Research Paper

JIS

Journal of Information Science
1–27
© The Author(s) 2020
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/0165551520911590
journals.sagepub.com/home/jis
SAGE

An ensemble clustering approach for topic discovery using implicit text segmentation

Muhammad Qasim Memon, Yu Lu and Penghe Chen
Advanced Innovation Center for Future Education, Faculty of Education, Beijing Normal University, China

Aasma Memon
School of Economics and Management, Beijing University of Technology, China

Muhammad Salman Pathan
College of Computer Science, Beijing University of Technology, China

Zulfiqar Ali Zardari
Faculty of Information Technology, Beijing University of Technology, China

Abstract

Text segmentation (TS) is the process of dividing multi-topic text collections into cohesive segments using topic boundaries. Similarly, text clustering has been renowned as a major concern when it comes to multi-topic text collections, as they are distinguished by sub-topic structure and their contents are not associated with each other. Existing clustering approaches follow the TS method which relies on word frequencies and may not be suitable to cluster multi-topic text collections. In this work, we propose a new ensemble clustering approach (ECA) is a novel topic-modelling-based clustering approach, which induces the combination of TS and text clustering. We improvised a LDA-onto (LDA-ontology) is a TS-based model, which presents a deterioration of a document into segments (i.e. sub-documents), wherein each sub-document is associated with exactly one sub-topic. We deal with the problem of clustering when it comes to a document that is intrinsically related to various topics and its topical structure is missing. ECA is tested through well-known datasets in order to provide a comprehensive presentation and validation of clustering algorithms using LDA-onto. ECA exhibits the semantic relations of keywords in sub-documents and resultant clusters belong to original documents that they contain. Moreover, present research sheds the light on clustering performances and it indicates that there is no difference over performances (in terms of *F*-measure) when the number of topics changes. Our findings give above par results in order to analyse the problem of text clustering in a broader spectrum without applying dimension reduction techniques over high sparse data. Specifically, ECA provides an efficient and significant framework than the traditional and segment-based approach, such that achieved results are statistically significant with an average improvement of over 10.2%. For the most part, proposed framework can be evaluated in applications where meaningful data retrieval is useful, such as document summarization, text retrieval, novelty and topic detection.

Keywords

Information retrieval; natural language processing; ontological similarity; text clustering; text mining; text segmentation

1. Introduction

Text segmentation (TS) is the process of detecting boundaries of text units (or documents) that contain multiple topics in accordance with some task-dependent criterion. One of TS example is topical segmentation, which identifies boundaries

Corresponding author:
Yu Lu, Advanced Innovation Center for Future Education, Faculty of Education, Beijing Normal University, Beijing 100875, China.
Email: luyu@bnu.edu.cn

and divides a document into fragments called segments. An extracted segment shows the subjective definition that a topic contains across different text units in a corpus. TS algorithms are used widely in the perspective of different Natural Language Processing (NLP) tasks: (1) Information Retrieval (IR) [1], (2) document summarization [2] and (3) automatic generation of E-Learning courses [3]. With the advancement in a large amount of data such as plain text collections (newswire and scientific articles), web documents (blogs and webpages), it has been become essential to provide relative and informative contents among such collections through TS and clustering approaches. In TS, a text unit is partitioned into cohesive sub-topics (i.e. segments) and topics are retrieved based upon user's demand. TS is not restricted to find accurate information to the user, but it also reduces the user's effort and time to read the text document. Conversely, a document is partitioned into fragments (i.e. topics) in the document summarization, wherein each topic contains a final summary to ensure that it provides all the unique topics in the text document. However, a document is segmented through the regions of words which is considered as weakest approach as suggested by the previous work [4]. In addition, segments are retrieved using lexical relationship among words, and hence, such segments do not possess enough cohesiveness. Therefore, TS approach is taken into account in this study due to the fact that retrieved segments are named and indexed using word frequencies rather topical segmentation; thus, it is indiscriminated and provoked as a major concern. On the contrary, NLP tasks are deemed with semantical and ontological methods that contain conceptual meaning of text and may find the conceptualisation in accordance with user's demands [5]. Thereby, TS is our major and primitive concern in order to resolve afore-mentioned issues through ontological relation among its constituents.

Text clustering is useful in detecting cohesive topics, such that it gathers text units into a meaningful and organised manner for topic identification. These text units use topic relevance and they are declared categorically to provide instructive data with comparative ease. However, existing works in this regard confer that agglomerative hierarchical clustering (AHC) and partitional clustering are well established and successful in the domain of text clustering [6]. In AHC, documents are clustered to classify topic similarity in the shape of clusters and computed using distance functions [7]. Partition clustering allows overlapping of clusters, where a data point contains a member of multiple clusters [8]. Overlapping of clusters is a constructive phase in real data, where semantics solutions are produced using k-means and probabilistic algorithms. However, different issues were always plodded on, such that afore-mentioned clustering approaches consider a text unit is explicitly associated with several topics. For instance, collection of text units with several topics can be related to multiple topical terms; a scientific research article can be related with text clustering that may contain the knowledge of the clustering, and it seems to be connected to techniques and algorithms that are categorised in supervised, unsupervised, hard and soft clustering. Similarly, collections of text units in scholarly research articles can be associated with medical, life sciences, engineering, economics, social sciences and humanities. Consequently, clustering approaches were applied to above-mentioned text collections and can identify the problem into two scenarios: (1) generating clusters in overlapping fashion and (2) evolve using TS methods.

In the former scenario, clustering approaches including fuzzy-based [9], generative-based models [10] and ensemble methods for clustering [11] assume a text unit as a single unit of information (i.e. single vector), and a cluster is considered as a single topic. However, this kind of document representation is not suitable for a collection of text units where nature of topics per document is cumbersome. Furthermore, their thematic association (i.e. topics) is being lost when incorrect and inaccurate assignment of a text unit occurs to its constituents. We also noticed that such document representation is deemed with vector space model, where a text unit is considered as a high-dimensional vector as it corresponds to unique feature (i.e. words). The outcome of such weighted resultant vector considers bag-of-words model, which may not highlight polysemy and synonymy. In order to overcome this gap, segment-based clustering [12] assumes text unit into different segments based on word frequencies through TextTiling. It follows a TS approach that is less comparative but better than traditional clustering approaches [9–11].

In the later scenario, TS focuses on similarity (or dissimilarity) between two adjacent blocks and it is based on merely semantic approach (word-based analysis). It considers shallow word-based parsing to identify the relationships between text units (i.e. synonymy and hyponymy). The characteristic of afore-mentioned two scenarios is still present, and thus, is taken into account when the nature of documents is biased to multiple topics. For the most part, traditional and segment-based clustering approaches are needed to be revised in a way that clustering problem could provide justified solutions with respect to the limitations as discussed in the both scenarios. Therefore, we refer former scenario is related with text clustering and later scenario with the text segmentation. To overcome the limitations of both scenarios, we try to deal and identify the problem into three steps.

First, text segmentation is applied to documents, where each document is estimated using semantical and ontological similarity through LDA-ontology (LDA-onto). In particular, semantic similarity is computed using keyword score in a sub-document (see Section 3.3), as it describes semantic relations (through words) between sub-documents by leveraging the segmentation in a linear form at different levels of coarseness. Second, sub-documents are clustered initially through an intra-level clustering resulting a cluster (known as a sub-document set), which associates with a relevant topic across different text units. Later, an inter-level clustering is performed on the resultant clusters, where each cluster associates with exactly unique topic exhibiting the aggregation of topics contained in overall text collections that they possess. Meanwhile, we also emphasised ontological similarly for the sake of bettering the TS, as there are several popular ontologies exist among different domains: (1) IR domain is used for bettering the accuracy and semantic indexing [13], and (2) NLP domain is used in various applications, such as synonym detection [14], analogical reasoning in sentiment analysis [15] and word sense disambiguation [16].

Recently, number of ontologies were employed in NLP domain, such as (1) DBpedia [17], a cross-domain knowledge base is a type of Linked Open Data (LOD) cloud [18]; (2) Wikidata [19] is a collaborative ontology and (3) YAGO [20] is derived from Wikipedia. Existing literatures have been providing with the task-oriented techniques and outcomes in both NLP and IR with respect to the above-mentioned ontologies. However, such ontologies are still need to be incorporated with TS methods in order to improve topical identification of documents. For instance, traditional model approaches use syntactic information [21, 22], where each text unit is represented in a vector space and words are assumed as dimensions. However, such approaches claimed inappropriate in finding the relationships among different semantic concepts such that ‘Pakistan have won the International Cricket Council (ICC) world cup only once and that victory came at the back of an excellent leadership’ and ‘Imran Khan had laid the foundation of first cancer hospital in Pakistan’. However, both of the sentences exhibit similar concepts, that is, Pakistan cricket team won the ICC world cup under the leadership of Imran Khan: (1) who is current Prime Minsiter of Pakistan, (2) founder of Pakistan Tahreek-e-Insaf (PTI) local party and (3) founder of Pakistan’s first cancer hospital. This knowledge can be found/extended through cross-domain ontologies. To do so, we improvised DBpedia and Wikipedia in this study through ensemble clustering approach (ECA) in order to estimate the similarity (i.e. ontological) of multi-topic documents (in the shape of words and topics), and they consider in-depth semantic analysis to find the coherence of documents among its constituents. In addition, similarity between sub-documents is computed to assess the cohesiveness through onltological and semantical too, where semantic similarity is performed conceptually [23] rather using lexical [24].

Underpinning the aims of this study is the notion that the research described sheds the light on clustering solutions in combination with TS through ontological and semantical similarity. LDA-onto is deemed to extract a segment, which deteriorates exactly one sub-topic in a single document. This allows a text unit is not being considered as a single unit, which is opposed to traditional clustering. Moreover, it divides a document in a way that it can recognise adhesive portions of a document into sub-documents and stick them together into groups. Eventually, clustering these groups to formulate various clusters, where each cluster represents a unique topic. As a result, descriptions of the resultant clusters may contain topics with higher cohesiveness of keywords within each cluster, and the existence of discriminating keywords is obvious to find the cluster appropriateness (see section 5.11). To achieve this, we follow the document representation approach as reported by [25], is referred to as sub-document-based document clustering which allows an efficient identification through overlapping and disjointed clustering solutions. Furthermore, experiments in this work provide an illustration of LDA-onto that estimate the extracted segments based on ontological and semantical similarity and generate the clusters through overlapping and disjointed clustering solutions. ECA demonstrates the following main contributions:

- ECA provides a demonstration of LDA-onto model through ontological and semantic similarity, which improves the quality of segmentation.
- ECA presents cluttering solutions of disjointed (non-overlapping) and overlapping for multi-topic documents.
- ECA is a novel topic-modelling clustering approach, which aims to encourage document representation in the form of cohesive fragments (i.e. sub-documents).
- ECA generates qualitative clustering and segmentation solutions are better than traditional and segment-based clustering.

The rest of this article is organised as follows. Section 2 briefly explains the related works of text clustering and text segmentation. Section 3 provides the preliminary background of the text segmentation and document representation models. Section 4 provides the evaluation metrics of text segmentation and clustering. Section 5 presents the experiment results. Finally, section 6 presents the conclusion.

2. Related work

2.1. TS

TS can be classified into four categories: (1) content and discourse based, (2) supervised and unsupervised based, (3) linear and hierarchical based and (4) borderline-based detection. Content-based TS depends upon content structure and estimates the word deviation based on sentence boundary. One of most famous content-based TS approach (i.e. TextTiling) is proposed by Hearst [26]. Discourse-based TS emphasises lexical and prosodic features in a discourse structure of story that tends to appear near the segment boundaries [27]. In the supervised TS, text is segmented into two adjacent segments through their lexical cohesiveness at their weakest level, such that a distant supervised approach is proposed to perform TS on multi-label documents using training data as reported by Saurva and George [28]. However, a Bayesian topical of lexical connectivity is proposed by C99, which is a type of an unsupervised TS [29]. The third category of TS is linear, which focuses on sequential approach and it analyzes topical variance among disjointed segments (i.e. non-overlapped fashion). Linear segmentation is also based on TextTiling approach and it has shown better results using domain knowledge [30]. Hierarchical TS is meant to find well-grained topics structure of coherent segments and it is based on the cohesion measurement that rendering sub-topics hierarchy [31]. Borderline-based detection can be performed through three methods: (1) similarity, (2) graphical and (c) lexical chains. Similarity method considers each text block as a single vector, and it estimates the proximity using cosine angle between two vectors. It also detects borderline using similarity matrix to categorise sentences and isolating the topical segments. C99 is an example of borderline-based detection TS through similarity. Graphical method finds coherent topics in text units and uses the term frequencies to detect borderline of the segments [32]. Lexical chains method combines semantic chunks of words in a sentence using a Roget's thesaurus [33]. The afore-mentioned categories of TS are well known in the applications of IR and NLP domain, such that they can enhance the performance of the user's retrieval experience by prevailing associated parts of a text unit. As a result, different methods were proposed in this regard [34], who claimed in finding the thematic parts of documents and identified the lexical chain information of associated words in cohesive segments through the boundaries between two regions in a text unit. However, boundaries are subject to change in vocabulary and topical structure too, and thus, it may not viable in finding exact boundaries with topical variations. This causes generating the segment boundaries with the disadvantage of words being repeated throughout the process of segmentation as reported in the existing works [35]. To overcome this issue, different approaches were also proposed [36, 37], but they do not rely on a training phase or directly applied to text data. However, existing works found comparative results using ontological similarity in conjunction with labelled data, but their document representation assume each text unit as single piece of information as well as their thematic information is being lost, and thus, segments might not be related or labelled with any topical information [38, 39].

2.2. Text clustering

Various clustering algorithms have been proposed, including partitional clustering (i.e. fuzzy clustering aimed to provide overlapping solutions) [40, 41]. In partitional clustering, each text unit's dissemination is based on a centroid, which represents a cluster and it is assigned by a distance measure or a heuristic function. Fuzzy k-means is composed of a function and it is related to each text unit through different clusters. Their fuzzy values are greater than the threshold values as defined by the user [9]. In probabilistic models, latent semantic analysis (LSA) analyzes the relationship between a set of text units and it performs dimensionality reduction to the terms-document matrix [42]. Probabilistic latent semantic analysis (PLSA) is an extension of LSA, where each document is represented as a mixture of topics distributed across the terms. Latent Dirichlet allocation (LDA) is also considered as a mixed model consisting of documents, corpuses and terms. Consequently, probabilistic models in IR domain meant to produce a distribution of possible outcomes, that is, ranking documents according to their similarity.

