An Analysis of Various Text Segmentation Approaches

Sumit Kumar Daroch and Pardeep Singh

1 Introduction

Text segmentation is method of separating written text into sections or parts, and these parts are known as segments. The text can be broken into sentences, topics, and words. Each section has a relevant meaning. The concept refers both to conceptual mechanisms used by humans when interpreting text, and to artificial processes that are the subject of natural language processing (NLP) applied in computers. Text segmentation is the procedure of splitting down a document into constituent portions based on its semantic structure. The difficulty in text segmentation varies depending on what is the type of that text and how it is written: informative, talkative, descriptive, and so on. The capacity of section archives dependent on theme would empower access and investigation of the subtopic in a report instead of access and examination of entire record. Section/Segment give us a superior comprehension of the archives.

We would like to draw your attention to the fact that it is not possible to modify a paper in any way, once it has been published. This applies to both the printed book and the online version of the publication. Every detail, including the order of the names of the authors, should be checked before the paper is sent to the Volume Editors.

S. K. Daroch (✉) · P. Singh
National Institute of Technology Hamirpur, Hamirpur, Himachal Pradesh 177005, India
e-mail: cs16mi518@nith.ac.in

P. Singh
e-mail: pardeep@nith.ac.in

B. Agarwal et al. (eds.), *Proceedings of International Conference on Intelligent Cyber-Physical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-16-7136-4_22

1.1 Why Text Segmentation

There are many reasons why we would want to do this, but maybe the most apparent is that surfing or looking for the results makes things much simpler for a person. Consider a long, continuous recording of a news program or a business conference. It can be difficult to find a specific news article or discuss a specific subject, especially if someone do not want to view or listen to the complete program. The first option is to check for similar words and keywords that are related to your interest, now a person can find the keyword, but it won't tell where is the starting of that section or topic and also there is no guarantee that someone will always be able to find the keywords or word you have selected, particularly if the error rates of the words are high. If the whole document is segmented according to the topic, then it is easier to find out the topic of your interest.

We will go far deeper than this: someone may be like to evaluate and identify the material of every section so that he can connect subjects from one session to another or record the evolution of news reports through multiple newscasts. He may wish to create a concise overview with the main points of every issue. In these cases, text segmentation helps a lot.

Text segmentation is a fundamental NLP challenge that is used in a wide range of activities including summarization, passage extraction, context comprehension, and emotion extraction, among others. Fine-grained text segmentation into many sections makes for a more accurate understanding of the construction of particular document which can be used to produce improved document representations.

Let us take a real-world example of a newspaper. In the newspaper, all the news are divided according to the topics like there is a separate page for the news related to sports. And this page is further divide into section which contain different news related to different sport like news related to Cricket and Football. This is all done to provide the better understanding of the news to the readers.

1.2 Type of Text Segmentation

Text segmentation is of mainly three types:

- **Word Segmentation**: The method of splitting a string or a text which is written in a specific language into its constituent words is known as word segmentation. It is the method by which computer algorithms decide the word borders in a sentence or text. For most higher level NLP functions, parsing and machine translation, POS tagging, word segmentation is the initial phase. It can be seen as the issue of correctly defining word types from a string of characters.

When we listen to speech, we hear a sequence of sentences, but when we talk, we cannot discern words through pauses. Then, eliminating vocabulary from continuous speech is the first step in learning a language's words. Our flow of speaking is also continuous, so for a machine to understand what the person is going to say, machine have to broke that text into meaning full words. Let us take an example that someone is asking that:

Example: Are you a "male or female"?

To give the correct answer to this question, person must have the understanding of each unit word. Here maleorfemale is a single string, machine can predict it as "maleor female" or "male or female". So, word segment help to correctly identify the words, so that one can understand proper meaning of given text. It is widely used in speech to text conversion and also used in to understand the language which doesn't have any delimiter to separate the words.

- **Sentence Segmentation**: The method of deciding the longer processing units composed of one or more words is sentence segmentation. This role includes the detection of sentence limits in various sentences between words. Since most written languages have punctuation marks that appear at sentence borders, phrase segmentation is sometimes referred to as identification of sentence boundaries, disambiguation of sentence boundaries, or recognition of sentence boundaries. All these words refer to the same task: to decide how a text can be separated for further processing into sentences.

In this form of segmentation, we break the written string or text into its Sentences part. We divide text into sentences for better understanding. In English language, we use full stop (.), to determine the ending of sentences, but in English language, we also use full stop (.) for the abbreviation. So with the help of full stop we cannot correctly identify the sentence ending. So sentence segmentation is used for this purpose. Let us take an example a text is written as follow:

Mr. Sumit is a student of NIT Hamirpur.

Here Mr. Sumit is a single entity. If we divide the sentence according to the full stop then Mr. is in different sentence and Sumit is in different sentence, which is wrong. So, to identify correct ending of sentence, firstly we have to understand the text and then divide accordingly. So here text segmentation plays an important role.

- **Topic Segmentation**: Topic segmentation is an important activity for semantic text analysis, which seeks to find the borders between topic blocks in a text. In this type of segmentation, we broke the written string or text into its component topics. This segmentation consists of two prime functions:

  – Topic Recognition.
  – Text Segmentation.

