Digital Scholarship in the Humanities, 2025, **40**, 477–486
https://doi.org/10.1093/llc/fqaf033
Advance access publication 30 April 2025
Full Paper

OXFORD

Improving text collations by local text resegmentation

Janis Dähne *[1], **Jörg Ritter** [1], **Paul Molitor** [1]

[1]Institute for Computer Science, Martin Luther University Halle-Wittenberg, Von-Seckendorff-Platz 1, Halle, 06120, Germany
*Corresponding author. Institute for Computer Science, Martin Luther University Halle-Wittenberg, Von-Seckendorff-Platz 1, Halle, 06120, Germany. E-mail: janis.daehne@informatik.uni-halle.de

Abstract

In almost all current approaches, the collation of large texts is applied to a fixed given segmentation of the two texts witnesses to be compared and consists of two consecutive steps. First, the segments of the two texts are aligned, and then the aligned segments are compared in detail. For larger manuscripts or books consisting of many pages, the segments are usually the paragraphs of the texts. When comparing two texts, where the second text is a revised version of the first, poor local alignments can arise. This occurs in places where paragraphs have been split into two smaller paragraphs to insert a new paragraph in between, or where several consecutive sentences have been moved from one paragraph to the previous or next paragraph. Most paragraph collation tools cannot handle these scenarios properly because they align each paragraph with at most one paragraph of the other text. In this paper, we discuss this problem in detail and present a heuristic for resegmenting the two texts to be compared in order to achieve a better collation.

Keywords: digital humanities; eHumanities; collation segment alignment; text segmentation; resegmentation.

1. Introduction

The task of finding differences and similarities between given texts is part of many scientific questions in the Humanities, for example, textual criticism. To address such questions, the collation of textual witnesses is typically the first step, nowadays often performed using digital tools like CollateX (Haentjens Dekker *et al.* 2015) or LERA (Pöckelmann *et al.* 2023). Nury and Spadini (2020) provide a comprehensive overview of available collation tools. For larger manuscripts and books, the collation process is generally divided into two steps: aligning paragraphs and subsequently performing a detailed comparison of the aligned paragraphs.[1]

There are several scenarios in which the (given) segmentation causes problems for the collation, for example, when a large part of a paragraph has been removed during the revision of a text. If the original text is to be collated with the revised text, it would make sense to split the paragraph of the original text in order to get matching pairs again. Another scenario occurs when a paragraph is divided during revision to insert a new paragraph with additional content. Again, splitting the original paragraph would similarly enable the collation tool to achieve better results. A third scenario is that the revision has moved several consecutive sentences from one paragraph to an adjacent paragraph. Again, the collation tool would not find a *good* collation unless it is allowed to (slightly) change the given segmentation. Even when collation is performed manually, resegmentation can be a complex task. Scholars must handle numerous segments containing many sentences, requiring a lot of scrolling through pages to identify similar segments or sentences. This process requires careful attention to individual sentences, while maintaining an overview of the overall segmentation. See Section 3.2 for such an example, albeit a small one, that illustrates the problem.

To the best of our knowledge, there are no collation tools that can change the segmentation of the given texts to be compared in order to achieve a better synoptic representation. The only tools that go a little bit in this direction are approaches for N-to-M (or many-to-many) sentence alignments, such as VecAlign (Thompson and Koehn 2019) and SentAlign (Steingrimsson, Loftsson, and Way 2023). The sentence alignments generated by these approaches do indeed imply a (new) segmentation of the sentences. However, these approaches only recognize the situation where a sentence has been

split into two or more smaller sentences without inserting a new thought between them. Moreover, these tools have so far only worked at the sentence level.

There are several approaches to text segmentation in the literature. To the best of our knowledge, they all aim to divide an *independent* text into meaningful units. They can be divided into two main categories, as described by Yaari (1997): lexical cohesion methods and multiple source methods. Lexical cohesion, an approach proposed by Halliday and Hasan (2014) and refined, for example, by Kozima (1993) and Hearst (1994), is the idea that two segments containing similar words may belong to the same topic and thus to the same text segment. Multiple source methods use a combination of lexical cohesion and other indicators, for example, prosodic features in the case of spoken text to be segmented (see, e.g. Litman and Passonneau 1995).

The goal of our approach is not to divide an *independent* text into semantically related segments, but to segment two versions of a text (or two witnesses of the same manuscript) in such a way that a good collation between the two witnesses is possible. The original segmentation should be preserved as much as possible, that is, the original segmentation should only be changed if necessary.

This article is structured as follows: Section 2 gives an overview over the proposed heuristic, its goals and assumptions. Sections 2.1–2.4 describe in detail the main steps of the heuristic (preprocessing, classification, and reconstruction). This article concludes with experimental results (Section 3) and some concluding remarks (Section 4).

2. The overall approach

In this article, we focus explicitly on collating *two* versions of a text. The two versions themselves are already segmented, with each segment consisting of one or more sentences. Our approach will work with two alignments: first, the alignment on the segment-level as originally given, and second, an additional alignment on the sentence-level, irrespective of the segmentations of both text versions. The goal of our approach is to locally improve the given segment alignment by looking at which sentences have been aligned in the sentence alignment. In fact, if the sentence alignment pairs two sentences, but the segments containing those sentences are not aligned, resegmentation of those two segments may be appropriate for better collation. The sentence alignment indicates where the sentences *want* to move (to which sentence they are aligned), and the segment alignment indicates where the sentences should be placed. To uniquely identify the parent segment of a sentence, our approach requires a one-to-one sentence alignment. We expect the segment alignment to have no text transpositions, since our computed sentence alignment does not have them either, and they would cause problems later by introducing conflicting classifications for sentences.

