Algorithm 3.1: Evaluation and Computation phases of DSCMS Model

End user (EUi) request Documents (SDn) from SCM (Pi) {
End users Query (Qi) is validated, Begin DSCMS,
Phase 1: SCM & Semantic Crawler
for user (i) {
    Key in Query Qn
    Check if Qn is new or old
    If (Qn == new) {
        Search for links in repository,
        If (Qn !found)
        Trigger semantic crawler ()
        If threshold >0.8
        Filter URLs based on meta-keywords MKi
        Read structure of web document
        If structure == Script [predefined structure]
        Store links in repository
        Update () repository periodically
    }
    Else if (Qn==old) {
        Create VDn with existing SDn
        Execute NER based concept_matching ()
        Annotate meaningful highlighted words
        Link the SDn with the top ten highlights
    }
    Select SCM(Pi) {
        If SCM(Pi==structured)
        Implement ETL {
            Extract harvested URLs
            Preprocess the content
            Store it in RDBMS repository
            Load into DFS
        }
        Else if SCM (Pi==semi-structured){
            Implement ETL {
                Extract harvested URLs
                Preprocess the content
                Store it in NOSQL repository
                Load into DFS
            }
        }
        Else if SCM (Pi==unstructured) {
Implement ETL {
  Extract harvested URLs
  Preprocess the content
  Store it in Distributed Data-model
  Load into DFS
}

Phase 2: RESCC on DFS
Domain expert User (j) {
  Initiate semantic k-means on DFS for high frequency keywords
  Identify topics using preliminary Tfidf method()
  Check domain specific keywords using concept_matching()
  Improve quality of cluster {
    Execute SHAC on DFS {
      Create 'n' number of clusters
      Match similarity score from NLP algorithms >0.5
      Iterate the process till minimum non-domain cluster
    }
    Fetch labels of cluster {
      Input labels to classifier1
      Perform sequence based classifier {
        Execute word_sense_disambiguation ()
        Execute SNBC()
        Identify sequence_words in VDn for threshold >0.8
      }
      Input labels to classifier2 {
        Generate intra and inter ontology
        Match similarity score >0.5
        Assign centrality (0-0.5) and authoritative scores (0.6-1.0)
        Rank the SDn matched with VDn using TOP-K
      }
    }
  }
}

Phase 3: Hadoop based Semantic Recommendation System (HBSRS)
Domain expert user(j) {
  If End user(EUi) mark the query and frequently visited links {
    Recommend the source links based on sequence words
    Personlize the SDn for user based on UFBC and CF
    Initiate ICTS based on marked links
    Preserve the timestamp Ts for the stored SD
  }
  Execute ABFS and DSBCD to track changes dynamically {
    Parse the content of URL
    Segregate to subject, predicate and object
    apply BFS to find the changes in literal and vertex
  }
  Execute DSBCD script in dynamic streamer {
    Check the timestamp ts of existing URL EURL
  }
if EURL change in ts==true
Check literal of object if (literal_change==true) {
Match the content with the NURL & EURL semantically
Generate RDF and Trigger alert periodically
Consolidate the alerts and email the end_user(i)
}

To accomplish the above algorithm in a successful manner there is a need for end to end semantic solution. A fourfold similarity measure is introduced in the DSCMS model which comprises of semantic annotation, concept matching, word sense disambiguation and sequence words based word sense mapping.