A document may contain multiple topics, we have to identify the different type of topic present in the document and then create segment accordingly. Numbers of segments that we create is equal to the numbers of topics in that document. Each segment contains a different topic. Sentences belong to the same topic are in same segment. Segmenting the text into topics or discourse turns might be useful in information retrieval. A whole book, for example, can be interpreted as one topically coherent portion. In a nutshell, chapters are subsegments of the book, and paragraphs are sub-segments of the chapters. A topically coherent portion is often formed by a single sentence or n-gram. As a consequence of the use of segmentation, the exact definition of a subject varies.

2 Various Approaches

Utiyama and Isahara [27] proposed an analytical procedure to determine the highest probability segmentation of a given text. Since it calculates probabilities from the provided document, this approach does not involve training data. Therefore, any text in any domain may be added to it. As a result, it can be used on any text in any domain. The experiment demonstrated that the procedure outperforms, or is at least as effective as, a cutting-edge text segmentation scheme. This approach determines the most likely segmentation of a given document. The probability of words in segments are naturally calculated using this approach. These probabilities, known as word densities, have been used to detect critical word meanings in documents. This approach is established on the assumptions that a word's density is large in a section where the word is addressed in detail. They experiment this approach on Choi's dataset and find Pk [1], in % is 10. Figure 1 shows the variation of results, where data is divided into various sets according to the number of sentences in a document. But in this method, error rate is high. LDA [22] can improve the result.

Brants et al. [4] introduce a new approach for topic-related text segmentation that incorporates the use of the Probabilistic Latent Semantic Analysis (PLSA) structure

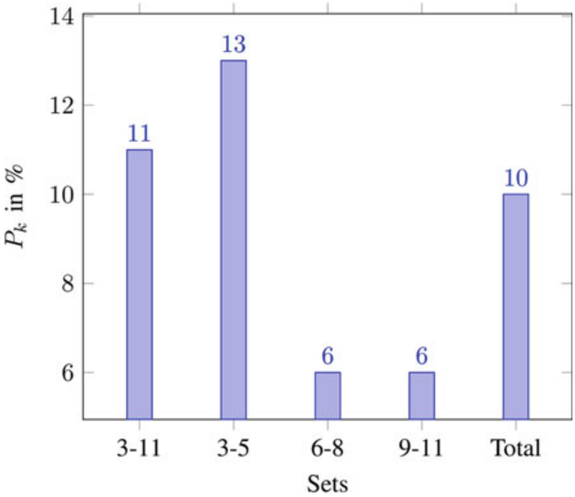


Fig. 1 Variation of Pk with respect to sets [27]

or representation with a procedure for choosing segmentation marks that depend on similarity values among neighboring lines. Latent means hidden, so we have to find the hidden features. Here, topics are hidden. PLSA authorize for a greater comprehension of fragmented knowledge in a chunk of text, like a single sentence or a list of sentences. Connecting or linking distinct occurrence of the identical model, by using various random occurrences or a various number of latent groups, improves segmentation efficiency even more. They evaluated this segmentation method on two corpus. The first corpus is Brown Corpus is which consists of 500 text samples with an average length of 2000 words each. They collect data by training and research, as defined in [3]. The second corpus is Router-21578, which was developed using the process described in [18]. The error reduction in Brown corpus is 31% and in Router-21578 is 71%. This model, however, cannot be used in real-world applications due to its high memory requirements and long execution time.

Athanasios Kehagias [16] presented a new text segmentation algorithm that is based on dynamic programming. This approach is based on unsupervised learning. It performs linear text segmentation by globally minimizing a segmentation cost function that integrates two factors: (a) similarity of the term within the segment and (b) prior segment length information. The algorithm's segmentation accuracy is measured by accuracy, recall, and Beeferman's segmentation metric. They experiment this approach on Choi's dataset and find Pk, in % is 6.40 (Fig. 2 where sets shows number of sentences in a document). Then they compared this methods with C99 [6], U00 [2], and CWM [7] methods and find out this method gives better results. The performance of this algorithm is very satisfactory. But this algorithm performs poor when number of sentences in a set is not between 9 and 11 as compared to topic modeling [22].

Chiru [5] purposed an unsupervised learning approach for text segmentation known as unsupervised cohesion-based segmentation. In this technique, they also

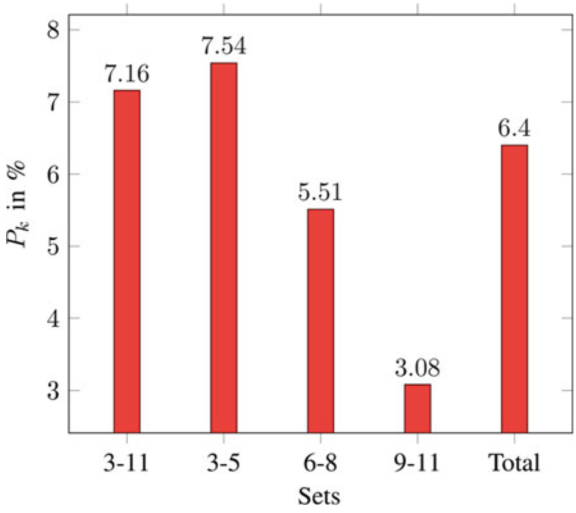


Fig. 2 Variation of Pk with respect to sets [16]

propose a voting system that improve the segmentation results. In this work they presented three modules, each with different methods and a voting mechanism is implemented in order to boost the outcomes achieved from the three processes. All the three modules are implemented using an lexical cohesion approach based on unsupervised learning and used to compare the results of various lexical cohesion-based approaches to find out that how they can be strengthened by integrating their outputs applying the voting system. The text is thought to be divided into subsections, with each column devoted to a single subject. Firstly, part-of-speech tagging is done. The first approach examines applicant limits to determine which are true limits. Any time a new candidate subject cap is encountered, it will be assessed to see if it is a genuine topic change. If it isn't, the present paragraph will be considered part of the previous subject, so this candidate topic cap will be skipped, and the review will go on to the next paragraph. In second module they use the concept of lexical chains and clustering algorithm. After part-of-speech tagging, the locations of the most.