The goal of this approach is to improve the collation of the two text versions as much as possible, while making as few changes to the original segmentation as possible, and thus preserving as much of the given segment alignment as possible.

Starting from an existing segment alignment, the approach consists of three consecutive steps: calculation of a sentence alignment (see Section 2.1), classification of the sentences (Sections 2.2 and 2.3), and calculation of the new segment alignment (Section 2.4).

2.1 Sentence alignment

The first step in our approach is to compute a sentence alignment. This can be done using existing tools such as TSAligner (Dähne *et al.* 2022), VecAlign (Thompson and Koehn 2019), SentAlign (Steingrimsson, Loftsson, and Way 2023), or CollateX (Haentjens Dekker *et al.* 2015). In order for each sentence to know which parent segment it is contained in, the aligner must compute a one-to-one alignment. Alternatively, a preexisting one-to-one sentence alignment can be used, if available.

We decided to use the aligner from *Putting collation of text witnesses on a formal basis* (Dähne *et al.* 2022). Their approach is to use a weighted graph to model the alignment problem, where two sentences can only be aligned if the distance between them is less than some constant. They use dynamic programming to compute an alignment, which in turn is based on Needleman and Wunsch's approach to aligning amino acid sequences back in 1970 (Needleman and Wunsch 1970). It is not limited to paragraphs and can work at any text level as long as there is a distance function. In principle, this allows us to experiment with different distance functions for the sentence alignment in the context of our problem.

For this article, we used the same distance function as in the original paper, which is the Jaccard distance (Jaccard 1901). The worst-case runtime of the algorithm is $\mathcal{O}(N \cdot M)$, where N and M are the number of sentences in the texts. The implementation takes about 53 s to compute a sentence alignment with about 8,000 sentences in each of the two texts.[2]

2.2 Sentence classification

In the second step, the original segment alignment is 'compared' with the sentence alignment calculated in the first step. This comparison allows conclusions to be drawn as to whether a sentence should remain in its segment in order to achieve a good alignment of the segments, or whether a resegmentation at this point might be appropriate.

More formally, and without loss of generality, let the first text to be compared be denoted by $\mathcal{T}_1$, and the second

text by $\mathcal{T}_2$. Let Φ be the function representing the given segment alignment. The function Φ maps each segment of $\mathcal{T}_1$ to either a segment of $\mathcal{T}_2$ or $\perp$, where $\perp$ indicates that the segment is not aligned with any of the segments of $\mathcal{T}_2$. Formally, for each segment S of $\mathcal{T}_1$, $\Phi(S)$ corresponds to its aligned segment in $\mathcal{T}_2$ or $\perp$:

$$\Phi(S) = \begin{cases} P, & \text{if segment } S \text{ is aligned with segment } P \text{ of } \mathcal{T}_2 \\ \perp, & \text{if } S \text{ is not aligned with any segment of } \mathcal{T}_2 \end{cases}$$

Φ also maps each segment of $\mathcal{T}_2$ either to a segment of T_1 or to $\perp$.

Similarly, let ϕ represent the alignment of the sentences computed in step one (see Section 2.1). It assigns each sentence of $\mathcal{T}_1$ either to a sentence of $\mathcal{T}_2$ or to $\perp$ and each sentence of $\mathcal{T}_2$ either to a sentence of $\mathcal{T}_1$ or to $\perp$.

Finally, we need a function that maps a sentence of a text to the segment in which the sentence is contained. Let *Seg* be this function. For each sentence X of text version $\mathcal{T}_1$ or $\mathcal{T}_2$, $Seg(X)$ denotes the segment of $\mathcal{T}_1$ or $\mathcal{T}_2$ containing sentence X.

Each sentence X is now classified by assigning it to one of three classes:

$$class(X) = \begin{cases} fitting, & \Phi(\mathrm{Seg}(X)) = \mathrm{Seg}(\phi(X)) \quad \phi(X) \neq \perp \\ non-fitting, & \Phi(\mathrm{Seg}(X)) \neq \mathrm{Seg}(\phi(X)) \quad \phi(X) \neq \perp \\ not-aligned, & \text{otherwise} \end{cases}$$

If a sentence is not aligned by the sentence alignment, the sentence is classified as *not-aligned*.

The function *class* classifies each sentence of both text versions. Knowing that the alignment functions Φ and ϕ are symmetric for aligned segments and sentences, we only have to evaluate the function *class* for the sentences of $\mathcal{T}_1$, the aligned sentences of $\mathcal{T}_2$ get the same classification. The remaining sentences of $\mathcal{T}_2$ get the classification *not-aligned*.

The idea of the *fitting* class is that these sentences are already fitting in a sense, that they must not be moved, that is, resegmentation is not necessary at this point. This also ensures that the improved alignment stays close to the original segment alignment.

The *non-fitting* class contains those sentences that match a sentence from $\mathcal{T}_2$, but their parent segments are not aligned. Therefore, it should be checked if it would be better to move these sentences to another segment.

Finally, the *not-aligned* sentences are inserts or deletes and have no partner in the sentence alignment.

Figure 1 illustrates the three classes.

