Investigation of Kernel Functions in EDA-GK

Ryoischi Hasegawa
Kindai Univeristy
Higashi Osaka
hasegawa0158@gmail.com

Hisashi Handa
Kindai Univeristy
Higashi Osaka
handa@info.kindai.ac.jp

ABSTRACT

We have proposed EDA-GK, Estimation of Distribution Algorithms with Graph Kernels. The EDA-GK is designed for solving graph-related problems, where individuals can be represented by graphs. By using graph kernels, the EDA-GK can be solved for graph-related problems well. The EDA-GK uses the graph kernels as probabilistic models in EDA. In this study, we examine the Weisfeiler-Lehman Kernel, and the mixture of two kernels. Experimental results on Graph Isomorphism problems showed the effectiveness of the proposed method:

CCS CONCEPTS

• **Theory of Computation**; • **Design and analysis of algorithms**; • **Mathematical optimization**; • **Discrete optimization**; • **Network optimization**;

KEYWORDS

Estimation of Distribution Algorithms, graph kernel, graph-related problems

ACM Reference Format:

Ryoischi Hasegawa and Hisashi Handa. 2018. Investigation of Kernel Functions in EDA-GK. In *GECCO '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan.* ACM, New York, NY, USA, Article 4, 2 pages. https://doi.org/10.1145/3205651.3208785

1 INTRODUCTION

We have proposed EDA-GK, Estimation of Distribution Algorithms with Graph Kernels. The EDA-GK is a sort of Esimation of Distribution Algorithms [2, 4], designed for solving the graph-related problems such that individuals are represented by graphs. Such graph-related problems are so time-consuming: it needs much time even if we only much two graphs. By using Graph Kernels, however, we can address this time-consuming difficulty.

In the proposed method, we use graphs as genotype, and use Kernel Density Estimation for estimating distribution of graphs. We can search on feature space that are like phenotype by use Graph-Kernel, as a result, can solve difficulty of search because of the roughness landscape. This will be discussed in the next section.

In order to improve the performace of the EDA-GK, we examine the Weisfeiler-Lehman Kernel in this paper. Moreover the mixture of the shortest path graph kernel and the the Weisfeiler-Lehman Kernel is introduced.

2 GRAPH KERNELS

Kernel methods have attracted much attention in the field of machine learning. In recent years, the notion of kernel functions is extended to graphs [3]. In this paper, we use the Shortest Path Graph Kernels and Weisfeiler-Lehman Kernels [5]. Moreover, a mixture of these kernels is introduced.

The procedure of the Shortest Path Graph Kernels [1] is carried out as follows: First, calculate shortest path distance between all of the nodes in two graphs. Next, compare the frequency of the shortest path distances of all the pairs in two graphs.

$$k_{sp}(G, G') = \sum_{v_i, v_j \in G} \sum_{v'_i, v'_j \in G'} k_l(d(v_i, v_j), d(v'_i, v'_j)),$$

where $d(v_i, v_j)$ is a shortest path distance between node v_i and v_j, and k_l is a function in order to compare two shortest path distances.

In the Weisfeiler-Lehman Graph Kernel, the proximity of two graphs is calculated over the similarity of the degrees of vertices in graphs.

(1) Labels corresponding to the degree of nodes are associated with each node in two graphs, where the degree of nodes means the number of connected edge with the node.
(2) In addition, update labels taking account the labels of neighbor nodes. We can say two graphs are of isomorphic if two graphs have the same label set as a result of the iteration of the re-labelling.

Note that this calculation is also an approximated one so that it rarely judges as the same graph, in spite of two graphs are different.

An extended kernel of combining two kernels in the previous subsections is introduced in this paper. By using a hyper-parameter $\alpha (0 \le \alpha \le 1)$, the mixture of two kernels k_{mix} is defined as follows:

$$k_{mix}(G, G') = \alpha k_{sp}(G, G') + (1 - \alpha) k_{wl}(G, G'),$$

where $k_{sp}(G, G')$ and $k_{wl}(G, G')$ are the shortest path graph kernel and the Weisfeiler-Lehman graph kernel, respectively.

"Extended" in the previous paragraph means this graph kernel k_{mix} is the equivalent to the shortest path graph kernel if $\alpha = 0$, and k_{mix} is the same as the Weisfeiler-Lehman Graph Kernel if $\alpha = 1$.

3 EDA-GK

This procedure is similar to conventional EDA. That is, it consists of generating initial population, evaluating population, selecting good individuals, estimating the probability distribution of

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07…$15.00
https://doi.org/10.1145/3205651.3208785

selected population, and generating offspring based on estimated probability distribution namely sampling. The difference between the EDA-GK and conventional EDA is using the graphs as individual representation. Because of such difference, the estimation and the sampling also differ.