Significant indexes are consumed into account for clustering. Then cohesion chains are made using repetition of words, synonyms, and words relationships. The clustering algorithm produces the first lexical chain with the first token, and so on. Topic changes have already been established, with a high density of chain begins and ends suggesting that the topic has shifted. Third module approach is same as second module approach, but in this module number of chains is equals to number of cluster/segment. An effort is made in voting to merge the results received from all the three modules to find out if some improvement in the last conclusion can be accomplished. The techniques will be tested to see whether they are similar, and whether scaling variables may be used to improve the probability of accurate outcomes. The scaling factors would be calculated empirically.

Eisenstein and Barzilay [9] presented an unsupervised learning-based approach that are using a novel Bayesian Lexical for text segmentation. In this method, unsupervised systems are guided by lexical cohesion: the propensity to cause a compact and coherent lexical distribution by well-formed segments. They demonstrate that we can put the lexical consistency in a Bayesian sense with help of modeling the terms in each subject section as generate from a multinomial language structure linked with the segment, and that maximizing the conclusional probability in such a model that generates lexically cohesive segmentation. This is in contrast to previous techniques, which focused on handcrafted harmony metrics. But this paradigm allows for the inclusion of additional functionality such as cue words, an important predictor of discourse architecture that has not traditionally been used in unsupervised learning-based segmentation frameworks. For both text and speech datasets, this model consistently outperforms a variety of state-of-the-art schemes. They further demonstrate that an entropy related testing and a previously existing method can be performed as particular instances of the Bayesian system. On all metrics, Bayesian schemes produce a raw output benefit of 2–3% over all baselines on the medical textbook corpus. On the ICSI meeting corpus, Bayesian structures outperform the best benchmark by 4–5% on the Pk metric and obtain a smaller gain on the WindowDiff metric. Table 1 shows the results for two datasets. This model performs better as compared to UI [27], LCSEG [12] and MCS [20].

Table 1 Values of Pk and WindDiff for different datasets [9]

Dataset	WinDiff (WD)	Pk Value
Medical Textbook	0.339	0.353
ICSI meeting	0.258	0.312

Misra [22] approached the task of text segmentation from a topic/subject modeling standpoint. They look at how the Latent Dirichlet Allocation (LDA) subject system can be used to splitting the text into the segments from a document. One important advantage of the suggested solution is that it outputs the subject distribution associated with each segment in addition to the segment boundaries. This data may be helpful in applications, for example, section retrieval and discourse inspection. The LDA-based approach introduced in this paper is depend upon the following assumption: if a section is build from just one story, there will be small amount of active topics, while if a segment is build from numerous story, there will be a considerably large amount of active subjects or topics. Expanding this logic, if a segment is rational (the subject circulation for that segment contains just a small amount of active topics), the log likelihood for that segment is normally high, as opposed to segment that is not coherent [21]. This discovery is vital to the effectiveness of the suggested LDA-based method for task of text segmentation, and it has remained undetermined excluding for its primary work in recognizing document consistency [21]. They experiment this approach on Choi's dataset and find Pk, in % is 15.5 (Unadapted LDA) for 3–11 sentences (Fig. 3, where sets shows number of sentences in a document). The main disadvantage of this approach is vocabulary Mismatch. To fix the issue of vocabulary overlap, they split Choi's dataset into two parts, and with the help of Part A (called it Adapted LDA) they find Pk, in % = 2.3 for 3–11 sentences (Fig. 4, where sets shows number of sentences in a document). This method performs better than Choi

Fig. 3 Variation of Pk with respect to sets (Unadapted LDA)

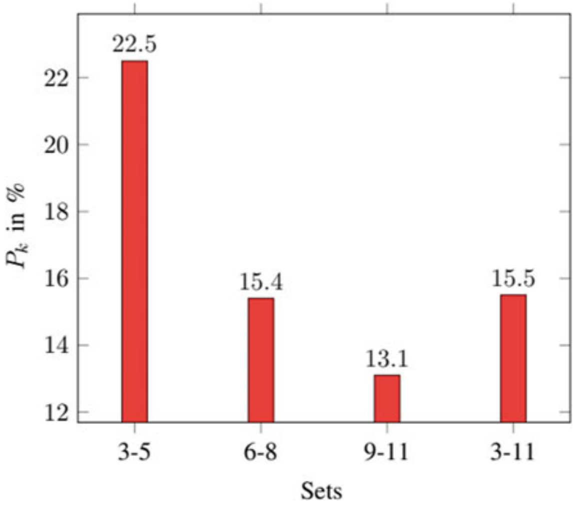
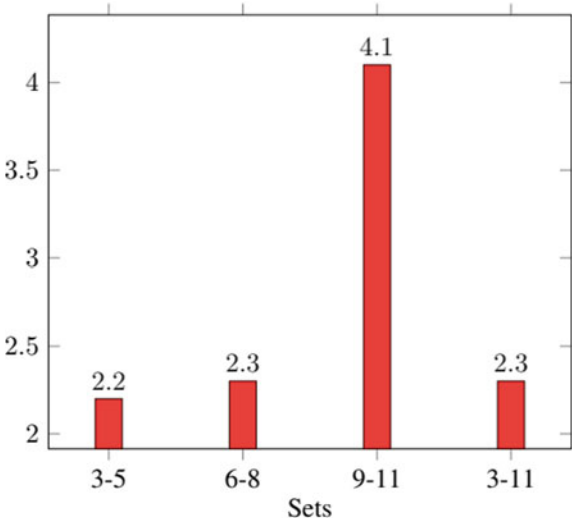