2.3 Reclassification of sentences

We will not try to move all *non-fitting* sentences to other segments, as we want to preserve as much of the original segment alignment as possible. For example, if there are aligned segments between the parent segments of two *non-fitting* sentences in the original segment alignment, we do not align the two *non-fitting* sentences as the user may have already manually changed the alignment and those changes would be lost. Therefore, we reclassify these sentences by putting them into the *not-aligned* class.

To make the sentences easier to handle, we group successive sentences of a segment into a *sentence block* if they belong to the same class. If they belong to the *not-aligned* class, we call them *not-aligned sentence blocks*, or *NABs* for short. If they belong to the *non-fitting* class, they must all be aligned to sentences with the same parent segment of $\mathcal{T}_2$ (see Figs 3 and 4, but without the *NAB*). This reduces the number of subsequent operations. In the following, we will only refer to *sentence blocks* instead of individual sentences.

After all sentences have been classified and grouped into *sentence blocks* as described, it is clear how to handle the *fitting* and *non-fitting sentence blocks*, keeping the former and moving the latter. However, it is not clear how to handle *sentence blocks* that contain sentences that are classified as *not-aligned*, because they don't belong anywhere intrinsically.

One way to handle *NABs* is to separate them from their parent segments by introducing a new segment containing only the *NAB*. We have chosen not to do this, as it would result in too many changes to the original segment alignment. Our approach attempts to reclassify the *NABs* according to the immediately adjacent sentence blocks. For this reclassification, only immediately adjacent sentence blocks belonging to the same parent segment as the *NAB* in question are considered. The following cases exist:

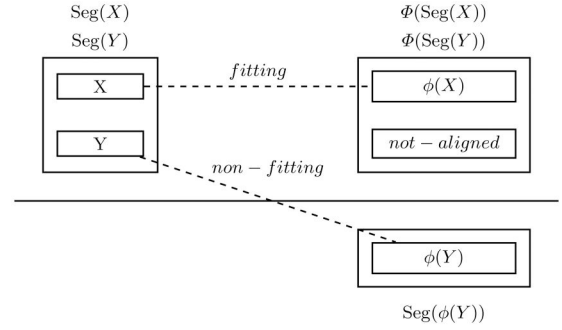

Figure 1. Illustration of the three classes *fitting*, *non-fitting*, and *not-aligned*: Sentence X is *fitting* because it is aligned and its two respective parent segments are also aligned. Sentence Y is *non-fitting* because it is aligned with a sentence $\phi(Y)$ contained in a segment that is not aligned with the parent segment of Y. The sentence in the middle of $\mathcal{T}_2$ is not aligned and is therefore classified as *not-aligned*.

- If both adjacent sentence blocks of a *NAB* have the same classification and the same target segment, that is, the sentences of both neighboring sentence blocks are aligned with sentences of the same parent segment, then we also assign this classification to the *NAB* and merge the three sentence blocks (see Figs 2 and 3).
- If the two adjacent sentence blocks are in the *non-fitting* class, but have different target segments, the three sentence blocks must not be merged. In this case, the *NAB* is only merged with the preceding sentence blocks. This makes it easier to recognize whether two sentence blocks are similar, since only the beginning needs to be read (see Fig. 4).
- If a *NAB* has only one direct adjacent sentence block because it is the first or last sentence block in its segment, we assign it the same classification as the adjacent sentence block and merge the two sentence blocks.
- If a *NAB* is the only sentence block in its segment, it retains its classification.
- The only remaining case is when a *NAB* has two directly adjacent sentence blocks assigned to different classes. In this special case, we decided to merge the *NAB* with the preceding sentence block, as in the case of two adjacent sentence blocks of class *non-fitting* and different target segments (see Fig. 5).

2.4 Construction of the improved alignment

The construction of the new alignment can be considered straightforward. It goes through the sentence alignment from top to bottom handling each *sentence block* individually. *Sentence blocks* that are *fitting* remain in their segments, and the given segment alignment is not changed at this point. *Non-fitting sentence blocks* are cut out of their segments and each form a new segment, which is equivalent to resegmentation at these points. The new segments of *non-fitting sentence blocks* which were assigned to each other by the sentence alignment, are aligned in the new alignment.

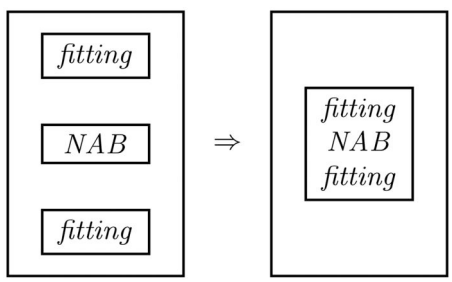

Figure 2. Illustration of the reclassification of *NABs* where both adjacent *sentence blocks* in a segment have the same *fitting* classification. The three sentence blocks are merged into one larger *fitting* sentence blocks.

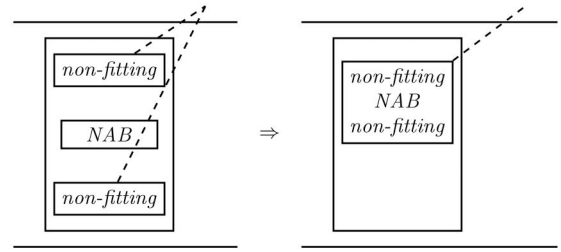

Figure 3. Illustration of the reclassification of *NABs* where both adjacent *sentence blocks* in a segment have the same *non-fitting* classification with the same target segment. The three *sentence blocks* are merged into one larger *non-fitting sentence block*.

