A Fast Approximation of the Weisfeiler-Lehman Graph Kernel for RDF Data

Advanced Algorithms and Graph Minings

Lorenzo Palloni Emilio Cecchini

Università Degli Studi di Firenze

lorenzo.palloni@stud.unifi.it

emilio.cecchini@stud.unifi.it

June 7, 2019

Introduction

- In order to apply machine learning algorithms on graphs it is necessary to develop algorithms to compute how similar two graphs are.
- Starting from the well-known Weisfeiler-Lehman isomorphism test, they have been developed kernel methods to measure the similarity between graphs.
- This paper proposes a fast approximation of a Weisfeiler-Lehman kernel applied to RDF data.

Graph Kernels

Kernel-based machine learning algorithms abandon the explicit vector representations of data items by means of the *kernel function*.

Definition (Graph Kernel)

Let $\mathbb{G}$ be a non-empty set of graphs. Any function $k : \mathbb{G} \times \mathbb{G} \to \mathbb{R}$ that takes as input two graphs G and G' and returns a real number that is equal to the scalar product between G and G' in a (even unknown) feature space is a valid kernel function.

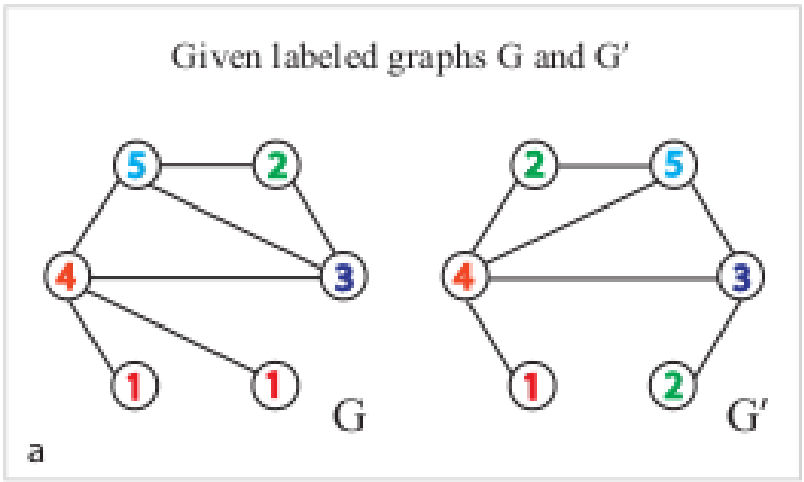
Graphs isomorphism

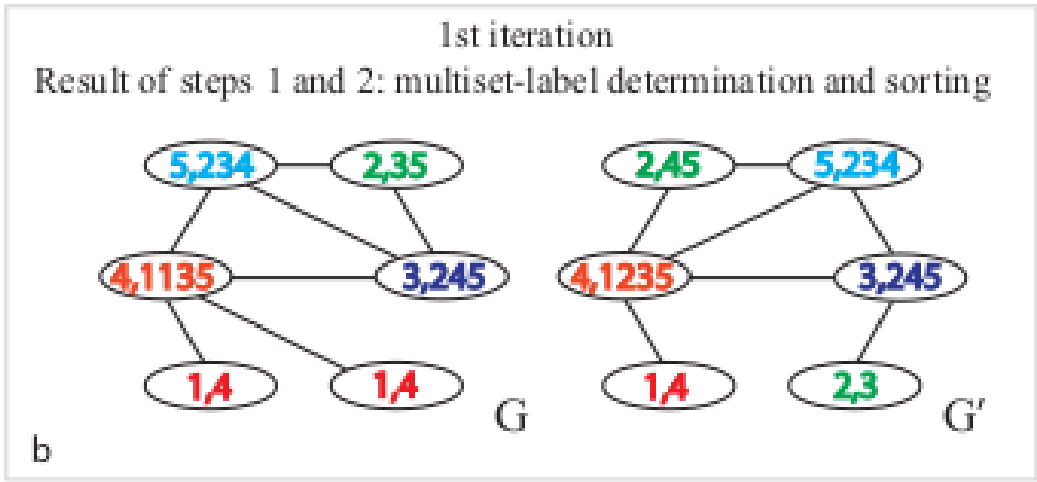
- Two graphs G and G' are isomorphic if exists a bijective mapping between the labels of G to the label of G'
- The graph isomorphism problem is NP.
- The graph kernel introduced in this paper uses concepts from the *Weisfeiler-Lehman test* of isomorphism.

