**What have we tried:**

1. Presented the summary of “Recovering Traceability Links between Code and Documentation”

* Learnings: Understood the problem for the project and identified the probably methodology to solve it.

1. Tried Implementing the Unibas coherence code base for Coffee Maker dataset :

* Problems: Understanding the initial code from Github took time.

1. Tried to run ML pipeline2 of coherence code on coffeemaker using jupyter.

* Problems: code integration issues and dependency issues. Understanding the output of pipeline took considerable time.

   4) Implemented cosine similarity for a pair of <String code , String comments>. Ran anova to check variance.

* Learnings: Understood how cosine similarity works for huge dataset.
* Cosine similarity for two groups are not different (proved by ANOVA)
* - we need something more definitive.

   5) Brainstorm on features to be implemented and Machine learning algorithm to be used.

List : Uncamel case, Java generic keywords ,Wordnet similarity, AST, Synonym detection , Generic comments detection.

* Problems: Identified a lot of corner cases where a particular feature may or may not be useful. These cases are discussed and implemented accordingly.

   6)  Implement the features and integrate the code for complete data set. (Benchmark.txt : 2881 data points)

* Learning: Understood the pipeline of features to be implemented.

   7) Use the word2vec model for calculating similarity. Use gensim word2vec as well as google news pretrained model.

* Learnings: Learnt gensim and word2vec for implementation. Also learnt about neural networks for identifying coherence.

  8 ) Implement k means unsupervised method for clustering (K =  2).

* Problems: Tried various thresholds and initial cluster mean values.

  9 ) Implement PCA on data and provide dimensionality reduced data to K-means as input. Plot graph representing the clusters.

* Learnings: Learnt about PCA and how dimensionality works. Understood the importance of plotting graphs and visualizations

 10 ) Annotate the coffeemaker dataset (47 data points) and manually check the accuracy with result from k means. (Accuracy :  73%)

* Learnings: Understood the clustering of example data points to see if it this is scalable.

**Plan for the future:**

* Explore more features -> see what really affects the usefulness of a comment - maybe through correlation.
* Manually annotate real apache projects.
* Create a background knowledge through custom w2v model. (turn java into human readable text + comment -> and train either glove or w2v model) - fine-tuned the other team’s code base