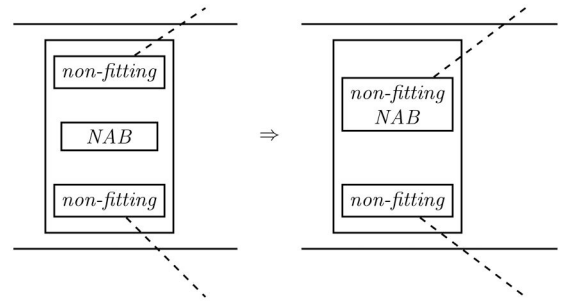

Figure 4. Illustration of the reclassification of *NABs* where both adjacent *sentence blocks* in a segment have the same *non-fitting* classification but different target segments. The *NAB* is merged with the preceding *non-fitting sentence block*.

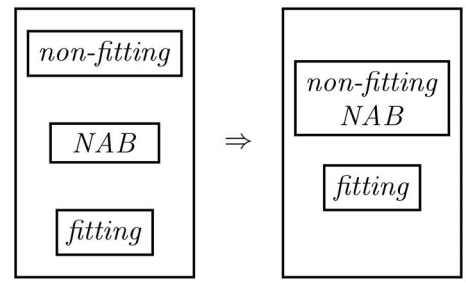

Figure 5. Illustration of the reclassification of *NABs* where both adjacent *sentence blocks* in a segment have different classifications. The *NAB* is merged with the preceding *sentence block*.

There are two special cases to consider:

As a result of the reclassification step, a segment either contains no *NAB* or consists only of this *NAB*. Thus, a *NAB* can usually be placed in its own alignment row (if it is not already). However, if the parent segment of a *NAB* is aligned in the original segment alignment (e.g. through manual intervention by the scholar) with another segment (which can only contain one *NAB* or one *non-fitting sentence block*), the *NAB* should not be placed in its own alignment row. This ensures that the

two segments are still aligned as they were in the original alignment. If the *NAB* was aligned with a *non-fitting sentence block*, the *non-fitting sentence block* is moved to its aligned partner block. When implemented in an interactive environment, it may be desired that the *non-fitting sentence block* is not moved in order to maintain the (manually adjusted) alignment. This can be achieved by marking the *non-fitting sentence block* and treating it as a *NAB*.

The other special case is that *NABs* can interfere with the alignment of two *non-fitting sentence blocks* that are aligned by the sentence alignment. In this case, the *NABs* must be moved either up with the lower *non-fitting sentence block* or down with the upper *non-fitting sentence block* to maintain the original sentence order.

3. Experimental results

In this section, we discuss our experimental results and show that our approach computes resegmentations that lead to significantly better alignments. We conducted our experiments on four freely available texts in two different versions. For the segment alignments, we used the segment alignments computed by LERA (Pöckelmann *et al.* 2023) when applied to the text versions with the original given segmentations.

3.1 Test data

We used four texts as test data. The first text *Der Sinn und Wert des Lebens* in its versions of 1907 (Eucken 1907) and 1914 (Eucken 1914)—we use siglum *SWL* for it in the following—was written by Rudolf Eucken. The book is the shortest text we used during the experiments. It is about 1,000 sentences long. In this treatise, Eucken shows that the closer the human race is connected to real life, the stronger its confidence becomes that it will find the meaning and value of life in the real world beyond religion and metaphysics.[3]

The second text is *Nahar* by Ernst Weiß in the 1922 (Weiß 1922) and 1930 (Weiß 1930) versions, each of which is about 2,000 sentences long and very similar in terms of the distance measures commonly used. It forms the second part of the novel *Tiere in Ketten* which tells the story of a prostitute and the transmigration of a woman's soul into a tiger.

The third text is book six of the *Histoire philosophique et politique des établissements et du commerce des Européens dans les deux Indes* by Guillaume-Thomas François Raynal which is discussed in (Bremer *et al.* 2015) with respect to its textual variants and their collation with LERA. The *Histoire* is one of the most widely read works of the Age of Enlightenment in which Raynal criticizes European colonization in Latin America. The Histoire was revised three times and further radicalized each time. We used the 1780 (Raynal 1780) and 1820 (Raynal 1820) versions as test data in our experiments, which were taken from the LERA demo page.[4] It has about the same number of sentences as *Nahar*.

The fourth text we used as test data is the novel *Der grüne Heinrich* (Keller 1854/1855; 1879/1880) by Gottfried Keller, in the 1854 and 1879 versions. This is the longest text we have used in the experimental analysis of our approach, with about 8,000 and 9,000 sentences, respectively. It is largely autobiographical and deals with the life of the title character, Heinrich Lee, how he grows up in poor circumstances without a father, and his dream of becoming a landscape painter.

We extracted the ePub versions of the texts from the Gutenberg project[5] and used publicly available converters[6,] to obtain the raw texts. The texts were normalized for distance calculation, taking into account different formatting and languages. Specifically, for the Jaccard distance used in sentence alignment, we first extracted words by splitting at spaces and removing stop words depending on the language used. The remaining words were then normalized by applying unicode canonical decomposition (NFD) and removing all characters that are not letters or spaces (using Unicode properties). At the text level, we preprocessed the paragraphs and converted tabs and line breaks to single whitespaces. This is not the most sophisticated normalization, but it is appropriate for our use case.

