Expedia

CRML
Control Robotics and Machine Learning Laboratory

הטכניון
מכון טכנולוגי
לישראל

Learning To Rank Hotels To Maximize Purchases

Abstract

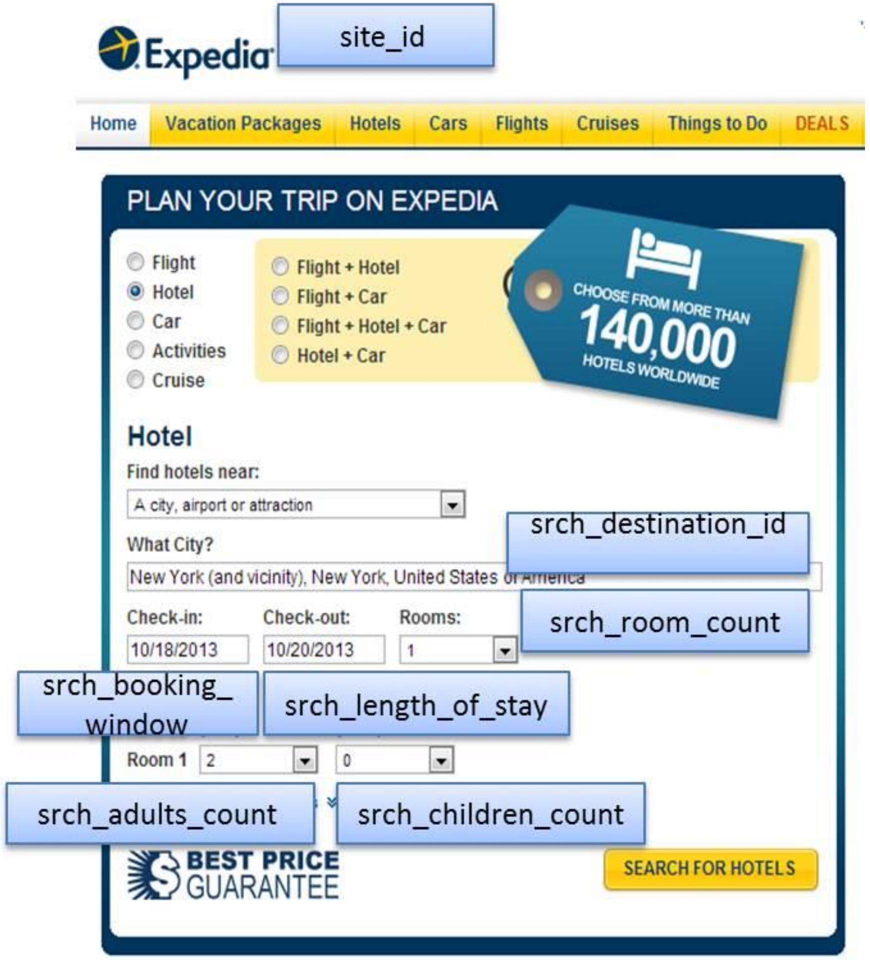
- Many customers search and purchase hotels online.
- Companies such as Expedia make their profit from purchases made through their sites.
- The ultimate goal – top of the list are the hotels that are most likely to be purchased by the user.

The Challenge

- Kaggle – Predictive Modeling competitions.
- Expedia – hotel ranking challenge through Kaggle.
- Data set used provided by Expedia.

Control Robotics and Machine Learning Laboratory

Searching at Expedia



site_id

Home | Vacation Packages | Hotels | Cars | Flights | Cruises | Things to Do | DEALS

PLAN YOUR TRIP ON EXPEDIA

○ Flight
● Hotel
○ Car
○ Activities
○ Cruise

○ Flight + Hotel
○ Flight + Car
○ Flight + Hotel + Car
○ Hotel + Car

CHOOSE FROM MORE THAN 140,000 HOTELS WORLDWIDE

Hotel

Find hotels near: A city, airport or attraction

What City? New York (and vicinity), New York, United States of America

Check-in: 10/18/2013 Check-out: 10/20/2013 Rooms: 1

Room 1 2 0

BEST PRICE GUARANTEE

SEARCH FOR HOTELS

srch_destination_id

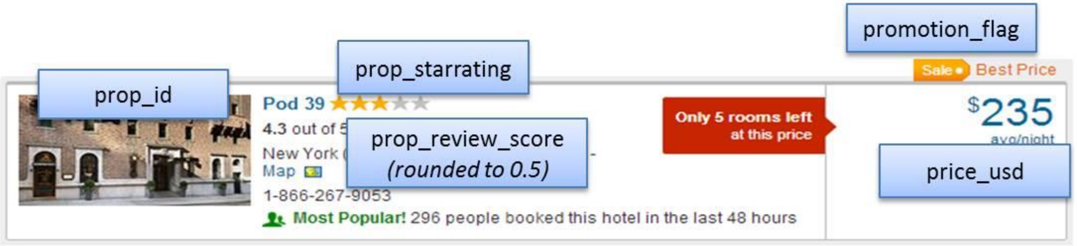
srch_room_count

srch_booking_window

srch_length_of_stay

srch_adults_count

srch_children_count



promotion_flag

prop_id

prop_starrating

prop_review_score (rounded to 0.5)

