ALGORITHM TO OBTAIN TOTAL ORDER FROM PARTIAL ORDERS FOR SOCIAL NETWORKS

Chandrika Satyavolu
Advisor: Dr. Chris Pollett
Committee: Dr. David Taylor, Dr. Robert Chun

December 17th, 2008

Outline

- Problem
- Existing Solutions
- Algorithm
- Working example
- Getting back accurate total order
- Experiments to analyze space complexity
- Experiments to analyze fault tolerance
- Software design of test website: 100 best movies!
- Special features of 100 best movies!
- Comparing existing systems
- Conclusion
- References

Problem

- Develop an algorithm to obtain total order from partial orders.
- Partial orders are human-generated and cannot be produced by a computer program.
- Harness human intelligence to create more accurate total order.
- Build the algorithm into a test application: 100 best movies!
- 100 top movies of the total order is listed.

Existing Solutions

- Generate total order from user inputs using:
  - Absolute ratings
  - Absolute voting (vote for single item)
  - Multiple voting (vote for multiple items)
  - Voting and Ratings

Algorithm

QuickSort

- Recursively sorts the input list based on the sorted partial order sequences.

Partition

- Partitions the portion of input list into two parts. The last element of the list is chosen as the pivot element. The list is partitioned such that first part has elements that have occurred before the pivot element in most of the partial orders and the second part has numbers that have occurred after the pivot element in most of the partial orders.

GetRank

- Gets the rank of an element in the input list with respect to the pivot element based on partial orders. If it occurs before pivot element more number of times than after in the partial orders, it returns a rank 1 otherwise it returns a rank 0 .

Working Example of Partial Order QuickSort

Input list:

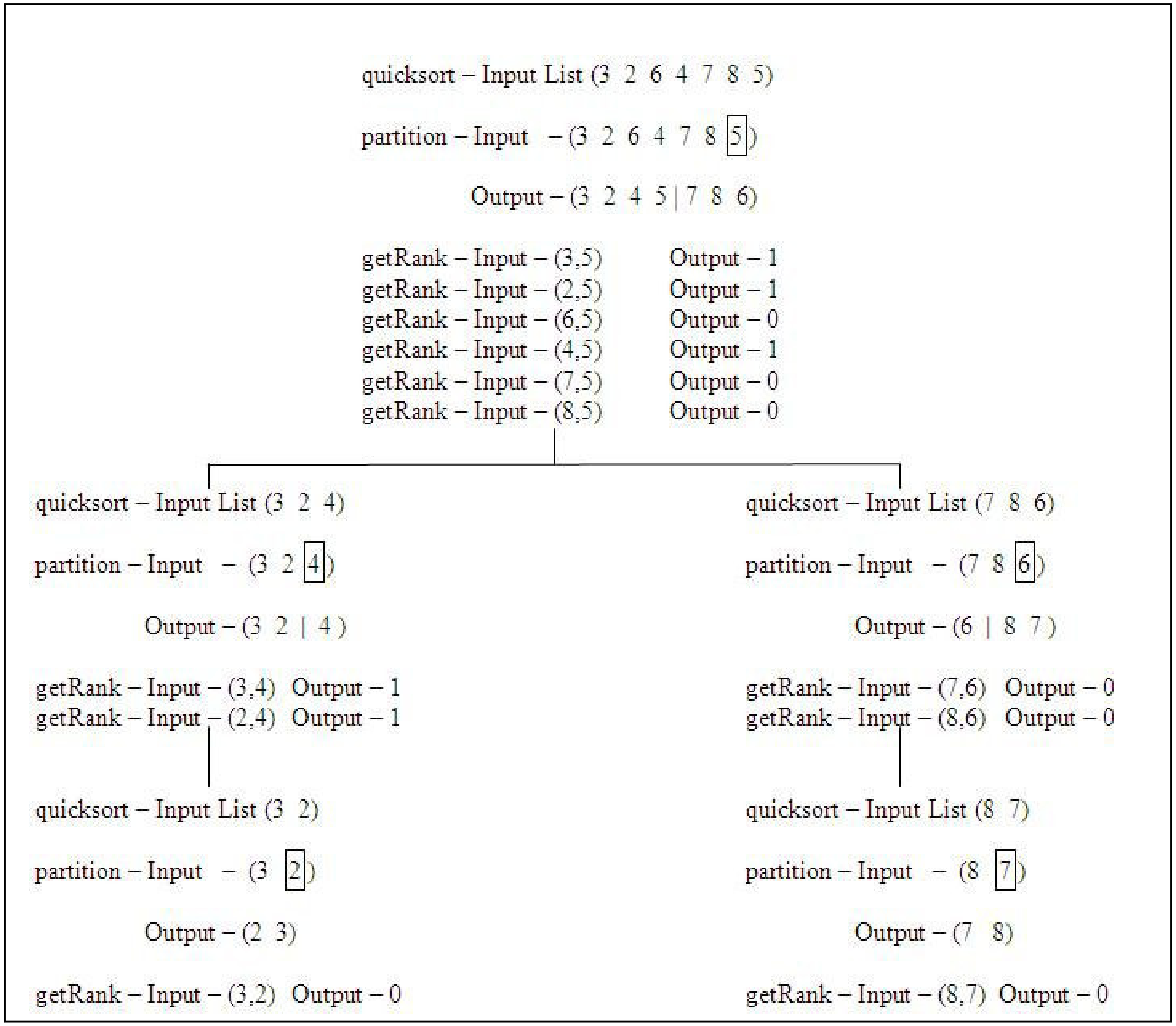
{3, 2, 6, 4, 7, 8, 5}

Partial orders:

{ {2, 3, 4, 5},
{2, 3, 4, 6},
{3, 4, 6, 7},
{5, 6, 7, 8},
{2, 5, 7, 8},
{3, 6, 7, 8},
{4, 6, 7, 8} }

Output list:

{2, 3, 4, 5, 6, 7, 8}



quicksort – Input List (3 2 6 4 7 8 5)

partition – Input – (3 2 6 4 7 8 $\boxed{5}$)

Output – (3 2 4 5 | 7 8 6)

getRank – Input – (3,5) Output – 1
getRank – Input – (2,5) Output – 1
getRank – Input – (6,5) Output – 0
getRank – Input – (4,5) Output – 1
getRank – Input – (7,5) Output – 0
getRank – Input – (8,5) Output – 0

quicksort – Input List (3 2 4)

partition – Input – (3 2 $\boxed{4}$)

Output – (3 2 | 4)

getRank – Input – (3,4) Output – 1
getRank – Input – (2,4) Output – 1

quicksort – Input List (3 2)

partition – Input – (3 $\boxed{2}$)

Output – (2 3)

getRank – Input – (3,2) Output – 0

quicksort – Input List (7 8 6)

partition – Input – (7 8 $\boxed{6}$)

Output – (6 | 8 7)

getRank – Input – (7,6) Output – 0
getRank – Input – (8,6) Output – 0

quicksort – Input List (8 7)

partition – Input – (8 $\boxed{7}$)

Output – (7 8)

getRank – Input – (8,7) Output – 0

Fig 1. Partial Order QuickSort Example

Getting back accurate total order

- There must be at least one partial order comparing every pair of elements in the list.

- There must be at least one partial order comparing every element in the list to every pivot element.

- Compare number of partial orders containing one element as opposed to other. (Compare popularity of two elements.)

Experiments to analyze space complexity

- **Terminology:**
  - n – total number of elements
  - k – number of partial orders
  - m – partial order set size

- Partial orders are stored in the database. Product of k and m *(k x m)* gives space complexity of the algorithm.

