Efficient approach for the maximum clique problem based on machine learning

Alexey Nikolaev

Laboratory of Algorithms and Technologies for Networks Analysis,
National Research University Higher School of Economics,
Nizhny Novgorod, Russia

Outline

- **Maximum Clique Problem**
- **Algorithm Selection Problem**
- **Our approach**

Definitions

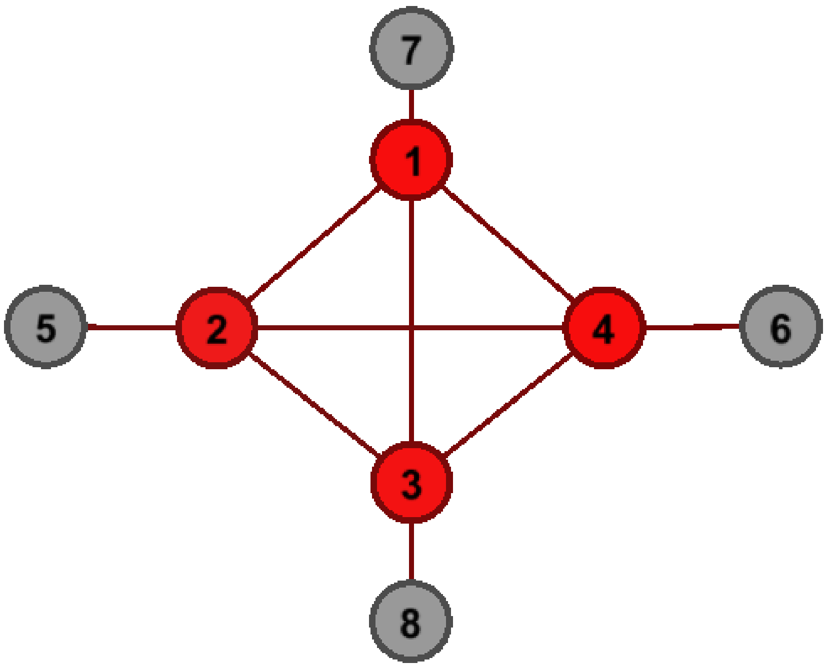
- $G=(V, E)$ is a **simple undirected graph** which consists of a finite set of vertices $V=\{v_1, v_2, \cdots, v_n\}$ and edges $E \subseteq V \times V$ that pair distinct vertices.

- A **clique** Q is a subset of V where all vertices are pairwise adjacent.

- A **maximum clique** is a clique of the maximum cardinality.

Maximum Clique Problem

- The maximum clique problem (MCP) is the problem of finding the maximum clique in a given graph *G*.



Exact algorithms

1957 – Harary and Ross

1973 – Bron and Kerbosch

1977 – Tarjan and Trojanowski

1990 – Carraghan and Pardalos

1986, 2001 – Robson

....

2010 – MaxCLQ (Li and Quan)

2010 – MCS (Tomita et al.)

2011 – BBMCI (Segundo et al.)

2013 – IncMaxCLQ (Li et al.)

2015 – BBMCX (Segundo et al.)

Modern review

Wu, Q., Hao, J.K.: A review on algorithms for maximum clique problems. European Journal of Operational Research 242, 693-709 (2015)

Motivation

Computational time (in milliseconds)

Instance	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4	Algorithm 5
C250.9	344516	1361335	1041168	987836	971229
brock400_1	259708	284586	244209	234715	235103
dsjc500.5	3532	1555	1412	1426	1487
Total	607756	1647476	1286789	1223977	1207819