price_usd

Pod 39

4.3 out of 5

New York

Map

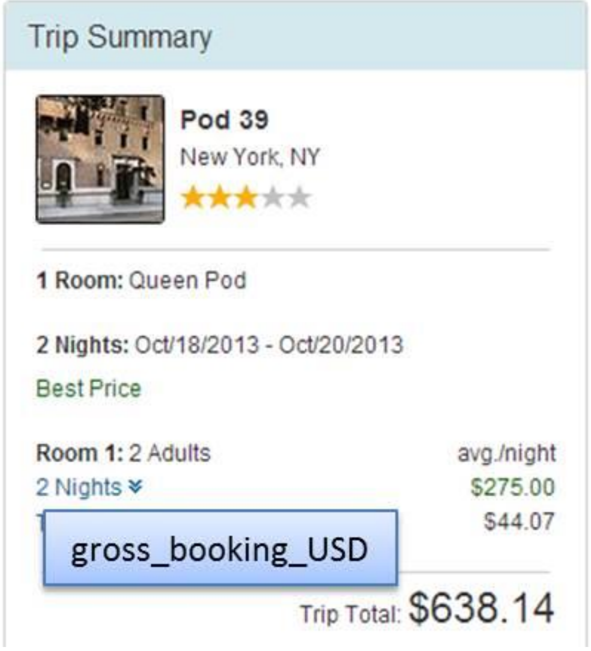
1-866-267-9053

Most Popular! 296 people booked this hotel in the last 48 hours

Only 5 rooms left at this price

Sale Best Price

$235 avg/night



Trip Summary

Pod 39

New York, NY

1 Room: Queen Pod

2 Nights: Oct/18/2013 - Oct/20/2013

Best Price

Room 1: 2 Adults avg./night

2 Nights $275.00

$44.07

Trip Total: $638.14

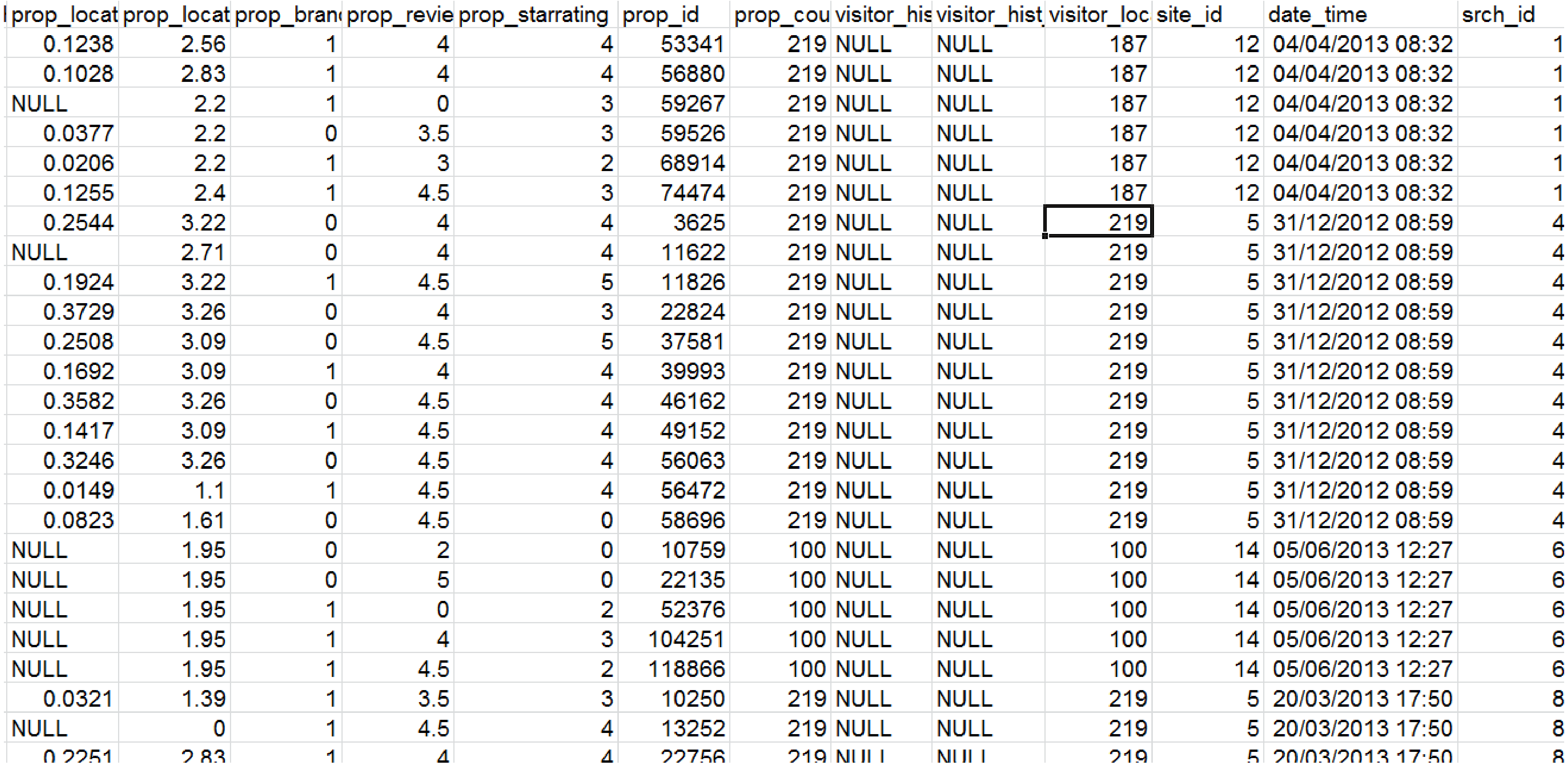
gross_booking_USD

The Data

- Each query shows multiple samples
- Each sample represents a hotel.
- A sample provides information on the hotel's cost, ratings etc.
- There are features that describe the search query and user history – same for all samples in query.

Control Robotics and Machine Learning Laboratory

The Data



l	prop_locat	prop_locat	prop_bran	prop_revie	prop_starrating	prop_id	prop_cou	visitor_his	visitor_hist	visitor_loc	site_id	date_time	srch_id
	0.1238	2.56	1	4	4	53341	219	NULL	NULL	187	12	04/04/2013 08:32	1
	0.1028	2.83	1	4	4	56880	219	NULL	NULL	187	12	04/04/2013 08:32	1
	NULL	2.2	1	0	3	59267	219	NULL	NULL	187	12	04/04/2013 08:32	1
	0.0377	2.2	0	3.5	3	59526	219	NULL	NULL	187	12	04/04/2013 08:32	1
	0.0206	2.2	1	3	2	68914	219	NULL	NULL	187	12	04/04/2013 08:32	1
	0.1255	2.4	1	4.5	3	74474	219	NULL	NULL	187	12	04/04/2013 08:32	1
	0.2544	3.22	0	4	4	3625	219	NULL	NULL	219	5	31/12/2012 08:59	4
	NULL	2.71	0	4	4	11622	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.1924	3.22	1	4.5	5	11826	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.3729	3.26	0	4	3	22824	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.2508	3.09	0	4.5	5	37581	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.1692	3.09	1	4	4	39993	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.3582	3.26	0	4.5	4	46162	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.1417	3.09	1	4.5	4	49152	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.3246	3.26	0	4.5	4	56063	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.0149	1.1	1	4.5	4	56472	219	NULL	NULL	219	5	31/12/2012 08:59	4
	0.0823	1.61	0	4.5	0	58696	219	NULL	NULL	219	5	31/12/2012 08:59	4
	NULL	1.95	0	2	0	10759	100	NULL	NULL	100	14	05/06/2013 12:27	6
	NULL	1.95	0	5	0	22135	100	NULL	NULL	100	14	05/06/2013 12:27	6
	NULL	1.95	1	0	2	52376	100	NULL	NULL	100	14	05/06/2013 12:27	6
	NULL	1.95	1	4	3	104251	100	NULL	NULL	100	14	05/06/2013 12:27	6
	NULL	1.95	1	4.5	2	118866	100	NULL	NULL	100	14	05/06/2013 12:27	6
	0.0321	1.39	1	3.5	3	10250	219	NULL	NULL	219	5	20/03/2013 17:50	8
	NULL	0	1	4.5	4	13252	219	NULL	NULL	219	5	20/03/2013 17:50	8
	0.2251	2.83	1	4	4	22756	219	NULL	NULL	219	5	20/03/2013 17:50	8

Modeling

- Rank(query,hotel) = q^TAh
- s.t:
  - q is the query features vector
  - h is the hotel features vector

Complexity And Difficulties

- Multi class problem – purchased, clicked, neither.
- Non-coherent data – some examples might be missing details that other has.
- Different features have different representation.
- Evaluation metric is NDCG – order is important.
    - See Appendix 1.