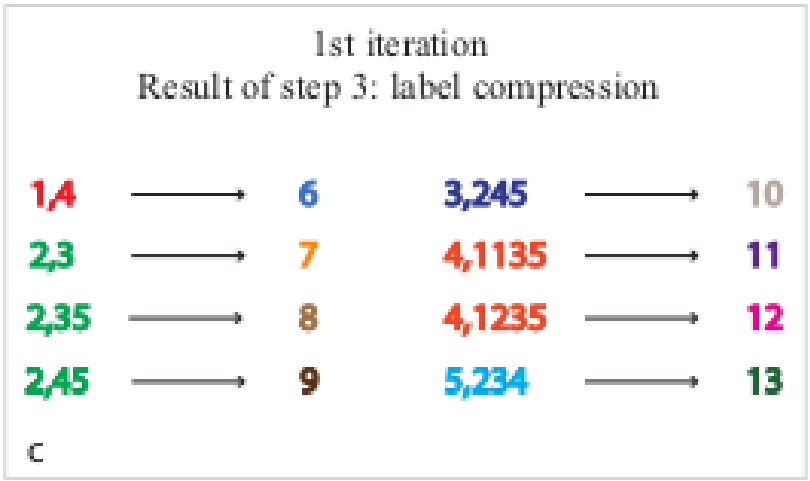
Weisfeiler-Lehman test

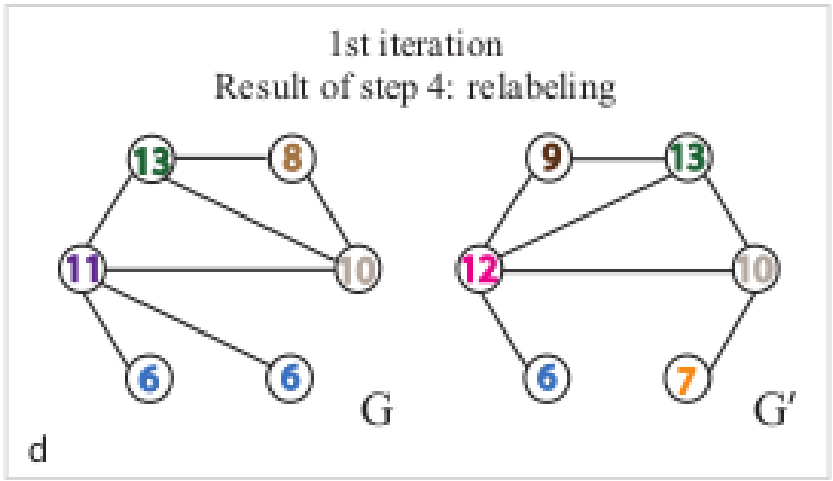
- Assume we are given two graphs G and G' and we would like to test whether they are isomorphic.
- The Weisfeiler-Lehman test performs h iterations.
- The key idea of the algorithm is to augment the node labels by the sorted set of node labels of neighbouring nodes, and compress these augmented labels into new, short labels.
- These steps are then repeated until the node label sets of G and G' differ, or the number of iterations reaches h.
- The runtime complexity of the Weisfeiler-Lehman algorithm with h iterations is $O(hk)$, where k is the number of labels in G and G'.

Weisfeiler-Lehman test









Weisfeiler-Lehman kernel

Definition (Weisfeiler-Lehman kernel)

Let $G_i = (V, E, l_i)$ and $G'_i = (V', E', l_i)$ be the i-th iteration rewriting of the graphs G and G' with the Weisfeiler-Lehman algorithm and h the number of iterations. Then the Weisfeiler-Lehman kernel is defined as:

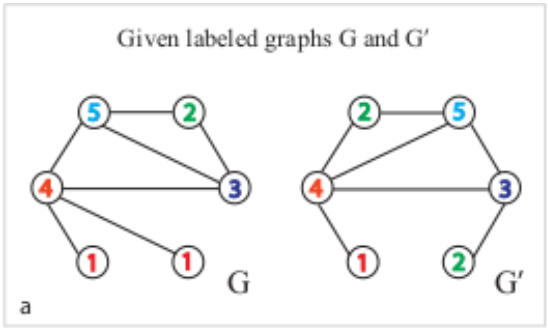
$$k^h_{\mathrm{WL}}(G, G') = \sum_{i=0}^{h} k_\delta(G_i, G'_i) \tag{1}$$

