# WikiCategorizer

## Group members:


- Thomas Ørkild - s154433@student.dtu.dk
- Christian Ingwersen - s154264@student.dtu.dk


## Problem description

Given a text, we want to use K-nearest neighbours to find similar wikipedia pages. To compute the KNN we will use the text2vec-algorithm to convert wikipedia articles from text to vectors. Text2vec produces a vector in a high-dimensional space, where similiar texts are located close to each other in this space. We will not implement text2vec ourselves, but rather use this python implementation.

To efficiently compute KNN on a big dataset like Wikipedia, we will implement it using MapReduce as described in the paper: The k-Nearest Neighbor Algorithm Using MapReduce Paradigm.

We will create a command-line tool that, given a text, utilizes this technique to find to k most similiar Wikipedia articles, and use majority voting on the categories of the k articles, to find which category it should belong to.

To easily organize the huge Wikipedia dataset we store it in an SQL-database, that we query from.

## Time plan

The weeks refer to the semester weeks.

- **Before official project start**
    - Before the project begins our plan is to download and investigate all of the data, so we are comfortable with the data before we have to work with it. In addition, we'll familiarize us with the library text2vec and start generating text vectors for the wiki pages.
- **Week 10**
    - Read and understand the paper *The k-Nearest Neighbor Algorithm Using MapReduce Paradigm*
    - Start implementing the KNN algorithm using MapReduce as described in the paper.
- **Week 11**
    - Improve the efficiency of the first basic implementation.
    - Finish and test the implementation.
- **Week 12**
    - Make a command-line tool utilizing the algorithm.
    - Document results and start writing the report.
- **Week 13**
    - Finalize project presentation and report.