In the context of afore-mentioned studies, researches were found less comparative and imparted below average results, such that irrelevant topics in terms of similarity and topical coherence produced less qualitative topics (i.e. through resultant clusters) as reported by [10, 11, 41]. Furthermore, clustering approaches, namely ensemble subspace clustering [21] and probabilistic-based model [10], consider each cluster as a unique topic. However, such resultant clusters are compromised with topical relevance, and thus, it is discriminating, as both of these approaches could not exploit discriminated words with/across original documents. To overcome these shortcomings, a segment-based clustering approach proposes a significant way to cluster multi-topic documents, and it found better results than traditional clustering approaches by segmenting a document into different portions that in turn generated qualitative resultant clusters. However, the authors could not justify their coherence segments in a document due to their generalised similarity computation among words as well as their TS approach contends to partition a document through word frequencies only (i.e. through TextTiling). However, topic-based models are most widely famous in the traditional clustering, and they found comparative results when clustering the multi-topic documents [43, 44], but they also consider a document as a single unit of information and it is not supported with any topical information (such as labelled information or an ontology). To overcome this, dimensionality reduction techniques using ontology is applied to terms within text units through clustering algorithms [45]. However, the primitive objective of topic-modelling-based models is to detect topics in documents, is thus generalised in

Table 1. Notations.

Symbol	Description
D	Set of documents
d	A document in a *D*
sd	Sub-document
sds	Sub-document set
S	Set of sub-document sets
N_d	# documents contain in **D**
N_{Sd}	# sub-documents contain in *Sd*
N_S	# sub-document sets contain in **S**
N_D	# contain in **D**
C_S	Sub-document set clustering
C	Document clustering
C_d	Document cluster
T	# Topics
V	Vocabulary size
J	Topic labels in set of documents **D**
K	# Average of sub-document sets
sdtf-idf	Sub-Document Set Term Frequency-Inverse Document Frequency
sdtf-isdf	Sub-Document Set Term Frequency-Inverse Sub-document Frequency
sdtf-isdsf	Sub-Document Set Term Frequency-Inverse Sub-document Set Frequency
α	Parameters of topics in a Dirichlet prior
γ	Parameters of words in a Dirichlet prior
W	Set of words (w_i) allocated to *W*
T	Set of topics (t_i) allocated to words in *T*
β	Probability of possible topics allocated $z = k \in d$
θ	Probability of possible words allocated $w = v \in z$

the afore-mentioned works, such that their thematic coherence of topics in the resultant clusters could not verify its semantic relations to original documents. In this study, ECA has been guided by the several studies to overcome above-mentioned shortcomings and follows the sub-document representation scheme in an agreement with the approach proposed by [25]. This study has been validated using the clustering algorithms that provide cluster solutions (i.e. overlapping and disjointed), which also validates the experimental results when compared against traditional and segment-based clustering.

3. Document representation and scoring

3.1. Notations

Let **D** be a set of documents, where $d \in \mathbf{D}$ contain sub-documents (*sd*) in a document (*d*). A set of sub-documents (*sds*) in a document corresponds to an individual sub-document set across **D**. Collection of sub-document sets (**S**) is denoted as $\mathbf{S} = \bigcup_{d \in \mathbf{D}} Sd$. Table 1 illustrates the notations used in this article.

3.2. LDA

LDA [45] finds topics among a set of documents, where each text unit is associated through topical sampling that contain word distributions. Considering a topic (*t*) and a topical sampling (β_{twi}) is chosen from a Dirichlet prior is inter-connected to vocabulary size. A text unit is selected from sampled topics $\theta_d = \{\theta_{dt}, t = 1...T\}$ using a Dirichlet prior, wherein words chosen are ranged in a mixture of topics for a model. The occurrence of words of a topic is chosen by θ_d and selected as the likelihood of w_i, where *i*th word of a topic is given below

$$P(w_i|\theta + \beta) = \sum_{t=1}^{T} P(t_i = t|\theta_d)P(w_i|t_i, \beta) \qquad [(1)$$

$$\sum_{t=1}^{T} \theta_{dt}\beta_{tw_i} \qquad [(2)$$

where $\mathrm{P}(\mathrm{t}_i = t|\theta)$ is associated with the probability of ith topic, which is selected for ith word token, and $\mathrm{P}(\mathrm{w_i}|\mathrm{t_i}, \beta)$ is selected as the probability of w_i from a set of topics. The likelihood of text document C_d is given as

$$\mathrm{P}(C_{\mathrm{d}}|\theta_{\mathrm{d}}, \beta) = \prod_{\mathrm{w}=1}^{\mathrm{W}} \left[\sum_{\mathrm{t}=1}^{\mathrm{T}} \theta_{\mathrm{dt}} \beta_{\mathrm{tw_i}} \right]^{C_w} \tag*{[(3)}$$

where C_w is number of words counted in a document. LDA-onto is trained once the sub-documents are extracted. However, model is not entirely relying on word redundancies and it incorporates a semantic arcanum in order to decompose documents. In particular, LDA-onto identifies topics in the sub-documents, and similarity measured in each sub-document with respect to adjacent number of sub-documents in original documents, which is opposed to TextTiling, such that there is no decrease in the similarity measure. Parameters settings (topic and word distributions) are measured for inference and their estimation is computed using Gibbs sampling from a set of documents. Assuming a collection of documents **D**, topic sampling is represented by $\{\theta_{dt}, t = 1...T, d = 1 \ldots \mathbf{D}\}$, and word sampling for a single topic is represented by $\{\beta_{twi}, t = 1...T, w = 1 \ldots W\}$. Two parameters ($\alpha$, γ) are employed to exploit the non-informative Dirichlet priors θ and β, respectively. The probability of allocating a current token to each topic is not applicable through a word. However, a topic is then comprised from a large distribution and it is allocated to current tokens. In Gibbs sampling, estimations parameters (θ and β) are taken into consideration in topic assignments, as follows

$$\beta_{tw_i} = \frac{J_{tw} + \gamma}{\sum_{k=1}^{W} J_k + w\gamma} \tag*{[(4)}$$

$$\theta_{dt} = \frac{K_{tw} + \alpha}{\sum_{k=1}^{w} K_k + T\alpha} \tag*{[(5)}$$

where J_{tw} is the number of times that a word is assigned to a topic (t) and K_{tw} is the number of times that a topic (t) is assigned to word tokens in a document (d). LDA-onto is also able to predict documents when same vocabulary exists as to the training corpus. To achieve this, topic distribution is computed to test a document using the iterative procedure based on the rule as follows

$$\theta_{\mathrm{dt}} = \frac{1}{\mathrm{L_d}} \sum_{\mathrm{w}=1}^{\mathrm{W}} \frac{C_{\mathrm{dw}} \theta_{\mathrm{dt}} \beta_{\mathrm{tw}}}{\sum_{\mathrm{t}'=1}^{\mathrm{T}} \theta_{\mathrm{dt}'} \beta_{\mathrm{t}'\mathrm{w}}} \tag*{[(6)}$$

where L_d is the length of document, which estimates the number of times of words' iterations. This rule is monotonically adjusted in either an increasing or decreasing manner. The likelihood of a document is obtained and executed under 10 times of iterations and can be computed once θ is achieved. Later, test estimation of the likelihood of sub-documents is computed in order to perform segmentation. Since LDA-onto is also trained using Wikipedia/DBpedia corpus and computed using semantic similarity. Semantic similarity is computed based on the sub-document score whether to test the sub-document, that is, finding topic distribution through adjacent keywords' scores in sentences in accordance with the distance as given by

$$D(s_i, s_{i+1}) = \sum_{t=1}^{T} \left[p((t|s_i) - p(t|s_{i+1}))^2 \right]^{\frac{1}{2}} \tag*{[(7)}$$

where $p(t|s_i = 1...n\}$ is the pairwise similarity of sentences in each sub-document, which resembles the probability of cohesive topics are generated using LDA-onto model.

3.3. Scoring of terms in the sub-document

The identification of words in a sub-document is computed using the *sdtf-isdf* score and sentiment score. In particular, words with the highest score are recognised as topical terms of a sub-document, where each term represents a unique topic in a sub-document. The word score of a term is calculated as follows

$$sdtf(t) = \frac{\mathrm{word}(t)\ \mathrm{occurrance\ in\ sub} - \mathrm{document}}{\mathrm{total\ number\ ofwords\ in\ sub} - \mathrm{document}} \tag*{[(8)}$$

$$isdf(t) = \frac{\#\text{sub} - \text{document sets}}{\#\text{ of sub} - \text{document sets of that word}(t)\text{ occurrences}} \qquad [(9)$$

$$Sc_{wi} = sdtfisdf(wi) \times Sm_{wi} \qquad [(10)$$

where Sc_{wi} is word-rating score of (wi), $sdtfisdf(wi)$ is the Sub-Document Term Frequency-Inverse Sub-Document Frequency (*sdtf-isdf*) score of the word, and Sm_{wi} is the sentiment-rating of wi. Once words are being identified within each sub-document, they are compared with identified words (i.e. keywords). This comparison is done on the basis of word-rating, that is, to discover remaining words whose score is close to remaining keywords. This closeness is defined through a certain limit of threshold value. The remaining words close or equal to such threshold value resemble closeness towards keyword. For the identification of each sub-document, the paragraph rating is given by

$$(P_i) = \frac{Sc(w_i) + \text{rating}(w_j)}{\#\text{ of keywords } Sc(w_i) + (\#\text{ of words close to keywords rating}(w_j))} \qquad [(11)$$

where P_i is the paragraph rating score of words that appeared in different sentences nearby to keywords in sub-document, rating (w_j) is rating score of word j, whose value is close to the Scw_i. Once the identification of paragraph is complete, average sub-document score through paragraph rating score is given by

$$Sd_i = \frac{1}{n}\sum_{i=1}^{n}\frac{P_i}{Scw_i} \qquad [(12)$$

where n is the total number of words used in a sub-document. Sub-document score is computed by averaging the score of all the words in the sub-document, that is, score of the keyword. However, occurrence of similar sub-documents can be represented by adding sub-document score sd_i and divide it with number of sub-documents being merged. The pseudo code of ECA is shown in Algorithm 1, which elaborates TS algorithm using two methods, such as LDA-onto and TextTiling. In LDA-onto, first removal of stop words and stemming is performed followed by the similarity of sub-documents (sd_i) is computed and compared for all consecutive sub-documents. As a result, we extract sub-documents in the sub-document set using the sub-document score as given in (equation (12)). Later, we performed soft and hard clustering on group of extracted sub-document sets that provide resultant clusters they contain.

3.4. Borderline detection

TS has been considered an essential task in the domain of NLP and IR. For the most part, it has laid determinant effort through different researchers into two ways: (1) finding topical boundaries within each block of text using TextTiling [30]; (2) finding topics based on extracted sentences (i.e. segments), which is proclaimed as 'speaking about' for a specific topic. However, traditional approaches [37–41] find topics underlie on boundaries by default: either they consider topical borderline is to be sketched in the perspective of no man's land between two distinguished topical areas, or a large variance in the vocabulary could occur. Furthermore, these approaches limit the data to words rather using topics and they are based on similarity or density measure, thus, letting go to discourse and semantic information contained in the text units. Initially, Choi showed improved results on C99 algorithm by decreasing the size of word capacity in a vector space with two approaches such as LSA and rank matrix. Later, DotPlotting, which is a graphical representation of the text into words with one or more dots on a bi-dimensional space, is considered. These dot positions are predicted on the basis of word appearance in the text. Conversely, borderline detection methods follow the concept of lexical chains by improvising an 'intended boundary detection'. As a result, a form of chain is occurred in the text of possible occurrences with a revised term/words. In this study, sub-document-based representation is considered in the text segmentation, where a sub-document is syntactically cohesive about a specific topic and succeeding sub-document sets are associated with different topics [25]. In doing so, we consider each sub-document as it contains at least three topics in order to understand wider spectrum of TS problem. Let n represents the number of sub-documents in a document, and θ_n represents the consecutive boundary points. We follow the parameters settings as discussed in Metropolis Hasting Green algorithm [46], which is a Markov Chain Monte Carlo (MCMC) technique. The computation of parameters is given by Algorithm 2, which finds the target distribution (P) with diffusion or jump, and later it computes uniform probability (A) across all the sub-documents in a document. It repeats until the probability distributions draws samples up to the (T) times (i.e. number of overall samples in the sub-document sets), as given by

Algorithm 1. Puedo code of ECA.

```
:
1. TS: Text Segmentation algorithm
        i. LDA-onto TS method
        ii. TextTiling method
2. LDA-onto TS method:
            Input: numbers of sub-documents
            Output: number of clusters
    a.    Begin
    b.        Removal of stop words.
    c.        Finding out the root/stem of a word. //stemming
    d.        For each $sd_i$, do
    e.            Compute $Sim(sd_i,sd_{i+1})$ and $Sim(sd_i,sd_{i-1})$
    f.        If $(Sim(sd_i,sd_{i+1}) \geq Sim(sd_i,sd_{i-1}))$ then
    g.            Combine $(sd_i,sd_{i+1})$
    h.        End
    i.        Else
    j.            Combine $(sd_i,sd_{i-1})$
    k.        End
    l.    End
    m.  End
    n.  compute the sdtf-isdf and sentiment score of each word   // representative keyword for each sub-document
    o.  For each $S_d$, compute $\sum_i^n sd_i$      // as given in (12)
    p.  n sub-documents randomly.
    q.  $S_d \overset{TS}{\leftarrow} sd$                          // retrieve sub-documents
    r.  until clusters become stable
    s.  According to the initial rating of sub-document in (11), choose sub-documents in the sub-document set nearby
        score to it.
    t.  for each possible sub-document (sd), Find θ, with $\theta_{sd_it} \leftarrow \frac{1}{n_i}\sum_w^W \frac{\mathbf{C}_{tw_i}\theta_{sd_i}\beta_{tw_i}}{\sum_{t=1}^T \theta_{sd_it'}\beta_{t'w_i}}$
    u.  Find its likelihood, with $P(w_i|\theta+\beta) = \prod_{w=1}^W [\sum_{t=1}^T \theta_{dt}\beta_{tw_i}]^{C_{w_i}}$
    v. -  Sub-document likelihood is computed by its rating: $\log P(w_i|sd_i) = \log P(w_i|\theta,\beta)$
    w.  Update the sub-document set score
3. SC: soft partition clustering
4. DC: hard partition clustering
5. Sub-document, $sd \leftarrow \varnothing$
6. $sd \leftarrow sd \cup S_d$
7. $S \leftarrow \varnothing$
8. For all retrieved, $S_d \in sd$, do
9. $S_d \overset{SC}{\leftarrow} (S_d)$
10. $s \leftarrow s \cup S_d$
11. $C_S \overset{DC}{\leftarrow} (S)$                             //cluster sub-documents sets
```

Algorithm 2. Parameters estimation.

```
1. Initialise the initial state of $n$
2. Find the $n'$ based on target distribution $p(n \rightarrow n')$ with a diffusion or jump
3. Compute $n'$ with uniform probability to $A(n,n') = \min\left\{1, \frac{p(n')g(n' \rightarrow n)}{p(n)g(n \rightarrow n')}\right\}$
4. If computed transition to $n'$ Then
        a. Accept initial state $n'$
5. Repeat Step 2 until T sample times
6. Commit $n$ as a sample distribution, and repeat step 2.
```

3.5. Similarity tagging

Semantic tagging is the process of inserting semantic tags in a document, which allows additional information to be processed according to semantic and relevance scores. A text unit is semantically annotated using traditional named entity recognition algorithm, which extracts text entities. Each entity is mapped to a single class in an ontology. The effectuality of annotation recognises the entities exist in the text unit that exactly generates a prototype of knowledge, and its semantics are extracted within a domain in the form of features. To do so, we deployed Wikipedia/DBpedia to construct features automatically, and thereafter, we can identify entities in the text. Entities are matched to their classes through knowledge-based ontology; thereby, similarity between adjacent sentences and entities in a sub-document is measured conceptually into related classes.

