|  |  |  |  |
| --- | --- | --- | --- |
| Name |  | | |
| KA | DA | N |
| deer | 1.0 | 1.0 | 4 |
| forest | 0.28 | 0.5194419168941462 | 2 |
| pangram | 0.07692307692307693 | 0.14285714285714285 | 2 |
| tree | 1.0 | 1.0 | 2 |
| woodm | 0.22727272727272727 | 0.27882037533512066 | 3 |
| 1984 | 0.1875 | 0.23076923076923078 | 4 |
| finnegan | 1.0 | 1.0 | 2 |

1. **How does the value of *n* affect the decipherment accuracy it achieves?**

When n =1, the decipherment accuracy is close to the decipherment accuracy which is gotten by frequency analysis.

When n increases, mostly time the decipherment accuracy will increase as well. The decipherment accuracy will attach the highest DA during the increment of n. And after attaching the highest, the decipherment accuracy will decrease.

1. **How does this hill-climbing solver compare to the textbook solver?**

The hill-climbing solver is better than the textbook solver. For the solver from the textbook, the KA and DA is low. For the hill-climbing solver, the AK and DK is higher by decrypting the same cipher text. And for he hill-climbing solver, the KA and DA can even reach 1.0, that means the hill-climbing solver can completely solve the cipher text.