Research purpose

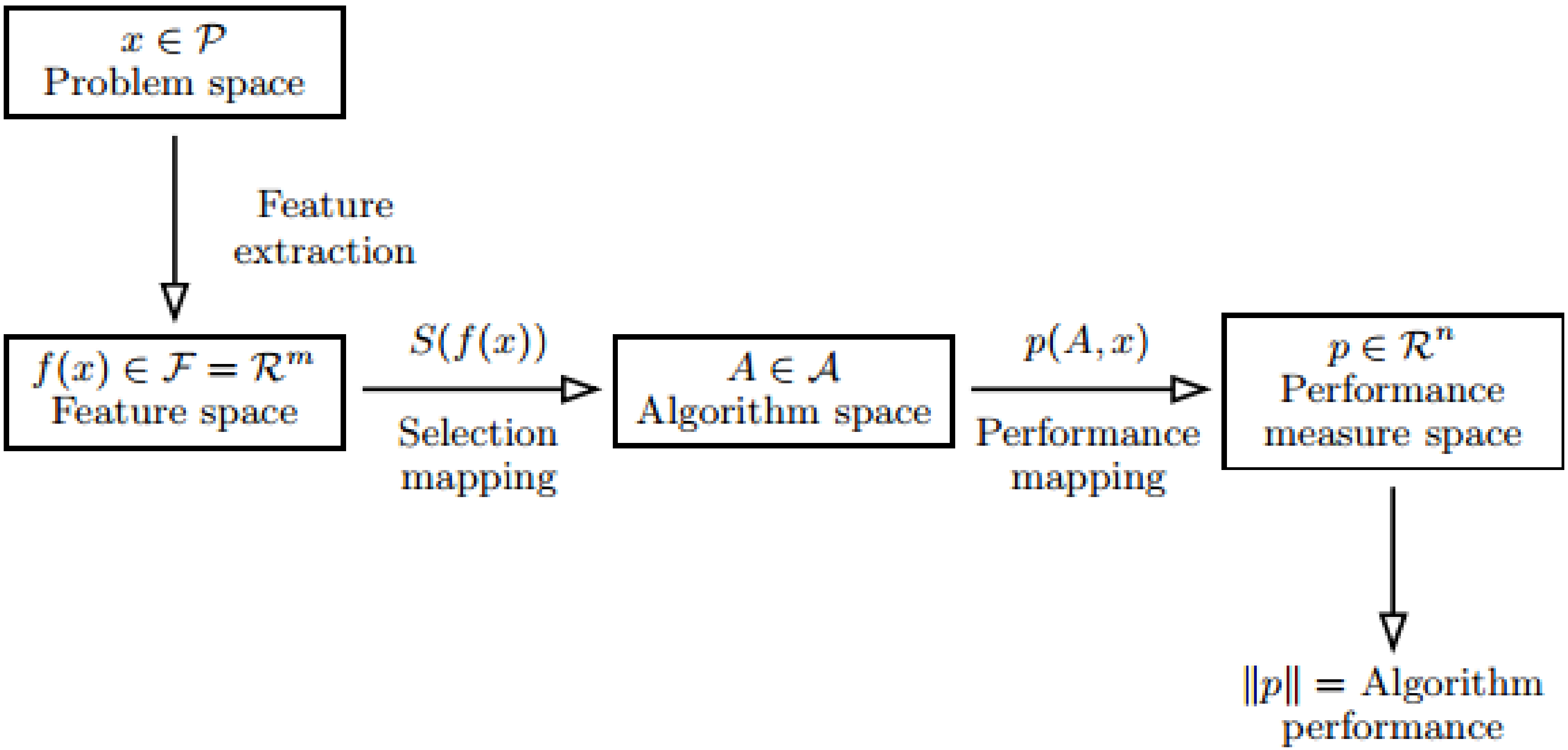
- **The purpose of the research is developing the algorithm that predicts the fastest algorithm from several algorithms for a given graph. Then the chosen algorithm is applied for solving the maximum clique problem in the graph.**

Algorithm Selection Problem

- **The algorithm selection problem consists of choosing the best algorithm from a predefined set to solve a problem instance.**

Algorithm Selection Problem

Model for the Algorithm Selection Problem with problem features (Rice*)



* Rice, J. R. (1976). The algorithm selection problem. Advances in Computers, 15, 65–118.

Algorithm Selectors*

- Case-based reasoning
  - k-NN
- Classification
  - SVM, decision tree, random forest
- Regression
  - linear regression, nonlinear regression

* Kotthoff, L., Gent, I., Miguel, I.: A Preliminary Evaluation of Machine Learning in Algorithm Selection for Search Problems. In Borrajo, D., Likhachev, M., Lopez, C., eds.: Procs. SoCS'11, AAAI Press (2011) 84–91

Our approach

Algorithm portfolio:

- **RPC (δ≥0)**

  Nikolaev A., Batsyn M., San Segundo P. *Reusing the same coloring in the child nodes of the search tree for the maximum clique problem*. Lecture Notes in Computer Science, 8994, 2015, 275-280

