Name: Peter Mortimer
GitHub: https://github.com/tonyromarock/city2vec

City2Vec

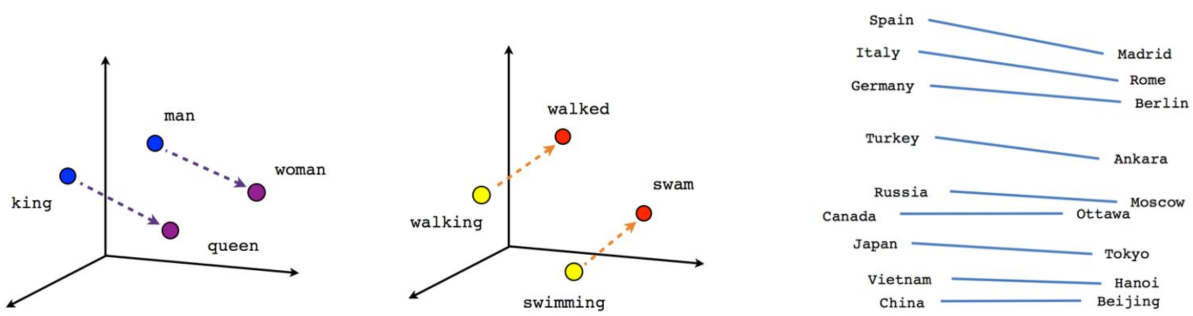
Analyzing the Analogical Properties of Cities in Word Vector Representations

Introduction

Throughout the course *From Complexity to Intelligence* we were introduced to different contexts were the measure of Kolmogorov Complexity could be applied to different domains of Artificial Intelligence and Computer Science in general.

In my project, I focused on Analogical Reasoning. I was interested in this topic, since there has not been one unified approach to create a learning algorithm to solve problems based on Analogical Reasoning. In particular, I will focus on the approach of using large text corpora in Natural Language Processing to train representations that encode some the analogical patterns commonly used in human texts.

The first time I encountered word vectors and their capabilities to encode analogical information was in “Distributed Representations of Words and Phrases and their Compositionality”[1] from Mikolov et al. This paper introduced the word2vec word embedding. In the paper they mention an interesting property about their word vectors called “Additive Compositionality”. As shown in Fig.1, the word vectors were distributed in the embedded space in such a way, that through simple vector arithmetic you could describe certain analogical relationships that commonly appear in human text.



Male-Female Verb tense Country-Capital

Figure 1: These are sketches of how the word vectors in word2vec are placed in higher dimensional space. Using the first image as an example: you could use word vectors to solve the analogical equation **man : woman :: king :: ?** by doing the word vector calculation: **vec(king) + vec(woman) - vec(man) = ?**.

Approach
I chose the domain of city names in Germany. With my project I wanted to see if an interesting relationship between the word vectors of the cities and the cities' geographical properties can be found and visualized. I decided to use the GloVe[2] word embeddings. The GloVe vectors are also known to have analogical properties encoded in the distribution of the word vectors.
I decided to use Multidimensional Scaling[3] (short MDS) as a technique to reduce my city data into a two-dimensional space. Multidimensional Scaling is a reasonable approach here, since it tries to preserve the distance between the points in the dataset, which is important when trying to find patterns from map plotting multiple cities.

Results
In Fig. 2, you can see the two plots I constructed based on the 10 largest cities in Germany. The first plot was constructed using the cosine distance between each word vector of a city and reducing this to a 2D plane using MDS. The second plot is based on the geographical distance between the 10 largest cities in Germany. The distance is measured using the great circle distance[4].
We can observe that in the word embedding cities with larger functions in Germany like Berlin, Frankfurt, Munich, and Hamburg are "semantically" closer than their actual geographical distance on the second plot would suggest.
Due to time constraints, I could not continue my analysis.

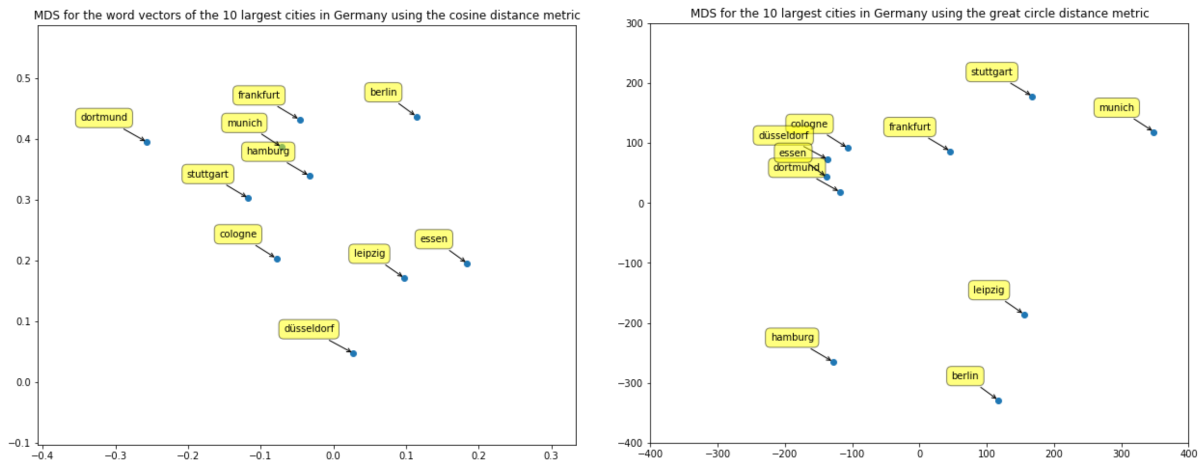


Figure 2: Comparing the MDS embedding of the cosine distance of the word vectors of cities to the MDS embedding of their geographical distance.

Web Links
[1] Mikolov et al. Distributed Representations of Words and Phrases and their Compositionality.
[2] GloVe: Global Vectors for Word Representation
[3] Wikipedia article on "Multidimensional Scaling"
[4] Wikipedia article on "Great Circle Distance"
Figure Links
Fig.1: TensorFlow Tutorial on "Vector Representations of Words"
Fig.2: Plots were generated using this Jupyter Notebook.