Pre Processing

Goals

- Unified representation of different types of features.
- Compensating missing data.
- Creating New features.
- Flexibility – easy to modify.

Pre Processing

Implementation

- Transforming into unified binary representation.
- Limit – predetermined value to define number of quantization levels.
- Boundaries – thresholds values for quantization.
- New features – Average, Median, Variance, Abs stars diff.

Algorithm 1 - Ranking SVM

Rank SVM

$$\underset{A,\xi_i,\zeta_j}{\arg\min}\{\frac{1}{2}\|A\|^2+C1\sum_{i=1}^{n}\xi_i+C2\sum_{j=1}^{m}\zeta_j\}$$

$$s.t: \qquad \forall i\in[1,n],\forall j\in[1,m]$$

$$q^TAh_2-q^TAh_1\geq 1-\xi_i$$

$$q^TAh_1-q^TAh_0\geq 1-\zeta_j$$

$$\xi_i\geq 0,\zeta_j\geq 0$$

Solution with Matpower – a matlab package.

Algorithm 1 - Ranking SVM

Setbacks

- Memory complexity $O(n^2)$ at best ☹
- High time complexity (empirically).
- Not optimizing NDCG evaluation metric directly.

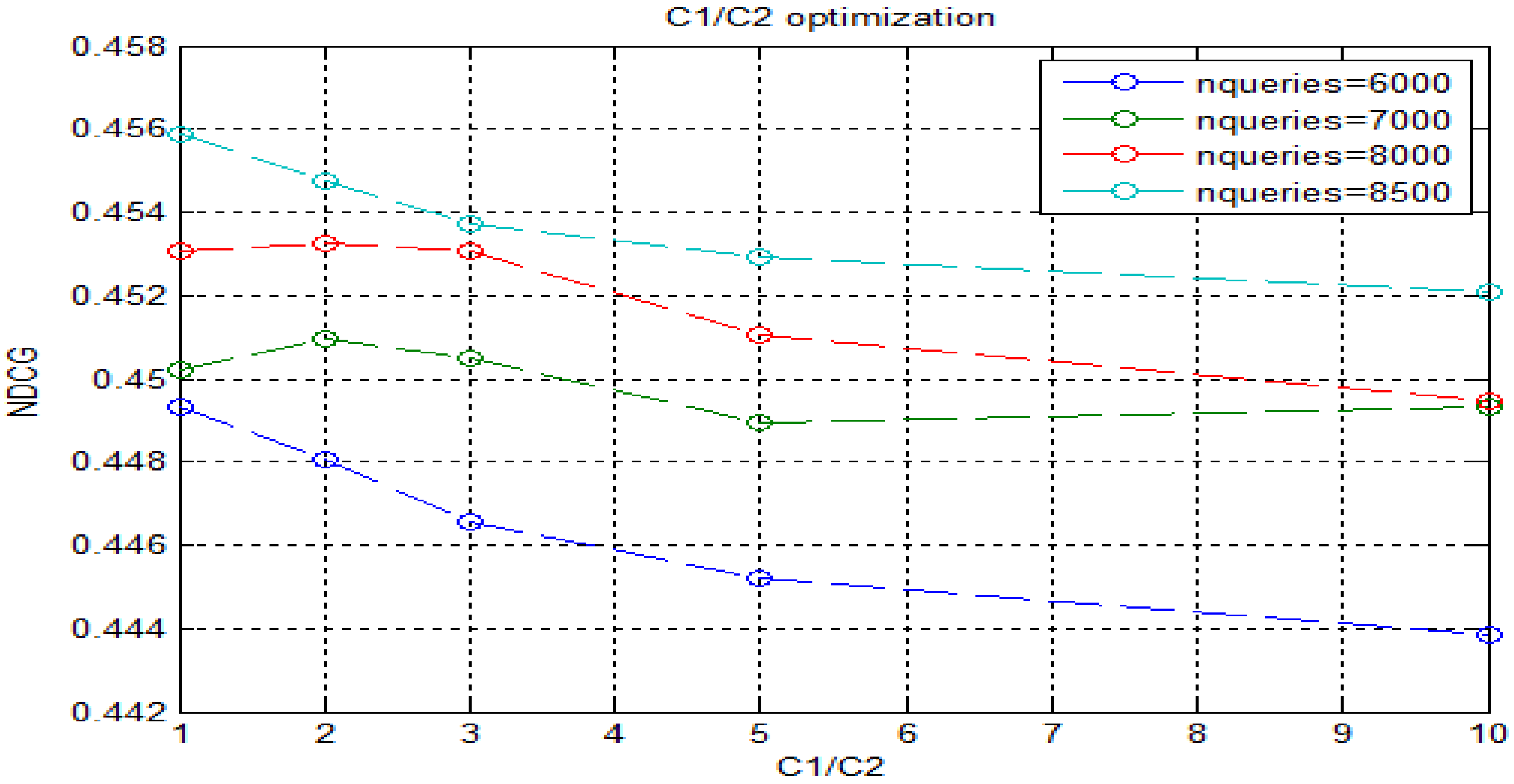
Algorithm 1 - Ranking SVM

Data Filtering

- Max no. of unclicked hotels (Max_Hotels).
    - Max no. of queries without bound was 9000.
    - Max no. of queries with Max_Hotels=5 was 18000
- Ignoring less effective features.