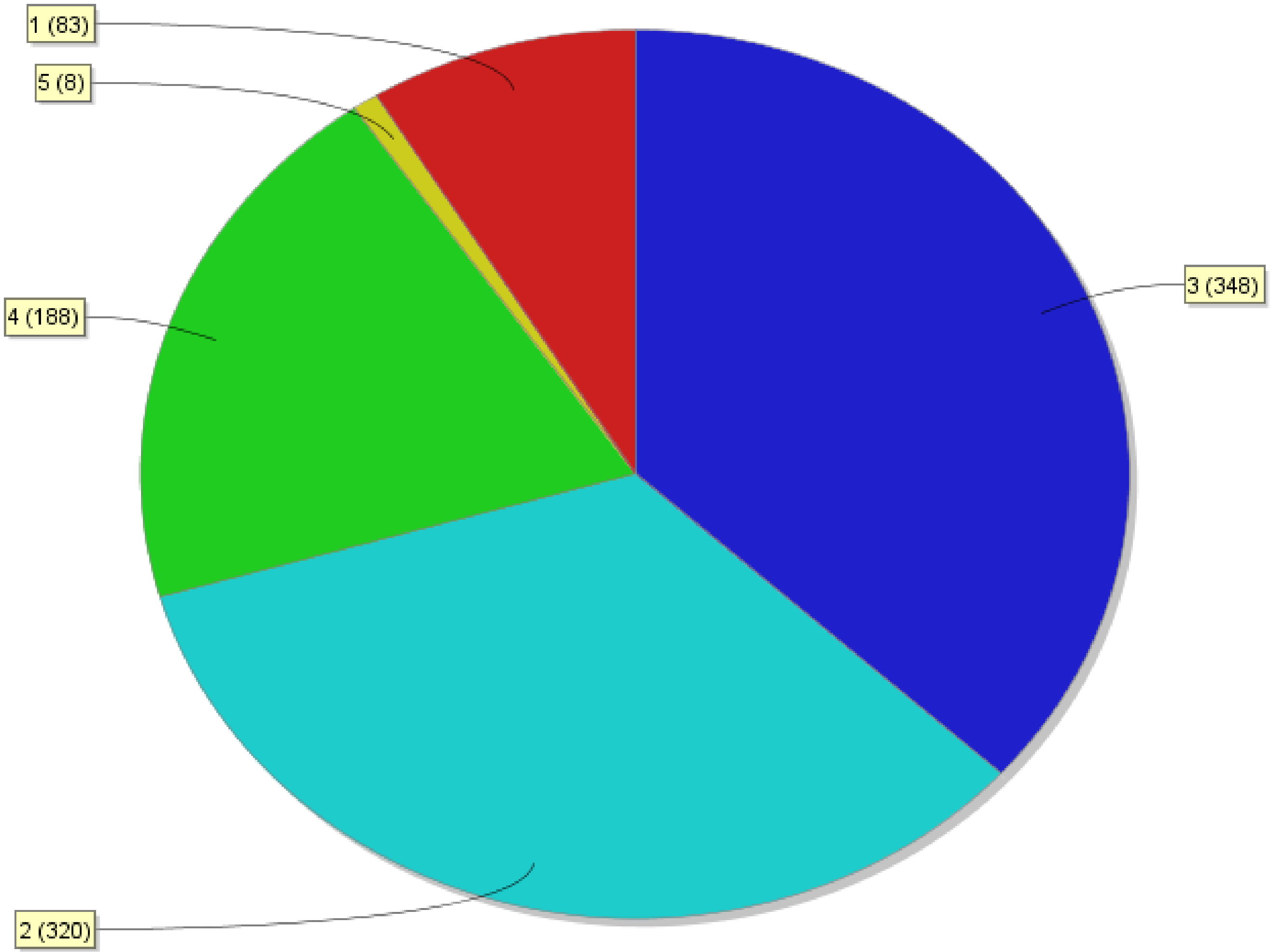
- **MaxCLQ**

  Li C.M., Quan Z. *Combining graph structure exploitation and propositional reasoning for the maximum clique problem*. Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence, Volume 01 (ICTAI'10), 2010, 344-351

Training set (947 instances)

n	***p***
150	**0.05-0.95**
200	**0.05-0.95**
300	**0.05-0.85**
400	**0.05-0.6**
500	**0.05-0.6**
1000	**0.05-0.4**
1500	**0.05-0.4**