- Number of partial orders is inversely proportional to partial order set size. *(k α 1/m)*

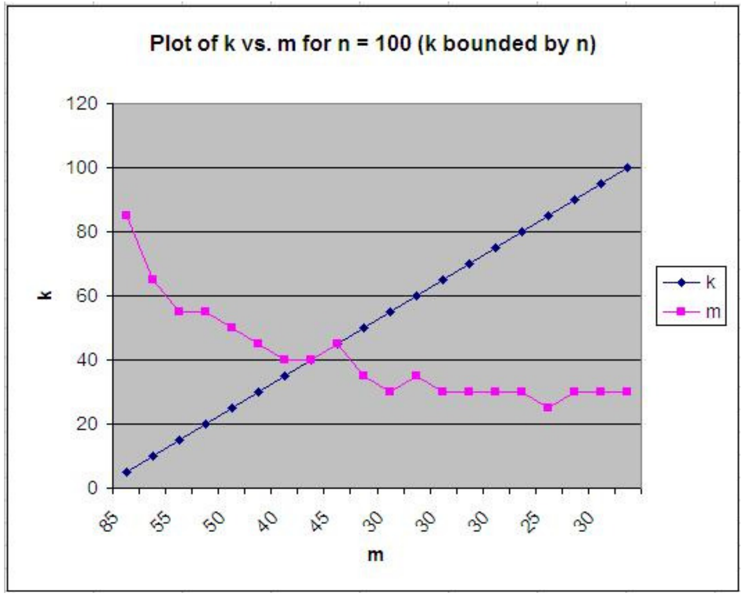
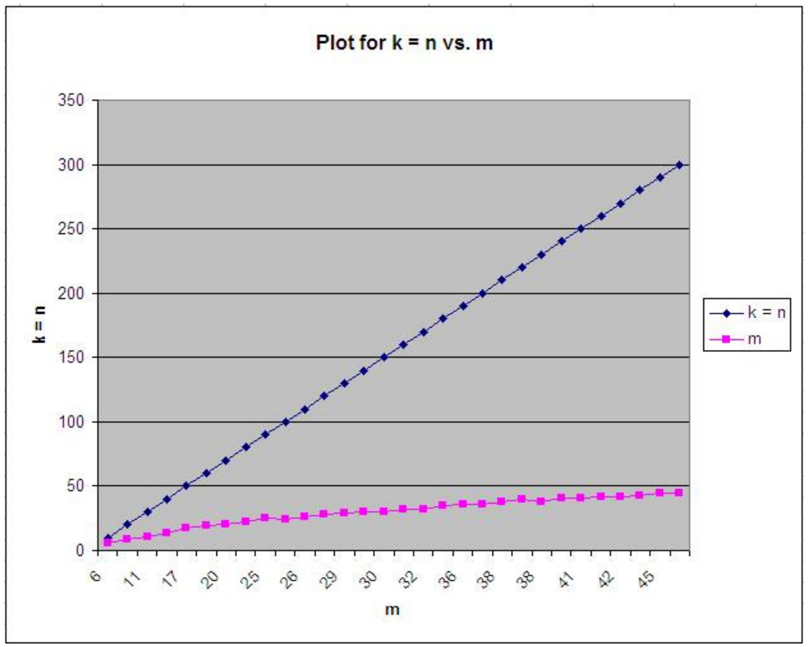


Fig 2. Plot of k vs. m for n = 100 (k bounded by n)

Fig 3. Plot for k = n vs. m



- Setting k=n, we can prove that partial order set size is a function of total number of elements.

- Not practical to have a large partial order set size. Larger is the partial order set size, lower is the accuracy of relative rankings in the partial order.

- Set partial order set size to a practical size (m=10) and proceed to derive the relationship between total number of elements (n) and number of partial orders (k) required to get back the accurate total order. *(k α f(n))*

Experiments to analyze space complexity (contd.)

Terminology:

- n – total number of elements
- k – number of partial orders
- m – partial order set size

Experiment 1: Derive the relationship between k and n for a fixed value of m (= 10).

