This video discusses the features of our project Image Understanding of GUI widgets for test reuse. The previous work on semantic matching for GUI event test transfer discusses semantic annotation based only on text. However, including the semantic annotations of GUI widgets and icons could probably boost the performance of certain techniques. The main objective of this work is to integrate semantic annotations of GUI widgets into the existing system for judging the variation in performance caused by it.

For the implementation of our project, we used the Rico dataset. Rico is A Mobile App Dataset for Building Data-Driven Design Applications. The Rico dataset[1] has a wide array of data samples of UI screenshots and related information such as trace of events, UI layout vectors, semantic annotations and descriptions of hierarchical relationships. The presence of these features in the dataset is essential for the task as information regarding different UI elements need to be collected step-wise as per the trace of the event. Also the semantic annotation of icons and widgets allows for easy integration of semantics of images. The relevant data was extracted from the dataset. Essential data (event\_index, label, type) as well as optional data (text, id, content\_desc, hint, parent\_text, sibling\_text, activity,atm\_neighbor, file\_name) were all taken from the dataset via mining of the dataset. The extraction of relevant data from the dataset is a critical step in data mining and is essential for obtaining meaningful insights and patterns from the data. By selecting essential data such as event\_index, label, and type, the extracted data can be used for a variety of purposes, such as training machine learning models, detecting anomalies, and understanding user behavior.

We have utilized a model provided by Leonardo Mariani that has readily been trained on various state of the art models like GLOVE, word2vec and so on. The same framework was utilized for testing the earlier obtained data as well after it was adjusted into suitable input format. The framework judged the performance of semantic matching of different events by the different test migration algorithm, descriptor algorithm, training set that was used to train the model and word embedding model.

The use of state-of-the-art models like GLOVE and word2vec in the training of the model provides a solid foundation for semantic matching of different events. By leveraging pre-trained word embeddings, the model can capture the semantic similarity between textual information of GUI widgets, enabling effective transfer of human-designed GUI tests from a source application to a target application with similar functionalities.

The testing of the earlier obtained data after adjustment into a suitable input format provides valuable insights into the performance of the framework. By evaluating the performance of semantic matching using different test migration algorithms, descriptor algorithms, training sets, and word embedding models, it becomes possible to identify the best approach for specific use cases.

The framework presented in [3] provides a robust and flexible solution for UI test reuse, leveraging state-of-the-art models and algorithms to achieve high accuracy in semantic matching. The insights gained from the evaluation of the framework can be applied to various domains, improving the efficiency and effectiveness of UI testing in software development.

Mean Reciprocal Rank (MRR) has been used for evaluation as a metric which has been used in [2] as well. The advantage of using MRR as an evaluation metric is that it takes into account the order of the results, rather than just whether or not the relevant item was present in the list. In other words, MRR rewards algorithms that place relevant items at the top of the list, and penalizes algorithms that bury relevant items lower down in the list.

According to the results, the top performing model combinations consisted of the SemFinder algorithm introduced by Mariani, ATM as a descriptor and Word Mover’s Distance for word embedding. These findings suggest that the aforementioned models can effectively enhance the accuracy of the model, thus making them the most suitable choices for the given task.