebrity are well documented all over the web. But what

Demo Link: http://youtu.be/eg4gRbHAFdE if someone wants to know the relation between the celebrities and at the same time wants to know the current affairs about them. Hence, there is a need for integration of the data from various sources on the web and consolidation of their results into one website that portrays all of it in a manner that sounds interesting to the user so that he/she is interested into exploring further.

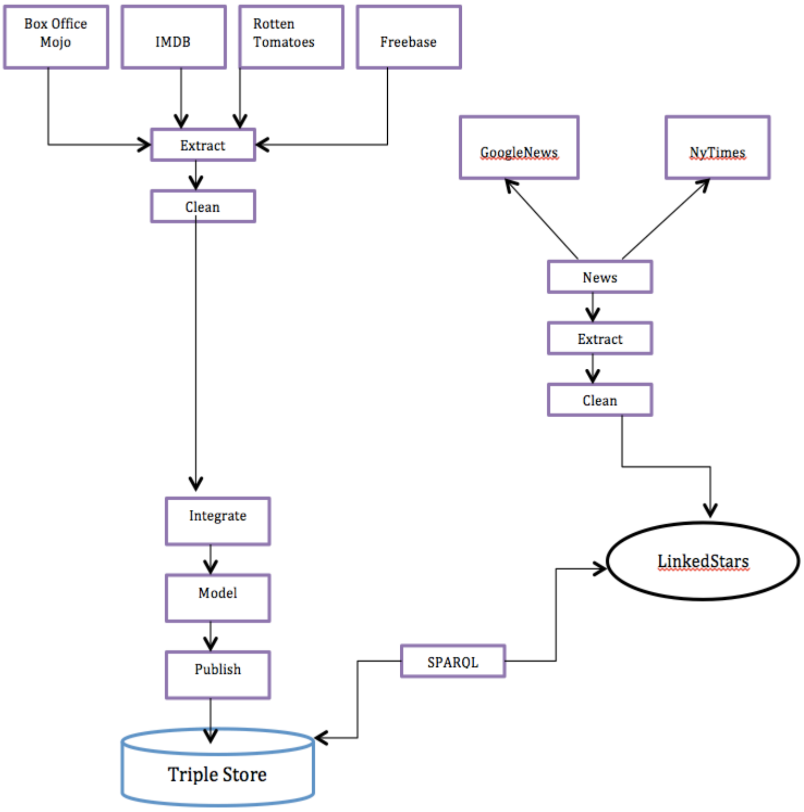
3 Approach

3.1 Architecture

Figure 1 shows the overall architecture of LinkedStars. As it can be seen that various sources have been used to collect data. Once the data is collected it is cleaned using Google Refine and the sanity of the data is verified as well. Some of the cleaned data was integrated using the Fril tool for generating record linkages. In this step we used the Jaro Winkler similarity for string matching of the names of movies. We also use the Year of production of the movie too as an additional matching criterion. The integrated data is then modeled using Karma. The rest of the data were modeled using Google refine. The generated RDF from Karma and Google refine is then uploaded to Triple Store (Sesame). News are extracted on the fly and cleaned using Open Calais.

We execute SPARQL queries on the triple store at runtime using sesame api. This way we fetch the required part from the triple store. News are extracted dynamically at runtime. The results are shown using Twitter Bootstrap framework and d3js.

Fig 1: Architecture



3.2 Data Sources, Cleaning and Record Linkage

We extracted the data from four major sources named box office mojo, freebase, imdb and rotten tomatoes. Around 696 celebrities with their respective revenues and 50,000 movies in which they have worked were extracted from box office mojo. These celebrities were then reconciled based on their names with the freebase to obtain all the required attributes of each of them. We extracted various attributes which includes people attributes, film worked, daily life, awards related, television and music attributes which summed up to around 175 attributes. Along with theses data, we extracted the imdb id of the movie in which the respective celebrity has ever worked. We successfully got around 24,653 movie's imdb id. We used omdbapi to extract the required movie's attributes from imdb, querying using the imdb id. Rotten tomatoes are used to extract the highest critic rating. Jsoup was used for scraping the data.

Then we used Google Refine to clean some of the malformed data using google refine expression language. Duplicity was removed using the same. Image url were put in the correct format to be fetched during application startup. Open calais was used to filter out the relevant news. After the news were fetched using google and new york times api, those news links were used to download the html source code and the results were passed to open calais. RDF returned from open calais was stored in triple store and were queried for person entities name. The obtained names were matched with the chosen celebrity names. For further efficiency, their other names were also used for matching. Finally, if both the celebrity names are present we accept the news otherwise we discard it.

Once all the data was collected we used FRIL to perform record linkage between the data extracted from imdb and box office mojo. In this step we used the Jaro Winkler similarity for string matching of the names of movies. We also use the Year of production of the movie too as an additional matching criterion.

3.3 Data Modeling

Once we got all the clean data, we constructed the mediated schema using Google refine and modeled the collected data. We used some freebase ontology to model our data but we needed some special classes and their relations, which were not defined by freebase Ontology, so we had to add some new classes and create their relations with existing classes. Finally we modeled the data and created the RDF. We used Karma as well to model the movie data. The entire set of RDF was stored in the triple store.

4 Application Details

The application is developed using various technologies. Java servlet was majorly used for back-end development. JSON format was used for exchanging data between front-end and back-end for maintaining consistency. Figure 2 shows the front page of the application where the user sees around 700 celebrity images with their names. Users are allowed to select any two favorite celebrities. An Ajax call is triggered as the page loads fetching the images from the backend. JQuery triggers the selection process as soon as the user selects his/her favorite celebrity.