Training set



Features

Graph size features:

1	**Number of vertices**
2	**Number of edges**
3	**Density**

Vertices degree statistics:

4	**Min**
5	**Max**
6	**Mean**
7	**Standard deviation**

Features

Sum of the degrees of the neighboring vertices statistics:

8	**Min**
9	**Max**
10	**Mean**
11	**Standard deviation**

Lower and Upper Bound:

12	**Lower bound (greedy heuristic)**
13	**Upper bound (greedy vertex coloring)**

Features

Search tree features:

14	The number of vertices to be considered on the first level of the search tree
15	Ratio of the number of vertices to be considered on the first level of the search tree (Feature 14) to the number of vertices (Feature 1)

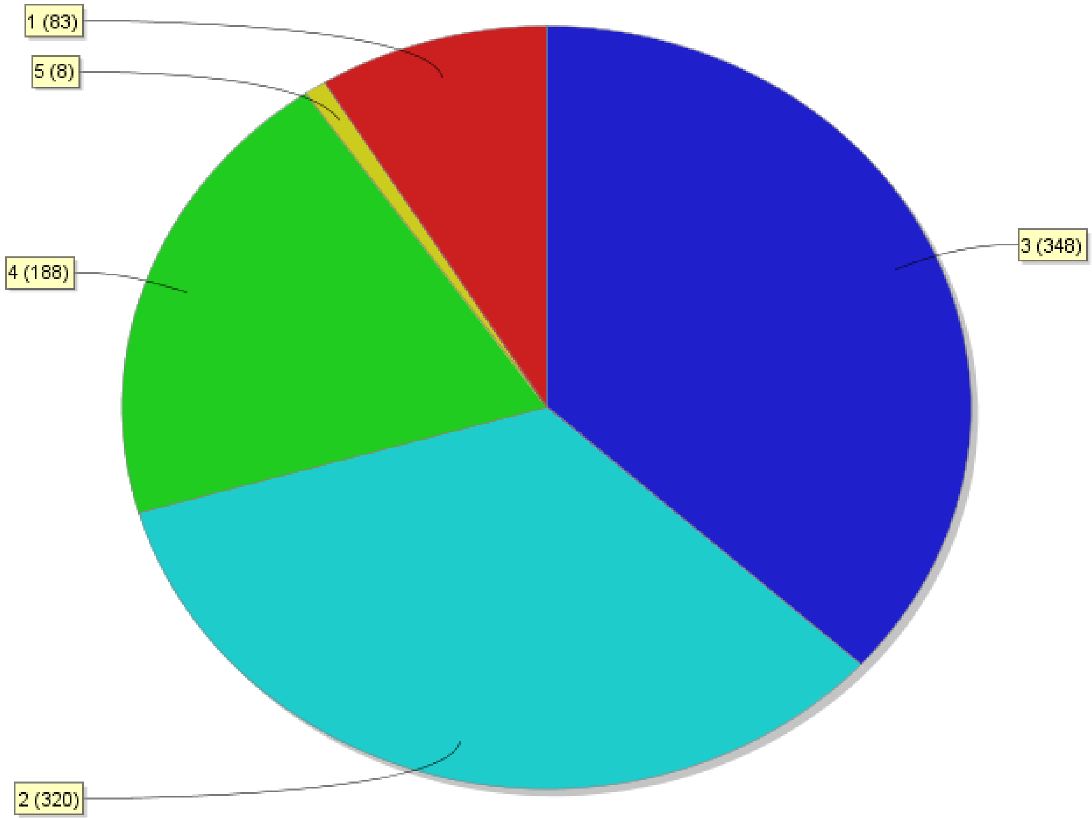
Decision trees

- **The strategy "one against all" was used to apply decision trees for multiclass classification.**
- **The strategy "one against all" is training *M* binary classifiers (where *M* is the number of classes). Classifier *i* separates class *i* from the other classes.**
- **It is believed that object *x* belongs to the class**

$$j^* = \arg\max_{j=\overline{1,M}} P(j \mid x)$$

where *P*(*i* | *x*) is a probability that object *x* belongs to class *i*.

Decision trees



Trained classifiers use only the following features for classification: 2, 3, 6, 8, 10, 12, 13, 14.