3.6. Similarity estimation

When the distance between classes is shorter, higher the similarity is achieved. The similarity is given by

$$\mathrm{Sim}\left(c_x, c_y\right) = \frac{2De^{-\lambda P/N}}{D1 + D2} \tag*{[(13)}$$

where P indicates the shortest path distance between two concepts. Weight of 1 is assigned when an edge traversed in the vertical side, and more than 1 in case of an edge traverse in different directions. N is the entire root of ontology tree. $D1$ and $D2$ indicate distance from original node to c_x and c_y. λ is assigned as '1' and '0' concepts associate 'different' and 'same' hierarchy, respectively. Thus, similarity of two entities e_x and e_y is given by

$$\mathrm{Sim}(e_x, e_y) = \sum_{x=1}^{a}\sum_{y=1}^{b} \mathrm{Sim}\left(c_x, c_y\right) \tag*{[(14)}$$

where a and b are the two groups of class entities (e_x and e_y) that they contain. Similarly, similarity between two paragraphs ($P1$ and $P2$) is as follows

$$\mathrm{Sim}(P_x, P_y) = \sum_{x=1}^{A}\sum_{y=1}^{B} \mathrm{Sim}\left(e_x, e_y\right) \tag*{[(15)}$$

where A and B are groups of entities in paragraph P_x and P_y that they contain. We can then compute the similarity between sub-documents sd_i and sd_{i+1}, contain a group of entities, are as follows

$$\mathrm{Sim}(sd_i, sd_{i+1}) = \frac{\sum_{x=1}^{M}\sum_{y=1}^{N} \mathrm{Sim}\left(P_x, P_y\right)}{M * N} \tag*{[(16)}$$

where M and N are group of entities.

3.7. Sub-document set clustering

The clustering of sub-document is performed through k-way clustering. The clustering algorithms, namely spherical k-means (*Sk-M*), and bisecting version of *Sk-M* produce disjointed (non-overlapping) clustering solutions. The overlapping clustering solutions are produced using LDA clustering algorithm and bisecting version of LDA. These clustering solutions correspond to collection of sub-document sets (**S**) in the formation of a unique cluster. The TF-IDF representation scheme is applied to document (d), sub-document (sd) and sub-document set (sds), respectively. However, adjusting term' weighting functions are matched to *tf-idf.* Consider $tf(w, Sd)$, where w is represented as index term and sd is represented as sub-document. The frequency functions for document, sub-document and sub-document-set correspond to *sdtf-idf, sdtf-isdf* and *sdtf-isdsf*, respectively, are given by

$$sdtf-idf(w, Sd) = tf(w, Sd) \times \log\left[\frac{N_D}{N_D(w)}\right] \tag*{[(17)}$$

$$sdtf-isdf(w, Sd) = tf(w, Sd) \times \exp\left[\frac{N_{Sd}(w)}{N_{Sd}(w)}\right] \times \log\left[\frac{n_S}{n_s(w)}\right] \tag*{[(18)}$$

$$sdtf-isdsf(w, Sd) = tf(w, Sd) \times \log\left[\frac{N_S}{N_S(w)}\right] \tag*{[(19)}$$

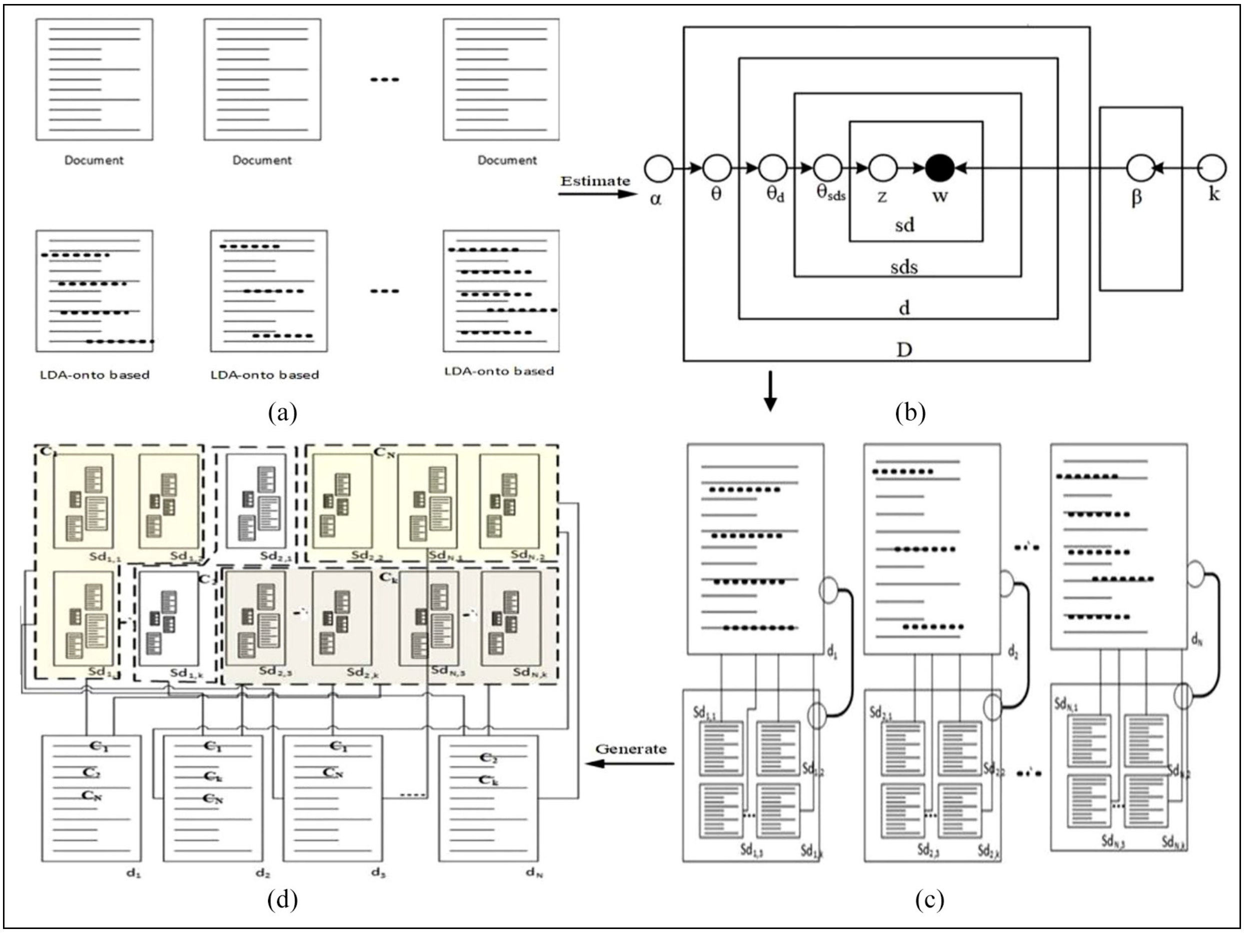


Figure 1. ECA: an illustration of sub-document-based representation using LDA-onto text segmentation. The description of all the labels used is provided in Table 1.

where N_D specifies the number of documents in $\mathbf{D}$ and $N_D(w)$ defines the part of $\mathbf{D}$ that contain w. N_{Sd} is the number of sub-documents and N_s is the number of sub-document set in S. $N_{Sd}(w)$ and $n_s(w)$ are distributions of Sd and S that contain w. N_s denotes the number of sub-documents sets in S and $N_S(w)$ denoted by the number of distributions of S that contain w. The exponential factor is included to improve the frequency of terms for sub-document sets.

3.7. ECA clustering mapping into traditional clustering

An illustration of ECA is presented with the first step of segmentation is pictorically shown in Figure 1(a), where sub-documents associated with a text unit are segmented based upon topics and words. Each sentence in a document is estimated using LDA-onto model (see Figure 1(b)). Specifically, boundary detection is based on similarity estimation as described (Section 3.4), which shows that it is not biased over sentences only rather it is based on terms score (i.e. words and topics) by computing semantical and ontological similarity. A sub-document set is assumed to have one unique topic, whereas a text unit may contain multi-topics in the entire corpus, such that a collection of sub-document $\{sd_{1,1}, \ldots Sd_{N,1} \ldots sd_{N,k}\}$ from a collection of text units $\{d_1, d_2, d_3, \ldots d_n\}$ is derived to perform inter level clustering (see Figure 1(c)). This is done by dividing whole corpus from text units $\{d_1, d_2, d_3, \ldots d_n\}$ into its subsequent sub-documents $\{sd_{1,1}, \ldots Sd_{N,1} \ldots sd_{N,k}\}$. Later, each cluster is generated (sub-documents clustered in disjoint and overlapping fashion) as shown in Figure 1(d), where each colour depicts exactly one topic (such as $C_1, C_2, \ldots C_N$, and C_K) inflicts the confirmation of topical cohesiveness within/across text units. This validates that ECA is suitable for clustering multi-topic documents and it resembles a fair representation of documents. Eventually, resultant clusters are mapped back to its respective text units in non-overlapping fashion. For instance, $\{C_1, C_2, \ldots C_N, C_K\}$ are clusters back to its respective text unit $\{d_1, d_2, d_3, \ldots d_n\}$ (see Figure 1(d)). A cluster refers to a sub-document set is mapped into its respective text unit in order to perform intra-level clustering; meanwhile, it generates non-overlapping clustering solutions (C_S). However, mapping of these clusters into their original text units was necessary to distinguish the performances achieved using ECA against traditional clustering. Performances of ECA are obtained using both the disjoint clustering (through spherical k-mean (*Sk-M*) and bisecting *Sk-M*), and the overlapping clustering (through LDA and bisecting LDA). The number of iterations in the experiments is restricted to 50 for each clustering algorithm.

4. Evaluation

4.1. LDA-onto text segmentation

P_k was introduced by Beeferman et al. [47] and WindowDiff was coined by Pavzner and Hearst [48], as they are the most popular text segmentation penalty measurement metrics. In order to compute P_k, given a window of fixed width k is assumed to derive one-half of the average of sub-document size in the reference partition and shifts it across the dividend text of consecutive sub-document. However, each step is tested whether the hypothesised accuracy of derived sub-document is manipulated correctly in the spite of fact that separation validates on both sides of the window. P_k metric is defined as

$$P_k = \frac{1}{N-n} \sum_{1 \le i \le j \le n}^{N-n} \left[\delta_{hyp}(i, j+n) \neq \delta_{ref}(i, j+n)\right] \qquad [(20)$$

where $\delta_{hyp}(i,j)$ and $\delta_{ref}(i,j)$ is an indicator function equal to 1 or 0, which indicates whether both sentences i and j are assigned to the same sub-document in hypothesised and reference segmentations, respectively. Sentences are related to same sub-document when function value is one and zero when sentences belong to different sub-document. n is indicated as ratio of one less than the integer nearby to half of the number of sentences and sub-documents in the reference segmentation. Lower scores show the better agreement between the two extracted segments. However, traditional methods [34–39, 49] assumed the segmentation that led below par results values such as (1) single pair segmentation, (2) assign a sentence its own segment and (3) assign constant boundaries and random boundaries. Unlike, we considered P_k dealt with manipulation of penalty of false positives at intense level than false negatives, which selects distribution of extracted segment with varying sizes. Hence, WindowDiff (WD) metric is given as

$$W_D = \frac{1}{N-n} \sum_{1=1}^{N-n} \left[b_{hyp}(i, j+n) \neq b_{ref}(i, j+n)\right] \qquad [(21)$$

where b_{hyp} indicates that the extracted segmentation generated by model, and b_{ref} indicates exact segmentation for reference, N is the number of sentences in the text, n is the size of the sliding window and $b_{i,j}$ is the number of boundaries between i and j.

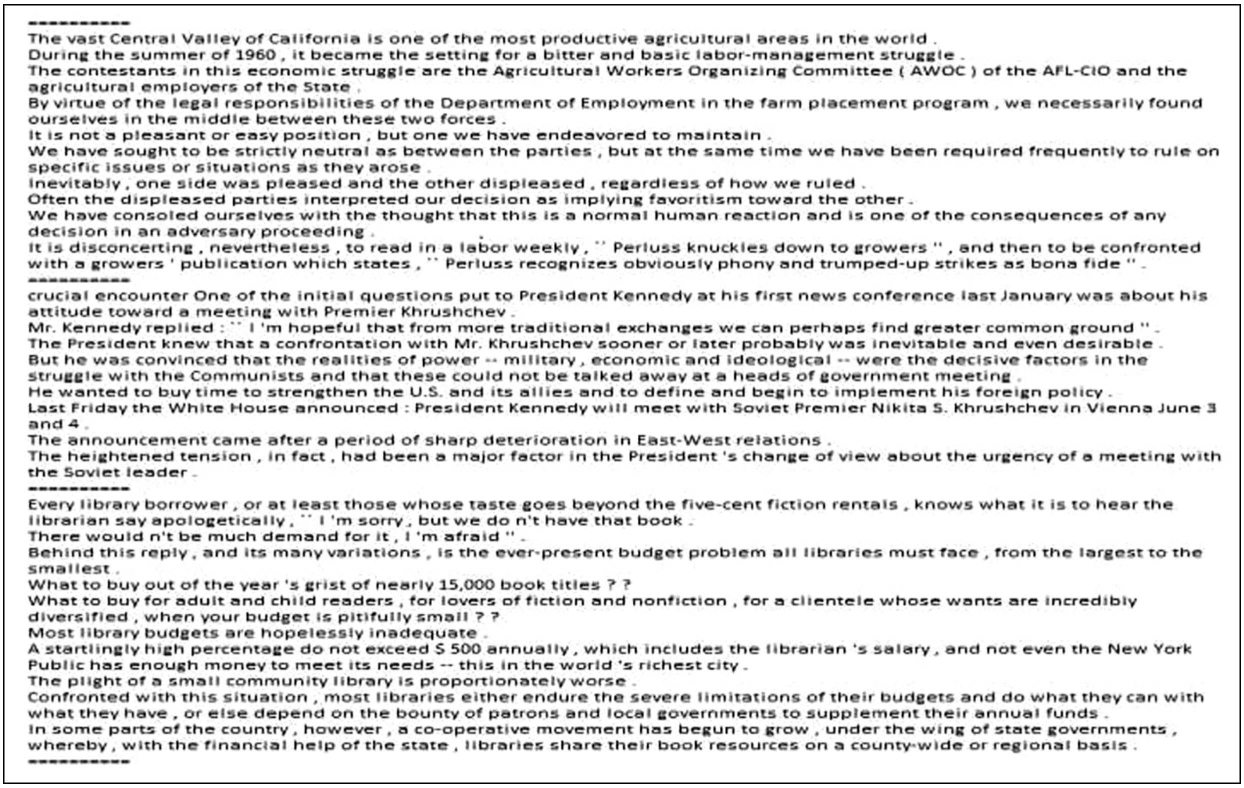
5. Results and analysis

5.1. Text segmentation

Efficacy of TS (LDA-onto) is assessed using Choi's dataset [50], which is the most popular in IR and NLP domain. The Choi's dataset is employed in this experiment to test whether a segmentation algorithm could exactly find natural topic boundaries. Choi's dataset includes several documents in the shape of segments. Each document contains 10 text segments and each segmented fragment is constituting of the first n sentences chosen from documents in the Brown Corpus. The Choi's dataset is divided into four subsets (namely 3–5, 6–8, 9–11 and 3–11) that specify the size of each segment. There are four subsets constituting of 700 documents; each of first three subsets contains 100 documents and final subset contains 400 documents. The details of Choi's dataset are given in Table 2. Each document is represented into a vector space model that represents text blocks as paragraphs. Each paragraph in the ontological vector space is

Table 2. Detail of Choi's dataset.