4 EXPERIMENTS

In this paper, we apply the EDA-GK to Graph Isomorphism problems such that algorithm must induce a graph with the same topology of the target graph. The target graph is supposed to be given by users. In this paper, the fitness function f_{iso} for solving the Graph Isomorphism problems is defined as follows :

$$f_{iso}(G) = k_{sp} \times k_{wl},$$

where k_{sp} and k_{wl} are graph kernels mentioned in Section 2. The reason for using graph kernels is that the order of calculating the isomorphism of two graphs is NP-hard. Each graph kernel varies from 0 to 1 so that the fitness function f_{iso} also varies from 0 to 1. Since these graph kernels are approximated computation, these graph kernels are combined for the fitness calculation.

The parameters of experiments are described in this subsection: The number of generations is set to be 5000, the number of nodes of Target and individuals is 40, and population size is 100. Plus Strategy is used for survivals selection.

The target graphs are generated randomly: The density of edges are set to be 90%. For each edge density, five target graphs are randomly generated. For each generated target graph, 10 runs are examined. Hence, totally, 50 runs are carried out for each edge density.

Initial individuals are sampled over uniform distribution. That is, for all the possible pairs, an edge is set with probability 0.5. The number of iterations of the sampling operation in Section **??** is 3.

For comparison, we examine evolutionary algorithms with a mutation operation. The mutation operation is set and unset of the pair of nodes randomly chosen. Note that the only difference with the EDA-GK is the use of the graph kernels.

Experimental results are shown in Fig. 1. In this plot, the horizontal axis shows the parameter α in the mixture of two kernels. The vertical axis shows the fitness at the final generation. These plots are boxplot: Horizontal lines denote upper and lower whiskers, respectively. The upper and lower side of boxes is upper and lower quartiles, respectively. A horizontal line in each box mens the median fitness value over 50 runs. The blue dots indicate outliers if exist.

The edge density of the target graph is 90 %, the EDA-GK with the mixture of two kernels ($\alpha = 0.8$) outperform other algorithms.

The distance of shortest path in the case of higher edge densities tends to 1. Hence, the performance deterioration of the EDA-GK with the shortest path graph kernel could be observed.

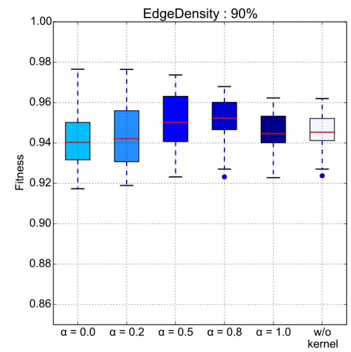

Figure 1: Experimental results on the Graph Isomorphism Problems: The densities of edges in target graph are 10% (Upper left), 30% (Upper right), 50% (middle left), 70% (middle right), and 90% (bottom); "w/o kernel" stands for evolutionary algorithms without graph kernels

5 CONCLUSIONS

In this paper, we investigated kernel function of EDA-GK. We newly examined the Weisfeiler-Lehman Graph Kernel, and proposed a mixture kernel of the Shortest Path graph Kernel and the Weisfeiler-Lehman Graph Kernel. The proposed methods are examined on the graph isomorphism problems with various edge densities of target graphs.

According to experimental results in Section 4, the proposed method performed well in Graph Isomorphism Problem of 40 nodes. In the case of that the edge density of the target graph is 10 %, the EDA-GK with the shortest path graph kernel outperformed others. For 90 % edge density, the EDA-GK with the mixture of two graph kernels outperformed other algorithms.

Future works are summarized as follows: We need to analyze the problem instances with middle edge densities. The graph kernels used in this paper could capture the features of randomized graphs. We may use another type of graph kernels which utilize partial small sub-graphs as features of kernel functions.

We need to examine other sorts of target graphs which have the typical structure of graphs, e.g. star, clusters, scale-free and so on. The must have other nature of randmized target graphs.

ACKNOWLEDGMENTS

This work was partially supported by the JSPS KAKENHI Grant Numbers JP23700267, JP26320291, JP17K00353.

REFERENCES

[1] Karsten M. Borgwardt and Hans-Peter Kriegel. 2005. Shortest-Path Kernels on Graphs. In *Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM '05)*. IEEE Computer Society, Washington, DC, USA, 74–81. https://doi.org/10.1109/ICDM.2005.132

[2] Hisashi Handa. 2009. EDA-RL: Estimation of Distribution Algorithms for Reinforcement Learning Problems. In *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation (GECCO '09)*. ACM, New York, NY, USA, 405–412. https://doi.org/10.1145/1569901.1569958

[3] H. KASHIMA. 2003. Marginalized kernels between labeled graphs. *Proceedings of the 20th International Conference on Machine Learning, San Francisco, CA, 2003* (2003). https://ci.nii.ac.jp/naid/10015408233/

[4] Pedro Larraanaga and Jose A. Lozano. 2001. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation.* Kluwer Academic Publishers, Norwell, MA, USA.

[5] Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn, and Karsten M. Borgwardt. 2011. Weisfeiler-Lehman Graph Kernels. *J. Mach. Learn. Res.* 12 (Nov. 2011), 2539–2561. http://dl.acm.org/citation.cfm?id=1953048.2078187