Semester- Thesis: Graph Embeddings with Differential Operators for Clustering and Classification of Brain Networks

Jakob Jakob

Description

When solved on graphs, the behavior of differential operators like the heat kernel will typically depend on the underlying topology of the graph. Thus , the graph characteristics will also be reflected in properties like the total heat flow, which can then serve as a representation of the graph. Being of much lower dimension than the original network, this embedding may be used beneficially in tasks like clustering or classification of a set of graphs.

Work packages

Note that sequence of work packages (WP) suggests a structure for the bachelor thesis but the work may require an iterative approach to solve the research problems. The dates indicate when noticeable first progress is expected to be reached for the respective WPs.

	Work Packages	Load	Date
1)	Study of related work	3-4x8h	14.March
2)	Implementation and testing of embedding on synthetic data	5-7x8h	4.April
3)	Implementation of brain network extraction from global tractography	5-7x8h	25.April
4)	Embedding of brain networks, clustering/classification	5-7x8h	16.May
5)	Refinement of embedding	3-4x8h	30.May
6)	Write up	3-4x8h	13.June

Grading

The requirements of each grade include all the requirements from lower grades.

>5.0

Extend approach by studying & implementing other differential operators (e.g. higher order time derivatives, divergence on graphs, etc) and investigate characterisation properties of different operators. Make operator models more flexible for learning e.g. with adaptive weights, etc.

5.0

Extraction of brain networks from global tractography results, i.e. create connectivity matrix between patches of the cortical surface. Apply graph embedding and perform clustering/classification on brain image data set.

4.0

Implementation of graph embedding with differential operators from the literature. Demonstration and testing of clustering/classification capacity on different types of synthetic random graphs.