Summary
This paper talks about the key challenges faced in longitudinal analytics on web archive data in a few aspects including time-travel indexing, querying and ranking, entity tracking, efficient analytics and scalability. For each category, it discussed the concrete aspects that need consideration, why these aspects make the problem itself challenging and how traditional data structures and algorithms are not suitable in this big data context.

Positive
a. The paper organization is clear and well formatted. Following the motivation part, each section addresses one aspect of the challenges.
b. Some part such as part discussing the challenge of time-travel indexing is detailed and nice-structured. The sub-problems are well explained. Concrete cases are given to explain the problems confronted such as applying multi-dimensional index structure like B-tree and TSB-tree for this problem.
c. The authors proposed some possible directions for the challenges and the limitations of these solutions, such as utilizing MapReduce for index building, using wikipedia for entity tracking, etc.

Negative/Potential extensions and improvements
a. Although directions/solutions are proposed for the challenging problems, some only works with the simplified assumptions. And the proposals are just too vague and big. It didn't offer meaningful technical details.
b. This paper is more descriptive and general. The intractability of phrase queries, for example, is not well explained. The complexities of cross-site retrieval are not clear.
c. It's better to offer some insights into the data to make it more illustrative, e.g. the distributions of time span of tracked-entities. In what way does it influence queries with the existence of indexing. Therefore, we can get more convinced about the big challenges.

Research Questions and Points for Discussion
a. This paper mentioned the role of MapReduce framework in tackling the scalability issues. So, how do they organize different components on this framework? How to parallel the different sub-tasks under their system architecture?
b. How to offer balanced workloads among reducers? What do they use as a unit to distribute the tasks?