Case: m=10

k = C x n { C = 1, 2, 5, 10, 50 }

Conclusion: As n increases, constant C must increase to get back correct total order. Therefore,

C = f(n) or k = f(n) x n

Experiment 2: Derive the functional dependency of k on n for fixed m (= 10).

Case: m=10

k = f(n) x n { f(n) = n, n/2, sqrt(n), log(n) }

Conclusion: As n increases, f(n) = sqrt(n) and log(n) yeilds poor results. f(n) = n and n/2 always get back correct total order.

Therefore, f(n) = C x n where 0 < C <= 1 or

k = C x n^2 where 0 < C <= 1

Experiment 3: Derive the closest constant relating k with n^2 for a fixed value of m (= 10).

Case: m=10

k = C x n^2 { C = 1/4, 1/6, 1/8 }

Conclusion: k = (1/4) n^2 gives back a correct total order every single time.

k = (1/6) n^2, give back the total order correctly too most of the time.

k = (1/8) n^2, does not give back the total order on many occasions.

Therefore, 1/4 < C < 1/8 or C ~ 1/6

Experiments to affirm fault tolerance

- **Terminology:**
  - n – total number of elements
  - k – number of partial orders
  - m – partial order set size

- **Experiment:** For different values of n (300 >= n >= 20), errors are introduced into partial orders. Here:

  p = % of error introduced.

  i.e p % of partial orders have single error in them.

- **Observations:**
  - A small percentage of error (p <= 10%) introduced cannot deduce the fault tolerance property of the algorithm.
  - As the error introduced is a two-element error, there is a good possibility of getting back the correct total order even if the error is large.
  - Even a 100% two-element error introduced had no greater than 5% chance of the total order being incorrect.

	Number of 'n' values giving incorrect total orders (Total no. of values = 30)	**% error in 30 values of n**
p = 5% error	2	0.6
p = 10% error	3	0.9
p = 20% error	4	1.2
p = 50% error	4	1.2
p = 75% error	10	3
p = 90% error	10	3
p=100% error	7	2.1

Table 1. Error introduced vs. Fault Tolerance

Software design of test website: 100 best movies!

- The software components and the functions associated with them are listed in Table 2.
- Non-registered users can only view homepage listing top 100 movies.
- Only registered users can use functionalities of 100 best movies!
- User must be signed in to:
  - Add movie
  - Rank movie
  - Remove Movie

Components	Functions
User	register, signin
Movie	add , rank , remove , group, select
Group	create, select, modify, rank
Partial Orders	create, sort

Table 2. Software Components

Software design of test website: 100 best movies!

- Movies,Users and Groups are unique.
- Group may or may not be saved while creating a partial order.
- Each user can request for a movie removal only once.
- Partial order stored as unique 4-tuples:

  order_id, user_id, movie_id, rank

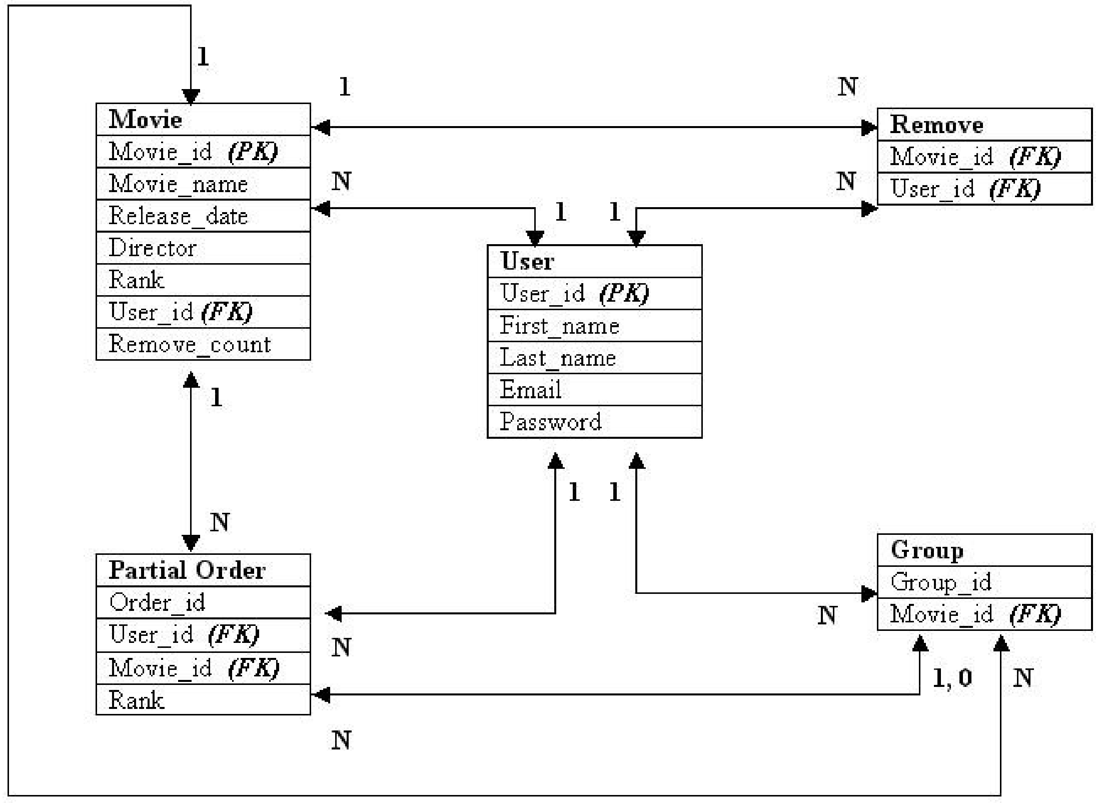