Fig. 4 Variation of Pk with respect to sets (Adapted LDA)



[6], Utiyama and Isahara [27], Choi [7], and Brants [4] methods. But Fragkou's [10] method perform better for a set where sentences are between 9 and 11.

Kazantseva and Szpakowicz [15] introduced a new linear text segmentation algorithm. It is a type of Affinity Propagation, a cutting-edge clustering algorithm in the context of factor graphs. Affinity Propagation for Segmentation (APS) takes a series of pairwise connection between data points and generates segment borderlines and segment centers data points that better represent all other data points inside the segment. APS transfers messages iteratively through a cyclic factor graph before convergence. Each iteration uses data from all available similarities to generate high-quality performance. For practical segmentation functions, APS scales linearly. This algorithm is derived from the original Affinity Propagation formulation and equate it to two state-of-the-art segmenters when it comes to topical text segmentation. In this technique they made four matrices: First matrix is similarity matrix, second is responsibility matrix, third is availability matrix, and last is criterion matrix. The higher values of each row of criterion matrix is set up as exemplar. Text whose exemplar are same are in same cluster. Sentences are data points in this context and exemplars are segment centers. They allocate each sentence in a text to a section center in order to optimize net similarity. The developers also made freely accessible implementations of Java. On three datasets, they tested the APS algorithm's accuracy. The first dataset contains AI lectures, the second dataset is collection of chapters from medicinal textbooks, and the third dataset contains 85 works related to fiction. For AI WindowDiff = 0.404, for Fiction dataset, it gives WindowDiff = 0.350, and for Clinical Dataset, WindowDiff = 0.371 (Table 2). They compared this method with BayesSeg [9] and MinCutSeg [20]. For clinical dataset, BayesSeg [9] perform better, where WindowDiff = 0.353.

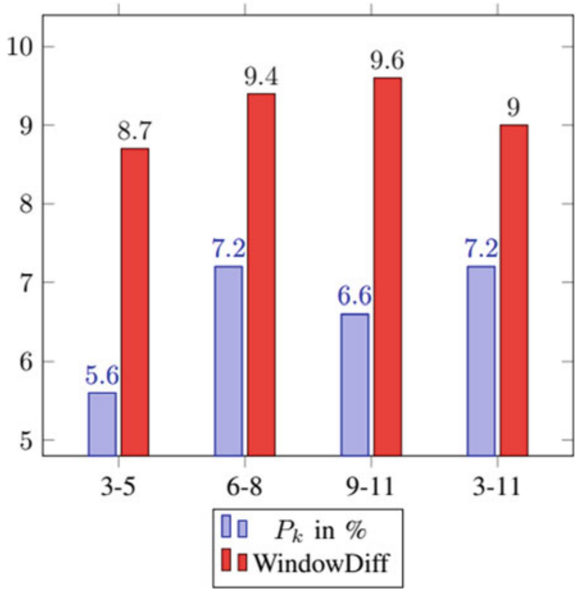
Table 2 Values of WindowDiff for different datasets [15]

Dataset	WindowDiff
AI	0.404
Clinical	0.371
Fiction	0.350

Glavas et al. [13] presented a novel algorithm for linear text segmentation (TS) based on unsupervised learning that constructs a semantic relatedness graph. In this approach, all the sentences become nodes of the relatedness graph G. The semantic similarity of all pairs of sentences in a given text is then determined. In this method, they used a graph data structure. A sentence is represented as node and if there is relation between two sentences or sentences are in same segment then there is edge between the nodes. To find the edges or semantic relation between sentences, they use greedy lemma alignment. Greedy Lemma Alignment use resemblance of their propagation vectors and greedily match background terms between sentences. If the words of two sentences have similar vector distribution, then it create the pair of the words. But this approach doesn't give optimal result, so they use new method that is building a weighted complete bipartite graph [11] between each pair of words of two sentences. A linked edge links each word in one sentence to different word in another sentence. And then running this algorithm [11], similarity between the words is calculated and then make pair of the words. Then create the set of the words pairs of two sentences and find out that there is a relation between two sentences or not. If there is a relation then edge is passes through that sentences or nodes. GRAPHSEG [13] was tested on four subsets of the Choi dataset, each with a different number of sentences. It gives Pk = 7.2 and WindowDiff = 9.0 for 3–11 sentences set as shown in Fig. 5 (where sets shows number of sentences in a document). On a synthetic dataset, this approach performs competitively with the best-performing LDA-based approaches [22]. But Riedl and Biemann [25] perform better for all the sets.