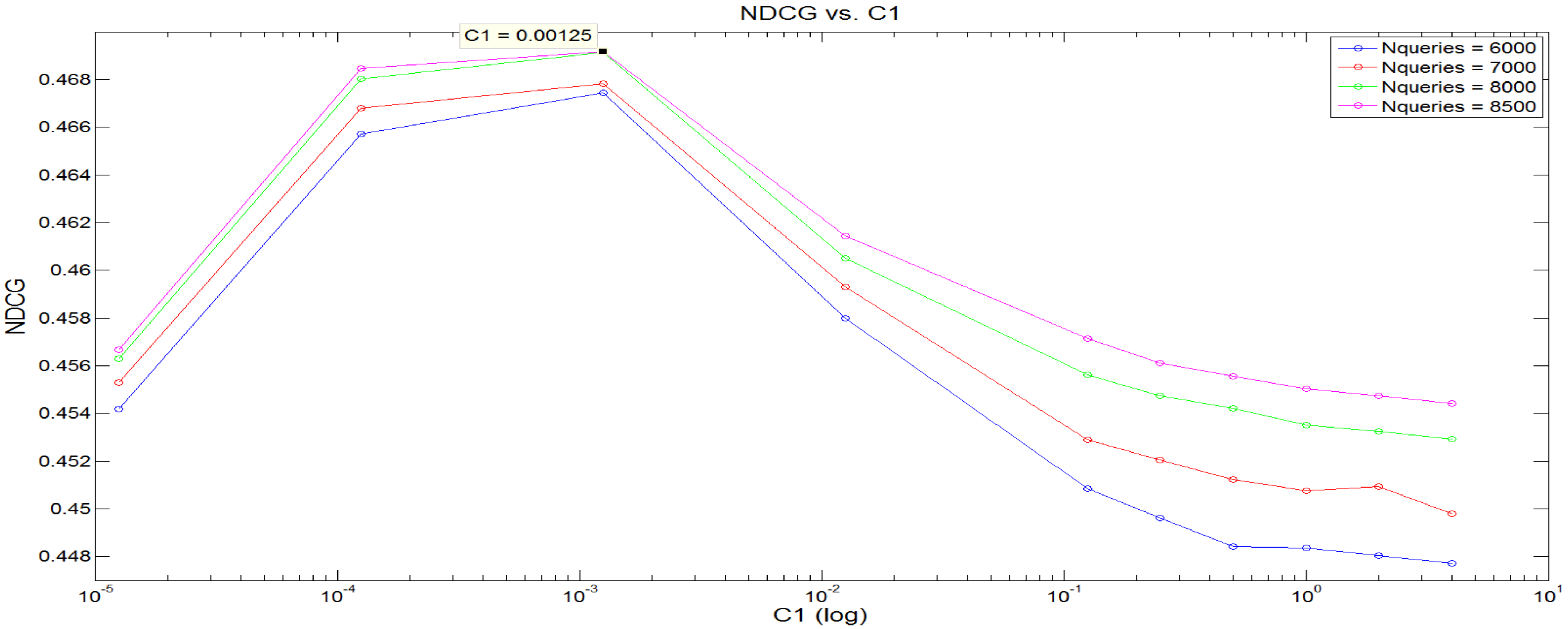
Algorithm 1 - Ranking SVM

Choosing C1/C2 Ratio



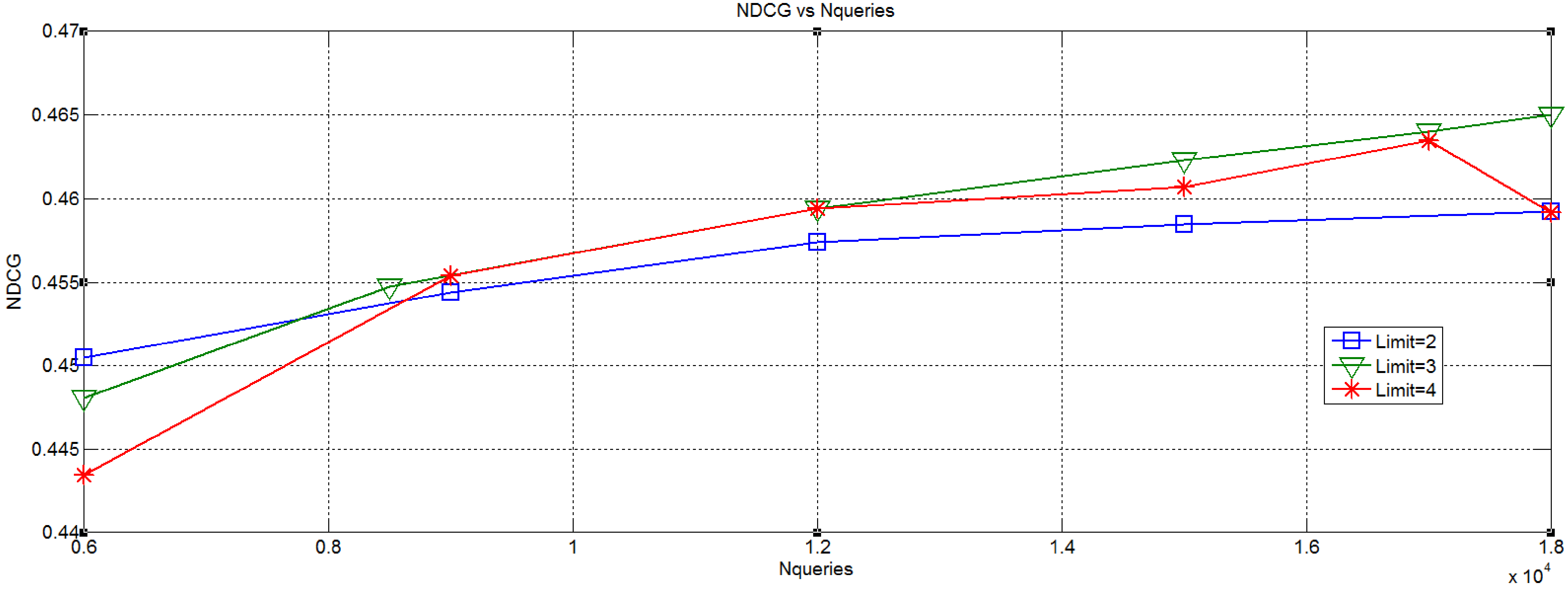
Algorithm 1 - Ranking SVM

Choosing C1,C2 values



Algorithm 1 - Ranking SVM

Limit



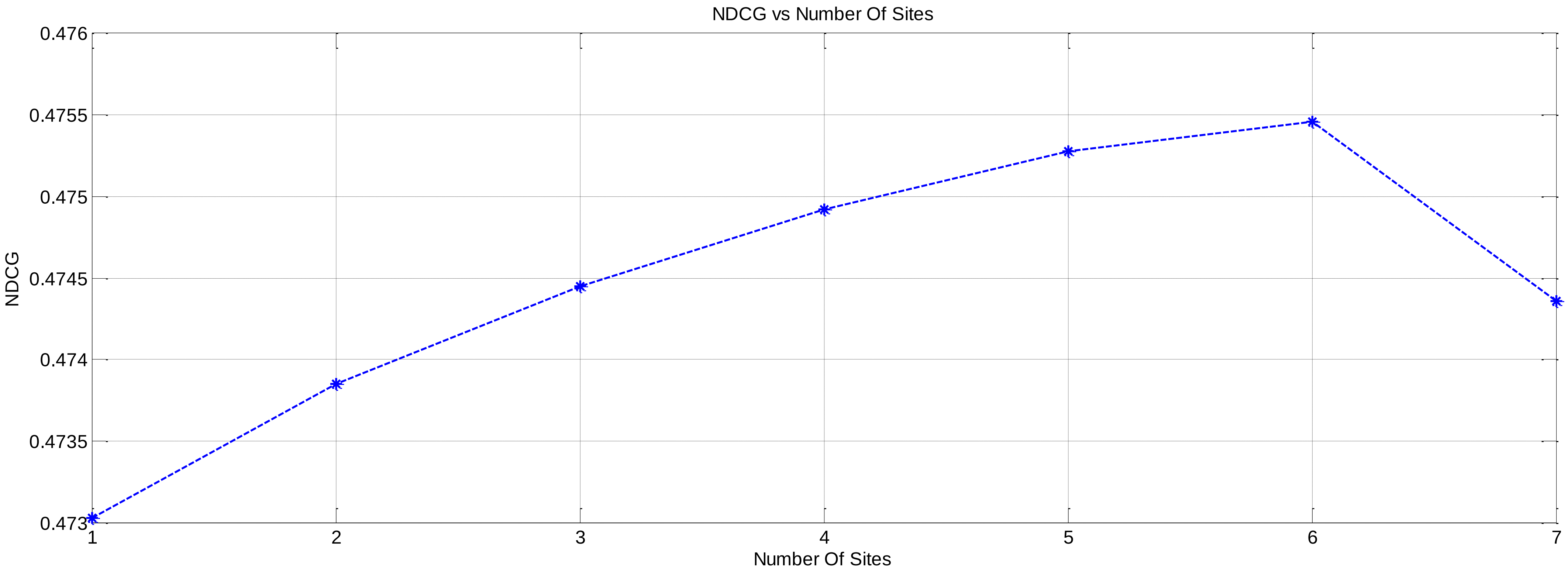
Algorithm 1 - Ranking SVM

Divide & Conquer - Explained

- Break into smaller similar problems.
- Select a feature to divide the train set into disjoint sets.