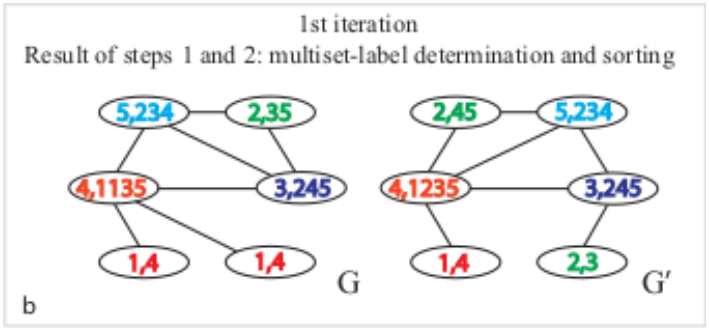
where

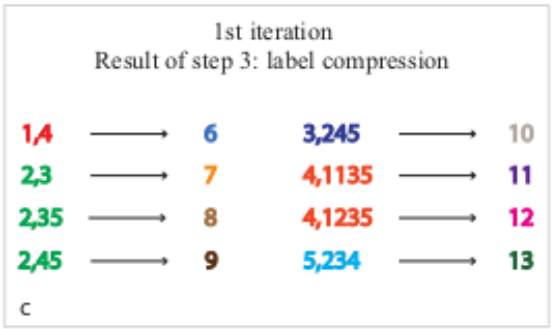
$$k_\delta((V, E, l), (V', E', l')) = \sum_{v \in V} \sum_{v' \in V'} \delta(l(v), l'(v')) \tag{2}$$

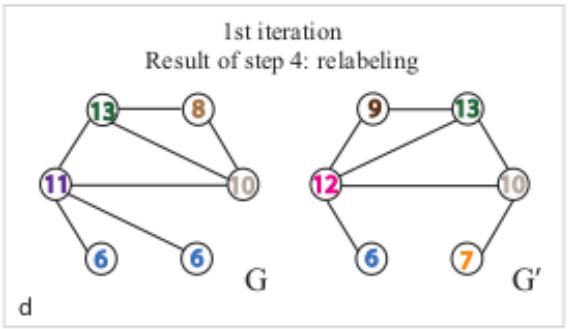
Here δ is the Dirac kernel, which tests for equality, it is 1 if its arguments are equal, and 0 otherwise.

Weisfeiler-Lehman Subtree Kernel









End of the 1st iteration
Feature vector representations of G and G′

$$\varphi^{(1)}_{WLsubtree}(G) = (2, 1, 1, 1, 1, 2, 0, 1, 0, 1, 1, 0, 1)$$

$$\varphi^{(1)}_{WLsubtree}(G') = (\underbrace{1, 2, 1, 1, 1}_{\text{Counts of original node labels}}, \underbrace{1, 1, 0, 1, 1, 0, 1, 1}_{\text{Counts of compressed node labels}})$$

