Discourse Segmentation

Complete Study Notes - Natural Language Processing

Topic	Covered
1	Discourse units and segments
2	Sentence-level vs discourse-level processing
3	Discourse segmentation techniques
4	Rule-based vs statistical segmentation
5	Architectures (pipeline, TextTiling, supervised, neural)
6	Step-by-step worked examples
7	Applications, evaluation, challenges, exam questions

1. Introduction and Definition

Discourse is a coherent, structured group of sentences (spoken or written) that together convey a meaning larger than any single sentence: an essay, a news article, a dialogue, an email thread, a lecture. **Discourse segmentation** is the task of automatically dividing a discourse (text or speech transcript) into smaller, coherent, non-overlapping units called *segments*. The boundaries mark where the topic, subtopic, speaker intention or rhetorical function changes.

> **Definition:** Discourse segmentation is the process of partitioning a text or conversation into contiguous, internally coherent units (discourse segments) by detecting boundaries at which topic, focus, or communicative purpose shifts.

Why is it needed?

- Long documents contain many topics; NLP systems (search, summarisation, QA) work better on topic-coherent chunks.
- It is the first step of higher-level discourse analysis such as discourse parsing (RST, PDTB), coreference resolution, and dialogue understanding.
- Human readers naturally perceive structure (paragraphs, sections, topic changes); the machine must recover it, often without explicit markup (e.g., ASR transcripts, chat logs).

Two broad flavours

Type	Goal	Unit produced	Example
Topic / linear segmentation	Split a document into topically coherent blocks	Topic segment (several sentences / paragraphs)	A news broadcast split into stories
Elementary Discourse Unit (EDU) segmentation	Split sentences into minimal clause-like units for rhetorical analysis	EDU (clause or clause-like phrase)	'Although it rained, we played' = 2 EDUs
Dialogue / utterance segmentation	Split a conversation into turns, dialogue acts or topic episodes	Utterance, dialogue act, episode	Customer-support chat divided into issues

2. Discourse Units and Segments

A discourse is organised hierarchically. Different theories define different 'basic units', but they share the idea that units combine into larger units through coherence relations.

2.1 Hierarchy of units

Level	Description	Example
Word / phrase	Lexical building block; not a discourse unit by itself	'because', 'the model'
Clause / EDU	Minimal unit of rhetorical analysis (RST). Usually a clause with a verb, or a clause-like phrase	'because it was raining'
Sentence	Grammatical unit; may contain one or several EDUs	'We stayed home because it was raining.'
Utterance / turn	Unit of speech by one speaker in dialogue	'Hi, I need to reset my password.'
Paragraph	Orthographic unit; usually one sub-idea, but not always topic-aligned	A 4-6 sentence block
Discourse segment	Contiguous span of sentences sharing a topic or purpose (Grosz and Sidner)	A section on 'Causes of inflation'
Whole discourse	Complete document or conversation	An entire article / meeting

2.2 Key theoretical views

- **Grosz and Sidner (1986) - Intentional structure:** A discourse consists of segments, each serving a *discourse segment purpose (DSP)*. Segments are related by dominance and satisfaction-precedence; attentional state is modelled with a focus stack.
- **Rhetorical Structure Theory (Mann and Thompson, 1988):** Text is split into EDUs; adjacent units are linked by relations (Elaboration, Contrast, Cause, Evidence...) into a tree with nucleus-satellite structure.
- **Penn Discourse Treebank (PDTB):** Focuses on discourse connectives (because, however, although) and their two arguments (Arg1, Arg2) rather than a full tree.
- **Hearst (1997) - Linear topic segmentation:** Text is a flat sequence of topically coherent blocks detected through lexical cohesion.
- **Halliday and Hasan - Cohesion:** Cohesive ties (reference, substitution, ellipsis, conjunction, lexical cohesion) bind sentences within a segment; the density of ties drops at a boundary.