- Solve for each set separately.
- Feature selected – Site ID.

Algorithm 1 - Ranking SVM

Divide & Conquer – Results

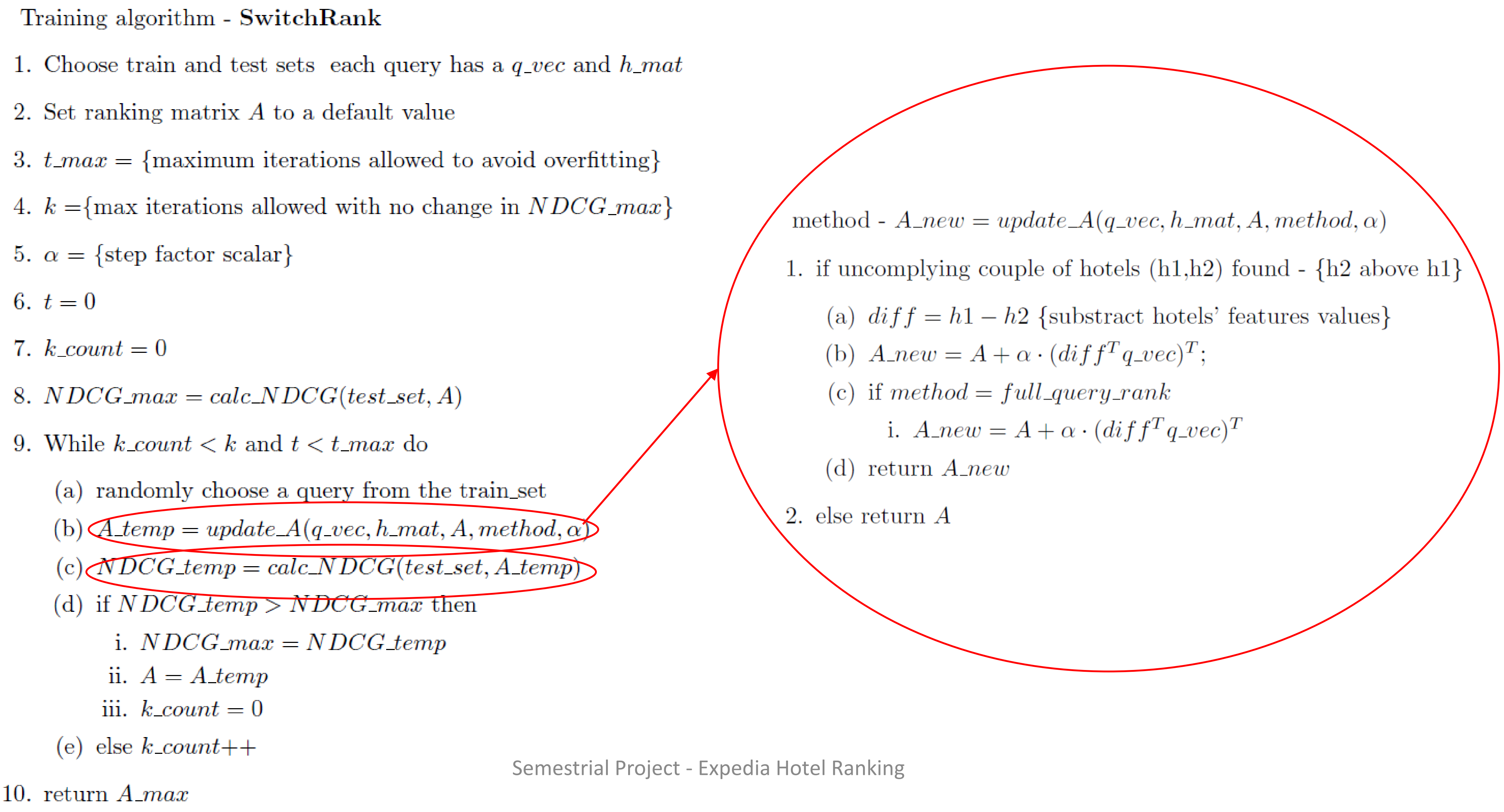


Algorithm 2 - SwitchRank

Concept

- Perceptron-like algorithm, without classifying.
- Find non-complying hotels.
- Update ranking matrix to reinforce correct ranking.
- Use updated matrix if NDCG value improves.