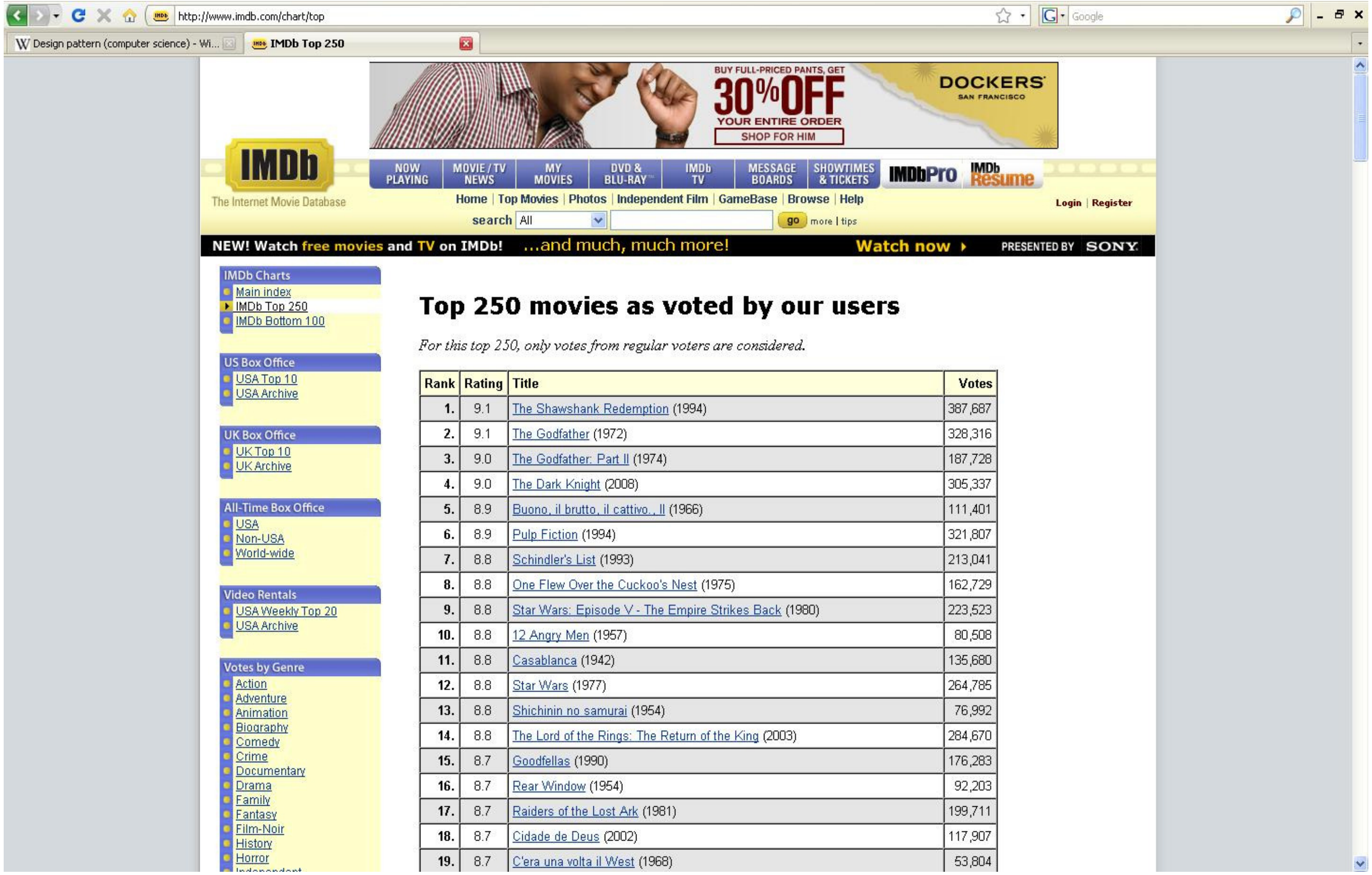
Fig 4. Entity Relationship diagram.