Segment size	3–5	6–8	9–11	3–11
Number of samples	100	100	100	400



```
----------
The vast Central Valley of California is one of the most productive agricultural areas in the world .
During the summer of 1960 , it became the setting for a bitter and basic labor-management struggle .
The contestants in this economic struggle are the Agricultural Workers Organizing Committee ( AWOC ) of the AFL-CIO and the
agricultural employers of the State .
By virtue of the legal responsibilities of the Department of Employment in the farm placement program , we necessarily found
ourselves in the middle between these two forces .
It is not a pleasant or easy position , but one we have endeavored to maintain .
We have sought to be strictly neutral as between the parties , but at the same time we have been required frequently to rule on
specific issues or situations as they arose .
Inevitably , one side was pleased and the other displeased , regardless of how we ruled .
Often the displeased parties interpreted our decision as implying favoritism toward the other .
We have consoled ourselves with the thought that this is a normal human reaction and is one of the consequences of any
decision in an adversary proceeding .
It is disconcerting , nevertheless , to read in a labor weekly , `` Perluss knuckles down to growers '' , and then to be confronted
with a growers ' publication which states , `` Perluss recognizes obviously phony and trumped-up strikes as bona fide '' .
----------
crucial encounter One of the initial questions put to President Kennedy at his first news conference last January was about his
attitude toward a meeting with Premier Khrushchev .
Mr. Kennedy replied : `` I 'm hopeful that from more traditional exchanges we can perhaps find greater common ground '' .
The President knew that a confrontation with Mr. Khrushchev sooner or later probably was inevitable and even desirable .
But he was convinced that the realities of power -- military , economic and ideological -- were the decisive factors in the
struggle with the Communists and that these could not be talked away at a heads of government meeting .
He wanted to buy time to strengthen the U.S. and its allies and to define and begin to implement his foreign policy .
Last Friday the White House announced : President Kennedy will meet with Soviet Premier Nikita S. Khrushchev in Vienna June 3
and 4 .
The announcement came after a period of sharp deterioration in East-West relations .
The heightened tension , in fact , had been a major factor in the President 's change of view about the urgency of a meeting with
the Soviet leader .
----------
Every library borrower , or at least those whose taste goes beyond the five-cent fiction rentals , knows what it is to hear the
librarian say apologetically , `` I 'm sorry , but we do n't have that book .
There would n't be much demand for it , I 'm afraid '' .
Behind this reply , and its many variations , is the ever-present budget problem all libraries must face , from the largest to the
smallest .
What to buy out of the year 's grist of nearly 15,000 book titles ? ?
What to buy for adult and child readers , for lovers of fiction and nonfiction , for a clientele whose wants are incredibly
diversified , when your budget is pitifully small ? ?
Most library budgets are hopelessly inadequate .
A startlingly high percentage do not exceed $ 500 annually , which includes the librarian 's salary , and not even the New York
Public has enough money to meet its needs -- this in the world 's richest city .
The plight of a small community library is proportionately worse .
Confronted with this situation , most libraries either endure the severe limitations of their budgets and do what they can with
what they have , or else depend on the bounty of patrons and local governments to supplement their annual funds .
In some parts of the country , however , a co-operative movement has begun to grow , under the wing of state governments ,
whereby , with the financial help of the state , libraries share their book resources on a county-wide or regional basis .
----------
```

Figure 2. Three successive extracted segments of 3–11 subset.

assumed as vector, that is, signifies a group of Wikipedia classes. Each group associates with an entity taken from the paragraphs. We then estimate the ontological similarity between these paragraphs vectors is computed based on the group of classes. Distance between two subsequent sub-documents is derived using equation (16). For each sub-document, *sdtf-isdf* and sentiment score of each word is also computed using equation (10) between each vector and its closest vectors. The resultant similarity rating between two sub-documents is computed by merging their ontological, *sdtf-isdf* and sentiment similarity score as derived in equation (12). For each collection of paragraphs, a sentence is combined with the one most similar to it from the remaining sentences. For example, two sentences 1 and 2 can be combined when the similarity rating of sentences 2 and 3 and 1 and 3 is lower. We focused on paragraphs as the elementary blocks in LDA-onto that assures the proximity test using the paragraphs than sentences, which affirms that enough lexical information is obtained as recommended by [33]. Each sub-document is assigned to an ontological vector, although the next sub-document is manipulated constantly in the successive iteration of the algorithm. This whole process is repeated until all sub-documents are assigned to their respective clusters and quality of the produced segments is achieved using ontological similarity. In addition, effect of training the LDA model through Wikipedia corpus using different sizes of datasets is investigated as well. However, size of the segmented fragment (i.e. sub-document) is an important and a critical phase in the segmentation process. As Hearst [26] computed similarity between text blocks, each text block consists of cumbersome number of sentences in fixed shape (i.e. window). Therefore, we applied same approach but sub-documents in LDA-onto are estimated by varying window to different sizes: 1, 2, 4 and 5 sentences. Figure 2 shows the three successive segments from the 3–11 subset.

Text segmentation is performed using LDA-onto and it is evaluated through P_k and WindowDiff (WD) error metric as derived in equation (20) and (21). Table 3 shows the results of varying window sizes of 1, 2, 4 and 5 sentences per text block. Error rates of all the subsets ranged from 0.71 to 3.50 indicate that segmentation using ontological connection among its constituents is convenient with reduced error rates in particular for window size 2 and 4. Moreover, changing the window size could improve the quality of the text segmentation. However, subsets are less improved in terms of error rates: subset (‘3–5’ and ‘3–11’) and (‘6–8’ and ‘3–11’) resembles higher and lower error rates, respectively. Results illustrated substantial abatement in error scores when segment size was 3–5 and 3–11 using window (w = 5 and w = 1),

Table 3. Performances of LDA-onto on Choi's dataset.

Window	Segment size							
	3–5		6–8		9–11		3–11	
	P_k	WD	P_k	WD	P_k	WD	P_k	WD
w = 1	3.50	3.97	2.65	2.79	2.61	3.13	2.58	3.25
w = 2	1.56	1.62	0.95	1.11	0.71	0.86	0.89	1.05
w = 4	2.96	3.16	1.12	1.27	0.92	1.15	0.96	1.23
w = 5	3.15	3.39	1.61	2.10	3.81	4.24	3.77	3.95

LDA: latent Dirichlet allocation; WD: WindowDiff.

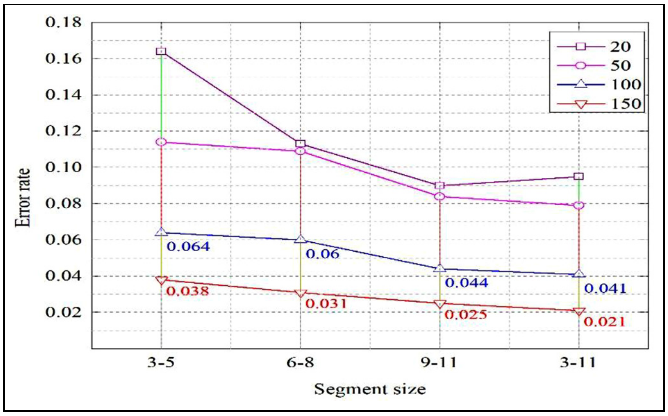


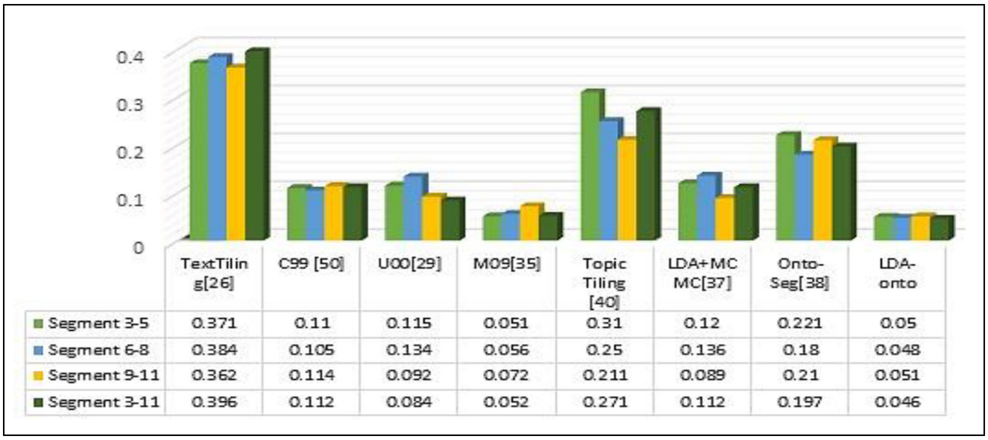
Figure 3. Performance of LDA-onto using Wikipedia ontology.

respectively. It can also be observed that larger segments lead better results when using w = 1 excluding small ranging sentences in subset 3–5.

Intuitively, it is obvious that efficiency of segmentation depends upon reference segments' length; as this outcome confirms to the achievability of proposed LDA-onto. This owes to, larger segments are exceptional, overlapped and conceptual connections are better among topics than smaller as discussed earlier. However, words are eliminated when training process is not counted in LDA-onto. As a consequence, a discrepancy in the vocabulary between training and testing data is appeared. This inflicted inferior segmentation performances due to huge amount of words in the testing phase are discarded and drastic reduction of information is generated. To overcome this issue, LDA-onto is trained through four Wikipedia corpus of varying sizes of 150, 100, 50, 20 entries. This validates LDA-onto's effectuality to avoid discrepancy of vocabulary. In doing so, we stick with only P_k error metric as our evaluation criteria, whereas standard parameters were taken as (1) Dirichlet priors (α = 1 and β = .01), and (2) number of topics and iterations set to 200 and 300, respectively. Figure 3 shows improved performances when the length of segment size changes from smaller (3–5) to larger (3–11). Results exhibited reliable estimation process of topic distribution through LDA-onto and showed improved segmentation performances when training corpus is large (i.e. 150 and 100).

5.2. LDA performances comparison against traditional text segmentation

LDA-onto is compared versus existing methods as shown in Figure 4, which indicates improved results performed on the benchmarked Choi's dataset. Results indicated that LDA-onto is a novel method in TS and compared against existing state-of-the-art methods. In contrast, LDA-onto is very robust and relies on lexical characteristics information. This implies that LDA-onto lends itself adequately into well-known NLP tasks for segmentation, such that results in terms of P_k values in comparison of traditional TS methods were found lower scores in different segment sizes. However, TextTiling and TopicTiling achieved very poor performances than other methods. It is noted that part of Choi's dataset is set to train LDA-onto model in order to reduce the problem of vocabulary mismatch. Lower the error rate gives better segmentation performances. In particular, segmentation performances are much improved when ontological similarity is



	TextTilin g[26]	C99 [50]	U00[29]	M09[35]	Topic Tiling [40]	LDA+MC MC[37]	Onto-Seg[38]	LDA-onto
🟩 Segment 3-5	0.371	0.11	0.115	0.051	0.31	0.12	0.221	0.05
🟦 Segment 6-8	0.384	0.105	0.134	0.056	0.25	0.136	0.18	0.048
🟨 Segment 9-11	0.362	0.114	0.092	0.072	0.211	0.089	0.21	0.051
■ Segment 3-11	0.396	0.112	0.084	0.052	0.271	0.112	0.197	0.046

Figure 4. Comparison of results of traditional segmentation approaches versus LDA-onto.

Table 4. The details of characteristics of dataset DS (1–5).

Dataset	Source	#docs	#topic labels	#terms	#docs per topic	#sub-docs per doc
DS1	Classic (CACM/CISI/CRANFIELD/MEDLINE)	5649	4	11,219	1412.2	2.3
DS2	RCV1 (Reuters Corpus Volume 1)	6165	23	36,147	278.8	2.1
DS3	K1b, webkb (webace, Web Knowledge Base)	7440	13	33,421	572.3	6.2
DS4	Ohscal (OHSUMED-233445)	8639	10	10,872	863.9	4.3
DS5	20Newsgroups	5400	20	24,641	281.8	5.3

DS: dataset.

combined with lexical distance, and thus, LDA-onto achieved slight improvement over M09. LDA-onto uses the topic association with several inference steps. This enables LDA-onto to stabilise the topic assignments using improvised ontological and semantic similarity, that is, scoring rating of sub-documents resemble improved performances. In addition, LDA-onto is linear with the varying number of sentences (in shape of paragraphs) to find natural topical boundaries. Unlike with lexical-based TS, LDA-onto could also locate sub-topical changes with larger segment size and thus, indicated low error rate than the other methods in the segment size (6–8, 9–11 and 3–11).

5.3. Dataset

ECA is evaluated with real-time cross-domain datasets derived from several databases denoted by DS (1–5). Datasets are retrieved from four different text databases. Researchers in IR and NLP domains are well recognised to these databases, such as DS1 is retrieved from CACM, CISI, CRANFIELD and MEDLINE abstracts available in Classic text database [51]. DS2 contains 6165 out of 21,850 documents are selected from the Reuters Corpus Volume 1 (RCV1) [52] after filtering short structured news due to less number of topics per document. DS3 is retrieved from both k1b and webkb, which contains documents correspond to webpages. In addition, k1b is derived from WebACE project that contains webpages in Subject hierarchy of Yahoo! (http://www.yahoo.com). Webkb (Web Knowledge Base Project) contains webpages across different universities [53]. DS4 is retrieved from Text REtrieval Conference (TREC) collections (http://trec.nist.gov) [54] and it is also available in CLUTO toolkit. DS5 is a set of approximately 5400 documents out of 18,800 are partitioned in 20 Newsgroups, where each newsgroup corresponds to different topics. Moreover, some criterions in the experiments with some form of restrictions are proclaimed as a pre-requisite for clustering. For instance, a text unit is supposed to contain minimum of one paragraph of few sentences, wherein sentences must exhibit the topics that are double the number of associated topics. Pre-processing steps are applied to each dataset like stop-words, removing strings of digits and words stemming. Table 4 summarises the characteristics of each dataset used in this study. In order to identify boundaries and extract the segments, two methods for text segmentation are adopted: (1) LDA-onto is used to segment the documents in the ECA, (2) TextTiling algorithm [26], which is used for segment-based clustering.