2.3 Segment vs unit - the difference

Discourse unit (EDU) is a *small*, fine-grained piece (clause level) used in discourse parsing. A **discourse segment** is a *larger* span (many sentences) identified by topical or intentional coherence. In many textbooks both are called 'discourse segmentation', so always state which granularity you mean.

2.4 Boundary signals

Signal type	Explanation	Examples
Lexical cohesion	Vocabulary repeats within a segment, changes at a boundary	Words 'bank, loan, interest' then 'river, bank, flood'
Cue phrases / discourse markers	Words that signal a new topic or shift	'Now', 'Next', 'Anyway', 'On the other hand', 'In conclusion'

Signal type	Explanation	Examples
Pronoun / reference chains	Pronouns without nearby antecedent signal a new entity	'He' after a long gap
Syntactic / punctuation cues	Paragraph breaks, headings, commas, conjunctions	New line, numbered heading
Prosodic cues (speech)	Pauses, pitch reset, speaking-rate change	Long pause before a new topic
Speaker / turn change	Change of speaker or dialogue act	Question after an answer
Temporal / spatial shift	Change of time or place in narrative	'Two days later', 'In Paris'

3. Sentence-level vs Discourse-level Processing

Traditional NLP (tokenisation, POS tagging, parsing) works on one sentence at a time. But meaning often depends on what came before or after. Discourse-level processing models relationships *across* sentences.

Aspect	Sentence-level processing	Discourse-level processing
Scope	Single sentence	Multiple sentences / entire document / dialogue
Main question	Is the sentence grammatical? What does it mean on its own?	How do sentences connect? What is the overall structure and purpose?
Typical tasks	POS tagging, parsing, NER, sentence classification	Segmentation, coreference, discourse parsing, coherence, summarisation
Dependencies	Syntactic (within sentence)	Cohesion, coherence relations, anaphora, topic flow
Context needed	Local (words nearby)	Global (earlier and later text)
Ambiguity example	'Bank' resolved using sentence words	'It' in sentence 5 resolved using sentence 2
Output structure	Parse tree, tag sequence	Segments, discourse tree, entity chains, graph
Complexity	Well-defined grammar; less ambiguity	Looser theory; annotation is subjective; larger inputs
Evaluation	Accuracy, F1, parse-eval	Pk, WindowDiff, boundary F1, relation accuracy

Example showing why discourse matters

'Ravi bought a laptop. It was expensive. He returned it the next day. The weather in Pune was hot.' Sentence-level analysis can parse each sentence correctly but cannot say that 'It' = laptop, nor that sentence 4 starts a new topic. A discourse segmenter would place a boundary before sentence 4 (shift from shopping to weather).

Where segmentation sits in the discourse pipeline

Discourse processing stages



4. Characteristics of Discourse Segments and of Segmentation

4.1 Characteristics of a good discourse segment

- **Coherence:** sentences inside a segment are semantically related and focus on one topic or purpose.
- **Cohesion:** strong lexical and referential ties (repeated words, synonyms, pronouns) inside, weak ties across boundaries.
- **Contiguity:** a segment is an unbroken span of text; segments do not overlap in linear segmentation.
- **Topical unity:** one main subtopic or communicative goal (DSP).
- **Hierarchical nature:** segments may contain sub-segments (section, subsection, paragraph).
- **Boundary markers:** usually signalled by cue phrases, paragraph breaks, speaker change, prosody.
- **Variable length:** there is no fixed size; segments range from a couple of sentences to many paragraphs.

4.2 Characteristics (and difficulties) of the segmentation task

- **Subjectivity:** human annotators often disagree on exactly where a boundary lies (inter-annotator agreement is moderate, kappa about 0.4-0.7).
- **Granularity dependence:** the 'right' segmentation depends on the desired level (section vs sub-topic).
- **Domain and genre dependence:** news, lectures, meetings, chats and legal texts all behave differently.
- **Language and modality dependence:** speech has no punctuation, so ASR errors propagate.
- **Gradual topic drift:** topics fade rather than switch abruptly, making boundaries fuzzy.
- **Class imbalance:** boundaries are rare (about 1 in 10-20 sentences), so naive classifiers predict 'no boundary' all the time.
- **Unsupervised-friendly:** because labelled data is scarce, many classical methods need no training.