Special features of 100 best movies!

- Partial order sorting is performed over database.
- Variable partial order set size allowed.
- User can influence removal of movie.
- Ranking techniques allowed by 100 best movies!:
  - Sort a set
  - Sort a set and save it as group
  - Select existing group and sort
  - Select existing group, add and/or remove movies and sort
  - Select existing group, add and/or remove movies, sort and save set as new group

Comparing existing systems

- Case 1: Total order is derived from voting and user ratings.



Top 250 movies as voted by our users

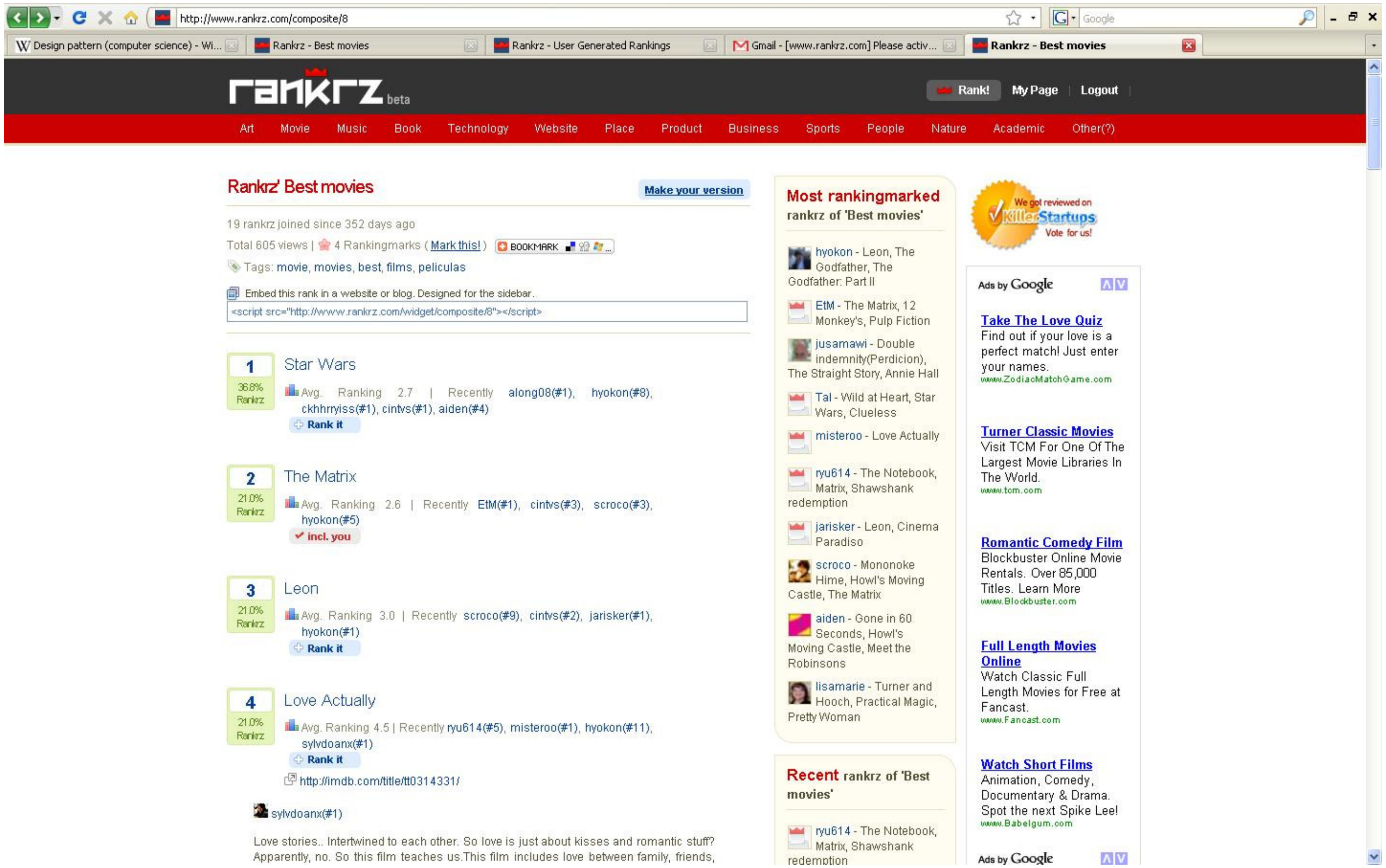
For this top 250, only votes from regular voters are considered.

Rank	Rating	Title	Votes
1.	9.1	The Shawshank Redemption (1994)	387,687
2.	9.1	The Godfather (1972)	328,316
3.	9.0	The Godfather: Part II (1974)	187,728
4.	9.0	The Dark Knight (2008)	305,337
5.	8.9	Buono, il brutto, il cattivo., Il (1966)	111,401
6.	8.9	Pulp Fiction (1994)	321,807
7.	8.8	Schindler's List (1993)	213,041
8.	8.8	One Flew Over the Cuckoo's Nest (1975)	162,729
9.	8.8	Star Wars: Episode V - The Empire Strikes Back (1980)	223,523
10.	8.8	12 Angry Men (1957)	80,508
11.	8.8	Casablanca (1942)	135,680
12.	8.8	Star Wars (1977)	264,785
13.	8.8	Shichinin no samurai (1954)	76,992
14.	8.8	The Lord of the Rings: The Return of the King (2003)	284,670
15.	8.7	Goodfellas (1990)	176,283
16.	8.7	Rear Window (1954)	92,203
17.	8.7	Raiders of the Lost Ark (1981)	199,711
18.	8.7	Cidade de Deus (2002)	117,907
19.	8.7	C'era una volta il West (1968)	53,804

Fig 5. www.imdb.com

Comparing existing systems (contd.)

- Case 2: Total order is derived from multiple voting.