Algorithm 2 - SwitchRank



Training algorithm - **SwitchRank**

```
1. Choose train and test sets each query has a $q\_vec$ and $h\_mat$
2. Set ranking matrix $A$ to a default value
3. $t\_max = \{$maximum iterations allowed to avoid overfitting$\}$
4. $k =\{$max iterations allowed with no change in $NDCG\_max\}$
5. $\alpha = \{$step factor scalar$\}$
6. $t = 0$
7. $k\_count = 0$
8. $NDCG\_max = calc\_NDCG(test\_set, A)$
9. While $k\_count < k$ and $t < t\_max$ do
    (a) randomly choose a query from the train_set
    (b) $A\_temp = update\_A(q\_vec, h\_mat, A, method, \alpha)$
    (c) $NDCG\_temp = calc\_NDCG(test\_set, A\_temp)$
    (d) if $NDCG\_temp > NDCG\_max$ then
        i. $NDCG\_max = NDCG\_temp$
        ii. $A = A\_temp$
        iii. $k\_count = 0$
    (e) else $k\_count{+}{+}$
10. return $A\_max$
```

method - $A\_new = update\_A(q\_vec, h\_mat, A, method, \alpha)$

```
1. if uncomplying couple of hotels (h1,h2) found - {h2 above h1}
    (a) $diff = h1 - h2$ {substract hotels' features values}
    (b) $A\_new = A + \alpha \cdot (diff^T q\_vec)^T$;
    (c) if $method = full\_query\_rank$
        i. $A\_new = A + \alpha \cdot (diff^T q\_vec)^T$
    (d) return $A\_new$
2. else return $A$
```

Algorithm 2 - SwitchRank

Full query ranking example

Iteration 1 the order is:

index	click	purch	rank
5	1	0	70.00
6	0	0	70.00
8	0	0	70.00
10	0	0	70.00
11	0	0	70.00
13	0	0	70.00
15	0	0	70.00
16	0	0	70.00
19	0	0	70.00
20	0	0	70.00
21	0	0	70.00
25	1	0	70.00
26	1	1	70.00
27	0	0	70.00
31	0	0	70.00

Iteration 2 the order is:

index	click	purch	rank
26	1	1	72.45
6	0	0	71.40
27	0	0	71.40
11	0	0	71.05
13	0	0	71.05
10	0	0	70.70
16	0	0	70.70
19	0	0	70.70
21	0	0	70.70
25	1	0	70.35
8	0	0	70.00
31	0	0	69.30
15	0	0	68.95
20	0	0	68.25
5	1	0	67.55

Iteration 3 the order is:

index	click	purch	rank
26	1	1	70.70
8	0	0	70.00
10	0	0	70.00
11	0	0	70.00
16	0	0	70.00
27	0	0	70.00
5	1	0	69.65
15	0	0	69.65
19	0	0	69.65
20	0	0	69.65
21	0	0	69.65
25	1	0	69.65
31	0	0	69.65
6	0	0	69.30
13	0	0	69.30

Iteration 4 the order is:

index	click	purch	rank
25	1	0	71.40
26	1	1	71.05
10	0	0	70.70
27	0	0	70.70
11	0	0	70.35
6	0	0	70.00
19	0	0	70.00
20	0	0	70.00
21	0	0	69.65
5	1	0	69.65
15	0	0	69.65
13	0	0	69.30
31	0	0	69.30
16	0	0	68.95
8	0	0	68.25

Iteration 5 the order is:

index	click	purch	rank
26	1	1	72.45
11	0	0	71.05
10	0	0	70.70
27	0	0	70.70
6	0	0	70.35
19	0	0	70.35
21	0	0	70.00
25	1	0	70.00
13	0	0	69.65
5	1	0	68.95
16	0	0	68.95
20	0	0	68.95
15	0	0	68.60
31	0	0	68.60
8	0	0	68.25

Algorithm 2 - SwitchRank

Full query ranking example

Iteration 6 the order is:

index	click	purch	rank
5	1	0	70.70
26	1	1	70.70
20	0	0	70.00
10	0	0	69.65
25	1	0	69.65
27	0	0	69.65
11	0	0	69.30
15	0	0	69.30
6	0	0	68.95
19	0	0	68.95
31	0	0	68.95
21	0	0	68.60
13	0	0	68.25
8	0	0	68.25
16	0	0	68.25

Iteration 7 the order is:

index	click	purch	rank
26	1	1	73.15
27	0	0	71.05
10	0	0	70.35
11	0	0	70.35
6	0	0	70.35
25	1	0	70.00
19	0	0	69.65
13	0	0	69.30
21	0	0	69.30
16	0	0	68.95
5	1	0	68.25
15	0	0	68.25
20	0	0	68.25
31	0	0	68.25
8	0	0	68.25

Iteration 8 the order is:

index	click	purch	rank
26	1	1	71.40
5	1	0	70.35
11	0	0	69.65
10	0	0	69.65
20	0	0	69.65
6	0	0	68.95
15	0	0	68.95
19	0	0	68.95
25	1	0	68.95
27	0	0	68.95
21	0	0	68.60
31	0	0	68.60
13	0	0	68.25
8	0	0	67.90
16	0	0	67.90

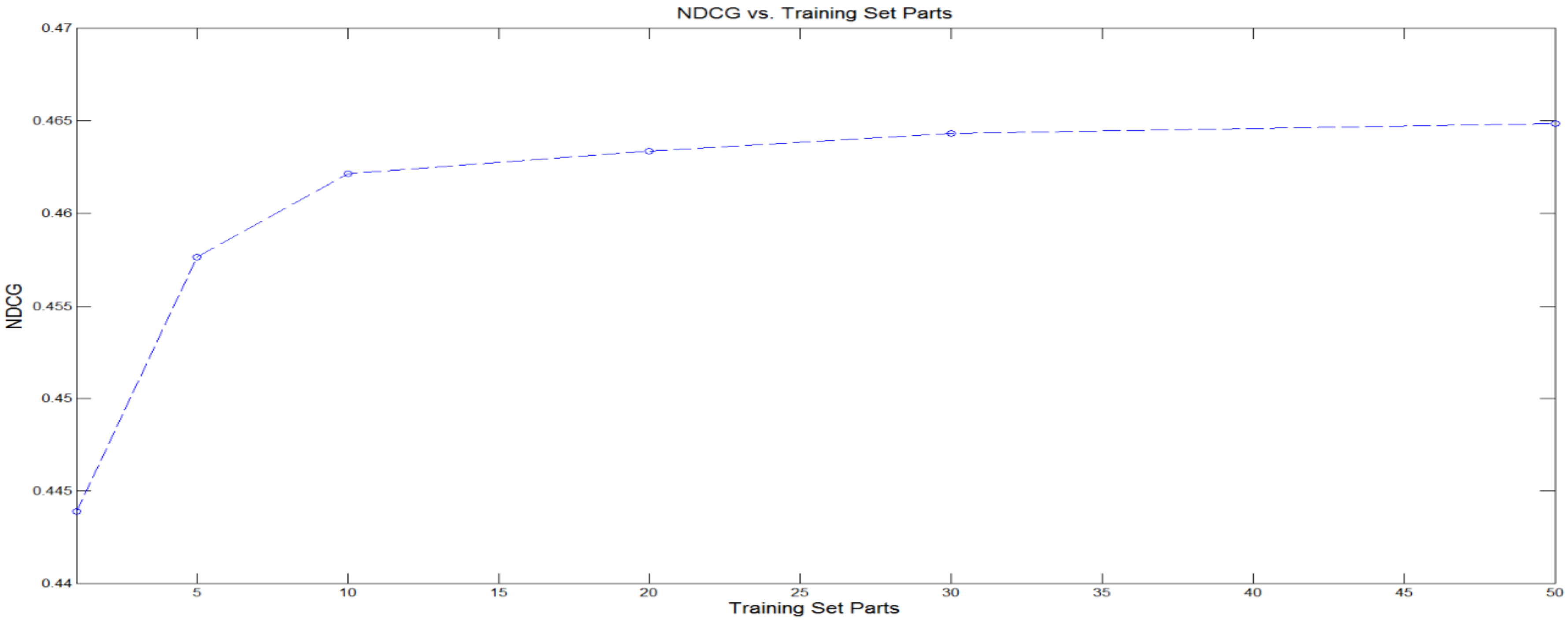
Iteration 9 the order is:

index	click	purch	rank
26	1	1	70.70
5	1	0	70.35
25	1	0	70.35
20	0	0	70.00
15	0	0	69.65
10	0	0	69.30
27	0	0	69.30
31	0	0	68.95
6	0	0	68.60
11	0	0	68.25
8	0	0	67.90
16	0	0	67.90
19	0	0	67.90
13	0	0	67.55
21	0	0	67.55

Algorithm 2 - SwitchRank

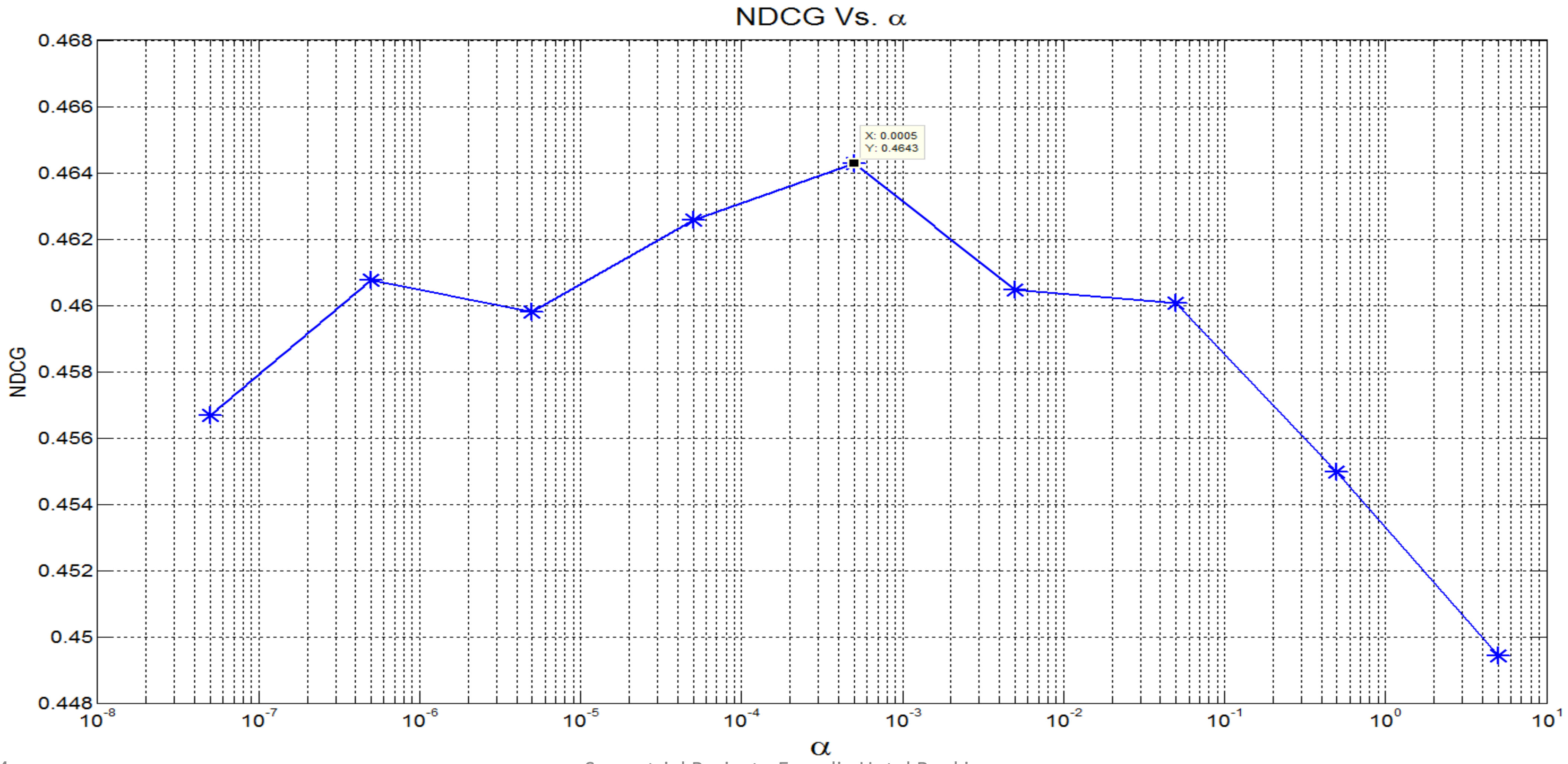
Cross Training

- Train on smaller parts.
- Calculate performance with average matrix.