4.3 Input and output

	Description
Input	Sequence of sentences s1, s2, ..., sn (or tokens / utterances)
Output	Set of boundary positions B = {b1, b2, ...} or label per gap (boundary / no-boundary), or per token (B-EDU / I-EDU)
Constraints	Segments are contiguous, non-overlapping and cover the whole text

5. Discourse Segmentation Techniques

5.1 Overview

Family	Key idea	Representative methods
Cue-phrase / marker based	Use discourse markers and formatting to hypothesise boundaries	Hirschberg and Litman cue phrases; rule lists
Lexical cohesion (similarity) based	Boundary where vocabulary similarity between adjacent blocks dips	TextTiling, C99, LCseg
Probabilistic / topic-model based	Each segment generated by a topic distribution; find best partition	Bayesian segmentation (BayesSeg), LDA-based TopicTiling, HMM
Graph / clustering based	Build similarity matrix or graph; cut it at weak links	C99 (rank matrix clustering), minimum-cut segmentation, dynamic programming (U00)
Supervised machine learning	Classify each sentence gap as boundary or not using features	Naive Bayes, SVM, CRF, decision trees
Neural	Learn representations that predict boundaries end-to-end	Hierarchical BiLSTM, BERT / Transformer segmenters (Text Segmentation as Supervised Learning, Cross-segment BERT)

5.2 Cue-phrase (marker) approach

Maintain a list of markers that frequently start or end a segment: *first, next, finally, now, meanwhile, in conclusion, moving on, by the way, so, okay*. When a marker occurs at the start of a sentence (and often after a paragraph break), a boundary is hypothesised. Precision is good in lectures or instructions, but recall is limited because many topic shifts have no marker.

5.3 Lexical cohesion - TextTiling (Hearst, 1997)

TextTiling assumes that when a topic changes, the set of words used changes. Algorithm outline:

- **Tokenise and normalise:** lower-case, remove stop words, stem.
- **Create pseudo-sentences / token sequences** of fixed length w (about 20 words).
- **Group into blocks** of k sequences (k about 6-10).
- **Compute similarity** at each gap between the block before and the block after using cosine similarity of word-frequency vectors: sim(b1,b2) = sum(w1*w2) / sqrt(sum(w1^2) * sum(w2^2)).
- **Compute depth scores** for each gap: (left peak - valley) + (right peak - valley), where peaks are found by climbing from the valley while scores rise.
- **Pick boundaries** where depth score exceeds a cutoff: mean - sigma/2 (liberal) or mean - sigma (conservative); boundaries are then aligned to the nearest paragraph break.

5.4 C99 and similarity-matrix methods

Choi's C99 builds a sentence-by-sentence similarity matrix, replaces each value by its *rank* within a local neighbourhood (to reduce noise), and then uses divisive clustering to find the partition that maximises the density of high-similarity regions along the diagonal.

5.5 Topic-model and probabilistic methods

- **LDA / TopicTiling:** Replace words by dominant topic IDs inferred by LDA; compute similarity between topic vectors of adjacent windows. Handles synonyms better than raw words.
- **Bayesian / Dynamic programming (U00, BayesSeg):** Find the segmentation that maximises the joint likelihood of the text under a generative model where each segment has a language model. Dynamic programming gives the global optimum.
- **HMM:** Hidden states are topics; a boundary occurs when the state changes; used heavily for broadcast news.

5.6 Supervised segmentation

Treat the problem as binary classification for every gap between sentences. Typical features: cosine similarity before/after, presence of cue words, sentence length, paragraph break, speaker change, pause length (speech), pronoun counts, change in noun phrases, and named-entity overlap. Sequence models (CRF) also capture dependence between neighbouring decisions.