Sehikh et al. [26] proposed a new method for topic segmentation in speech recognition transcripts that uses bidirectional Recurrent Neural Networks (RNNs) to calculate lexical cohesion. The bidirectional RNNs catch meaning from the previous set of words as well as the next set of words. To identify subject transitions, the prior and subsequent contexts are contrasted. This method does not use a segmented corpus subject design for preparation, unlike previous research focused on arrangement and discriminative systems for topic segmentation. This model is learned by reading news stories on the internet. Based on DNN-HMM [19] acoustic models, ASR transcriptions are obtained from French ASR method. The feasibility of this solution is shown by this ASR transcripts. The bidirectional RNNs gathered meaning in the previous and subsequent sets of words, and compared the two sets of contexts to identify subject shifts. Concatenating news stories from the internet was used to train these models discriminatively. This RNN models outperformed the C99-LSA [6] and TopicTiling [25] models on ASR transcripts of French television news programs. They use the traditional subject segmentation appraisal steps Pk and WindowDiff [23] for comparison of the proposed and base line approaches. WindowDiff score is

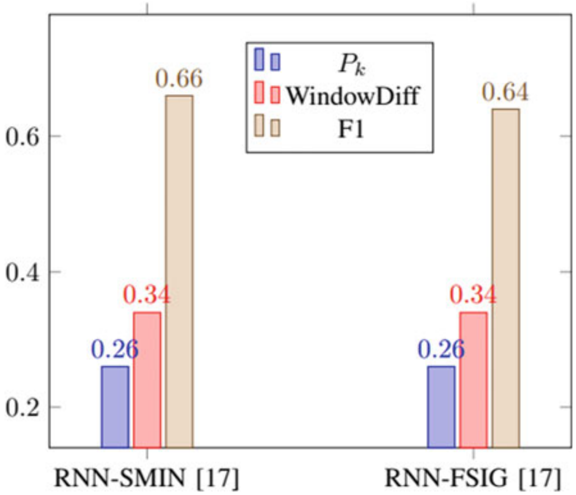
Fig. 5 Variation of Pk and WindowDiff with respect to sets for semantic relatedness graph model



evaluated for TV5 dataset and found 0.34 and Pk is 0.26, which means it has less loss of information. Figure 6 shows the result for two techniques in this paper.

Koshorek et al. [17] presented a massive new dataset for text segmentation that is automatically take out and tagged from Wikipedia, and formulated text segmentation as a supervised learning problem. They also build a model (Fig. 7) that established on this dataset and prove that it generalizes effectively to common text that hasn't been used before. In this dataset, input x is a document which consist n sentences

Fig. 6 Results for two different techniques in [26]



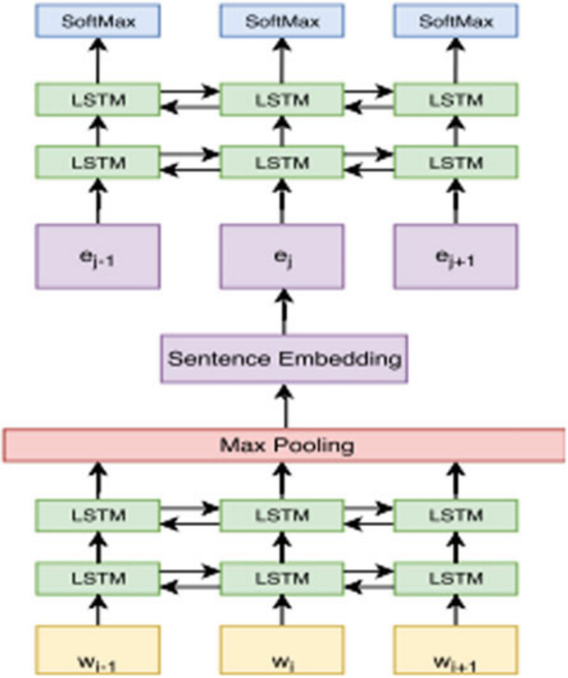


Fig. 7 Architectural Diagram of LSTM to based Supervised Model [17]

and n-1 binary value which represent whether that sentence is end of segment or not. Model (Fig. 7) has two sub-networks. First sub-network is used to generate sentence representation and second sub-network is used to predict the text segmentation. First sub-network generates sentence representation using BiLSTM cells, it take words from sentence as input and max-pooling over the LSTM outputs yields the final sentence representation. Then these embedding fed up to the second sub-network as an input and this sub-network feeds a two-layer bidirectional LSTM with a series of sentence embeddings as data. After that, they added a fully connected layer to each of the LSTM outputs to generate a series of n vectors. To obtain n-1 segmentation probabilities, they disregard the last vector and use a softmax function. This model can be run on modern GPU hardware and has a linear runtime in terms of text length. They tested this approach on WIKI-50 and CHOI datasets and find out the Pk value = 18.24 for WIKI-50 dataset. Figure 8 shows the variation of Pk for two models for CHOI's dataset. But GRAPHSEG [13] provides better results on the synthetic Choi dataset on comparing with this approach, but this performance does not carry over to the natural Wikipedia data, where they underperform the random baseline.

