The architecture of the prediction model is shown in Figure 1. The model architecture is inspired by Hsieh, Moreira, and Ouyang. However, we do not separate the input into dynamic and static features.

One input consists of a 2-dimensional event tensor containing integers. The second input is a 3-dimensional tensor containing the remaining feature attributes. The first dimension in each layer represents the variable batch size, and *None* acts as a placeholder.

The next layer is primarily concerned with preparing the full vector representation. We encode each activity in the sequence into a vector space. We chose a dense-vector representation instead of a one-hot representation. We also create positional embeddings. Then we concatenate the activity embedding, positional embedding and the event attribute representation to a final vector representation for the event that occurred.

Afterwards, we pass the tensor through a Long Short-Term Memory (LSTM) module. We use the output of the last step to predict the outcome of a sequence using a fully connected neural network layer with a sigmoid activation as this is a binary classification task.

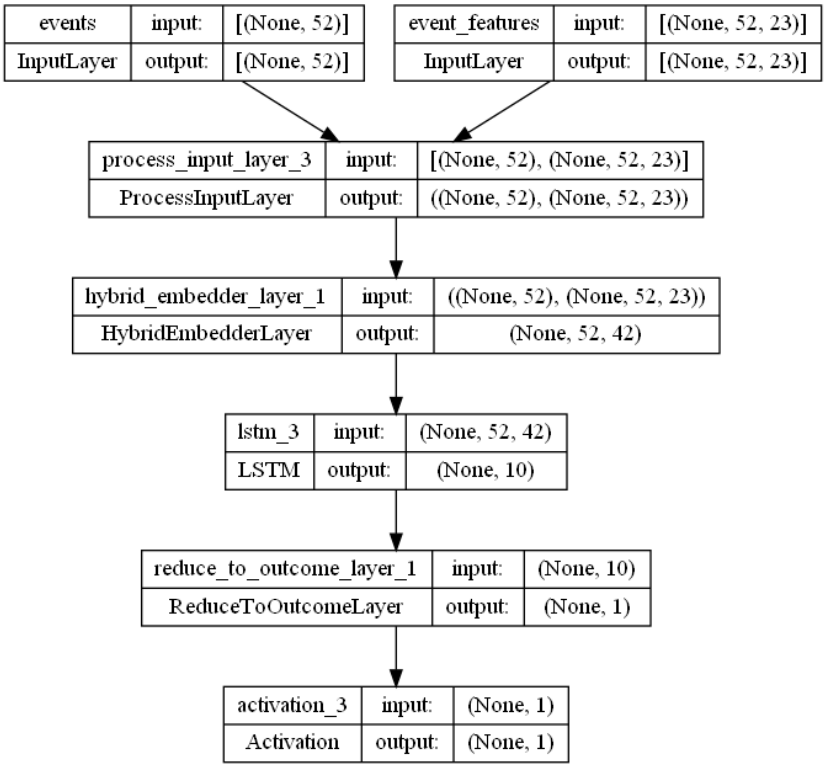

Figure 1: The different components of the LSTM architecture. Each element contains information about the input and output of a layer. None is a placeholder for the batch size.