Fig 2. Landing Page, shows 700 celebrities

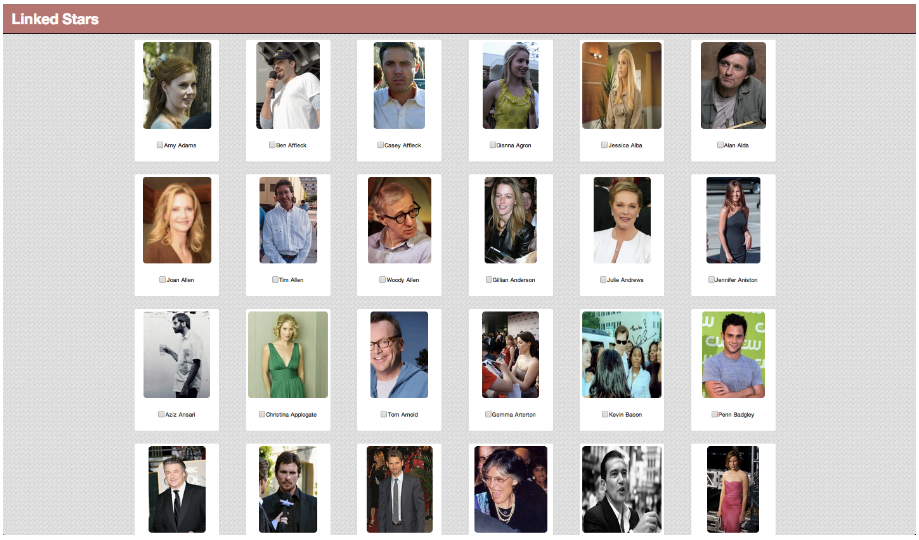
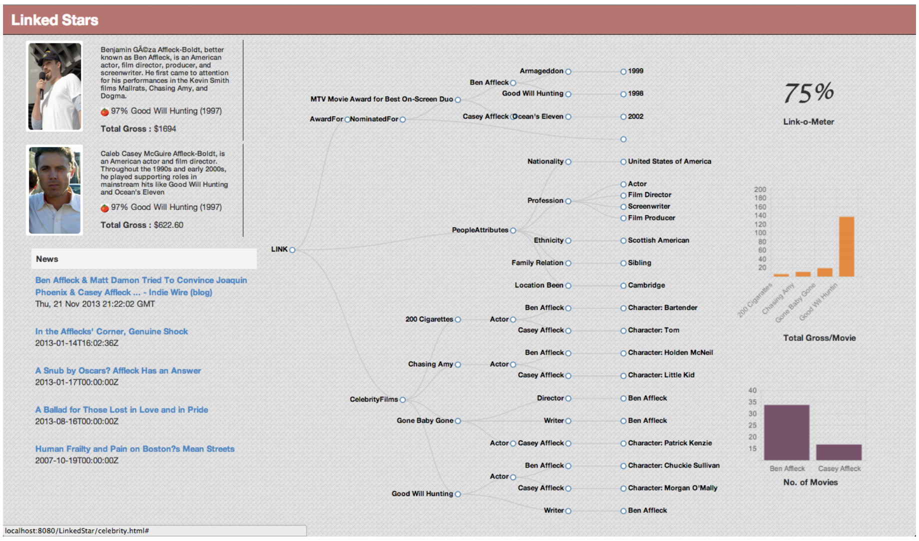


Fig 3. Resulting Page


Linked Stars

Benjamin GÃ©za Affleck-Boldt, better known as Ben Affleck, is an American actor, film director, producer, and screenwriter. He first came to attention for his performances in the Kevin Smith films Mallrats, Chasing Amy, and Dogma.

97% Good Will Hunting (1997)

Total Gross : $1694

Caleb Casey McGuire Affleck-Boldt, is an American actor and film director. Throughout the 1990s and early 2000s, he played supporting roles in mainstream hits like Good Will Hunting and Ocean's Eleven

97% Good Will Hunting (1997)

Total Gross : $622.60

News

Ben Affleck & Matt Damon Tried To Convince Joaquin Phoenix & Casey Affleck ... - Indie Wire (blog)
Thu, 21 Nov 2013 21:22:02 GMT

In the Afflecks' Corner, Genuine Shock
2013-01-14T16:02:36Z

A Snub by Oscars? Affleck Has an Answer
2013-01-17T00:00:00Z

A Ballad for Those Lost in Love and in Pride
2013-08-16T00:00:00Z

Human Frailty and Pain on Boston?s Mean Streets
2007-10-19T00:00:00Z

localhost:8080/LinkedStar/celebrity.html#

Figure 3 shows the result of the selection. The page is subdivided into three sections. The leftmost section depicts certain information about the individual celebrity. Below that is the news bulletin showing the news involving both. The middle one shows the linkage network emphasizing on common attributes between them. We used d3js technology for the developing linkage network. On the right hand side we have few charts for analyzing the best movie done so far based on total gross and the individual popularity. Link-O-meter defines the closeness factor between the chosen celebrities.

5 Conclusion and Future Works

"LinkedStars" successfully portrayed the relationships between celebrities with the help of a simple and clear linkage network graph. It could successfully put together the various interesting relationships between celebrities that we initially had set out to document.

The future work can include the social media feeds from twitter. Since the application is currently restricted to actors, we can integrate other celebrities as well such as directors, writers, musicians etc. Application can be made more flexible by facilitating it to search linkages between more then two celebrities.

6 Acknowledgement

The work on this project has been done as a part of class work for Information Integration on the Web (CSCI – 548) course at University of Southern California under the guidance of Professor Craig Knoblock and Professor Pedro Szekely.

7 References

1. http://www.boxofficemojo.com/
2. http://www.freebase.com
3. http://www.rottentomatoes.com
4. http://www.imdb.com
5. http://news.google.com/
6. http://www.nytimes.com/