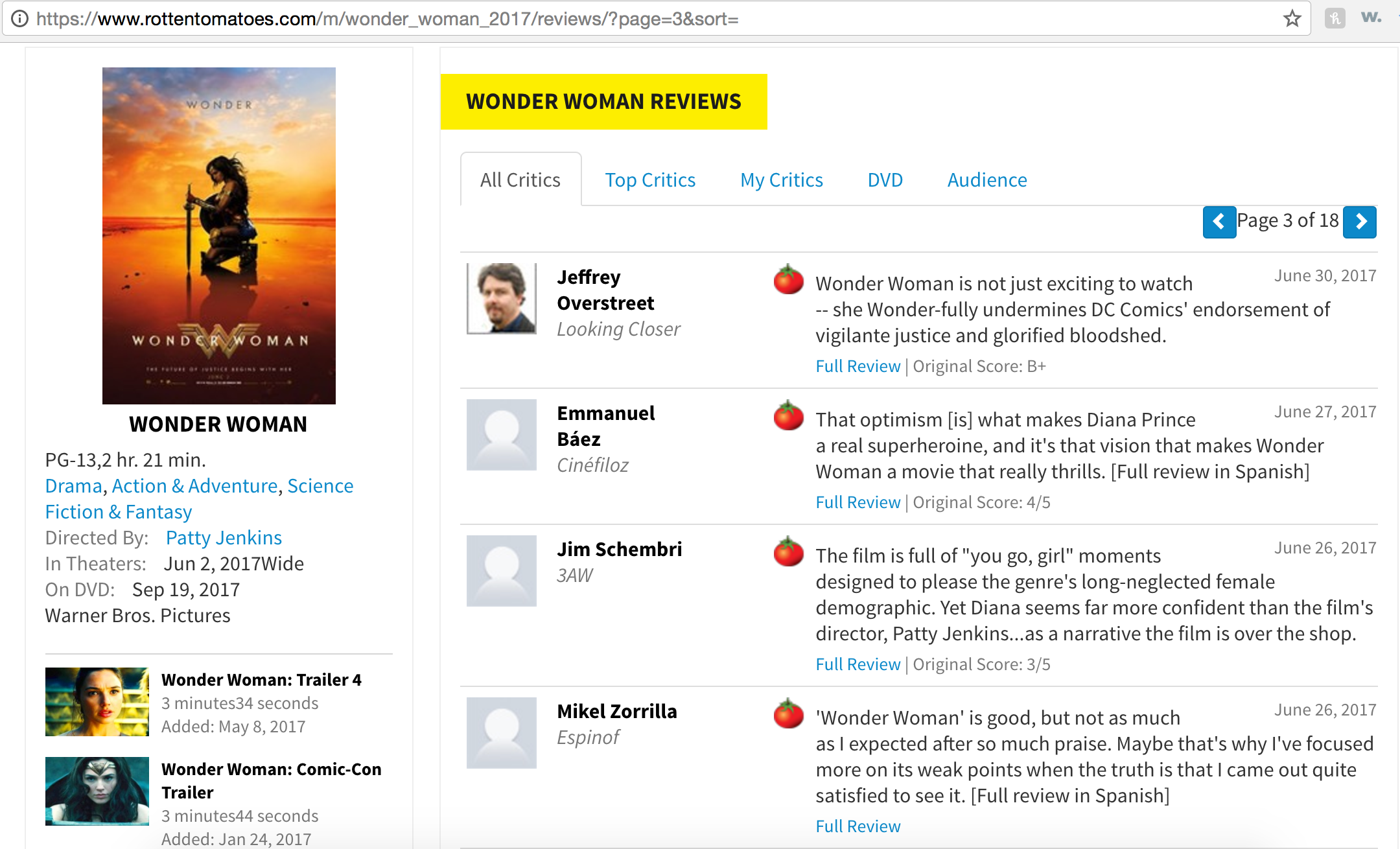
**Exercise 9: Web page mining**

**Assignment Specification**

**Description**: Choose a popular movie you like and find the id of this movie at rottentomatoes.com

* Example: https://www.rottentomatoes.com/m/wonder\_woman\_2017/reviews/
* Movie id = “wonder\_woman\_2017”
* Wonder Woman has a total of 356 reviews in 18 Pages of reviews. You need to collect reviews from all the pages.
* Create a list of lists [review, score], where “review” is the review text content and “score” is the rate given for each review. Pay attention that this score can be a nominal grade, fraction or can be missing – see picture.
* Rank reviews by score.
* Test your program with different movie ids to make your script generic enough



**Input**: Data will be collected from the website www.rottentomatoes.com. User can be prompt to insert movie id, but it’s not required.

**Output**: Print the 20 better and 20 worst rated reviews. See details in the Procedure.

**Procedure**:

* Import the needed libraries: urllib2 and BeautifulSoup
* Define the target URL and open it, using urllib2
* Load the page into your “soup”
* Identify the total number of pages
* Loop through each page, creating a new soup every time
* Collect review and respective score
* Generate a list of lists for the reviews
* Standardize scores and clean the reviews that do not hold a rating

Hint: replace nominal scores for percentages -> A+ = 1, A = 0.96, A- = 0.92, B+ = 0.89, B = 0.86, B- = 0.82, C+ = 0.79, C = 0.76, C- = 0.72, D+ = 0.69, D = 0.66, D- = 0.62

* Sort list of lists by ranking scores
* Print Top 20 and Bottom 20 reviews lists
* BONUS: Create TWO word clouds, one for the top 20 and other for the bottom 20 reviews.
  + Use only the text part of the reviews list of lists
  + Clean stop words using the stopword\_en.txt file