5.7 EDU segmentation (clause-level)

Used for RST parsing. It is typically a **sequence labelling** problem: for each token decide whether it begins a new EDU (tag B) or continues (tag I). Early systems used hand-written rules based on punctuation and connectives; modern systems use BiLSTM-CRF or BERT taggers and reach about 93-96 F1 on the RST-DT corpus.

5.8 Neural approaches

- **Hierarchical BiLSTM:** a lower BiLSTM encodes each sentence from words; an upper BiLSTM encodes the sentence sequence; a softmax predicts whether each sentence ends a segment.
- **Transformer / BERT:** contextual embeddings of sentences; cross-segment attention across a window of neighbours; fine-tuned on Wiki-727K or Choi datasets.
- **Coherence-based training:** models are trained to distinguish natural from shuffled sentence order, which implicitly learns boundaries.
- **LLM prompting:** large language models can be prompted to mark topic changes zero-shot, useful in low-resource domains but costly for long documents.

6. Rule-based vs Statistical Segmentation

Criterion	Rule-based	Statistical / machine-learning
Core idea	Hand-crafted patterns: punctuation, cue phrases, formatting, syntactic rules	Learn patterns or probabilities from data (counts, features, embeddings)
Knowledge source	Linguist expertise	Annotated or raw corpora
Training data	Not required	Required for supervised; raw text for unsupervised (TextTiling, topic models)
Transparency	High - every decision is explainable	Lower, especially for neural models
Accuracy	High precision, low recall in open domains	Better balance, higher overall F1 in most benchmarks
Domain portability	Poor; rules must be rewritten	Moderate; needs retraining or adaptation (transfer learning helps)
Robustness to noise	Brittle (ASR errors, informal text)	More robust, particularly with embeddings
Development cost	Labour-intensive rule engineering	Annotation / compute cost
Speed	Very fast	Fast (classical) to slow (large neural)
Examples	Marker lists; regex for 'Chapter N'; EDU rules splitting at ',' + 'which/because'	TextTiling (statistical, unsupervised), C99, LDA, SVM, CRF, BiLSTM, BERT
Best for	Structured documents, manuals, legal text, bootstrapping labels	Open-domain text, lectures, news, conversational data

Hybrid approach (common in practice): Rules supply strong, high-precision cues (headings, speaker changes, markers) as *features* to a statistical or neural model, which then decides on ambiguous cases. This combines interpretability with generalisation.

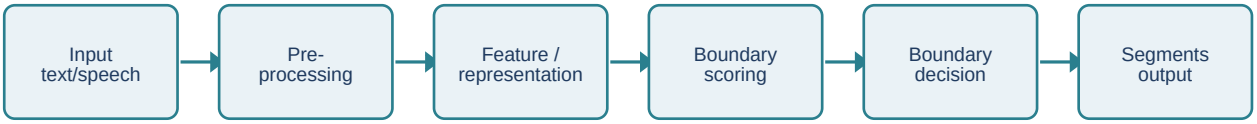
Quick comparison of statistical sub-types

Sub-type	Supervision	Example	Pros	Cons
Unsupervised statistical	None	TextTiling, C99	No labelled data; simple	Parameter tuning; coarse
Supervised classical	Labelled boundaries	SVM / CRF	Uses rich features	Needs annotation; feature engineering
Neural supervised	Large labelled corpora	BiLSTM, BERT	State-of-the-art; semantic similarity	Data and compute hungry; less interpretable