Pinkesh Badjatiya [1] proposed an attention-dependent bidirectional LSTM [14] model for in which CNNs are used to learn sentence embeddings and the segments are concluded based on contextual data (Fig. 9). Variable-sized background information can be managed dynamically by this model. This model takes sentence s and its K-sized left context (K numbers of left sentences) and K-sized right context (K

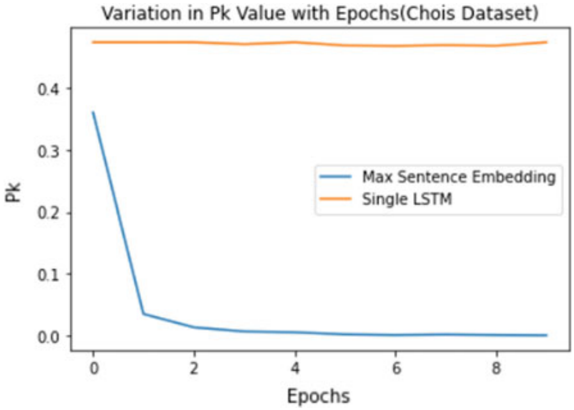


Fig. 8 Variation of Pk with respect epochs for models in paper [17]

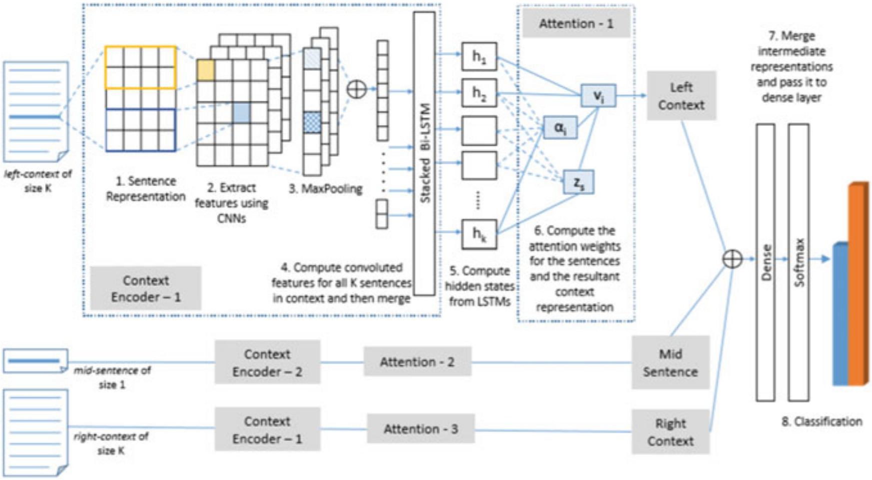

left-context of size K
1. Sentence Representation
2. Extract features using CNNs
3. MaxPooling
4. Compute convoluted features for all K sentences in context and then merge
Stacked Bi-LSTM
h1
h2
hk
5. Compute hidden states from LSTMs
Attention - 1
vi
αi
zs
6. Compute the attention weights for the sentences and the resultant context representation
Context Encoder – 1
Left Context
7. Merge intermediate representations and pass it to dense layer
Dense
Softmax
8. Classification
mid-sentence of size 1
Context Encoder – 2
Attention - 2
Mid Sentence
right-context of size K
Context Encoder – 1
Attention - 3
Right Context

Fig. 9 Architectural diagram of attention-based model [1]

number of right sentences) as input and with help of this information it predicts that whether this sentence s is a beginning of new segment or not. In this model, sentences embedding is done with the help of word2vec model and that feature are extracted by using CNNs. Max-pooling is used to maximize the result. All the features from all the K sentences are merged and then stacked-BiLSTM is used to compute hidden states. After this, results are fed into attention layer.

Attention layer help to get more information from a text. And finally, to produce results, they used a softmax layer as an activation function and to get the highest value from the target attribute, they use the arg max function. They trained and tested this model on three different datasets Clinical [20], Fiction [15] and Wikipedia [1].

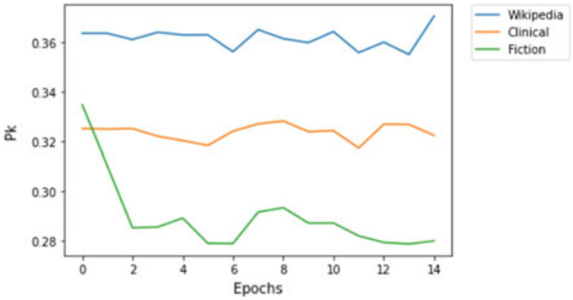


Fig. 10 Pk value's variation with respect to epochs for different datasets for model in [1]

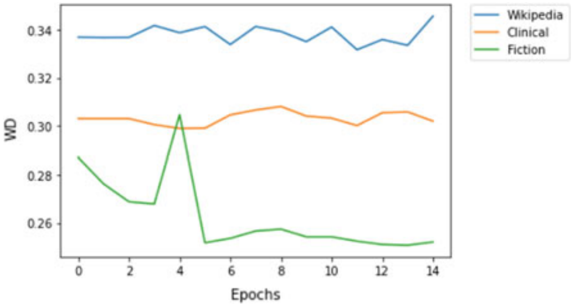


Fig. 11 WD value's variation with respect to epochs for datasets for model in [1]

Table 3 Results of different datasets [1]

Dataset	WindowDiff	Pk Value
Clinical	0.294	0.318
Fiction	0.308	0.378
Wikipedia	0.315	0.344

Figure 10 and Fig. 11 shows the variation of Pk and WindowDiff for different datasets respectively. Table 3 shows the results for various datasets. They also compare this model with some baseline models like PLDA [24], TSM [8] and find out this model gives better WindowDiff for three datasets, but some models perform better for Pk values.