Finally, we extracted the sentences using pySBD (Sadvilkar and Neumann 2020) and aligned them with the approach presented by Dähne *et al.* (2022).

3.2 The quality of resegmentation demonstrated by an example

There is no mathematically formal (or algorithmic) method for evaluating the quality of an alignment (without a ground truth) that meets the requirements of (all) philologists, especially since the quality of an alignment also depends on the humanities research question to be investigated. The philologist is responsible for the final evaluation of a collation and cannot be replaced by a formal, computable measure.

For this reason, we first show the result of our approach on a real example by explicitly presenting the segmentation and leaving it to the reader to judge whether the resegmentation performed by our approach has significantly improved the alignment.

The synopsis in Table 1 shows an excerpt from the segment alignment computed by LERA for the original segments of the two versions of the text *Der grüne Heinrich*, in which we have added geometric icons to the (not yet aligned) text passages found by our resegmentation approach that should be aligned. The presented heuristic correctly identifies several well-matching parts and locally resegments the originally

Table 1. Excerpt of the original segment alignment of the two versions of *Der grüne Heinrich*.

Row	Der grüne Heinrich (1855)	Der grüne Heinrich (1879)
1192	Nachdem nun, was eine Stadt baut und ziert und von ihr liebend gehegt wird, vorangegangen, trat gewissermaßen die Stadt selbst auf, wenn der nun folgende Zug von jenem irgend noch zu trennen ist; denn beide zusammen machten ja das Ganze aus, und sein rühmliches Wohl kannte nur einen Boden für seine Wurzeln. ■	
1193	Von zwei bärtigen Hellebardierern begleitet wurde das große Stadtbanner getragen. Hoch trug der kecke Träger im weiß und roten, üppig geschlitzten Kleide die wallende Fahne, die eine Faust stattlich in die Seite gestemmt und anmutig den Fuß vorsetzend. […] ♠	
1194	Ihm folgten gleich die beiden Bürgermeister, staatsmännischen und weisen Ansehens, dann der Syndikus und die Ratsherren, unter denen manch ein im weiten Reich angesehener und demselben ersprießlicher Mann war. ★	
1195	Von den beiden Stadtschreibern, welche nebeneinander gingen, war der eine schmächtige Schwarzgekleidete, mit der schön geschnitzten Elfenbeinbrille auf der Nase, in Wirklichkeit der Literator der Künstlerschaft und der gelehrte Beschreiber des Festes. […]	
1196		[…] Unmittelbar voran ging ihm der Edelknabe mit dem Wappenschilde, der in blauem Felde drei silberne Schildchen zeigt und von Maximilian dem großen Meister für die ganze Künstlerschaft gegeben worden ist. ◆
1197	Den Schluß bildeten nun die festlichen Reihen der ehrbaren Geschlechter. Seide, Gold und Juwelen glänzten hier in schwerem Überfluß. Diese kaufmännischen Patrizier, deren Güter auf allen Meeren schwammen, die zugleich in kriegerischer Haltung mit dem selbst gegossenen trefflichen Geschütze ihre Stadt verteidigten und an Reichskriegen teilnahmen, übertrafen den Adel an Pracht und Reichtum und unterschieden sich von ihm durch Gemeinsinn und sittliche Würde, vom gemeinen Bürger aber durch weitsehenden Blick und umfassenden erhaltenden Sinn. […]	Nachdem nun, was eine Stadt baut und ziert, vorangegangen, trat gewissermaßen die Stadt selbst auf. ■
		Von zwei bärtigen Hellebardieren begleitet, wurde ihr das große Banner vorgetragen. Hoch trug der kecke Fähndrich die wallende Fahne, im üppig geschlitzten Kleide, die linke Faust stattlich in die Seite gestemmt. […] ♠
		Ihm folgten Bürgermeister, Syndikus und Ratsherren, unter ihnen manch ein im weiten Reich angesehener und ersprießlicher Mann, und endlich die festlichen Reihen der Geschlechter. ★
		Seide, Gold und Juwelen glänzten hier in schwerem Überfluß. Die kaufmännischen Patrizier, deren Güter auf allen Meeren schwammen, die zugleich in streitbarer Haltung mit dem selbstgegossenen Geschütze die Stadt verteidigten und an den Reichskriegen teilnahmen, übertrafen den mittleren Adel an Pracht und Reichtum wie in Gemeinsinn und sittlicher Würde. […]

Note: The segments with a geometric shape represent the text passages found by our approach that should be aligned after a resegmentation of the versions. The segment with the diamond icon (green box) on the right corresponds to a text passage of the 1855 version from a line before row 1192.

Table 2. Improved segment alignment of the two versions of *Der grüne Heinrich* after resegmentation and realignment.