Computational results (DIMACS)

Confusion matrix

	true 1	true 3	true 4	true 5	true 2	class precision
pred. 1	**2**	1	0	0	0	66.67%
pred. 3	0	**1**	0	0	0	100.00%
pred. 4	0	0	**13**	8	0	61.90%
pred. 5	0	0	0	**0**	0	0.00%
pred. 2	0	0	0	0	**0**	0.00%
class recall	100.00%	50.00%	100.00%	0.00%	0.00%	

The total prediction accuracy is 64% (16 out of 25)

Computational results (DIMACS)

Instances	MaxCLQ	RPC, δ=0	RPC, δ=1	RPC, δ=2	RPC, δ=3	Perfect Algorithm Selector	Our approach
C250.9	344516	1361335	1041168	987836	971229	**344516**	344517
MANN_a45	34148	43230	31159	34574	85558	**31159**	34154
brock400_1	259708	284586	244209	234715	235103	**234715**	234716
brock400_2	118908	125398	107746	103844	103952	**103844**	103845
brock400_3	204222	193386	164518	158995	159121	**158995**	158996
brock400_4	130754	92893	79692	76856	77071	**76856**	76858
brock800_1	5606592	3914626	3430660	3294158	3300223	**3294158**	3294165
brock800_2	4889039	3395345	2971610	2845765	2842208	**2842208**	2845772
brock800_3	3222601	3461488	3020965	2899545	2898460	**2898460**	2899552
brock800_4	2438408	1602096	1405760	1341436	1333342	**1333342**	1341442
dsjc500.5	3532	1555	1412	1426	1487	**1412**	1415
dsjc1000.5	317877	140509	127070	123663	127732	**123663**	123675
frb30-15-1	655244	721617	440375	339713	289979	**289979**	339715
frb30-15-2	951654	533285	296789	208696	160926	**160926**	208698
frb30-15-3	580959	473354	278528	210579	176525	**176525**	210581

Computational results (DIMACS)

Instances	MaxCLQ	RPC, δ=0	RPC, δ=1	RPC, δ=2	RPC, δ=3	Perfect Algorithm Selector	Our approach
frb30-15-4	1155555	1327562	765939	574914	481628	**481628**	574915
frb30-15-5	873662	489109	266500	187442	147959	**147959**	187443
p_hat300-3	1387	1245	1051	1044	1089	**1044**	1045
p_hat500-3	49829	60698	51642	49310	49779	**49310**	49313
p_hat700-2	3586	2726	2319	2250	2371	**2250**	2255
p_hat700-3	1082242	1063763	894519	841329	854369	**841329**	841333
p_hat1000-2	117828	106392	87986	82428	83927	**82428**	82439
p_hat1500-1	11408	2257	1993	1916	1922	**1916**	1936
sanr200_0.9	5604	13855	10712	10107	10122	**5604**	5604
sanr400_0.7	97663	79299	68261	66440	68368	**66440**	66442
Total	23156926	19491609	15792583	14678981	14464450	**13750666**	14030826

Computational results (DIMACS)

The average time reduction of the proposed approach with respect to each of considered algorithm

MaxCLQ	RPC, δ=0	RPC, δ=1	RPC, δ=2	RPC, δ=3	Perfect Algorithm Selector	Our approach
35,74%	28,97%	11,83%	4,41%	3,21%	-4,97%	0,00%

Computational results (protein alignment graphs)

Confusion matrix

	true 5	true 4	true 3	true 2	true 1	class precision
pred. 5	382	11	0	0	0	97.20%
pred. 4	0	0	0	0	0	0.00%
pred. 3	0	0	0	0	0	0.00%
pred. 2	0	0	0	0	0	0.00%
pred. 1	0	0	0	0	0	0.00%
class recall	100.00%	0.00%	0.00%	0.00%	0.00%	

The total prediction accuracy is 97.2% (382 out of 393)

Computational results (protein alignment graphs)

The average time reduction of the proposed approach with respect to each of considered algorithm

MaxCLQ	RPC, δ=0	RPC, δ=1	RPC, δ=2	RPC, δ=3	Perfect Algorithm Selector	Our approach
96,08%	32,69%	13,18%	0,56%	-4,46%	-4,47%	0,00%

Conclusion

- **In this research decision trees were used to predict the fastest algorithm from a set of algorithms.**
- **For this purpose some features were proposed for the MCP.**
- **Computational results show that the considered approach is effective.**

Thank you for your attention!