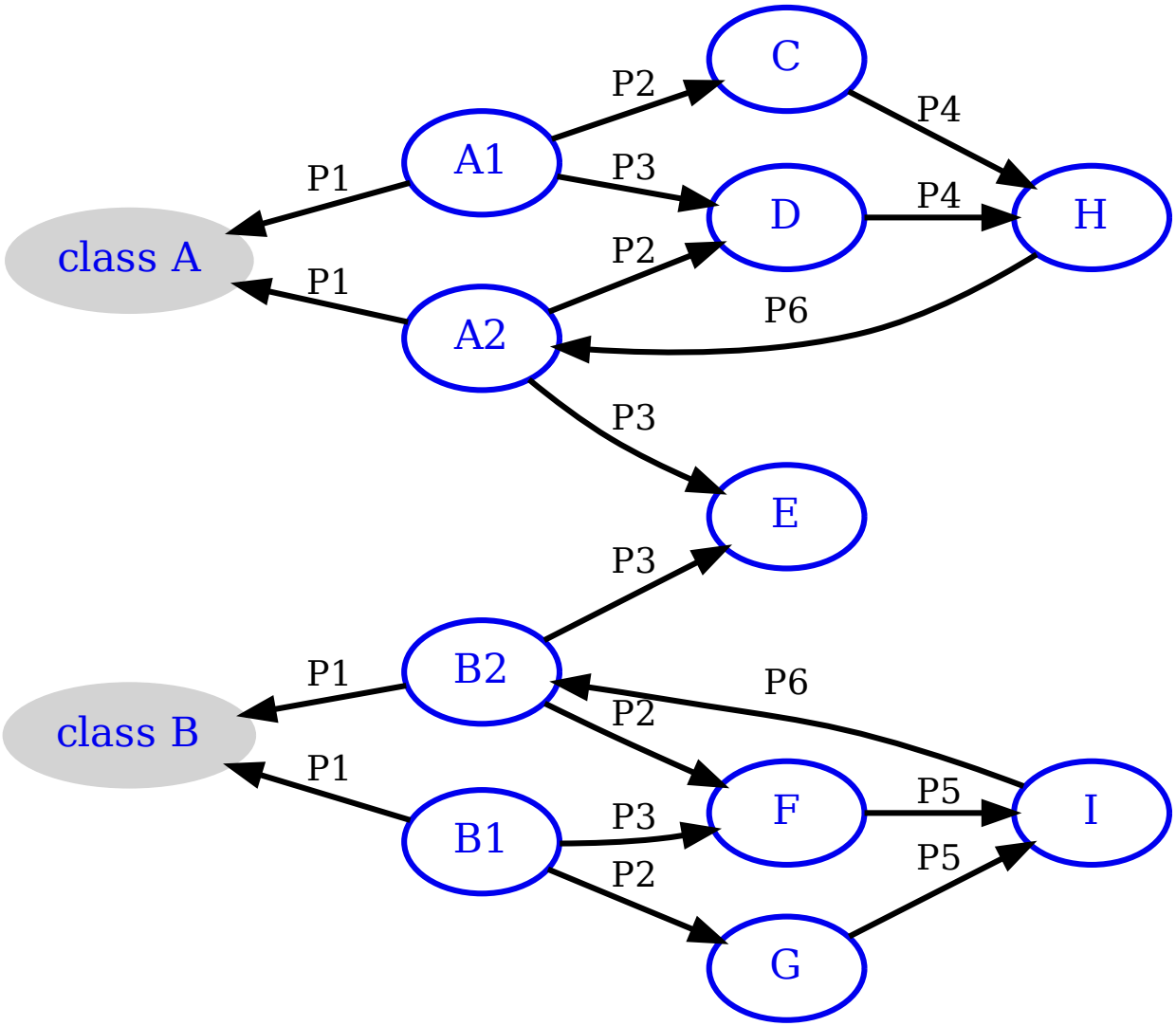
$$k^{(1)}_{WLsubtree}(G,G') = \langle \varphi^{(1)}_{WLsubtree}(G), \varphi^{(1)}_{WLsubtree}(G') \rangle = 11.$$

e

The Resource Description Framework

- The Resource Description Framework (RDF) is the foundation for knowledge representation on the semantic web.
- It is based on the idea of making statements about resources in a *subject-predicate-object* form, called *triples*
- A set of triples represents a graph, that has subjects and objects as nodes and predicates as edges (note that is a *directed multigraph with labeled edges*).

The Resource Description Framework



Fast Weisfeiler-Lehman for RDF

- The most immediate approach to apply graph kernels to RDF is to extract subgraphs for the instances that we are interested in and to compute the kernel on these subgraphs.
- Potentially it can be more efficient to do the kernel computation directly on the larger underlying RDF graph, instead of extracting many subgraphs.
- This paper proposes an approximation of the Weisfeiler-Lehman kernel designed for RDF data.

Weisfeiler-Lehman RDF graph

Definition (Weisfeiler-Lehman RDF graph)

A Weisfeiler-Lehman RDF graph is a graph $G = (V, E, l)$, where V is a set of vertices, E a set of directed edges, and $l : (V \cup E) \times N \rightarrow \Sigma$ a labeling function from vertices V or edges E and a depth index $j \in \mathbb{N}$ to a set of labels Σ.

Definition (Neighborhood)