3 Conclusion

We compare the performance of the various models that are trained using unsupervised learning and use same Choi's Dataset. Figure 12 shows the variation in Pk values for this comparison, where x-axis shows the references of various paper accordingly and x-axis shows the Pk value. From this we conclude that for Choi's dataset, best model was proposed in [22]. But in this method they divided the dataset

Fig. 12 Variation of Pk with respect to different models (Choi's Dataset)

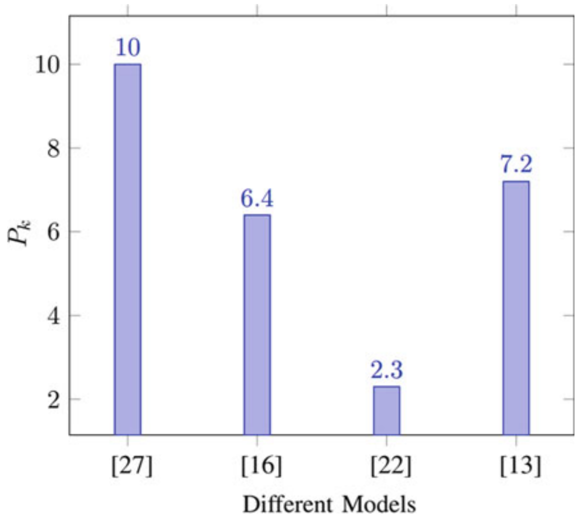
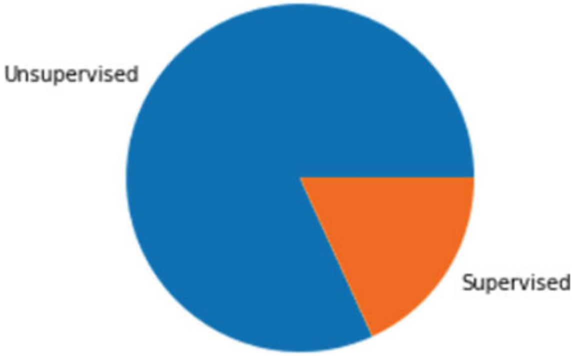


Fig. 13 Pie chart showing Unsupervised and Supervised techniques compared in this paper



to remove vocabulary mismatch and use only half dataset. So we can say that best result is given by model in [16].

Text segmentation is critical for activities like context Fig. 11. WindowDiff (WD) value's variation with respect to epochs for different datasets for model in [1] comprehension, summarization record indexing, and noise elimination in NLP. There are many techniques available for text segmentation but most of the text segmentation approaches are unsupervised learning-based (as shown in Fig. 13), this may be due to the unavailability of supervised data. After comparing these approaches (from Table 4) for text segmentation, we find out that Attention-based text segmentation is best approach for text segmentation. This experiment gives best values.

Table 4 Comparison table for various approaches for text segmentation

References	Year	Learning approaches	Methodology	Performance	Shortcomings
[27]	2001	Unsupervised Learning	Maximum-probability segmentation. In terms of the likelihood specified by a statistical model, it selects the optimum segmentation	Choi's dataset: Pk in %: 10	LDA [22] can improve the result. Error rate is high
[4]	2002	Unsupervised Learning	Use of Probability Latent Semantic Analysis system with the procedure of choosing segment point	Brown corpus: Error reduction: 31% Router-21578 Error Reduction :71%	Big memory constraints and large execution time make this approach impractical to implement in the real world
[10]	2003	Unsupervised Learning	Dynamic programming algorithm universal minimization of the cost function of segmentation	Choi's dataset: Pk in %: 6.40	Perform poor when number of sentences in a set is not between 9 and 11
[5]	2007	Unsupervised Learning	Three different module based on cohesion. Proposed a Voting System	Proposed a Voting System	Due to the use of statistical technique, this system give inaccuracy. Error rate high
[9]	2008	Unsupervised Learning	By modeling the terms in each topic segment, lexical cohesion is put in a Bayesian space	Medical textbook corpus Pk: 0.339 ICSI meeting corpus: Pk: 0.258	Due to the use of statistical technique, this system give inaccuracy. Error rate high
[22]	2009	Unsupervised Learning	Usage of the subject model of latent Dirichlet allocation (LDA) to produce the segment from the text	Choi's dataset: Pk in %: 2.3	Dynamic programming-based [10] approach for text segmentation performs well than LDA. Vocabulary Mismatch

(continued)

Table 4 (continued)

