Finding Inexact Quotations Within a Tibetan Buddhist Corpus

Benjamin Eliot Klein[1], Nachum Dershowitz[1], Lior Wolf[1],
Orna Almogi[2], and Dorji Wangchuk[3]

[1] School of Computer Science, Tel Aviv University, Ramat Aviv, Israel
beni.klein@gmail.com; nachum.dershowitz@cs.tau.ac.il; wolf@cs.tau.ac.il
[2] Centre for the Study of Manuscript Cultures,
Universität Hamburg, Hamburg, Germany
orna.almogi@uni-hamburg.de
[3] Department of Indian and Tibetan Studies,
Universität Hamburg, Hamburg, Germany
dorji.wangchuk@uni-hamburg.de

Abstract. Text matching can be a powerful tool in exploring historical texts. Here, we compare two Tibetan texts and identify pairs of nearly identical strings. A maximal-path algorithm, previously used for matching biological sequences, lies at the heart of our method. The matches found have been verified by Tibetan scholars. They were shown to be of concrete value for Tibetan studies and open up previously inaccessible research avenues.

1 Introduction

One thing that literary scholars routinely look for – regardless of the specific field – is textual citations, where one work quotes or paraphrases another work. In historical works, even quotations are frequently quite inexact. To complicate matters further, there is often no clear indication that a passage is being quoted, let alone which work is being cited. It is, therefore, only natural to use algorithmic tools to search for such occurrences in texts and present the results to scholars for consideration.

One such corpus is the Tibetan Buddhist canon, currently being digitized. Altogether, there are more than 300 volumes, averaging about 800 pages (400 folios) of 200 words each. Words in Tibetan are predominantly monosyllabic. In addition to the canon, there are many other important collections in the Tibetan Buddhist literary corpus. And, of course, there are many Buddhist corpora in other languages. Scholarly editions are still wanting for many of these works, so we have set out to design computerized tools to help deal with the masses of data.

The most relevant previous work is by Prasad and Rao [2] who search for citations within Sanskrit texts. They break the text into units (lines, say) and then compare each potential citation with each unit in the cited corpus (Smith-Waterman-Gotoh), using approximate match. To constrain the search, they first sort the units, so they only need compare units that begin similarly. We approach the problem of reducing the complexity of the search differently. We borrowed an algorithm designed for finding *all* (sufficiently long) approximate subsequence matches in genomic data and adapted it for finding common approximate subtexts between two large corpora. This involved parallelizing the algorithm, adding some simple preprocessing, and some less trivial post-processing.

In this exploratory work, we compared two major Buddhist texts, transliterated into Latin characters from the Tibetan, the *Sūtrasamuccaya* and the *Śikṣāmuccaya*. Each is over 150,000 words long. Preliminary results are extremely encouraging and hold promise for furthering scholarship in this field.

After briefly describing the corpus in the next section, we explain the method we used in Sect. 3. Results are summarized in Sect. 4, and are followed by a discussion.

2 The Corpus

The Sūtrasamuccaya. The "Compendium of [Citations from Mahāyāna] Sūtras" is a compilation ascribed to the famous Nāgārjuna (2nd century CE). The work is arranged according to various philosophical topics. The Sanskrit original is not available; the text has survived only in its Tibetan (P5330, D3934) and Chinese translations. It is an important source for early Mahāyāna sūtras, and it is invaluable for the study of the early phase of Mahāyāna. The Tibetan translation was done by the Tibetan famous translator Ye-shes-sde (8th century), in collaboration with two Indian Scholars, Jinamitra and Śīlendrabodhi. This Tibetan translation has been critically edited by Bhikkhu Pāsādika (1989). (It also includes a reproduction of the Chinese translation, and there is also a modern translation by the same author "in installments").

The Śikṣāmuccaya. The "Compendium of Teachings" (i.e. citations from mostly early Mahāyāna sūtras) was compiled by the famed Indian scholar Śāntideva (7th c.). Also in this case, the work is arranged by philosophical topic. Sanskrit original has survived and was critically edited and translated (1897–1902 & 1922, respectively, by Cecil Bendall). It was translated into Tibetan by the same Tibetan translator Ye-shes-sde (8th c.), in collaboration with the Indian scholars Jinamitra and Danaśīla. Later on the translation was revised by the Tibetan translator Blo ldan shes rab (1059–1109) in collaboration with the Kāśmirian scholar Tilakakalaśa. No critical edition of the Tibetan is available.

3 Method

We describe now the algorithm and the different techniques that we use to solve the problem of finding local regions with high similarity in the two texts.

We worked with transliterations of the Tibetan texts. The texts contain multiple spaces, line breaks, page numbers, punctuation marks and the like. In a preprocessing step, we clean the texts and remove all such.

The main workhorse is an efficient algorithm for solving the "threshold all against all" variant of the problem. Let s and t be two strings, with lengths n and m respectively. The edit distance of s and t is the minimum number of the standard edit operations (insertion of a letter, deletion of a letter, and substitution of a letter by another) needed to transform s to t. The result of the algorithm is all pairs (s', t') of maximal substrings of s and t, respectively, such that their length is at least L_0, and the edit distance between them is at most k.

We used the algorithm of Barsky et al. [1]. They recast the original problem into the problem of finding "maximal paths" in a "matching" graph, paths which correspond to solutions of the original string problem. Taking advantage of properties of this graph, their algorithm runs in $O(mnk^2)$ time and requires $O(m|\Sigma| + n)$ memory, where Σ is the alphabet.