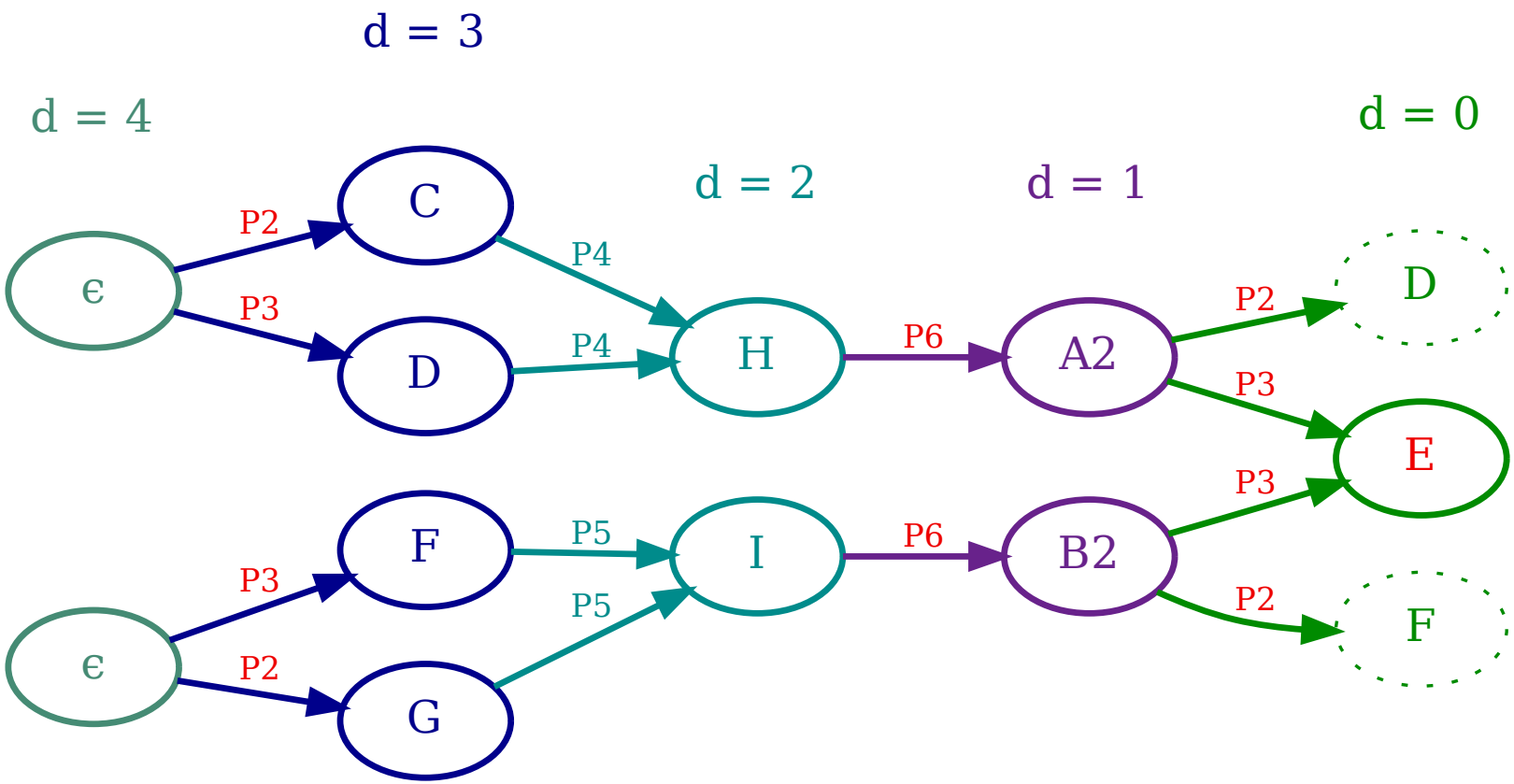
The neighborhood $N(v) = \{(v', v) \in E\}$ of a vertex is the set of edges going to the vertex v and the neighborhood $N((v, v')) = v$ of an edge is the vertex that the edge comes from.

Graph extraction from RDF

- Given a set of RDF triples and a set of instances I, there is an algorithm to build a Weisfeiler-Lehman RDF graph.
- For each instance i a subgraph up to depth d is extracted from the RDF dataset and this subgraph is added to the total graph G that the algorithm is building. Thus, vertices and edges are only added if they have not been added to the graph already.
- Next to the graph G we also construct mappings $\mathcal{V}_i$ and $\mathcal{E}_i$ for each instance i, which records which vertices and edges belong to the subgraph of instance i at which depth.

Graph extraction from RDF

Extraction of the instances A1 and B1.