7. Architectures

7.1 General discourse segmentation pipeline

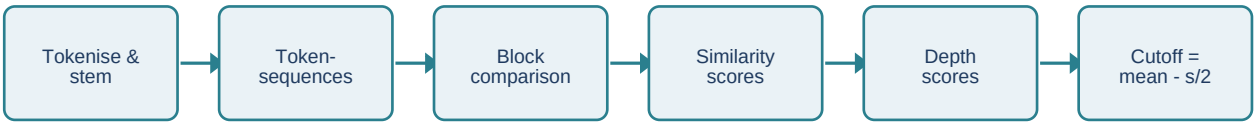
Generic architecture



Stage	What happens
Input	Plain text, HTML, ASR transcript, chat log
Pre-processing	Sentence splitting, tokenisation, lower-casing, stop-word removal, stemming / lemmatisation; for speech: ASR, pause and speaker information
Representation	Bag-of-words / TF-IDF vectors, topic distributions (LDA), sentence embeddings (SBERT), cue-word flags, prosodic features
Boundary scoring	Similarity between adjacent windows (unsupervised) or classifier probability (supervised) at each gap
Boundary decision	Threshold, depth-score cutoff, dynamic programming, or argmax; optional smoothing and paragraph alignment
Output	List of segments (start, end) and optional labels or titles

7.2 TextTiling architecture

TextTiling (Hearst 1997)



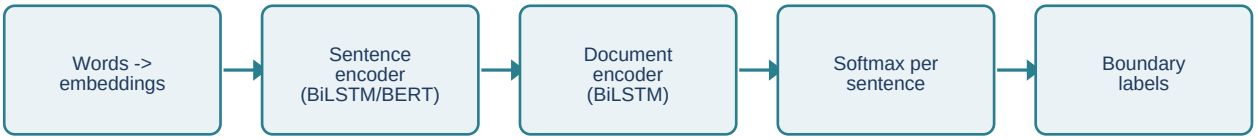
7.3 Supervised (feature-based) architecture

Supervised segmentation



7.4 Neural hierarchical architecture

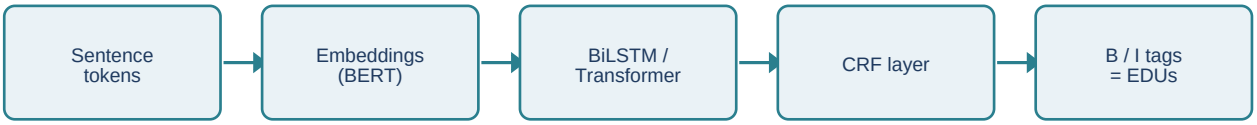
Hierarchical neural segmenter



The lower encoder turns each sentence into a vector; the upper encoder contextualises the sentence vectors across the document; a classifier outputs 1 if the sentence is the *end of a segment*. Loss: cross-entropy over sentences. Decoding may use a threshold or CRF layer.

7.5 EDU segmentation architecture (sequence tagging)

Clause-level EDU segmenter



8. Step-by-Step Worked Examples

Example A - Topic segmentation with TextTiling (numerical)

Step 1 - Input. A 20-sentence passage; after pre-processing we group sentences into blocks and compute similarity at each of the gaps. Sentences 1-8 discuss *cricket match*, sentences 9-14 discuss *stock market*, sentences 15-20 discuss *cricket again* (or another topic).

Step 2 - Build vectors. For each gap, form bag-of-words vectors for the block on the left and on the right. Example for gap 3: left block {match, bat, wicket, over, bat} and right block {stock, share, price, market, bat}.

Step 3 - Cosine similarity. Suppose we obtain the following scores at gaps g1...g7 (sample illustrative data):

Gap	g1	g2	g3	g4	g5	g6	g7
Similarity	0.62	0.58	0.12	0.55	0.60	0.15	0.50

Step 4 - Depth scores. For each gap, climb left and right while the score increases to find the peaks; depth = (left peak - gap score) + (right peak - gap score).

- g3 (0.12): left peak 0.62, right peak 0.60 -> (0.62 - 0.12) + (0.60 - 0.12) = **0.98**
- g6 (0.15): left peak 0.60, right peak 0.50 -> (0.60 - 0.15) + (0.50 - 0.15) = **0.80**
- g2 (0.58): left peak 0.62, right side falls -> 0.04 + 0 = **0.04**
- g4 (0.55): left falls, right peak 0.60 -> 0 + 0.05 = **0.05**
- g1, g5, g7 are peaks or end points -> **0**

Gap	g1	g2	g3	g4	g5	g6	g7
Depth	0	0.04	0.98	0.05	0	0.80	0

Step 5 - Cutoff. Mean depth = 1.87 / 7 = 0.267; standard deviation sigma is about 0.40. Liberal cutoff = mean - sigma/2 = 0.267 - 0.20 = **0.07**.