In TextTiling, sub-documents are collected using non-contiguous parts of document and they are not assumed to have inter-link connection related to respective topics. In addition, it decomposes a text into different portions in the shape of contiguous blocks (passages and subtopics). Each block finds boundaries in documents correspond to topics using terms and words. The identified patterns of lexical co-occurrences and scattered diffusion in the text units among contiguous blocks are measured using the dot-product in the vector space.

5.4. Parameter settings in segmentation

In order to assess the quality of segmentation of DS (1–5), we selected segmentation methods whose performance was better on Choi's dataset, such as U00, C99, M09 and LDA-onto. We performed LDA-onto segmentation to extract sub-documents across different text units in accordance with parameters mentioned in Section 5.2. A dataset is divided into its coherent subsets that consist of sentences in a sub-document: For instance, a subset 'M-N', a sub-document is generated randomly using a sequence of sentences from corpus of a story, wherein first S sentences are selected (an integer S represents a number between *M* and *N* and is chosen from the story). In contrast, sentence end' information in a sentence is to be E node for each sub-document in a document as suggested by existing researches [35, 49]. A subset contains number of documents, where each dataset is divided into four subsets. We chose the subsets ('S1–S3', 'S4–S7' and 'S8–S12') are partitioned based on the number of sentences in a sub-document. The fourth subset ('S1–S12') contains the whole dataset. Each subset (excluding forth) is associated with sub-documents per document such as 20, 40 and 80, respectively. In order to maintain the efficacy of TS and to avoid discrepancy of vocabulary, training set for datasets (DS1–5) are set to 98, 97, 95, 90, and 82 K, respectively. The word token for DS (1–5) ranged from 2.0 to 20M. Standard parameters are taken as (1) Dirichlet priors (α = 50/T and β = .01) and (2) number of topics and iterations were set to 40 and 200, respectively. When training the LDA-onto model, number of topics (*t*) chosen is 40. Computational cost of the segmentation algorithm is moderated as *t* increases; the computational cost is also increased.

5.5. Parameter settings in text clustering

There are two kinds of clustering solutions laid in ECA to find sub-documents related to similar classes in each text unit [25]. First, a text unit includes same number of clusters as to classes, which is known as *h*-way clustering solution. Second, number of clusters is twice multiplied with number of classes, known as h^2-way clustering solutions. *F*-measure is featured to assess the quality of clustering in terms of precision and recall. Given a collection **D** of documents, clustering solutions $C = \{C_1, \ldots, C_h\}$ and $c = \{c_1, \ldots, c_k\}$, where C is the classification of document **D** and c is clustering across **D**. Assuming, C_i, c_j and P_{ij} is denoted by $P_{ij} = |C_j \cap c_i|/|C_j|$. Similarly, C_i, c_j and R_{ij} is denoted by $R_{ij} = |C_j \cap c_j|/|c_j|$. The *F*-measure is measured by harmonic means using both precision and recall in order to compute the quality of C with c. The *F*-measure is computed using macro-average F^M and micro-average F^{μ}. Macro *F*-measure considers equal weight to each class, whereas micro *F*-measure considers equal weight for each sub-document in a text unit. F^M is aimed to assess the quality of clustering, which concludes the main analysis of the experimental results. In order to compute micro-average (F^{μ}), macro-average is computed as [25]

$$F^M = \frac{2PR}{(P+R)} \qquad [(22)$$

and

$$P = \frac{1}{h}\sum_{i=1}^{h} P_i \qquad [(23)$$

$$R = \frac{1}{h}\sum R_i \qquad [(24)$$

where $i = 1 \ldots h$, $P_i = P_{ij'}$ and $R = R_{ij'}$. Here, $j' = argmax_{j=1\ldots k}\{P_{ij}, R_{ij}\}$. The micro-average is then computed as

$$F^{\mu} = \sum_{i=1} \left[\frac{|c_i|}{|\boldsymbol{D}|}\right] \max_{j=1\ldots k} F_{ij} \qquad [(25)$$

where $F_{ij} = 2 * P_{ij}R_{ij}/(P_{ij} + R_{ij})$. However, clustering' performances are relied at random initialization of different parameters such as number of iterations and parameter values, as these parameter values may vary over algorithm.

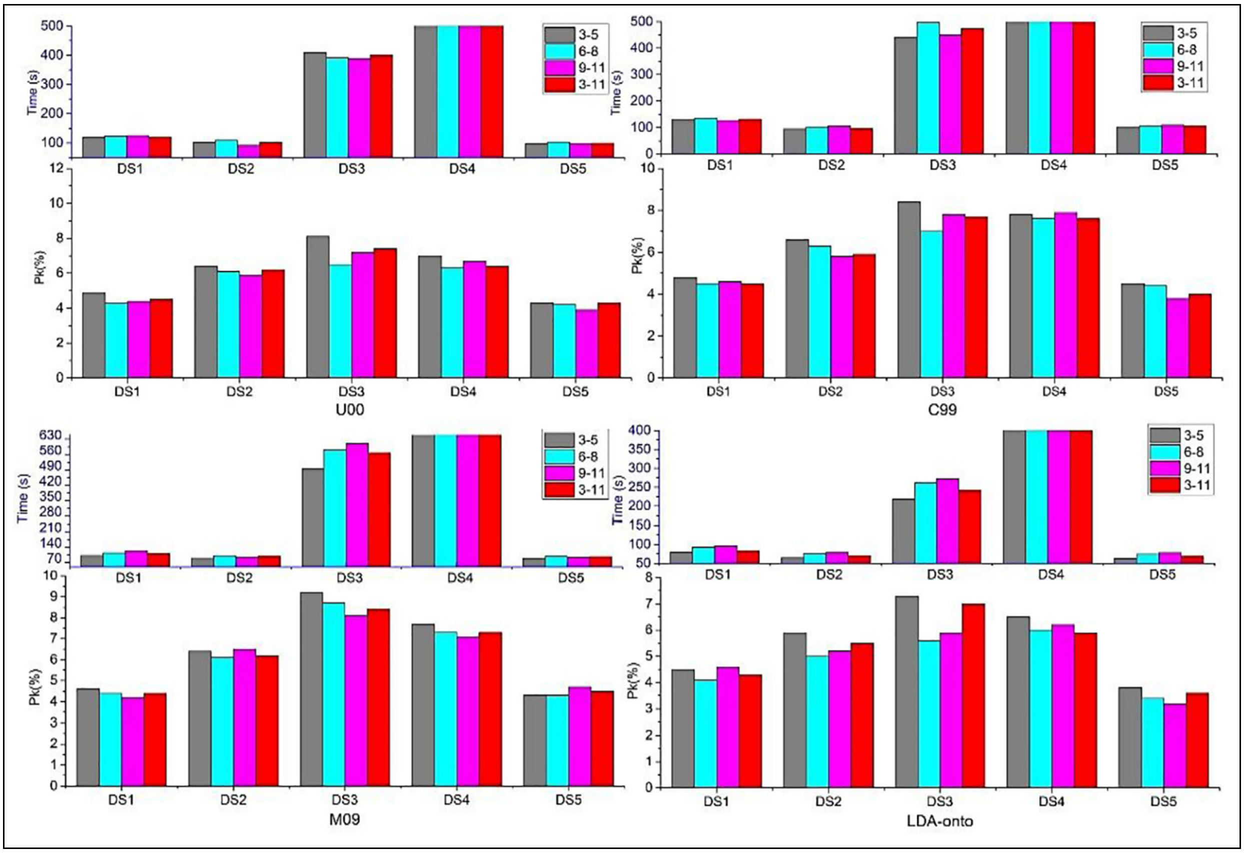


Figure 5. Results performances of four segmentation methods (when sub-document per document = 20).

Overlapping clustering (LDA and bisecting) algorithm' probability threshold value may vary from 5.00E–05 to 0.30. However, each dataset tends to contain different ranges of F-measure scores are normalised using maximum F-measure. In case of disjoint clustering, algorithms (Sk-M and Bisecting Sk-M) parameters, value is computed and examined over the maximum averaged micro F-measures across the dataset.

5.6. Performances of segmentation in real-time dataset

LDA-onto is experimented on more realistic and domain-independent datasets (DS1–5) to assess the wider spectrum of the segmentation problem. Each dataset has separate train set so that it must contain sufficient train documents with minimum vocabulary size and reliable parameters estimation. Figures 5–7 show the results performances on dataset DS1–5 in terms of P_k for segmentation methods such as U00, C99, M09 and LDA-onto, respectively. The lower value of P_k is taken as an indication of better performance. During training, we stick to choose half of the documents for each dataset in order to find the optimal parameter settings. However, we varied the training size, in particular, for dataset (DS3 and DS4) due to large number of sub-documents associated with the original document. Results showed in Figures 5–7 correspond to an average number of sub-documents per document such as 20, 40 and 80, respectively. When the sub-documents per document were equal to 20, error rate (3.6) and time (68 in seconds) conceded for DS5 using LDA-onto was much improved than U00, C99 and M09 for DS-1, as shown in Figure 5.

The overall results indicated that performances of LDA-onto consistently improved, leveraging the segment size (3–5, 6–8–9–11 and 3–11). Moreover, results achieved on DS (1, 2 and 5) were given best segmentation performances and conceded less time than DS (3 and 4). This implies better estimation of parameters could achieve better results with longer segments size as mentioned in Section 5.1. Figures 5–7 depicted that 3–5 segment size obtained higher error rates in overall results performances. Result indicates that LDA-onto produced better performances against U00, C99 and M09 with number of observations such as (1) underlying consideration of topic and words distribution in each dataset showed better segmentation when compared with results versus other methods, (2) traditional TS (U00, C99) were outperformed and (3) topic-based TS (M09) is less robust, whereas LDA-onto is more accurate and constant in error rate.

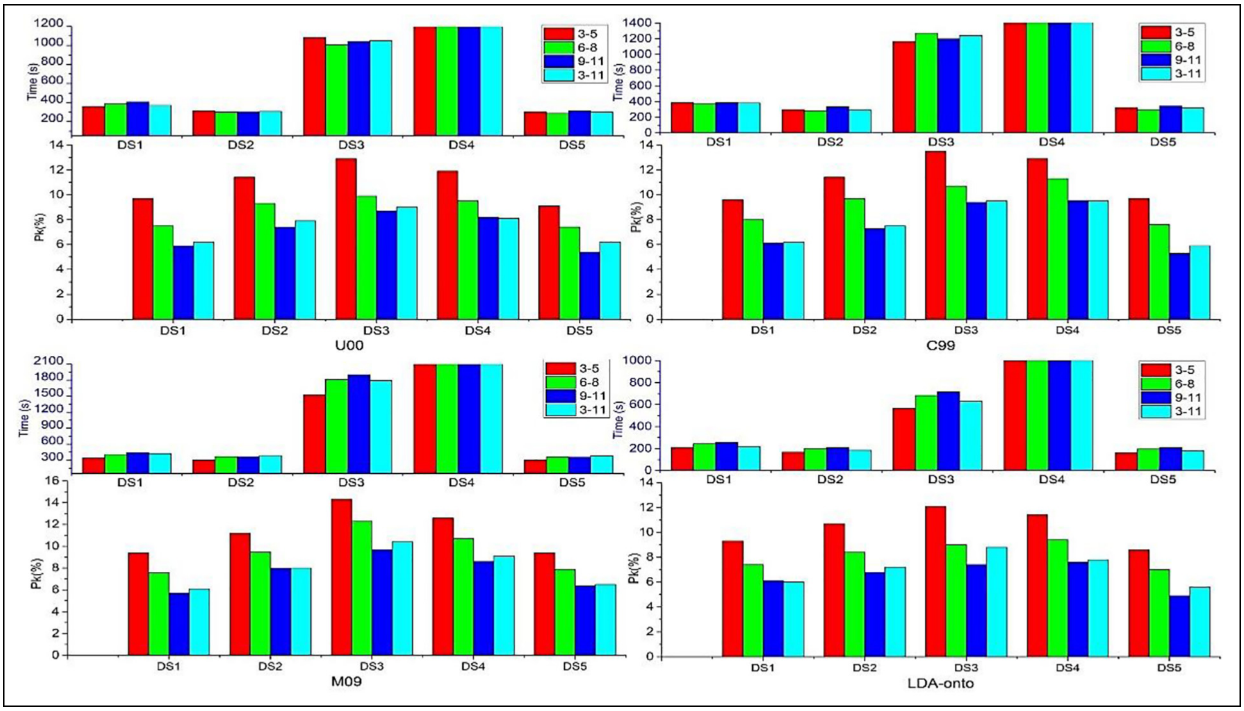


Figure 6. Results performances of four segmentation methods (when sub-document per document = 40).

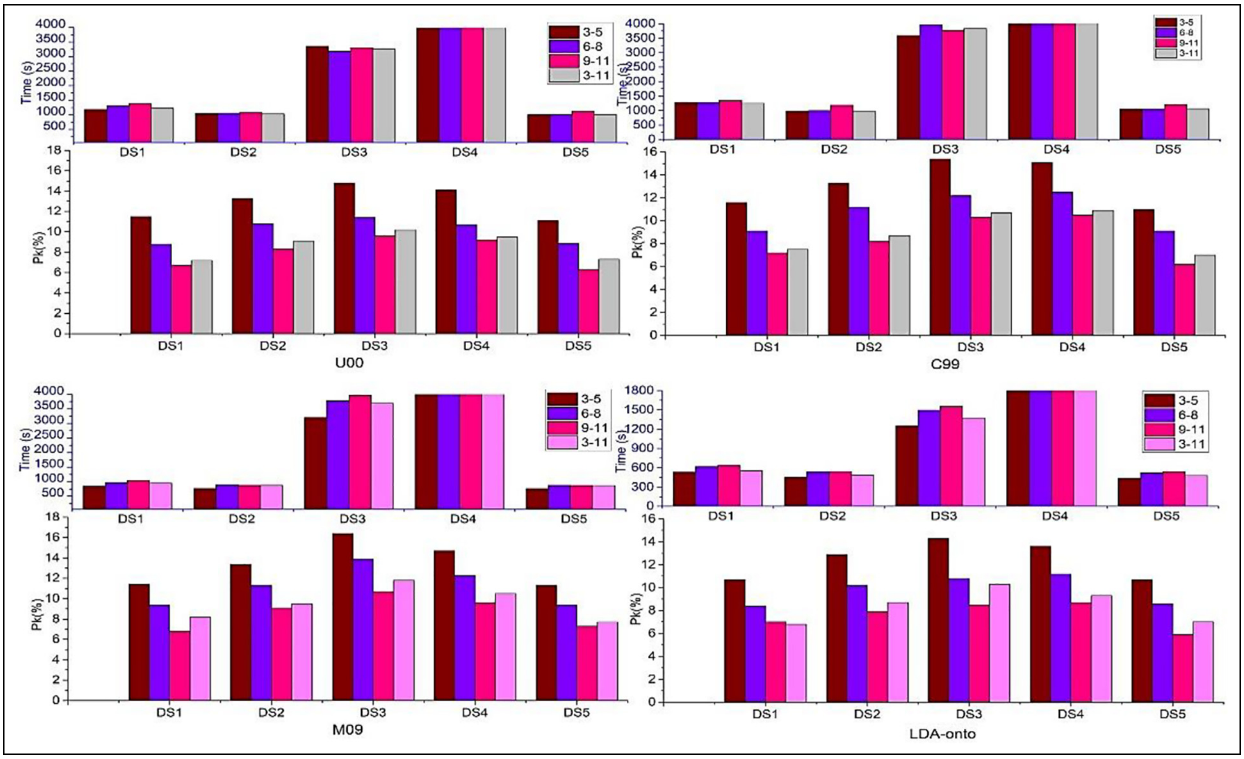


Figure 7. Results performances of four segmentation methods (when sub-document per document = 80).