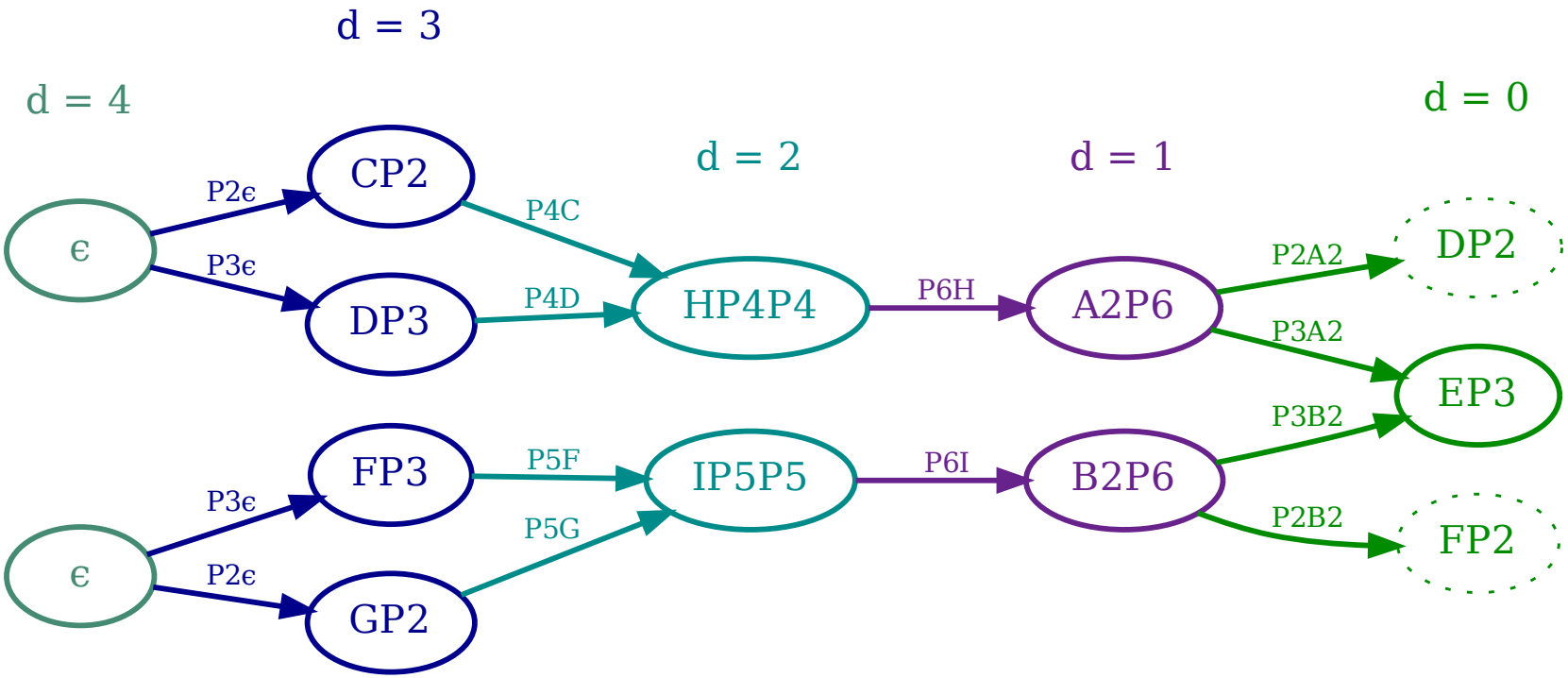


Relabeling of the Weisfeiler-Lehman RDF graph

- The relabeling process is quite similar to the standard one.
- It is extended to directed and labeled edges.
- The augmented labels are constructed taking into account the new definition of neighborhood and the depths.

