

Title : Deep Learning Approaches for Identifying Compiler and Optimization Options from Binary Code.

Group Members :

20BCE515

20BCE518

20BCE519

Abstract—When compiling a source file, several flags can be passed to the compiler. These flags, however, can vary between debug and release compilation. In the release compilation, in fact, smaller or faster executables are usually preferred, whereas for a debug one, ease-of-debug is preferred over speed and no optimization is involved. After the compilation, however, most of the flags used cannot be inferred from the compiled file. These flags could be useful in case we want to classify if an older build was made for release or debug purposes, or to check if the file was compiled with flags that could expose vulnerabilities. In this paper we present a deep learning network capable of automatically detecting, with function granularity, the compiler used and the presence of optimization with 99% accuracy. We also analyze the change in accuracy when submitting increasingly shorter amounts of data, from 2048 up to a single byte, obtaining competitive results with less than 100 bytes. We also present our process in the huge dataset creation and manipulation, along with a comparison with other less successful networks using functions of varying size.