Row	Der grüne Heinrich (1855)	Der grüne Heinrich (1879)
1177	[…] Einzeln ging jetzt ein schöner Edelknabe mit dem Wappen, das in himmelblauem Felde drei silberne Schildchen zeigt und von Maximilian dem großen Meister für die ganze geehrte Künstlerschaft gegeben worden ist. ◆ […]	[…] Unmittelbar voran ging ihm der Edelknabe mit dem Wappenschilde, der in blauem Felde drei silberne Schildchen zeigt und von Maximilian dem großen Meister für die ganze Künstlerschaft gegeben worden ist. ◆
	[…]	[…]
1181	Nachdem nun, was eine Stadt baut und ziert und von ihr liebend gehegt wird, vorangegangen, trat gewissermaßen die Stadt selbst auf, wenn der nun folgende Zug von jenem irgend noch zu trennen ist; denn beide zusammen machten ja das Ganze aus, und sein rühmliches Wohl kannte nur einen Boden für seine Wurzeln. ■	Nachdem nun, was eine Stadt baut und ziert, vorangegangen, trat gewissermaßen die Stadt selbst auf. ■
1182	Von zwei bärtigen Hellebardierern begleitet wurde das große Stadtbanner getragen. Hoch trug der kecke Träger im weiß und roten, üppig geschlitzten Kleide die wallende Fahne, die eine Faust stattlich in die Seite gestemmt und anmutig den Fuß vorsetzend. […] ♠	Von zwei bärtigen Hellebardieren begleitet, wurde ihr das große Banner vorgetragen. Hoch trug der kecke Fähndrich die wallende Fahne, im üppig geschlitzten Kleide, die linke Faust stattlich in die Seite gestemmt. […] ♠
1183	Ihm folgten gleich die beiden Bürgermeister, staatsmännischen und weisen Ansehens, dann der Syndikus und die Ratsherren, unter denen manch ein im weiten Reich angesehener und demselben ersprießlicher Mann war. ★	Ihm folgten Bürgermeister, Syndikus und Ratsherren, unter ihnen manch ein im weiten Reich angesehener und ersprießlicher Mann, und endlich die festlichen Reihen der Geschlechter. ★
1184	Von den beiden Stadtschreibern, welche nebeneinander gingen, war der eine schmächtige Schwarzgekleidete, mit der schön geschnitzten Elfenbeinbrille auf der Nase, in Wirklichkeit der Literator der Künstlerschaft und der gelehrte Beschreiber des Festes. […]	
1185	Den Schluß bildeten nun die festlichen Reihen der ehrbaren Geschlechter. Seide, Gold und Juwelen glänzten hier in schwerem Überfluß. Diese kaufmännischen Patrizier, deren Güter auf allen Meeren schwammen, die zugleich in kriegerischer Haltung mit dem selbst gegossenen trefflichen Geschütze ihre Stadt verteidigten und an Reichskriegen teilnahmen, übertrafen den Adel an Pracht und Reichtum und unterschieden sich von ihm durch Gemeinsinn und sittliche Würde, vom gemeinen Bürger aber durch weitsehenden Blick und umfassenden erhaltenden Sinn. […]	Seide, Gold und Juwelen glänzten hier in schwerem Überfluß. Die kaufmännischen Patrizier, deren Güter auf allen Meeren schwammen, die zugleich in streitbarer Haltung mit dem selbstgegossenen Geschütze die Stadt verteidigten und an den Reichskriegen teilnahmen, übertrafen den mittleren Adel an Pracht und Reichtum wie in Gemeinsinn und sittlicher Würde. […]

given text segments so that the new segmentation allows for a better segment alignment (probably for all humanities questions). Table 2 shows the alignment after resegmentation.

In the two tables, we have shortened long passages of text with […] to keep the example clear. However, this is exactly the problem if you want to do manual resegmentation. A lot of scrolling is necessary to find similar text segments, if a person can keep track at all. The problem can be seen in Table 1. The text in the segment with the diamon icon (green box) in the 1879 version of *Der grüne Heinrich* is aligned after resegmentation with a segment (see Table 2) that is much higher in the original segment alignment and cannot be seen on the screen at the same time (here Table 1 is limited by the size of a page).

3.3 Measuring the improvement in alignments after resegmentation using statistical analysis

As already discussed in Section 3.2, it is difficult (if not impossible) to define a measure that objectively measures the improvement in segment alignment after resegmentation. Nevertheless, for the sake of completeness, we will present and discuss some statistical analyses in this section. First, however, we will briefly

discuss why, in the context of our application, statistical analyses show a deterioration of segmental alignment in some text passages, even though objectively the alignment looks better.

The main reason for this are the *NABs*. It often happens that a segment contains only one *non-fitting sentence block* and at least one *NAB*. If we move the *non-fitting sentence blocks* from both sides to a separate alignment row, the *NABs* will also be moved, because the segment as a whole will be moved. Of course, before the *improvement*, the segments containing the *non-fitting sentence blocks* were not aligned, and after the *improvement*, the segments are aligned. The now aligned segments can be formally very different depending on the size of the *NABs* that are part of the segments, depending on which formal measure is used to measure the similarity of two text segments. The same happens if the *non-fitting sentence block* is part of a segment that also contains a *fitting sentence block*. Moving the *non-fitting sentence block* out of the segment usually results in a better distance for the *fitting* part, but the new alignment row with the *non-fitting sentence blocks* may have a very poor distance, depending on whether *NABs* were moved with the *non-fitting sentence block*. Since the *NABs* do not fit any part of the other text, they degrade the overall distance no matter where they are placed. We could put them in their own segments. However, this would not be in line with our goal to change the original segmentation as little as possible.

Since most of the problems are caused by the *NABs*, we decided to measure only the segments with *non-fitting sentence blocks* and not to include the *NABs* in the distance calculation between two text segments, that is, to simply ignore them in the distance calculation. This should allow the statistical approaches of the analysis to better represent the actual improvements achieved by our heuristics.