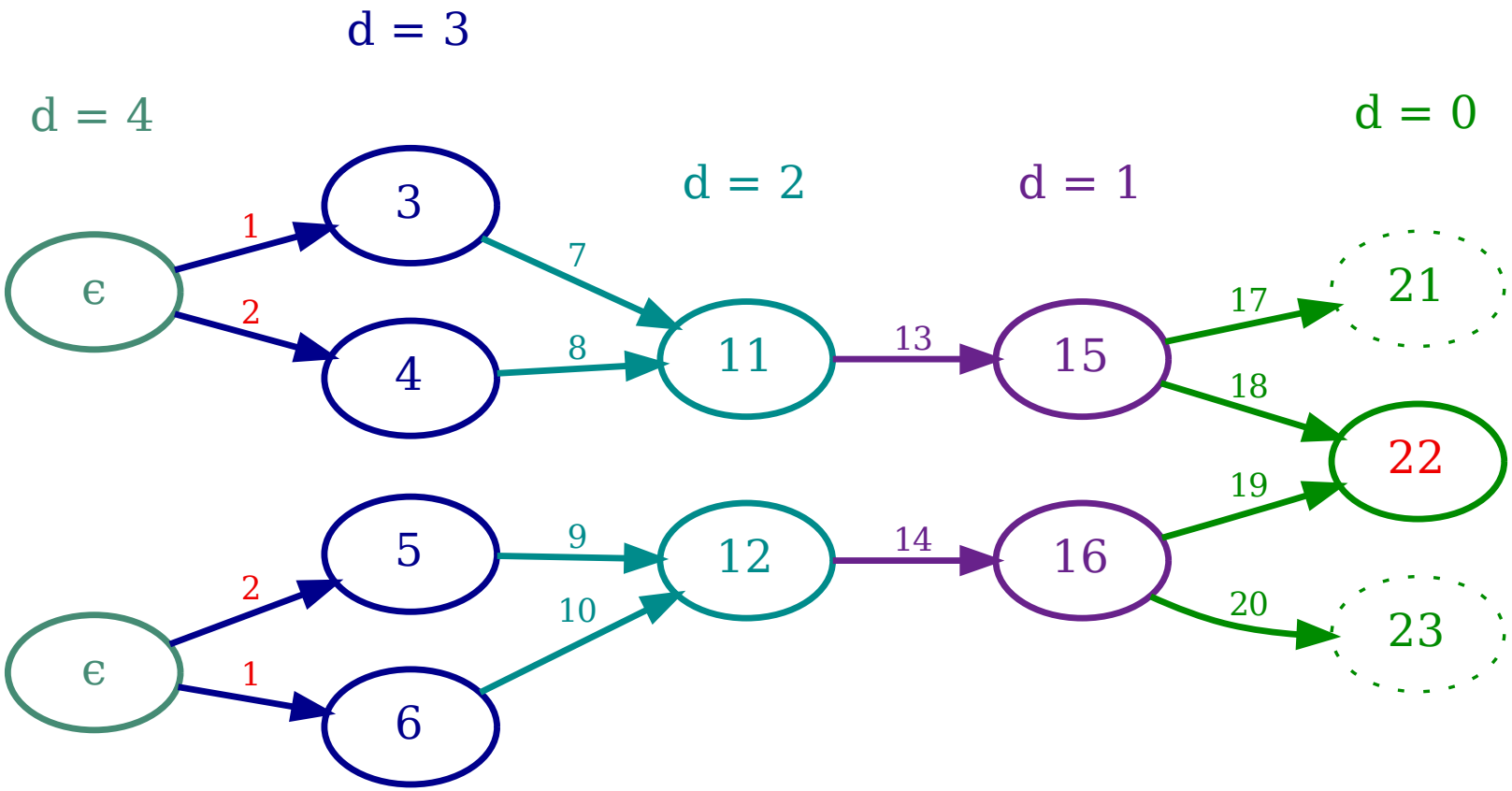
Relabeling of the Weisfeiler-Lehman RDF graph

Label propagation.



Relabeling of the Weisfeiler-Lehman RDF graph

Relabeling.



Weisfeiler-Lehman kernel for RDF

Definition (Weisfeiler-Lehman kernel for RDF)

Let G be a Weisfeiler-Lehman RDF graph and rewritten for h iterations, and l_0 to l_h the resulting label functions. Then we compute a kernel between two instances i, $i' \in I$, as:

$$k^h_{\mathrm{WLRDF}}(i, i') = \sum_{n=0}^{h} \frac{n+1}{h+1} k^n_{\delta,\mathrm{RDF}}\left((\mathcal{V}_i, \mathcal{E}_i), (\mathcal{V}_{i'}, \mathcal{E}_{i'})\right) \tag{3}$$