Algorithm 2 - SwitchRank

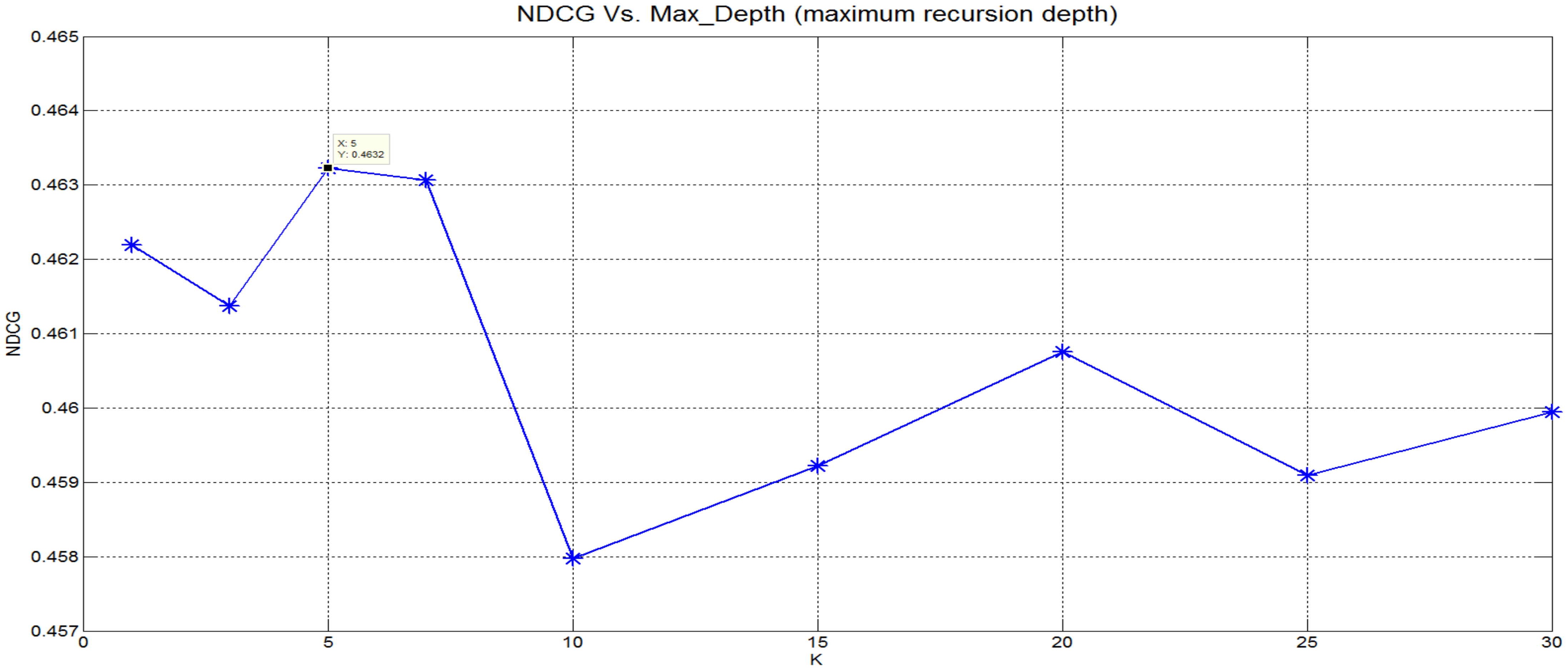
Choosing alpha factor



Control Robotics and Machine Learning Laboratory

Algorithm 2 - SwitchRank

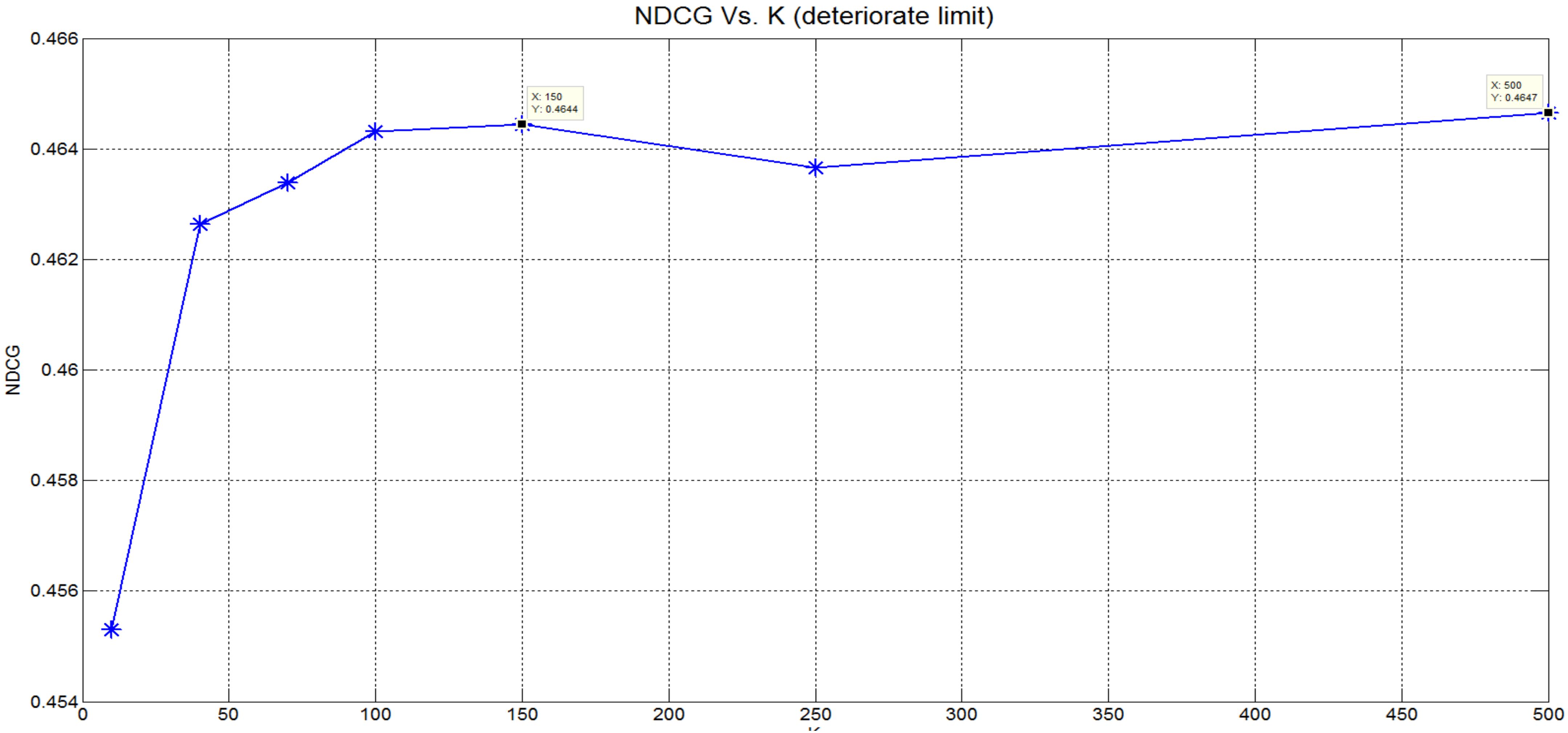
Choosing K



Control Robotics and Machine Learning Laboratory

Algorithm 2 - SwitchRank

Deteriorate limit (K)



Final Results

Algorithm	NDCG at Kaggle	Details
Best Score	0.54075	Best score reached on Kaggle by competitors.
Position Benchmark	0.49748	Properties ranked according to the position they were shown on Expedia.com
SVM-Rank	0.47546	Algorithm 1
SwitchRank	0.46805	Algorithm 2
Basic Python Benchmark	0.40356	Some basic benchmark created for competitors use
Random Order Benchmark	0.34958	Properties are recommended in a random order

Future Development

- Combining different algorithms.
- Finding a better quantization method.
- Creating more features.
- Divide & Conquer with different features.

Appendix 1 – Evaluation Metric NDCG

- NDCG - Normalized Discounted Cumulative Gain

$$DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

- Where **K** is the maximum number of entities that can be recommended and rel_i is the graded relevance of entity i.

$$rel_i \in \{0,1,5\}$$

- $IDCG_k$ is the maximum possible (ideal) DCG_k for a given set.
- The final score is calculated by:

$$nDCG_k = \frac{DCG_k}{IDCG_k}$$