|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Activity** | **Bernhard** | **Shinho** |
| 1 | Research | Reading the original paper, understanding the algorithm and looking for some first built-in Spark utilities that could be useful | |
| 2 | - Prototyping  - Data collection for evaluation  - Prepare Intermediate presentation | Searching for the test data used in the paper and transforming the files to .nt to ensure Sansa compatibility. Furthermore several modifications of the data are required (e.g. modifying literals that are longer than one line).  Preparing the first presentation. | Setting up our GitHub repository.  Implementing a first version of the algorithm based on the existing java code. However the original algorithm is using some unique indexing and several nested operations that are not possible on RDD. |
| 3 | Implementation | Researching alternatives to the nested operations in form of completely different approaches using the built-in Spark utilities. Also trying to think of a different data representation that could allow us to use different techniques. | |
| 4 |
| 5 | Implementing a conversion from tree per instance to feature vector using Spark ML. | Implementing a way to store maps and indexes in Scala that allows us to generate the trees per instance in a single dataframe. |
| 6 | Test and evaluation  Optimization | First tests of our code and some adaptations. Trying to optimize the computation by avoiding non-Spark functions as much as possible. This results in a completely new simplified algorithm. | |
| 7 | Optimization  Documentation |  |  |