5.7. ECA against segment-based clustering

The characteristics of ECA are described into six different clustering methods categorised either within a document or across document. These six clustering methods (produce overlapping and disjoint clustering solutions) are taken into consideration in the experiments, and they compared against both the traditional (partitional clustering) and segment-based clustering. Performances are carried using the algorithms namely, Sk-M, LDA and Bisecting (Sk-M, LDA). In addition, clustering method composed of disjoint (within sub-document) and overlapped (across sub-document) clustering in order to assess intra and inter clustering similarity, respectively. This was the necessary step to estimate clustering in the perspective of qualitative and quantitative aspect (such as derived number of clusters and their mapping to original text units). To do so, sub-documents are extracted in preliminary step, which is performed using LDA-onto (see Section 5.6). In segment-based clustering, TextTiling is used for extracting the segments. Results in Tables 5 and 6 report the performances of segment-based and ECA, respectively. These performances are shown in terms of *F*-measure, precision and recall. The overall results suggested non-overlapping solutions (through Sk-M and Bisecting Sk-M) obtained poor performances and may not be considerable for clustering multi-topic documents. Hence, this verifies that Sk-M and bisecting Sk-M algorithms need further improvements to cluster multi-topic documents in ECA. Results produced overlapping solutions (through LDA and bisecting LDA) indicated ECA outperformed against segment-based approach. In addition, highest *F*-measure is obtained with an average improvement of over 13.2% in the dataset (DS1, 2 and 5). An average improvement of 11.7% is obtained through each clustering method in DS (1–5). Precision and recall are much better with an average improvement of over 54%. The inter-comparison and the differences obtained through recall and precision values indicate better improvements in each dataset (DS1–5) using ECA. This implies that sub-documents containing low substantial recall and high precision values obtained best topics (i.e. clusters). Unlike, Table 5 reports low recall and low precision values resulting each cluster that may not relate or associate with a specific topic. However, higher precision and low recall achieved in DS (1, 2, 5), indicated that ECA allocates documents properly within the clusters. In contrast, DS (3 and 4) involved cumbersome number of documents along with the larger length as well, and thus, clusters contain huge number of documents that render large amount of proportions of sub-documents. An average number of extracted sub-documents per document in DS (3, 4) were perceived 6.2 and 4.3, respectively.

5.8. Clustering assessment

The assessment of clustering performances is conducted to account clustering algorithm' efficiency using both ECA and segment-based clustering in terms of time, memory and number of h-way clustering solutions. Figure 8 shows the runtime conceded for DS (1–5) in each clustering algorithm using segment-based approach is greater than ECA. Results also indicated that clustering algorithms using ECA were efficient with respect to time and gained an average improvement of 20% over segment-based clustering. This is due to cumbersome number of sub-documents in original text units. In Figure 9, numbers of clusters (20-, 40- and 80-way) in ECA conceded less time than segment based in dataset DS (1–5). This concluded that ECA is suitable and justifiable when compared with traditional based clustering as well. Our findings give us credence to the claim made here that ECA is much improved and efficient for managing multi-topic documents. Moreover, all the algorithms implemented using Java 1.6 and CLUTO kit, as they were performed on Windows OS 64-bit platform with a 2.8 GHz CPU and 8 GB memory. However, both the ECA and segment-based clustering inflicted high computational cost in case of disjoint clustering only. For this reason, memory consumption of Bisecting LDA and the LDA is shown in Figure 10. In doing so, we used a process-monitoring tool in order to capture memory. Results showed the memory consumption in datasets DS (1–5) using ECA, which consumed less time in comparison to segment-based clustering. In segment based, topic distribution is not associated with each sub-document and its boundary detection is based on the terms occurred in number of adjacent blocks. Unlike, ECA's boundary detection is based on ontological and semantic word score (i.e. word and topics), and thus, it is an efficient clustering approach for multi-topic documents.

5.9. Clustering performances over varying topics

As discussed in Section 5.4, number of topics in ECA is an important factor, which affects the clustering performances and likelihood as well when the number of topics (*T*) varies. We tried different numbers of values of *T* ranged from 5 to 40 using Bisecting LDA. However, the value with optimal likelihood might not resemble better clustering solutions. Actually, value of *T* using Bisecting LDA is distinct and varies as per data set. Thereby, the impact of topic varies with respect to DS (1–5) on clustering performances is limited. The results in Figure 11 show *F*-measure of varying numbers of topics in DS (1–5). Evidently, impact of varying number of topics *T* is minor and sustainable in the proposed ECA.

Table 5. Clustering performances of segment-based clustering using disjointed and overlapping solutions.

Clustering method	Clustering algorithm	Dataset														
		DS1			DS2			DS3			DS4			DS5		
		P	*R*	F^M	*P*	*R*	F^M	*P*	*R*	F^M	*P*	*R*	F^M	*P*	*R*	F^M
Non-overlapping document	Sk-M	0.602	0.164	0.275	0.721	0.397	0.531	0.46	0.247	0.337	0.658	0.137	0.244	0.689	0.357	0.489
	LDA	0.184	0.657	0.283	0.476	0.658	0.552	0.265	0.637	0.376	0.386	0.527	0.446	0.447	0.621	0.52
	BS(Sk-M)	0.567	0.521	0.565	0.738	0.414	0.56	0.604	0.293	0.411	0.737	0.137	0.243	0.694	0.372	0.506
	BS(LDA)	0.573	0.642	0.601	0.53	0.702	0.598	0.391	0.622	0.473	0.519	0.592	0.549	0.489	0.67	0.559
Non-overlapping sub-document	Sk-M	0.622	0.224	0.347	0.66	0.517	0.599	0.48	0.307	0.387	0.678	0.197	0.323	0.621	0.486	0.564
	LDA	0.224	0.688	0.335	0.662	0.561	0.607	0.305	0.667	0.416	0.426	0.558	0.483	0.628	0.551	0.587
	BS(Sk-M)	0.597	0.531	0.584	0.665	0.613	0.693	0.634	0.303	0.431	0.767	0.147	0.259	0.627	0.5	0.581
	BS(LDA)	0.593	0.677	0.628	0.675	0.59	0.627	0.411	0.652	0.493	0.539	0.627	0.575	0.637	0.511	0.564
Overlapping sub-document	Sk-M	0.653	0.234	0.363	0.62	0.535	0.593	0.51	0.317	0.407	0.709	0.207	0.338	0.586	0.567	0.595
	LDA	0.268	0.727	0.389	0.616	0.634	0.625	0.355	0.707	0.466	0.47	0.597	0.526	0.593	0.599	0.596
	BS(Sk-M)	0.577	0.601	0.611	0.626	0.548	0.61	0.614	0.373	0.481	0.747	0.217	0.351	0.592	0.577	0.625
	BS(LDA)	0.624	0.762	0.681	0.561	0.681	0.61	0.441	0.742	0.543	0.57	0.712	0.628	0.527	0.655	0.579
Overlapping document	Sk-M	0.641	0.223	0.349	0.82	0.378	0.536	0.5	0.307	0.397	0.697	0.196	0.323	0.776	0.341	0.492
	LDA	0.256	0.706	0.373	0.483	0.81	0.605	0.335	0.687	0.456	0.458	0.576	0.51	0.445	0.762	0.561
	BS(Sk-M)	0.561	0.551	0.578	0.816	0.484	0.68	0.594	0.323	0.441	0.731	0.167	0.285	0.779	0.352	0.502
	BS(LDA)	0.573	0.741	0.641	0.63	0.659	0.64	0.391	0.722	0.493	0.519	0.691	0.587	0.588	0.602	0.591
Overlapping sub-document set	Sk-M	0.664	0.264	0.396	0.712	0.438	0.561	0.52	0.347	0.437	0.72	0.237	0.374	0.668	0.459	0.563
	LDA	0.332	0.757	0.46	0.67	0.473	0.554	0.415	0.737	0.526	0.534	0.627	0.577	0.647	0.436	0.52
	BS(Sk-M)	0.597	0.571	0.606	0.71	0.455	0.573	0.634	0.343	0.461	0.767	0.187	0.314	0.67	0.474	0.574
	BS(LDA)	0.644	0.772	0.698	0.72	0.623	0.665	0.461	0.752	0.563	0.59	0.722	0.644	0.681	0.592	0.63
Overlapping sub-document and sub-document set	Sk-M	0.685	0.274	0.41	0.664	0.686	0.693	0.54	0.357	0.447	0.741	0.247	0.388	0.636	0.654	0.611
	LDA	0.364	0.777	0.494	0.604	0.698	0.648	0.445	0.757	0.566	0.566	0.647	0.604	0.578	0.669	0.614
	BS(Sk-M)	0.667	0.551	0.625	0.664	0.701	0.69	0.704	0.323	0.461	0.837	0.167	0.291	0.637	0.669	0.649
	BS(LDA)	0.656	0.812	0.698	0.643	0.827	0.695	0.471	0.792	0.583	0.602	0.762	0.647	0.617	0.723	0.661

DS: dataset; LDA: latent Dirichlet allocation.

Table 6. Clustering performances of ensemble clustering approach (ECA) using disjointed and overlapping solutions.

Clustering method	Clustering algorithm	Dataset														
		DS1			DS2			DS3			DS4			DS5		
		P	*R*	F^M	*P*	*R*	F^M	*P*	*R*	F^M	*P*	*R*	F^M	*P*	*R*	F^M
Non-overlapping document	Sk-M	0.493	0.235	0.388	0.518	0.445	0.582	0.352	0.245	0.374	0.518	0.155	0.294	0.477	0.395	0.532
	LDA	0.269	0.692	0.415	0.511	0.597	0.601	0.364	0.646	0.503	0.52	0.552	0.588	0.461	0.598	0.567
	BS(Sk-M)	0.541	0.527	0.6	0.696	0.481	0.643	0.551	0.326	0.469	0.762	0.225	0.401	0.654	0.457	0.629
	BS(LDA)	0.606	0.713	0.709	0.541	0.649	0.643	0.486	0.646	0.605	0.675	0.569	0.68	0.504	0.612	0.605
Non-overlapping sub-document	Sk-M	0.513	0.295	0.453	0.439	0.575	0.633	0.372	0.305	0.431	0.538	0.215	0.371	0.442	0.479	0.579
	LDA	0.309	0.723	0.463	0.722	0.546	0.684	0.404	0.677	0.545	0.56	0.583	0.625	0.714	0.486	0.644
	BS(Sk-M)	0.571	0.537	0.619	0.674	0.695	0.685	0.581	0.336	0.484	0.792	0.235	0.416	0.61	0.516	0.644
	BS(LDA)	0.626	0.748	0.735	0.705	0.545	0.68	0.506	0.681	0.63	0.695	0.604	0.708	0.679	0.461	0.618
Overlapping sub-document	Sk-M	0.544	0.305	0.468	0.398	0.56	0.606	0.403	0.315	0.447	0.569	0.225	0.386	0.37	0.592	0.607
	LDA	0.353	0.762	0.515	0.725	0.577	0.704	0.448	0.716	0.592	0.604	0.622	0.667	0.681	0.547	0.668
	BS(Sk-M)	0.551	0.607	0.646	0.584	0.609	0.671	0.561	0.406	0.533	0.772	0.305	0.492	0.654	0.526	0.675
	BS(LDA)	0.657	0.833	0.786	0.68	0.611	0.705	0.537	0.766	0.679	0.726	0.689	0.767	0.644	0.57	0.666
Overlapping document	Sk-M	0.532	0.294	0.455	0.559	0.389	0.548	0.391	0.304	0.435	0.557	0.214	0.372	0.53	0.361	0.517
	LDA	0.341	0.741	0.499	0.603	0.755	0.719	0.436	0.695	0.577	0.592	0.601	0.651	0.589	0.69	0.686
	BS(Sk-M)	0.535	0.557	0.613	0.756	0.447	0.62	0.545	0.356	0.491	0.756	0.255	0.435	0.705	0.412	0.593
	BS(LDA)	0.606	0.812	0.744	0.657	0.585	0.68	0.486	0.745	0.634	0.675	0.668	0.73	0.619	0.554	0.646
Overlapping sub-document set	Sk-M	0.555	0.335	0.499	0.458	0.521	0.611	0.4142	0.345	0.475	0.5802	0.255	0.422	0.439	0.493	0.586
	LDA	0.417	0.792	0.582	0.78	0.409	0.623	0.512	0.746	0.661	0.668	0.652	0.715	0.743	0.382	0.641
	BS(Sk-M)	0.571	0.577	0.641	0.649	0.57	0.655	0.581	0.376	0.517	0.792	0.275	0.463	0.645	0.529	0.699
	BS(LDA)	0.677	0.843	0.803	0.82	0.555	0.751	0.557	0.776	0.71	0.746	0.699	0.782	0.778	0.533	0.766
Overlapping sub-document, and sub-document set	Sk-M	0.576	0.345	0.513	0.51	0.683	0.723	0.4352	0.355	0.488	0.6012	0.265	0.435	0.437	0.654	0.672
	LDA	0.449	0.812	0.616	0.673	0.633	0.749	0.5435	0.766	0.706	0.6995	0.672	0.755	0.637	0.614	0.806
	BS(Sk-M)	0.641	0.557	0.661	0.686	0.704	0.766	0.651	0.356	0.518	0.862	0.255	0.447	0.641	0.692	0.729
	BS(LDA)	0.689	0.883	**0.805**	0.7	0.796	**0.841**	0.569	0.816	**0.751**	0.758	0.739	**0.798**	0.693	0.711	**0.875**

DS: dataset; LDA: latent Dirichlet allocation.

Bold values refer to highest macro *F*-measure.

