Justification

Several solutions were considered, namely:

- Simple implementation in Java 8 based on signatures (natural ordering of the word) and LinkedHashMap, the result: 10 Million words / 4 seconds. With this approach, it was necessary to implement Scalability by own. For example: sharding based on signatures. Also we have to implement transport and other components, that leads to the idea to take a look to other solutions.

- Replace LinkedHashMap to external storage such as a relational database (MySQL) or a key / value store (Redis). This solution allows to increase the volumes of raw data, but also require a lot of manual work, and it might lead to increase Scalability, but decrease Maintainability.

- Elasticsearch another solution, and everything would be fine, but in this case we need to prepare a data (insert, at least).

Spark is quite nice solution and satisfy the requirements: Maintainability, Scalability, Performance. And with Hadoop, will increase the volume of processed data up to 100 Billion Words easily.