where

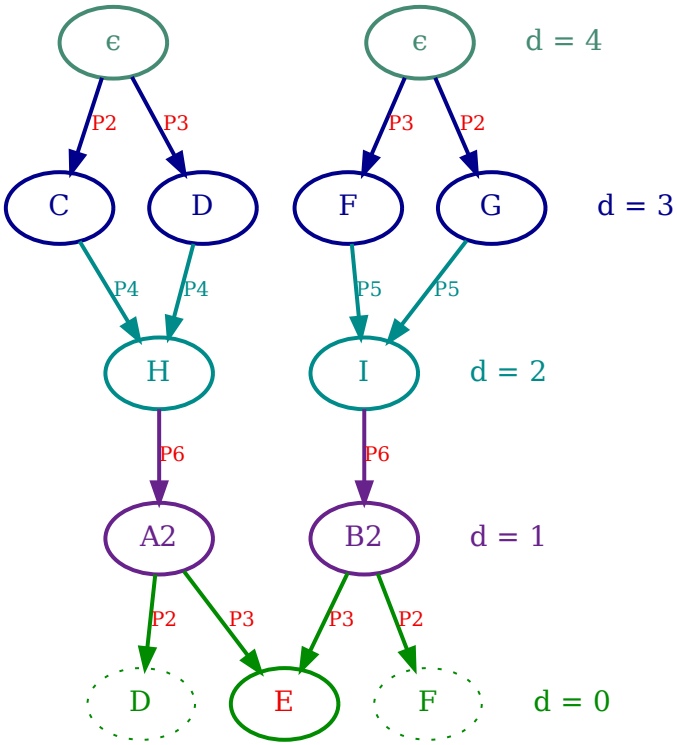
$$k^n_{\delta,\mathrm{RDF}}\left((\mathcal{V}_i, \mathcal{E}_i), (\mathcal{V}_{i'}, \mathcal{E}_{i'})\right) = \sum_{(v,d)\in\mathcal{V}_i (v',d')\in\mathcal{V}_{i'}} \delta\left(l_n(v,d), l_n(v',d')\right) \tag{4}$$

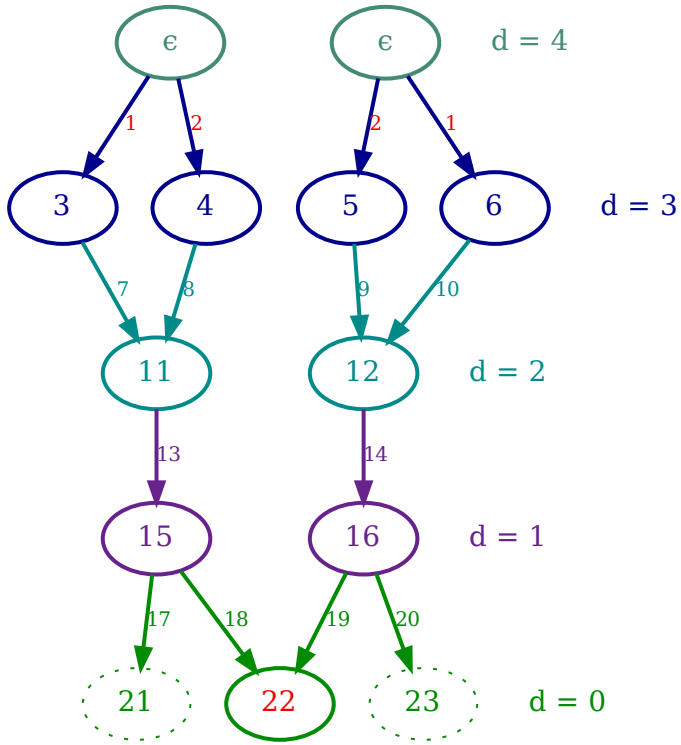
$$+ \sum_{(e,d)\in\mathcal{E}_i (c',d')\in\mathcal{V}_{i'}} \delta\left(l_n(e,d), l_n(e',d')\right) \tag{5}$$

Weisfeiler-Lehman kernel for RDF

Example

$$k^h_{\mathrm{WLRDF}}(A1, B1) = \sum_{n=0}^{h} \frac{n+1}{h+1} k^n_{\delta,\mathrm{RDF}}\left(\left(\mathcal{V}_{A1}, \mathcal{E}_{A1}\right), \left(\mathcal{V}_{B1}, \mathcal{E}_{B1}\right)\right) = \frac{1}{2} \cdot 10 + \frac{2}{2} \cdot 3 = 8$$





Complexity

- The complexity of the standard relabeling algorithm on a set of graphs is $O(Nh(n+m))$, where N is the number of graphs, h is the number of iterations and n and m are the number of vertices and edges per graph.
- This new relabeling method does not have N graphs, but it introduces d labels per vertex/edge, where d is the extraction depth.
- If the WL RDF graph has k nodes and edges the complexity of this new algorithm is $O(dhk)$
- The new proposed method is faster than the regular one if $hk < N(n+m)$

References


Vries Gerben Klaas Dirk, A Fast Approximation of the Weisfeiler-Lehman Graph Kernel for RDF Data, 2013


Shervashidze, N., Schweitzer, P., van Leeuwen, E.J., Mehlhorn, K., Borgwardt, K.M. Weisfeiler-lehman graph kernels, 2011