Learning An Embedding Model for Spatial Information

Why

Machine learning on knowledge graphs is becoming a popular area of research which studies how statistical pattern recognition approaches, such as (deep) neural networks, can be applied to complex graphs with rich semantics. Whereas past research in this field largely focussed on the connectivity between the nodes, more recent work is also looking at how we can include the attributes belonging to these nodes in the learning process. One example of this concerns spatial information, such as coordinates, boundaries, floor plans, and 3D point clouds. To include this information in the learning process, the related attributes have to be vectorized and fed to the machine learning model. However, to vectorize these attributes, it is common practice to just use whatever strategy seems to work best, which leads to vastly different methods across the field.

What

The goal of this project is to develop, test, and evaluate an embedding model for spatial information. This model should follow the same principles as the well-known *word2vec*[1]embedding model, which means that you will need to train an autoencoder that returns a sensible embedding vector for each generated geometry.

How

This project will follow the follow steps:

- Develop a method to generate synthetic geometries (i.e. polygons)
- Develop a machine learning pipeline that takes geometries as input and returns embedding vectors
- Train the model on large amounts of synthetic geometries and use this to generate pretrained embeddings
- Evaluate your approach by testing the embeddings in a machine learning task

[1] See `https://israelg99.github.io/2017-03-23-Word2Vec-Explained/`

Who

This project aims at students who are interested in machine learning and representation learning. A basic familiarity with programming (Python) and machine learning and/or data mining is required.

Supervisors

Xander Wilcke & TBD