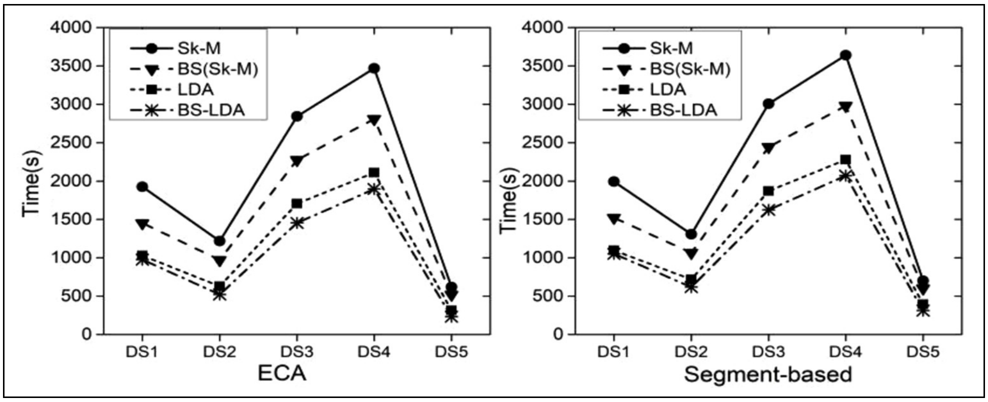


Figure 8. Time cost of ECA and segment-based segmentation in DS (1–5) through four different clustering algorithms.

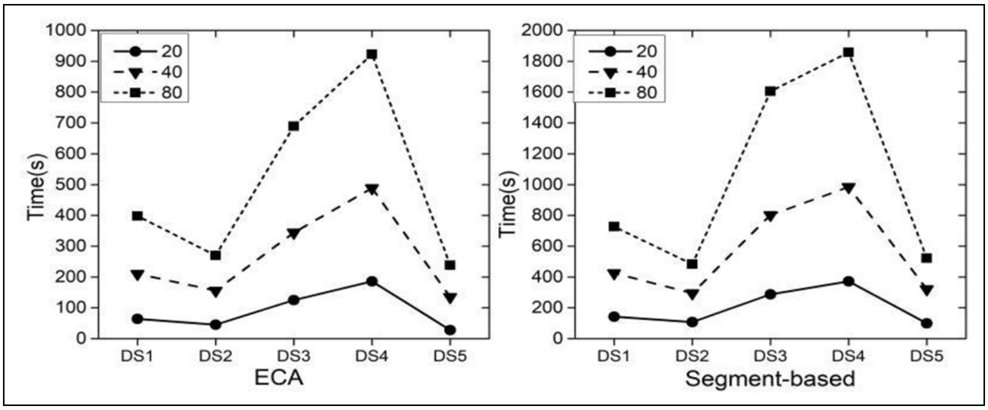


Figure 9. Time cost of ECA and segment-based segmentation in DS (1–5) using 20-, 40- and 80-way clustering solutions.

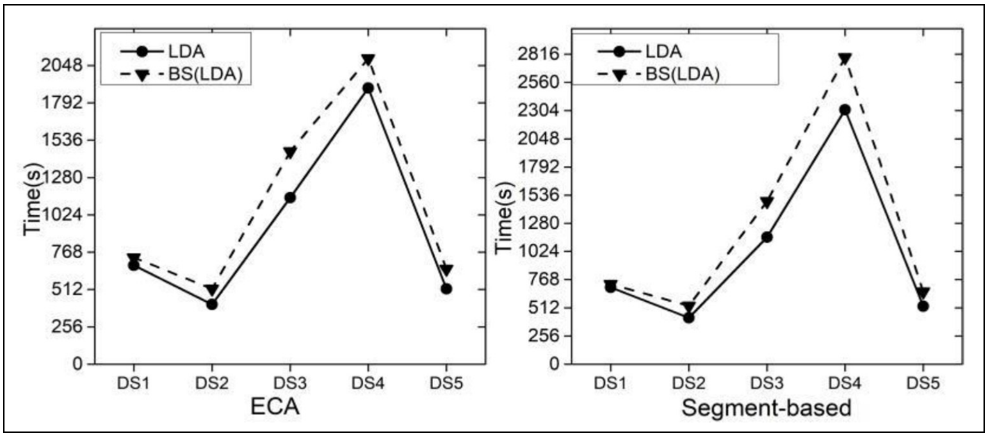


Figure 10. Memory consumption of ECA and segment based in DS (1–5) using LDA and bisecting LDA, respectively.

5.10. Statistical testing of ECA versus segment-based clustering

The statistical significance of ECA versus segment-based clustering is obtained using unequal variances, wherein unpaired *T* test of achieved results is measured with null hypothesis of no difference. As mentioned in Section 3.7, we

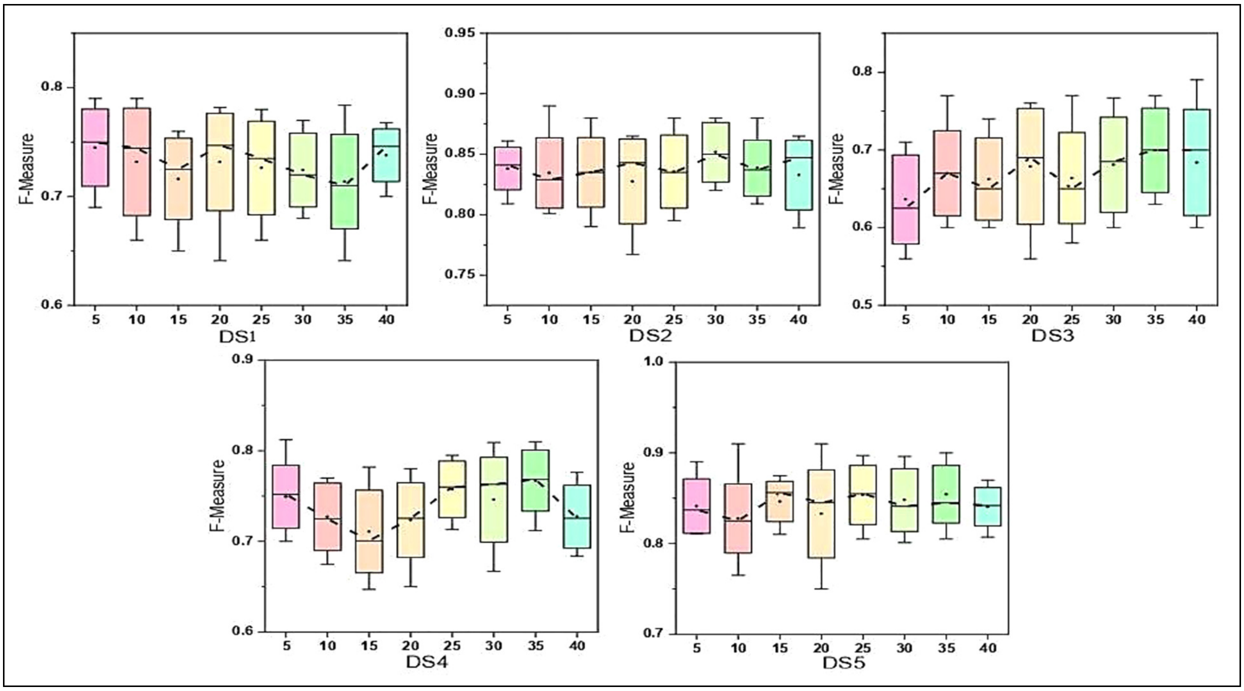


Figure 11. Effect of varying number of topics on clustering performances in terms of *F*-measure in DS (1–5).

Table 7. Statistical significance of achieved results.

Dataset	ECA versus segment-based clustering					
	Non-overlapping document	Non-overlapping sub-document	Overlapping sub-document	Overlapping document	Overlapping sub-document set	Overlapping sub-document, and sub-document set
DS1	2.26E–28	4.26E–36	2.31E–18	4.11E–16	5.89E–12	5.22E–13
DS2	3.16E–18	5.21E–26	3.12E–21	5.10E–12	4.89E–11	4.20E–12
DS3	3.10E–42	4.26E–33	2.40E–15	7.18E–14	2.11E–10	4.15E–14
DS4	3.33E–51	5.10E–60	5.57E–26	3.92E–23	9.11E–19	3.10E–23
DS5	2.17E–38	5.46E–40	3.13E–22	5.32E–16	3.29E–13	5.12E–18

ECA: ensemble clustering approach; DS: dataset.

counted 50 iterations of each clustering algorithm into six clustering methods. Table 7 reports *p* values of *t*-test of each dataset, which reported ECA' *p*-values were low, and the obtained values are associated with *T*-values (two-tailed) with degree of freedom ($df = 98$) at significant level is set to $\alpha = 0.01$. The statistical significance of achieved results using ECA versus segment-based approach reports that null hypothesis is rejected in terms of *F*-measure and validated the evidence that ECA is significant over segment-based clustering.

5.11. Qualitative assessment

The qualitative evaluation of ECA shows the descriptions of the clusters and score of keywords. Clusters are manipulated based on high-intra and low-inter cluster similarity. Table 8 reports the score of words of various sub-documents in DS (1–5) was obtained using equation (10) for only four documents d (1–4), and score of sub-document itself was obtained using equation (12). Description of top four clusters is provided in each dataset in Table 9. Results indicate that topical terms are represented by the list of keywords and their evaluation is done using macro *F*-measure. These terms in a document are related to different categories (i.e. domain-dependent), such that their proper nouns could discriminate the illustration of clusters through keyword score in their sub-document as reported in Table 8. It can be observed that keywords

Table 8. Word score in respective sub-document for four documents in DS (1–5).

doc-id.	DS1		DS2		DS3		DS4		DS5	
	Sub-doc.-id	Keyword (score)	Sub-doc.-id	Keyword (score)	Sub-doc.-id	Keyword (score)	Sub-doc.-id	Keyword (score)	Sub-doc.-id	Keyword (score)
d-1	Sd-1	subtract(0.873)	Sd-1	account(0.431)	Sd-1	scholar(0.620)	Sd-1	action(0.781)	Sd-1	heavy(0.541)
	Sd-2	squar(0.792)	Sd-2	credit(0.401)	Sd-2	teach(0.610)	Sd-2	muscle (0.682)	Sd-2	sound(0.129)
	Sd-3	error(0.820)	Sd-3	balance(0.491)	Sd-3	assist(0.571)	Sd-3	calcium(0.705)	Sd-3	top(0.341)
	Sd-4	curv(0.639)			Sd-4	california(0.361)	Sd-4	antagonist (0.652)	Sd-4	controller(0.422)
	Sd-5	comput(0.831)			Sd-5	science(0.482)	Sd-5	effect(0.794)	Sd-5	dram(0.291)
	Sd-6	algebra(0.581)			Sd-6	professor(0.572)			Sd-6	crowd(0.218)
					Sd-7	program(,479)				
d-2	Sd-1	command(0.794)	Sd-1	transport(0.391)	Sd-1	credit(0.891)	Sd-1	scan(0.860)	Sd-1	data(0.621)
	Sd-2	function(0.742)	Sd-2	airport(0.411)	Sd-2	course(0.881)	Sd-2	cerebr(0.779)	Sd-2	spam(0.532)
	Sd-3	improve(0.693)	Sd-3	flight(0.327)	Sd-3	assignment(0.701)	Sd-3	tomograph(0.644)	Sd-3	email(0.487)
	Sd-4	formul(0.621)	Sd-4	date(0.591)	Sd-4	professor(0.702)	Sd-4	radiation(692)	Sd-4	technology(0.301)
	Sd-5	algorithm(0.662)	Sd-5	passenger(0.401)	Sd-5	syllabus(0.890)	Sd-5	measure(0.686)	Sd-5	internet(0.291)
	Sd-6	variabl(0.734)			Sd-6	solution(0.781)	Sd-6	reson(0.608)	Sd-6	information(0.510)
	Sd-7	counter(0.674)			Sd-7	grade(0.672)			Sd-7	communication(0.481)
	Sd-8	type(0.852)			Sd-8	cornell(0.532)			Sd-8	online (0.503)
d-3	Sd-1	order(0.738)	Sd-1	company(0.623)	Sd-1	school(0.671)	Sd-1	surgery(0.855)	Sd-1	sensor(0.261)
	Sd-2	russian(0.582)	Sd-2	seller(0.581)	Sd-2	univers(0.730)	Sd-2	patient(0.951)	Sd-2	electronic(0.192)
	Sd-3	supervisori(0.593)	Sd-3	profit(0.601)	Sd-3	washington(0.704)	Sd-3	stage(0.769)	Sd-3	ftp(0.110)
	Sd-4	shift(0.630)	Sd-4	year(0.481)	Sd-4	life(0.595)	Sd-4	morta(0.783)	Sd-4	response(0.234)
	Sd-5	built(0.782)			Sd-5	languag(0.781)	Sd-5	eye(0.842)	Sd-5	location(0.201)
					Sd-6	team(0.790)	Sd-6	complication(0.732)	Sd-6	router(0.231)
d-4	Sd-1	synovial(0.349)	Sd-1	supply(0.612)	Sd-1	publications(0.821)	Sd-1	baby(0.843)	Sd-1	innovative(0.131)
	Sd-2	nephron(0.462)	Sd-2	U.S.A (0.682)	Sd-2	search(0.903)	Sd-2	labor(0.731)	Sd-2	handle(0.210)
	Sd-3	unifect(0.593)	Sd-3	taleban(0.494)	Sd-3	index(0.870)	Sd-3	pregnant(0.792)	Sd-3	project(0.187)
	Sd-4	anti(0.621)	Sd-4	troops(0.381)	Sd-4	research(0.976)	Sd-4	mother(0.778)	Sd-4	success(0.190)
	Sd-5	ureter(0.533)	Sd-5	pakistan(0.411)	Sd-5	abstract(0.980)	Sd-5	deliveri(0.762)	Sd-5	lead(0.130)
	Sd-6	haemagglutin(0.425)	Sd-6	afghanistan(0.496)			Sd-6	care(0.769)	Sd-6	implement(0.231)

DS: dataset.

Table 9. Descriptions of top four clusters in DS (1–5).

