

図 6.1: 純正の Cowrie と修正済の Cowrie のスコアリングによる比較

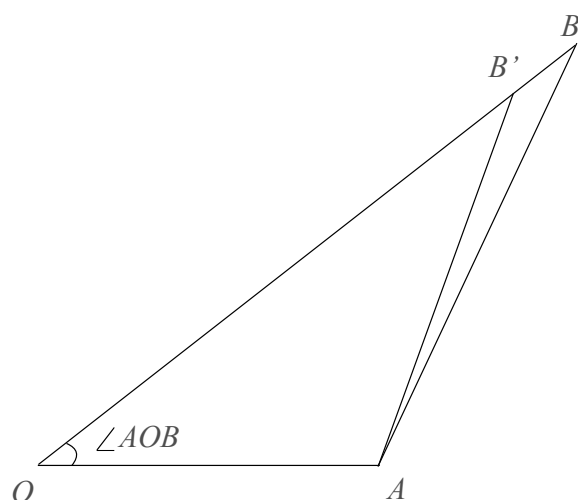
6.2 評価手法

本研究の仮説の検証手法としての評価として、[3.2](#) 節で述べた要件に対して評価を行う。予備実験では、素の低対話型 Honeypot よりも、コマンドを拡張した Honeypot の方がコマンドパターンが多く収集できることを示した。本研究では拡張した Honeypot で収集したコマンドログが、どれほど一般的な UNIX ユーザーの実行するコマンド [?] から離れたのかを評価した。

本研究では、以下の三種類の Honeypot を設置する。

1. 広く利用されている SSH の低対話型 Honeypot
2. 実際の Shell には実装されているが、1. の Honeypot で未実装のコマンドを実装した Honeypot
3. 広く利用されている高対話型 Honeypot

これ以降、1. の広く利用されている SSH の低対話型 Honeypot のことを ”純正の低対話型 Honeypot” ， 2. の実際の Shell には実装されているが、1. の Honeypot で未実装のコマンドを実装した Honeypot のことを ”修正済の低対話型 Honeypot” ， 3. の広く利



したがって, $0 < n$ を満たす n と任意のベクトル $\vec{\alpha}$ を使うと, \vec{OB} は $\vec{OB} = (1+n)\vec{OA} + \vec{\alpha}$ と表すことができる. したがって, 修正済の Honeypot のコマンドログは一般ユーザーのコマンドログを始点とすると, 素の Honeypot のコマンドログのベクトル方向よりも正の向きに遠くに位置することが分かった.

また, SCDV によって獲得した各々の Honeypot の攻撃ログにおける文章ベクトルを, t-SNE で次元削減することで可視化を行なった結果を図 6.4, 図 6.5, 図 6.6 に示す.

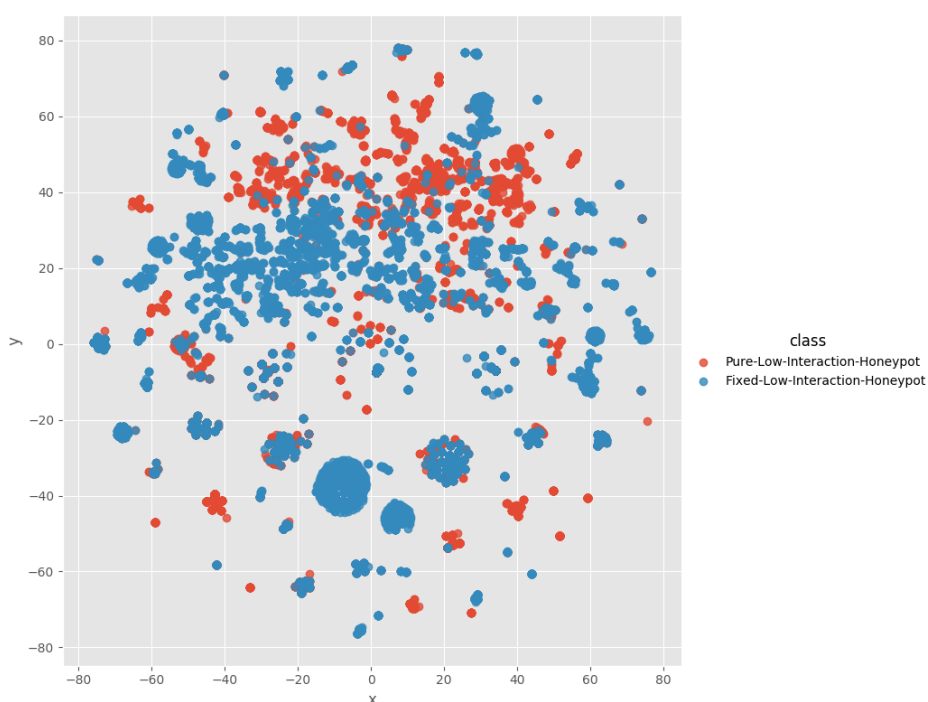


図 6.4: 修正前と後の honeypot のコマンドログの文章ベクトルの可視化