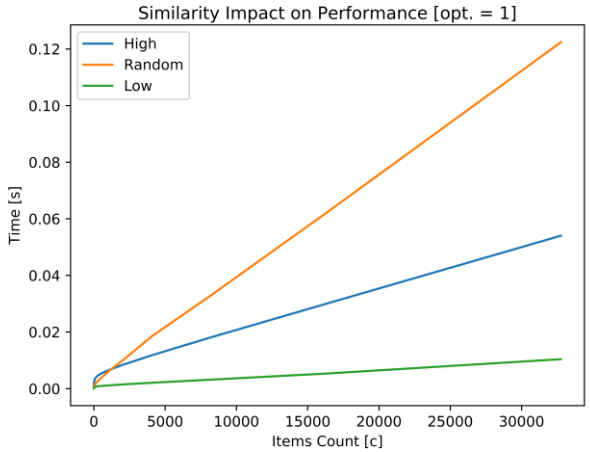

Fig. 3. Algorithm search performance in the case of low, random and high similarity data, with only the first optimization enabled.

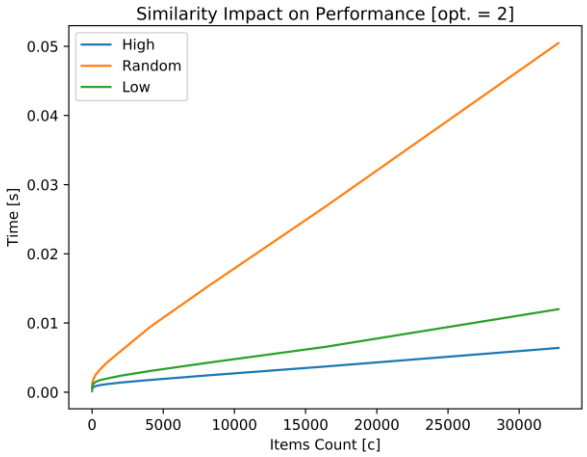

Fig. 4. Algorithm search performance in the case of low, random and high similarity data, with only the second optimization enabled.

The last two tests have been designed to test the asymptotic behavior of the contextual similarity function in function of the input size and search area. Fig.5 and Fig.6 show the results of the named tests.

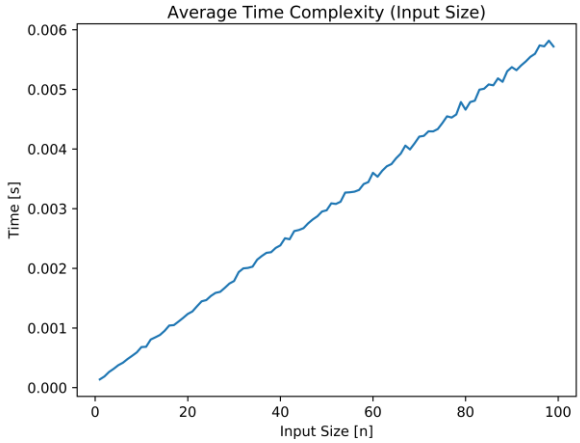

Fig. 5. Asymptotic behavior of the average time execution in function of the length of the first sequence.

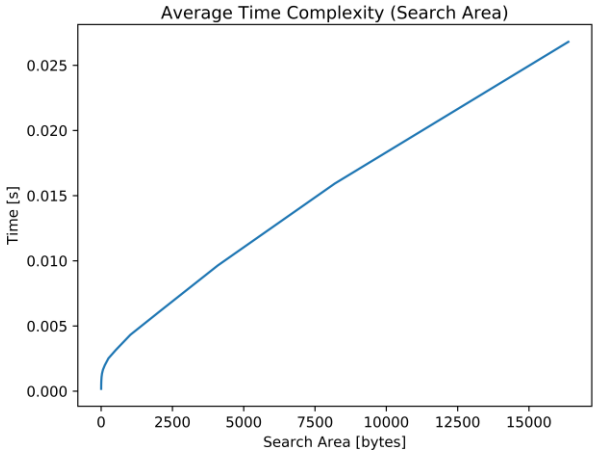

Fig. 6. Asymptotic behavior of the average time execution in function of the number of sequences to search through. All sequences have been generated randomly.

7 Conclusion

We conclude that our algorithm both from the theoretical and practical perspective is indeed quasilinear in nature. Notice how other similarity functions such as the **Jaccard Index**, **Cosine Similarity** or **Sørensen–Dice** are different. The **Jaccard Index**, for instance, can only operate on two sets at a time meaning that to calculate the similarity for **n** sets one would have to perform n^2 Jaccard calculations, which would result in **quadratic time** complexity. The proposed method provides a unique scoring which awards position, order and structure of the sequence. The algorithm allows for a unique way of performing a **dichotomic search** over inherently random and unsorted arrays of data which allows us to quickly search, find and match sequences based on their similarity.

References:

[1] X1. Konrad Rieck, Pavel Laskov, "Linear-Time Computation of Similarity Measures for Sequential Data", Journal of Machine Learning Research 9 (2008) 23-48 pp. 1

[2] S. T. Piantadosi, "Zipf's word frequency law in natural language: A critical review and future directions", Psychonomic Bulletin & Review, vol. 21, 2014, pp. 1112-1130