Step 6 - Decide boundaries. Gaps with depth above 0.07 are boundaries: **g3 and g6**. (g2 and g4 are below the cutoff.)

Step 7 - Align to paragraphs and output. Move each boundary to the nearest paragraph break. Result: three segments - Segment 1 (cricket) | Segment 2 (stock market) | Segment 3 (cricket/other).

> **Result:** TextTiling found two topic shifts exactly where the vocabulary similarity dipped (g3 and g6), producing three coherent segments.

Example B - Cue-phrase / rule-based segmentation of a lecture transcript

```
1 Welcome everyone. Today we study sorting.
2 First, let us look at bubble sort.
3 It repeatedly swaps adjacent elements.
4 Now, moving on to merge sort.
5 Merge sort divides the list into halves.
6 Finally, let us compare their running times.
7 Bubble sort is O(n^2) while merge sort is O(n log n).
```

Step 1: Split into sentences (7). **Step 2:** Load marker list {first, now, moving on, finally, next}. **Step 3:** Scan each sentence start: S2 'First' -> boundary before S2; S4 'Now, moving on' -> boundary before S4; S6

'Finally' -> boundary before S6. **Step 4:** Output segments:

Segment	Sentences	Topic
1	S1	Introduction
2	S2-S3	Bubble sort
3	S4-S5	Merge sort
4	S6-S7	Comparison of complexity

Example C - EDU segmentation (clause level)

Sentence: *'Although the team trained hard, they lost the final because the captain was injured.'*

- **Step 1 - Tokenise:** Although | the | team | trained | hard | , | they | lost | the | final | because | the | captain | was | injured | .
- **Step 2 - Find candidate split points:** subordinating conjunctions (Although, because) and the comma after a subordinate clause.
- **Step 3 - Apply rules / tagger:** 'Although' starts a clause -> tag B; the comma + main clause 'they' -> tag B; 'because' -> tag B. All other tokens -> I.
- **Step 4 - Output EDUs:**
  [EDU1: Although the team trained hard,] [EDU2: they lost the final] [EDU3: because the captain was injured.]
- **Step 5 - Use downstream:** A discourse parser links EDU1-EDU2 with *Concession* and EDU2-EDU3 with *Cause*.

Example D - Supervised classification (feature-based) mini-walkthrough

Gap	Cosine sim.	Cue word next?	Paragraph break?	Speaker change?	Predicted
s3 | s4	0.10	Yes ('Now')	Yes	No	Boundary (p=0.93)
s4 | s5	0.71	No	No	No	No boundary (p=0.05)
s6 | s7	0.45	Yes ('Finally')	No	No	Boundary (p=0.61)

Steps: (1) extract features for every gap; (2) feed to the trained classifier (e.g., logistic regression: p = sigmoid(w.x + b)); (3) apply threshold 0.5; (4) cut at gaps with p > 0.5; (5) evaluate with Pk / WindowDiff against gold boundaries.

Example E - Dialogue segmentation

```
A: My internet is not working.
B: Did you restart the router?
A: Yes, still nothing.
B: I will raise a ticket. [boundary]
A: Also, can you tell me my bill amount?
B: Sure, your bill is Rs 899.
```

Segmenter detects change of intent (technical fault -> billing), triggered by the marker 'Also' and a drop in word overlap. Segment 1 = fault resolution; Segment 2 = billing query.