Rankrz' Best movies

Make your version

19 rankrz joined since 352 days ago

Total 605 views | 4 Rankingmarks (Mark this!) BOOKMARK

Tags: movie, movies, best, films, peliculas

Embed this rank in a website or blog. Designed for the sidebar.

<script src="http://www.rankrz.com/widget/composite/8"></script>

1 36.8% Rankrz	Star Wars	Avg. Ranking 2.7 | Recently along08(#1), hyokon(#8), ckhhrryiss(#1), cintvs(#1), aiden(#4) Rank it
2 21.0% Rankrz	The Matrix	Avg. Ranking 2.6 | Recently EtM(#1), cintvs(#3), scroco(#3), hyokon(#5) ✓ incl. you
3 21.0% Rankrz	Leon	Avg. Ranking 3.0 | Recently scroco(#9), cintvs(#2), jarisker(#1), hyokon(#1) Rank it
4 21.0% Rankrz	Love Actually	Avg. Ranking 4.5 | Recently ryu614(#5), misteroo(#1), hyokon(#11), sylvdoanx(#1) Rank it

http://imdb.com/title/tt0314331/

sylvdoanx(#1)

Love stories.. Intertwined to each other. So love is just about kisses and romantic stuff? Apparently, no. So this film teaches us.This film includes love between family, friends,

Most rankingmarked rankrz of 'Best movies'

hyokon - Leon, The Godfather, The Godfather: Part II

EtM - The Matrix, 12 Monkey's, Pulp Fiction

jusamawi - Double indemnity(Perdicion), The Straight Story, Annie Hall

Tal - Wild at Heart, Star Wars, Clueless

misteroo - Love Actually

ryu614 - The Notebook, Matrix, Shawshank redemption

jarisker - Leon, Cinema Paradiso

scroco - Mononoke Hime, Howl's Moving Castle, The Matrix

aiden - Gone in 60 Seconds, Howl's Moving Castle, Meet the Robinsons

lisamarie - Turner and Hooch, Practical Magic, Pretty Woman

Recent rankrz of 'Best movies'

ryu614 - The Notebook, Matrix, Shawshank redemption

Fig 6. www.rankrz.com

Comparing existing systems (contd.)

Comparing 100 best movies! and rankrz.com with an example case:

Input list:	5, 8, 1, 2, 4, 6, 10, 3, 7, 9
User input:	1, 2, 3, 4, 5, 6
	1, 2, 3, 4, 5, 7
	1, 2, 3, 4, 5, 8
	1, 2, 3, 4, 5, 9
	1, 2, 3, 4, 5, 10
	5, 6, 7, 8, 9, 10

Note: 100 best movies! sees the user inputs as relative partial orders whereas rankrz.com sees the user inputs as list movies users voted for.

100 best movies! total order: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

rankrz.com total order: 5, 1, 2, 3, 4, 6, 7, 8, 9, 10

Conclusion

- Partial Order QuickSort algorithm sorts elements that are not possible to be ordered by a computer program.
- It harnesses human intelligence.
- It generates an accurate total order.
- Our work can be extended to have the total ordering done by variation of other sorting algorithms. The performance of using different sort algorithms can be compared and documented.

References

[1] Introduction to Algorithms. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Prentice-Hall, 2nd edition, 2002.

[2] Intelligent Planning - A Decomposition and Abstraction Based Approach. Qiang Yang, Springer-Verlag, 1997.

[3] Artifical Intelligence: A Modern Approach. S. Russell and P. Norvig, Prentice-Hall, 2nd edition, 2002.

[4] Depth Optimal Sorting Networks Resistant to k Passive Faults, M Piotrów, Proc. 7th SIAM Symposium on Discrete Algorithms, 1996.

[5] Quicksort example, http://www.cise.ufl.edu/~ddd/cis3020/summer-97/lectures/lec17/sld003.htm

[6] http://en.wikipedia.org/wiki/Quicksort

[7] www.imdb.com

[8] www.rankrz.com

Thank you.

Q&A