We use the Jaccard similarity (or Jaccard coefficient) as a measure of the similarity of two text segments. Roughly speaking, the Jaccard similarity of two text segments is defined as the ratio of the number of unique words that occur in both the one and the other text segment to the total number of unique words in the two text segments after normalization.

Table 3 shows the statistical characteristics of the segment alignments for the four text examples obtained by calculating the final segment alignment after resegmentation. The *non-fitting sentence blocks* found during the resegmentation step and aligned by our alignment algorithm do indeed match well, especially for *Nahar*, *Heinrich*, and *Histoire* with an average Jaccard similarity of 89%, 60%, and 46 %, respectively.

One way to improve the (statistical) result is to choose a higher similarity threshold—only segments whose similarity is greater than the threshold can be aligned. On the other hand, this might lead to fewer *non-fitting sentence blocks* and, from a humanities perspective, to worse results of our resegmentation approach.

While the distance function evaluates all sentence pairs uniformly, this uniformity may not meet the nuanced needs of humanities scholars. It is impractical to propose a distance function that fits every humanities research question. Therefore, it would be beneficial to implement this approach in an interactive environment. In such an environment, humanities scholars can manually review and adjust the proposed segment movements on a case-by-case basis, tailoring the alignments to their specific needs.

We did not conduct runtime experiments because the approach is linear in the total number of sentences in both text versions with a small constant. This does not include segment and sentence alignment time, as these are not part of the heuristic and can be computed in advance. For the largest text, *Heinrich*, the resegmentation procedure takes less than a second on a 3.7 GHz machine.

The detailed experimental results on all four texts, in particular the alignments before and after resegmentation, can be viewed on our project page.[7] We invite the reader to take a closer look at the alignments before and after resegmentation to better evaluate the effectiveness of our approach. The page also includes a link to our source code.

Table 3. Statistics on the Jaccard similarity of the improved alignments of the four text examples, considering only *non-fitting sentence blocks* and ignoring *NABs*.

	new/modified segments	average similarity	median similarity	standard deviation	worst similarity	best
Nahar	72	0.89	1.0	0.18	0.33	1.0
SWL	11	0.48	0.48	0.11	0.32	0.67
Heinrich	298	0.60	0.58	0.26	0.21	1.0
Histoire	34	0.46	0.44	0.12	0.31	0.88

Note: The column 'new/modified segments' shows the number of newly created segments (alignment rows) with *non-fitting sentence blocks*.

4. Conclusion and future work

We have presented an approach that can be applied to a given segment alignment of two versions of a text, with the goal of improving the segment alignment by changing the segmentation where necessary and leaving the rest of the segmentation unchanged.

The approach consists of three steps: First, preprocessing is performed to compute a sentence alignment if one does not already exist. This step requires a threshold value that defines a lower bound for the similarity of two sentences so that they can be considered for alignment by the alignment algorithm. In the second step, all sentences of the two text versions are classified as *fitting*, *non-fitting* or *not-aligned*. In addition, consecutive sentences that are classified the same are grouped into *sentence blocks*. *Fitting* sentences must not be moved, *not-aligned* sentences represent insertions or deletions, and *non-fitting* sentences should be moved to improve the alignment. The last step uses the classification and constructs an improved alignment by moving each of the *non-fitting sentence blocks* into their own segment.

The experiments show that resegmentation has a great impact on the quality of the collation of two versions of a text. The runtime is also very fast for medium or large texts, that is, tens of thousands of sentences, if an existing sentence alignment is provided.

The approach presented in this article greatly reduces the work involved in manually post-processing the alignment of two versions of a text, that is, going through insertions and deletions and identifying those that have matching parts in the other version of the text and need to be moved.

Our approach is not limited to monolingual texts and theoretically can be extended to bilingual alignments. However, when taking full advantages of the capabilities of multilingual aligners such as VecAlign and SentAlign to compute good bilingual alignments, they often produce many-to-many sentence alignments. In these alignments, the parent segment of a sentence may not be clearly defined, especially when multiple sentences on one side of the alignment span multiple segments. Another challenge is obtaining a good segment alignment of texts in different languages, since multilingual sentence aligners have not yet been tested at the segment level.

The next steps are to integrate the tool into LERA (Pöckelmann *et al.* 2023), so that resegmentation is available to the many edition projects that work with LERA, and to generalize the approach to the case of more than two versions of a text to be collated. For this generalization we expect several challenges. First, there are many ways to define an alignment of more than two texts. One possible definition could involve pairwise aligned sentences or segments, which would allow our classification function to be applied without modification. This approach introduces the possibility of certain sentences having conflicting classifications across different pairwise alignments (see Fig. 6). Second, the process of resegmentation itself presents challenges. If resegmentation is done pairwise, the order in which pairs are processed becomes significant, as earlier classifications may influence subsequent ones. Conversely, if resegmentation is not performed pairwise, changing the segmentation of a single sentence requires simultaneously considering and potentially resegmenting all connected sentences across the texts. This complexity grows with the number of text versions, as the number of possible class combinations increases correspondingly.

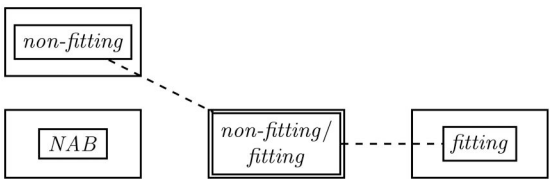

Figure 6. Illustration of conflicting classifications across pairwise alignments.