References	Year	Learning approaches	Methodology	Performance	Shortcomings
[15]	2011	Unsupervised Learning	Adaptation of Affinity Propagation, takes series of pairwise connections between data points and generates segment borderline	Fiction WindowDiff: 0.350	The choice for objects to use as examples needs to be defined
[13]	2016	Unsupervised Learning	Builds a semantic relatedness graph, all the sentences become nodes and if there is similarity between sentences then a edge is created between them	Choi's Dataset: Pk: 7.2 WinDiff: 9.0	It is used for short text. Topic Modeling give better results
[26]	2017	Unsupervised Learning	Bidirectional RNN with LSTM cells to calculate cohesion. Word Embedding	Choi's Dataset: Pk: 7.2 WinDiff: 9.0	The entire utterance may not be usable for such implementations, such as real-time speech recognition, and Bidirectional RNN may not be satisfactory
[17]	2018	Unsupervised Learning	First sub-network generates sentence representation using BiLSTM cell. Second sub-network feeds a two-layer BiLSTM with a series of sentence embeddings as data	Wiki-50 Dataset Pk: 18.24	Not perform good in Choi's Dataset as compared from GraphSeg [26]
[1]	2018	Unsupervised Learning	Extract features using CNNs MaxPooling, BiLSTM, Attention	Fiction: WinDiff: 0.308 Clinical WinDiff: 0.294 Wikipedia: WinDiff: 0.315	We have to decide number of sentences (Right and Left Context) for training. Due to this documents dropout may occur

References

1. Badjatiya, P., Kurisinkel, L.J., Gupta, M., Varma, V.: Attention-based neural text segmentation. In: European Conference on Information Retrieval, pp. 180–193. Springer (2018)
2. Blei, D.M., Moreno, P.J.: Topic segmentation with an aspect hidden markov model. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 343–348 (2001)
3. Brants, T.: Test data likelihood for plsa models. Inf. Retrieval **8**(2), 181–196 (2005)
4. Brants, T., Chen, F., Tsochantaridis, I.: Topic based document segmentation with probabilistic latent semantic analysis. In: Proceedings of the Eleventh International Conference on Information and Knowledge Management, pp. 211–218 (2002)
5. Chiru, C.-G.: Unsupervised cohesion-based text segmentation. EUROLAN 2007 Summer School Alexandru Ioan Cuza University of Ias¸i, p.93
6. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. arXiv preprint cs/0003083 (2000)
7. Choi, F.Y.Y., Wiemer-Hastings, P., Moore, J.D.: Latent semantic analysis for text segmentation. In: Proceedings of the 2001 Conference on Empirical Methods In Natural Language Processing (2001)
8. Du, L., Buntine, W., Johnson, M.: Topic segmentation with a structured topic model. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 190–200 (2013)
9. Eisenstein, J., Barzilay, R.: Bayesian unsupervised topic segmentation. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pp. 334–343 (2008)
10. Fragkou, P., Petridis, V., Kehagias, A.: A dynamic programming algorithm for linear text segmentation. J. Intell. Inform. Syst. **23**(2), 179 197 (2004)
11. Frank, A.: On kuhn's hungarian method a tribute from hungary. Naval Res. Logist. (NRL) **52**(1), 2–5 (2005)
12. Galley, M., McKeown, K., Fosler-Lussier, E., Jing, H.: Discourse segmentation of multi-party conversation. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp. 562–569 (2003)
13. Glavas, G., Nanni, F., Ponzetto, S.P.: Unsuper-vised text segmentation using semantic relatedness graphs. Association for Computational Linguistics (2016)
14. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
15. Kazantseva, A., Szpakowicz, S.: Linear text segmentation using affinity propagation. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 284–293 (2011)
16. Kehagias, A., Fragkou, P., Petridis, V.: Linear text segmentation using a dynamic programming algorithm. In: 10th Conference of the European Chapter of the Association for Computational Linguistics (2003)
17. Koshorek, O., Cohen, A., Mor, N., Rotman, M., Berant, J.: Text segmentation as a supervised learning task. arXiv preprint arXiv:1803.09337 (2018)
18. Lee, L.: Measures of distributional similarity. arXiv preprint cs/0001012 (2000)
19. Li, L., Zhao, Y., Jiang, D., Zhang, Y., Wang, F., Gonzalez, I., Valentin, E., Sahli, H.: Hybrid deep neural network–hidden markov model (dnn-hmm) based speech emotion recognition. In: 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction, pp. 312–317. IEEE (2013)
20. Malioutov, I.I.M.: Minimum cut model for spoken lecture segmentation. Ph.D. thesis, Massachusetts Institute of Technology (2006)
21. Misra, H., Cappe, O., Yvon, F.O.: Using lda to detect' semantically incoherent documents. In: CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning, pp. 41–48 (2008)

22. Misra, H., Yvon, F., Jose, J.M., Cappe, O.: Text' segmentation via topic modeling: an analytical study. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 1553–1556 (2009)
23. Purver, M.: Topic segmentation. Spoken language understanding: systems for extracting semantic information from speech, pp. 291– 317 (2011)
24. Purver, M., Kording, K.P., Griffiths, T.L., Tenenbaum, J.B.: Unsupervised topic modelling for multi-party spoken discourse. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 17–24
25. Riedl, M., Biemann, C.: Topictiling: a text segmentation algorithm based on lda. In: Proceedings of ACL 2012 Student Research Workshop, pp. 37–42 (2012)
26. Sehikh, I., Fohr, D., Illina, I.: Topic segmentation in asr transcripts using bidirectional rnns for change detection. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp. 512–518. IEEE (2017)
27. Utiyama, M., Isahara, H.: A statistical model for domain independent text segmentation. In: Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, pp 499– 506 (2001)