Dataset	Cluster ID (F^M)	Cluster description
DS1	4(0.861)	catheter, vacuolar, angiocardiography, septum, intraventricular, sinus, artery, fluxa
	3(0.746)	weapon, disappear, florida, bureaucrat, advis, atlant
	1(0.712)	experi, skill, grow, difficult, quick, train
	2(0.689)	increas, input, power, larg, volum, time, signific, data
DS2	22(0.906)	market, trade, currency, stock, dollar, year
	4(0.813)	account, cash, amount, bank, purchase, company
	19(0.764)	afghanistan, war, nato, dead, security, taleban
	8(0.812)	price, program, iraq, east, oil, arab
DS3	10(0.842)	morphologi, metaphor, insignific, cue, cerebr, infecti
	13(0.762)	model, softwar, system, fault, technique, design
	1(0.745)	balloon, rad, cyte, oxygen, morbid, quir
	6(0.719)	transact, societi, ieee, accept, press, public
DS4	9(0.789)	cancer, stage, tumor, cure, dosage, therapi
	4(0.762)	alpha, factor, stimul, cell, gamma
	2(0.729)	hiv, mg, sera, virus, infection
	8(0.692)	infant, health, cholesterol, risk, pregnant, alcohol
DS5	4(0.952)	people, drink, eat, effects, chinese, diet
	10(0.937)	nasa, energy, gov, space, launch, apr
	15(0.912)	window, client, screen, file, lib, server, version
	5(0.854)	turkish, war, armenian, muslims, russian, survivor, death

DS: dataset.

score for a document (d-1) is segmented into six sub-documents sd (1–6) in DS-1, such that sd (1, 3, 5) obtained higher *F*-measure (0.873, 0.820, 0.831), respectively. The resultant clusters explicitly show that the semantic relationship of keywords and their association with original documents that they contain, such that cluster id-4 obtained the highest *F*-measure (0.861) in DS-1. Similarly, words in the cluster description indicated that cluster id-4 is associated with 'Medline', that is, a classic dataset (DS-1). This is due to LDA-onto model, as it does not consider a text document as a single unit, rather divides it into sub-documents using both semantical and ontological similarity. Furthermore, DS-4 contains short documents, such that clusters (4, 10, 15) achieved highest *F*-measure and they exhibit their association with topics such as Food, Technology, and Computers, respectively. It can be noted that qualitative assessment of ECA is providing clear significance through resultant clusters, which confirms that unique topic is easily discoverable in each cluster as shown in Table 9. This validates that ECA found above par results in both TS and text clustering, and as a whole it fully contributed when finding the solutions for document clustering in a broader spectrum without applying dimension reduction techniques over high sparse data. Our experiments provide an illustration for estimating the extracted segments first, and later it generates clusters through both overlapping and disjointed clustering solutions as well. In this study, ECA is proximated and associated with the characteristics of feasible attributes, such that appropriate similarity score is defined in each sub-document during segmentation, that in turn provide adequate identification of the gathered structure in the whole corpus. In addition, LDA-onto detects the topical boundaries that are being generated using coherent topical sub-documents as well as within/across text units, and thus, validating both text segmentation and clustering are combined into a single framework namely ECA. Regardless of not going for searching for lexical breaks rather expediting the nature of a transition among topics, we also noted that ECA followed the sub-document representation scheme as suggested by [25] could detect topical properties in the paragraph (through sentences) with the mainstream of searching boundaries better than existing traditional clustering and segment-based approach.

5.12. Implications for practice

ECA can produce clustering solutions in a variety of informative contents among text collections where manipulation of numerous topical construction of documents is necessary. ECA can also be useful in NLP tasks such as information retrieval and document summarization. ECA can also be evolved with new methods in text clustering and segmentation. In addition, it can identify and explore the topically coherent sub-documents to its respective documents including document hierarchy construction, clustering web-log data to identify similar groups of data, topic/novelty detection and extractive segmentation. However, existing topic-modelling-based approaches [12, 25, 42–45] either find the resultant clusters without having semantic relations between topics due to text segmentation model, or they could not exhibit the combination of document generative models and TS as established using ECA. In contrast, ECA could be effective to ease the process of conceptual relations in sub-documents through semantical and ontological similarity in ontology-based clustering tasks such as document comparison, document categorisation and document selection.

6. Conclusion

ECA is a novel topic-modelling-based approach, which improvises both text segmentation and clustering solutions for multi-topic document collections over high dimensional sparse data. First, text segmentation is performed using LDA-onto (through topics and words). Later, text clustering solutions in combination with LDA-onto are compared against traditional and segment-based clustering. Specifically, we derived clustering solutions in overlapping and disjointed way such that effectuality of LDA-onto is validated, and thus, improved text segmentation as well through ontological and semantical similarity. In LDA-onto, words rating is induced to rank sub-document to their respective document in an effective manner, such that it validates extracted sub-documents has achieved topical coherence over segment-based clustering (TextTiling). Overall, ECA achieved above par results and indicated better performances than traditional clustering in domain-independent dataset namely DS (1–5), as it significantly ameliorates the identification of different topics within/across documents. Our findings also indicated that the critical information from large collection of documents can be found in the resultant clusters, and it is obvious to control their linear measure along with topical cohesiveness. ECA is beneficial when documents tend to have less paragraph structure and useful in document recognition where topical structure of documents is not sufficient.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research is partially supported by the Humanities and Social Sciences Foundation of the Ministry of Education of China (no. 17YJCZH116), the National Natural Science Foundation of China (nos 61702039, 61807003) and the Fundamental Research Funds for the Central Universities.

ORCID iD

Muhammad Qasim Memon https://orcid.org/0000-0001-6380-9315

References

[1] Gennady S, Polina K, Nikita N et al. Applying topic segmentation to document-level information retrieval. In: *Proceeding of the 14th conference on Central and Eastern European software engineering*, Moscow, 12–13 October 2018, p. 3484. New York: Association for Computing Machinery.

[2] Cai X and Li W. A spectral analysis approach to document summarization: clustering and ranking sentences simultaneously. *Inform Sciences* 2011; 181(18): 3816–3827.

[3] Riken S, Deesha S and Lakshmi K. Automatic question generation for intelligent tutoring systems. In: *Proceedings of the 2nd international conference on communication systems, computing and it applications (CSCITA)*, Mumbai, India, 7–8 April 2017, pp. 127–132. New York: IEEE.

[4] Xiao L and Cornoy N. Discourse relations in rationale-containing text segments. *J Assoc Inf Sci Tech* 2017; 66(12): 2783–2794.

[5] Zulkefli NSSB, Rahman NBA, Puteh MB et al. Effectiveness of Latent Dirichlet allocation model for semantic information retrieval on Malay document. In: *Fourth international conference on information retrieval and knowledge management (CAMP)*, Kota Kinabalu, Malaysia, 26–28 March 2018, pp. 101–106. New York: IEEE.

[6] Li X and Lei L. A bibliometric analysis of topic modelling studies (2000–2017). *J Inform Sci*. Epub ahead of print 20 September 2019. DOI: 10.1177/0165551519877049.

[7] Bouguettaya A, Yu Q, Liu X et al. Efficient agglomerative hierarchical clustering. *Expert Syst Appl* 2015; 42(50): 2785–2797.

[8] Ganguly D. A Fast partitional clustering algorithm based on nearest neighbours heuristics. *Pattern Recogn Lett* 2018; 112: 198–204.

[9] Rathore AS and Roy D. Performance of LDA and DCT models. *J Inform Sci* 2014; 40(3): 281–292.

[10] Damien M and Claire GI. Model based clustering for mixed data: clustMD. *Adv Data Anal Classi* 2016; 10(2): 155–169.

[11] Zhang P and He Z. Using data-driven feature enrichment of text representation and ensemble technique for sentence-level polarity classification. *J Inform Sci* 2015; 41(4): 531–549.

[12] Tagarelli A and Karypis G. A segment-based approach to clustering multi-topic documents. *Knowl Inf Syst* 2013; 34(3): 563–595.

[13] Cunnings I and Sturt P. Retrieval interference and semantic interpretation. *J Mem Lang* 2018; 102: 16–27.

[14] Tugba Y, Banu D and Savas Y. Turkish synonym identification from multiple resources: monolingual corpus, mono/bilingual online dictionaries and WordNet. *Turk J Electr Eng Co* 2017; 25(2): 752–760.

[15] Schwarz C. Ldagibbs: a command for topic modeling in Stata using Latent Dirichlet allocation. *Stata* 2018; 18(1): 101–117.

[16] Corrêa EA, Lopes AA and Amancio DR. Word sense disambiguation: a complex network approach. *Inform Sciences* 2018; 442–443: 103–113.

[17] Auer S, Bizer C, Kobilarov G et al. DBpedia: a nucleus for a web of open data. In: *Proceedings of the 6th international the semantic web and 2nd Asian conference on Asian semantic web*, Busan, South Korea, 11–15 November 2007, pp. 722–735. Cham: Springer.

[18] Bizer C, Heath T and Berners–Lee T. Linked data – the story so far. *Int J Semant Web Inf* 2009; 5(3): 1–22.

[19] Vrandecic D and Krotzsch M. Wikidata: a free collaborative knowledge base. *Commu ACM* 2014; 57(10): 78–85.

[20] Suchanek FM, Kasneci G and Weikum G. Yago: a core of semantic knowledge. In: *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*, New York, 8–12 May 2007, pp. 697–706. New York: Association for Computing Machinery.

[21] Zhao H, Salloum S, Cai Y et al. Ensemble subspace clustering of text data using two-level features. *Int J Mach Learn Cyb* 2017; 8(6): 1–16.

[22] Capó M, Pérez A and Lozano JA. An efficient approximation to the K-means clustering for massive data. *Knowl-Based Syst* 2017; 117: 56–69.

[23] Anuar FM, Setchi R and Lai Y-K. Semantic retrieval of trademarks based on conceptual similarity. *IEEE Trans Syst Man Cyb* 2016; 46(2): 220–233.

[24] Chris B, Stefano F, Alexander P et al. A framework for enriching lexical semantic resources with distributional semantics. *Nat Lang Eng* 2018; 24(2): 265–312.

[25] Memon MQ, He J, Lu Y et al. An Improvised Sub-Document Based Framework for Efficient Document Clustering. *Journal of Internet Technology* 2019; 20(4): 1191–203.

[26] Hearst MA. TextTiling: segmenting text into multi-paragraph subtopic passages. *Comput Linguist* 1997; 23(1): 33–64.

[27] Madhusudanan N, Amaresh C and Gurumoorthy B. Discourse analysis based segregation of relevant document segments for knowledge acquisition. *AI EDAM* 2016; 30(4): 446–465.

[28] Saurav M and George K. Text segmentation on multilabel documents: a distant-supervised approach. In: *Proceedings of the 18th IEEE international conference on data mining, ICDM*, Singapore, 17 November 2018, pp. 1170–1175. New York: IEEE.

[29] Utiyama M and Isahara H. A statistical model for domain-independent text segmentation. In: *Proceedings of the 39th annual meeting on association for computational linguistics*, Toulouse, 6–11 July 2001, pp. 499–506. New York: Association for Computational Linguistics.

[30] Sunil WR and Deepa A. A novel approach of augmenting training data for legal text segmentation by leveraging domain knowledge. In: *Proceedings of the 4th international symposium on intelligent systems, technologies and applications, advances in intelligent systems and computing*, 24 February 2019, vol. 910, pp. 53–63. Cham: Springer.

[31] Mostafa B and Séamus L. C-HTS: a concept-based hierarchical text segmentation approach. In: *Proceedings of the 11th international conference on language resources and evaluation*, Miyazaki, Japan, 7–21 May 2018, pp. 1519–1528. European Language Resources Association (ELRA)

[32] Taeho J. Using K nearest neighbors for text segmentation with feature similarity. In: *Proceedings of the International Conference on Communication, Control, Computing and Electronics Engineering, ICCCCEE*, Khartoum, Sudan, 16–18 January 2017.

[33] Doina T, Diana I and Gabriela C. Text segmentation using Roget-based weighted lexical chains. *Comput Informat* 2013; 32(2): 393–410.

[34] Ji-Wei W, Judy TCR and Wen-Nung T. A hybrid linear text segmentation algorithm using hierarchical agglomerative clustering and discrete particle swarm optimization. *Integr Comput-Aid Eng* 2014; 21(1): 35–46.

[35] Misra H, Yvon F, Cappé O et al. Text segmentation: a topic modeling perspective. *Infor Process Manag* 2011; 47(4): 528–544.

[36] Kaimin Y, Zhe L, Genliang G et al. Unsupervised text segmentation using LDA and MCMC. In: *Proceeding of 10th Australian data mining conference (AusDM, 2012), Conferences in research and practice in information technology series*, Sydney, NSW, Australia, 5–7 December 2012, vol. 134, pp. 21–26. New York: Association for Computational Linguistics.

[37] Bayomi M, Levacher K, Ghorab MR et al. OntoSeg: a novel approach to text segmentation using ontological similarity. In: *Proceeding of the 15th international conference on data mining workshops*, Atlantic City, NJ, 14–17 November 2015, pp. 1274–1281. New York: IEEE.

[38] Goran G, Federico N and Paolo PS. Unsupervised text segmentation using semantic relatedness graphs. In: *Proceedings of the 5th joint conference on lexical and computational semantics*, Berlin, 11 August 2016, pp. 125–130. New York: Association for Computational Linguistics.

[39] Riedl M and Biemann C. Topic tiling: a text segmentation algorithm based on LDA. In: *Proceedings of ACL 2012 student research workshop*, Jeju Island, South Korea, 9–11 July 2012, pp. 37–42. New York: Association for Computational Linguistics.

[40] Bagheri A, Saraee M and de Jong F. ADM-LDA: an aspect detection model based on topic modelling using the structure of review sentences. *J Inform Sci* 2014; 40(5): 621–636.

[41] Omar M, On B-W, Lee I et al. LDA topics: representation and evaluation. *J Inform Sci* 2015; 41(5): 662–675.

[42] Kamal AS, Zuping Z and Yang K. Latent semantic analysis approach for document summarization based on word embeddings. *KSII Trans Internet Inf Syst* 2019; 13(1): 254–276.

[43] Gutiérrez-Batista K, Campaña JR, Vila M-A et al. An ontology-based framework for automatic topic detection in multilingual environments. *Int J Intell Syst* 2018; 33: 1459–1475.

[44] Jui-Feng Y, Yi-Shang T and Chen-Hsien L. Topic detection and tracking for conversational content by using conceptual dynamic latent dirichlet allocation. *Neurocomputing* 2016; 216: 310–318.

[45] Rifki A, Retno K and Rahmat G. Topic labeling towards news document collection based on latent dirichlet allocation and ontology. In: *1st international conference on informatics and computational sciences (ICICOS)*, Semarang, Indonesia, 15–16 November 2017, pp. 247–251. New York: IEEE.

[46] Green P. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 1995; 82(4): 711–732.

[47] Beeferman D, Berger A and Lafferty J. Statistical models for text segmentation. *Mach Learn* 1999; 34(1–3): 177–210.

[48] Pevzner L and Hearst MA. A critique and improvement of an evaluation metric for text segmentation. *Comput Linguist* 2002; 28(1): 19–36.

[49] Misra H, Cappe O and Yvon F. Using LDA to detect semantically incoherent documents. In: *Proceedings of the 12th conference on computational natural language learning*, Manchester, 16–17 August 2008, pp. 41–48. New York: Association for Computational Linguistics.

[50] Choi FYY. Advances in domain independent linear text segmentation. In. *Proceedings of the conference of 1st north American chapter of the association for computational linguistics conference (NAACL 2000)*, Seattle, WA, 27 April-4 May 2000, pp. 26–33. New York: Association for Computational Linguistics.

[51] Classic text database, https://www.dataminingresearch.com/index.php/2010/09/classic3-classic4-datasets/

[52] Lewis DD, Yang Y, Rose TG et al. RCV1: a new benchmark collection for text categorization research. *J Mach Learn Res* 2004; 5: 361–397.

[53] Han EH, Boley D, Gini M et al. WebACE: a web agent for document categorization and exploration. In: *Proceedings of the 2nd international conference autonomous agents*, Minneapolis, MN, 1 May 1998, pp. 408–415. New York: Association for Computational Linguistics.

[54] Voorhees E and Harman D. The text retrieval conferences (TRECS). In: *Proceedings of a workshop*, Baltimore, MD, 13–15 October 1998, pp. 241–273. New York: Association for Computational Linguistics.