Author contributions

Janis Dähne (Investigation, Software, Writing—original draft, Writing—review & editing), Jörg Ritter (Writing—review & editing), and Paul Molitor (Supervision, Writing—review & editing)

Notes

1. In the following, we use the terms text paragraph and segment as synonyms.
2. On a 3.7 GHz machine.
3. See https://www.projekt-gutenberg.org/eucken/sinnwert/chap019.html.
4. https://lera.uzi.uni-halle.de/?lang=en.
5. https://www.projekt-gutenberg.org/
6. http://www.epub2go.eu; https://calibre-ebook.com/
7. https://tsaligner.uzi.uni-halle.de/resegmentation.

References

Bremer, T. *et al.* (2015) 'Zum einsatz digitaler methoden bei der erstellung und nutzung genetischer editionen gedruckter texte mit verschiedenen fassungen: Das fallbeispiel der histoire philosophique des deux indes von guillaume-thomas raynal', *Editio*, **29**: 29–51.

Dähne, J. *et al.* (2022) 'Putting Collation of Text Witnesses on a Formal Basis', *Digital Scholarship in the Humanities*, **37**: 375–90.

Eucken, R. (1907) 'Der sinn und wert des lebens', https://www.projekt-gutenberg.org/eucken/sinnwert/.

Eucken, R. (1914) 'Der sinn und wert des lebens', https://www.projekt-gutenberg.org/eucken/sinnwer1/.

Haentjens Dekker, R. *et al.* (2015) 'Computer-Supported Collation of Modern Manuscripts: Collatex and the Beckett Digital Manuscript Project', *Digital Scholarship in the Humanities*, **30**: 452–70.

Halliday, M. A. K., and Hasan, R. (2014) *Cohesion in English*. London: Routledge.

Hearst, M. A. (1994) 'Multi-paragraph Segmentation Expository Text', in *32nd Annual Meeting of the Association for Computational Linguistics*, pp. 9–16. Las Cruces, New Mexico, USA, June. Association for Computational Linguistics. https://aclanthology.org/P94-1002/

Jaccard, P. (1901) 'Étude comparative de la distribution florale dans une portion des alpes et des jura', *Bulletin de la Société Vaudoise des Sciences Naturelles*, **37**: 547–79.

Keller, G. (1854/1855) 'Der grüne heinrich', https://www.projekt-gutenberg.org/keller/heinric1/.

Keller, G. (1879/1880) 'Der grüne heinrich', https://www.projekt-gutenberg.org/keller/heinrich/.

Kozima, H. (1993) 'Text Segmentation Based on Similarity Between Words', in *31st Annual Meeting of the Association for Computational Linguistics*, pages 286–288, Columbus, Ohio, USA, June. Association for Computational Linguistics. doi: 10.3115/981574.981616. URL https://aclanthology.org/P93-1041/.

Litman, D. J., and Passonneau, R. J. (1995) 'Combining Multiple Knowledge Sources for Discourse Segmentation', in *33rd Annual Meeting of the Association for Computational Linguistics*, pp. 108–115, Cambridge, MA: Association for Computational Linguistics. https://aclanthology.org/P95-1015/.

Needleman, S. B., and Wunsch, C. D. (1970) 'A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins', *Journal of Molecular Biology*, **48**: 443–53.

Nury, E., and Spadini, E. (2020) 'From Giant Despair to a New Heaven: The Early Years of Automatic Collation', *IT-Information Technology*, **62**: 61–73.

Pöckelmann, M. *et al.* (2023) 'Lera—An Interactive Platform for Synoptical Representations of Multiple Text Witnesses', *Digital Scholarship in the Humanities*, **38**: 330–46.

Raynal, G.-T. (1780) 'Histoire philosophique et politique des établissements et du commerce des européens dans les deux indes', https://lera.uzi.uni-halle.de/editions/1/edit.

Raynal, G.-T. (1820) 'Histoire philosophique et politique des établissements et du commerce des européens dans les deux indes', https://lera.uzi.uni-halle.de/editions/1/edit.

Sadvilkar, N., and Neumann, M. (2020) 'PySBD: Pragmatic Sentence Boundary Disambiguation', in E. L. Park *et al.* (eds) *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pp. 110–14. Association for Computational Linguistics. https://aclanthology.org/2020.nlposs-1.15/.

Steingrimsson, S., Loftsson, H., and Way, A. (2023) 'SentAlign: Accurate and Scalable Sentence Alignment', in Y. Feng and E. Lefever (eds), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 256–63. Singapore: Association for Computational Linguistics. https://aclanthology.org/2023.emnlp-demo.22/.

Thompson, B., and Koehn, P. (2019) 'Vecalign: Improved Sentence Alignment in Linear Time and Space', in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1342–1348.

Weiß, E. (1922) 'Nahar', https://www.projekt-gutenberg.org/weisse/nahar/.

Weiß, E. (1930) 'Nahar', https://www.projekt-gutenberg.org/weisse/nahar2/.

Yaari, Y. (1997) 'Segmentation of Expository Texts by Hierarchical Agglomerative Clustering', in *Proceedings of RANLP*, pp. 59–65. Tzigov Chark, Bulgaria.

Digital Scholarship in the Humanities, 2025, 40, 477–486
https://doi.org/10.1093/llc/fqaf033
Full Paper