The problem definition

Connection in knowledge graphs are different from social network graphs, since connections in knowledge graphs usually have a direction. Thus, traditional link prediction methods from social networks cannot be used.

In the paper, they embed entities and relations of the zhishi.me knowledge graph into a low dimensional vector space. The vector representation of the entities and relations will contain semantic relationships among them.

Goal

The training part aims to learn the semantic relationships among the entities and relations with the negative entities, and the goal of the prediction part is to give a triplet score with the vector representations of entities and relations.

Core Architecture of knowledge Graph Embedding

For a given triplet (h, r, t) in the training set, the model will learn the vector representations of h and t as well as the r, denoted as $\mathbf{h}$, $\mathbf{t}$ and $\mathbf{r}$.

The core idea

Their core idea is to transform the link prediction problem into a question and answer mode, i.e. $\mathbf{h} + \mathbf{r}$ expresses the question, and $\mathbf{t}$ is the answer, or $\mathbf{t} - \mathbf{r}$ is the question, and $\mathbf{h}$ expresses the answer.

- They use the cosine similarity to judge the matching degree of question and answer.
- During the training process, at every epoch, they randomly sample a wrong entity which is from the whole entity set to each correct triplet in the training set.
- As a result, the four tuple $(h, r, t^+, t^-)(or(h^-, h^+, r, t))$ forms a training sample.

- The vector representation after the pooling layer is treated as the final embedding of the entity or relation which will be used in the loss function.

Parameter settings

They state in section 3.1 that the embedding dimension of entities and relations is set to 100, which I think is quite high.