9. Evaluation Metrics

Metric	Meaning	Note
Precision / Recall / F1	Exact boundary match against gold	Harsh: a boundary one sentence off counts as an error
Pk (Beeferman et al.)	Probability that two sentences k units apart are wrongly judged as in the same / different segment	Lower is better; k = half of average segment length
WindowDiff (Pevzner and Hearst)	Slides a window; counts windows where predicted and reference boundary counts differ	Fixes Pk's bias to false negatives; lower is better
Boundary Similarity (B)	Edit-distance-based, tolerant to near-misses	More recent and fine-grained
Inter-annotator agreement	Kappa between human annotators	Sets the upper bound on performance

Benchmark datasets: Choi (synthetic), Wiki-727K and Wiki-50 (Wikipedia sections), Moonshot / ICSI meeting corpus, Broadcast News, RST-DT (for EDUs), PDTB.

10. Applications of Discourse Segmentation

Application	How segmentation helps	Example
Text summarisation	Summarise each topic segment, then combine, giving balanced coverage	News digest with one sentence per sub-topic
Information retrieval and passage retrieval	Index and return the relevant segment rather than the entire document	Search engine showing the exact section answering a query
Question answering / RAG	Topic-coherent chunks give better retrieval contexts for LLMs	Chunking manuals before embedding in a vector database
Meeting and lecture browsing	Automatic chaptering of transcripts and videos	YouTube chapters, meeting minutes
Dialogue systems and chatbots	Detect topic or intent switches; manage context and slot filling	Customer support bots handling multiple issues
Discourse parsing (RST / PDTB)	EDU segmentation is the mandatory first step	Building discourse trees for essays
Machine translation	Translate with document-level context; avoid cross-topic confusion	Consistent pronoun translation within a segment
Text classification and topic detection	Assign topics per segment rather than per document	News stream topic tracking
Anaphora / coreference resolution	Restrict antecedent search to the current segment	Resolving 'it', 'they'
Sentiment and opinion mining	Separate aspects discussed in a review	Review: camera (positive) | battery (negative)
Speech processing	Segment ASR output of podcasts, broadcast news	Story boundary detection
Legal, medical and scientific text	Identify sections (facts, ruling; symptoms, treatment)	Clinical note structuring

Application	How segmentation helps	Example
Essay scoring and writing support	Assess coherence and organisation	Automated essay evaluation
Text generation / LLM data prep	Split long documents into coherent chunks for training or prompting	Long-document processing

11. Challenges and Future Directions

- Ambiguity and subjectivity of boundaries; need for multiple reference annotations.
- Hierarchical (multi-level) segmentation is harder than linear segmentation.
- Low-resource languages and domains lacking annotated data.
- Noisy input: ASR errors, informal chat, code-mixed text.
- Long documents exceed transformer context length; efficient long-range models are needed.
- Integrating multimodal cues (video, slides, prosody).
- Explainable and controllable segmentation using LLMs.

12. Quick Revision Summary

Point	Key takeaway
Definition	Dividing a discourse into coherent contiguous segments by detecting topic or purpose shifts
Units	EDU (clause), sentence, utterance, paragraph, discourse segment
Sentence vs discourse	Sentence-level = local grammar and meaning; discourse-level = cross-sentence coherence and structure
Techniques	Cue phrases, lexical cohesion (TextTiling, C99), topic models, supervised ML, neural
Rule vs statistical	Rules = precise, explainable, brittle; statistical = adaptable, data-driven, better recall
TextTiling formula	Cosine similarity -> depth score -> cutoff (mean - sigma/2)
Metrics	Pk, WindowDiff, boundary F1
Applications	Summarisation, IR, QA/RAG, dialogue, discourse parsing, MT, meeting browsing

13. Likely Exam Questions

1. Define discourse segmentation. Explain discourse units and segments with examples.
2. Differentiate between sentence-level and discourse-level processing.
3. Explain the TextTiling algorithm with a numerical example.
4. Compare rule-based and statistical discourse segmentation.
5. Describe the architecture of a neural discourse segmenter.
6. What are EDUs? How are they identified? Give an example.
7. Explain Pk and WindowDiff.
8. List and explain five applications of discourse segmentation.