The texts we are using are very long and therefore we created a parallelize version of the algorithm. Each text is divided into overlapping chunks of size ℓ such that the range of the first chunk is $[0, \ell]$, the range of the second chunk is $[\ell/2, 3\ell/2]$, etc. By choosing ℓ to be large enough, and due to the overlapping chunks, the quality of the results of the parallelized algorithm does not fall below the original one for our purposes. By dividing the texts to chunks, we get $O(mn/\ell^2)$ different pairs of chunks. The running time for each pair is $O(\ell^2 k^2)$, and therefore the total running time is $O(mnk^2)$. By using a cluster of r cores, we get a running time of $O(mnk^2/r)$ on each core. Hence we achieve almost perfect parallelization, while significantly reducing the memory requirements for each core to $O(\ell|\Sigma|)$.

After collecting all the results, some post-processing steps are required in order to build a non-redundant and meaningful collection of local regions with high similarity; the splitting to overlapping chunks causes some results to overlap and in fact, even the original algorithm may return some overlapping results. In addition, some results do not overlap, but are near each other and therefore could be merged into a longer match. We address these issues by uniting every pair of overlapping or nearby results.

The second problem arises when we have a meaningful result with length that is smaller than L_0 and a very small edit distance. In this scenario, the algorithm extends the result in order for the minimal length constraint to be satisfied, resulting with a less meaningful result. We solve this issue by applying local alignment on each result, which removes these uninformative extensions.

From the final collection of matched substrings we get two main outcomes; the trivial one is finding the local regions of high similarity in the two texts. We built a designated interface that has tools for investigating the matches. It presents the two texts side by side and a list of all the matches in descending order of their edit distance. Using it, one can focus on a specific match, and see the relevant substrings in both texts. The substrings are also presented in another window, aligned to each other for convenient comparison. Additionally, by selecting a substring in one of the texts, one can see all the matches that overlap with the selection. See the screenshot in Fig. 3.

The second outcome is computing statistics on all the results (character confusion matrix, common words that appears in one text but not in the other, etc.). This requires that we carefully align each result, as the quality of the statistics depends on the alignment of each single word. The alignment is done by a variant of global alignment that penalizes gaps that occur between words (or in the beginning/end of the string) differently than gaps that occur within a word. Let $D[0..m, 0..n]$ be the dynamic programming matrix, let p and q be the penalty for an inter-word gap and the penalty for an intra-word gap, respectively, and let $\delta(a, b)$ be the match score function for characters $a, b \in \Sigma$. We define $D[i, j]$ inductively:

- Initialization

$$\begin{aligned} D[0,0] &:= 0 \\ D[i,0] &:= i \cdot p \\ D[0,j] &:= j \cdot p \end{aligned}$$

- Recursive Step

$$D[i,j] := max \begin{cases} D[i-1,j-1] + \delta(s_i, t_j) & \\ D[i-1,j] + p & \text{if } j = m \text{ or } t_{j+1} = \text{space} \\ D[i-1,j] + q & \text{if } j \neq m \text{ and } t_{j+1} \neq \text{space} \\ D[i,j-1] + p & \text{if } i = n \text{ or } s_{i+1} = \text{space} \\ D[i,j-1] + q & \text{if } i \neq n \text{ and } s_{i+1} \neq \text{space} \end{cases}$$

This simple alignment allows us to derive meaningful statistics on differences from the collection of matched substrings.

4 Results

The matching software was applied to the two texts using the following parameters: minimum length of match is $L_0 = 60$, maximum number of errors in a match is $k = 10$, and size of each chunk is $\ell = 25000$ characters. Overall, 2514 matches were found between the texts. These matches cover a significant fraction of the texts and 9.15% of the Sūtrasamuccaya as well as 10.85% of the Śikṣāmuccaya comprise of regions for which at least one match in the other text was found. Some of the matches are quite long, as can be seen in the histogram in Fig. 1.

A research tool was built, in which the texts are displayed side by side, such that text regions for which matches exist are marked as bold. A user can select text on either corpus and request all matching text regions from the second one. The user can also browse all matching quotations, sorted by score from the highest score (meaning longest match and fewest substitutions) to the lowest. Lastly, the user can export specific quotations, including their immediate context, to files (as in Fig. 2).

Sample matches were verified by Tibetan scholars who found the quotations to be correct. In some cases, the scholars were less conservative than the matching system and would have extended the quotations to regions before or after the

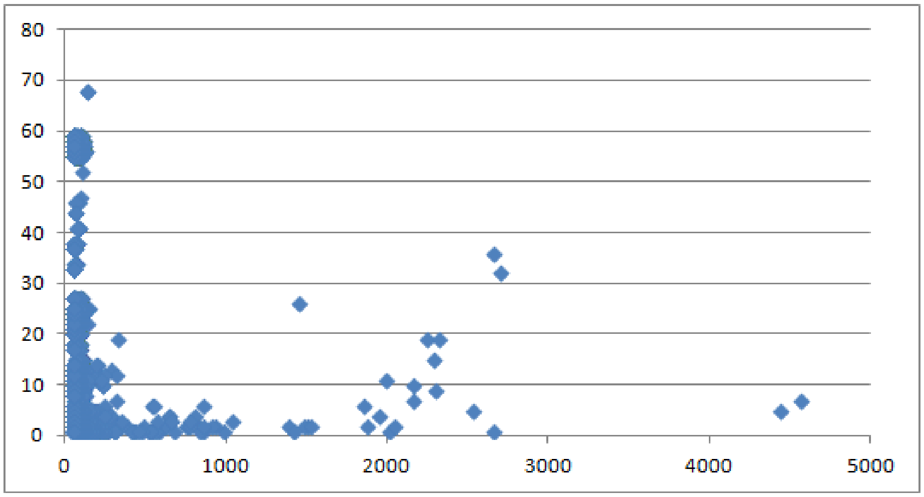


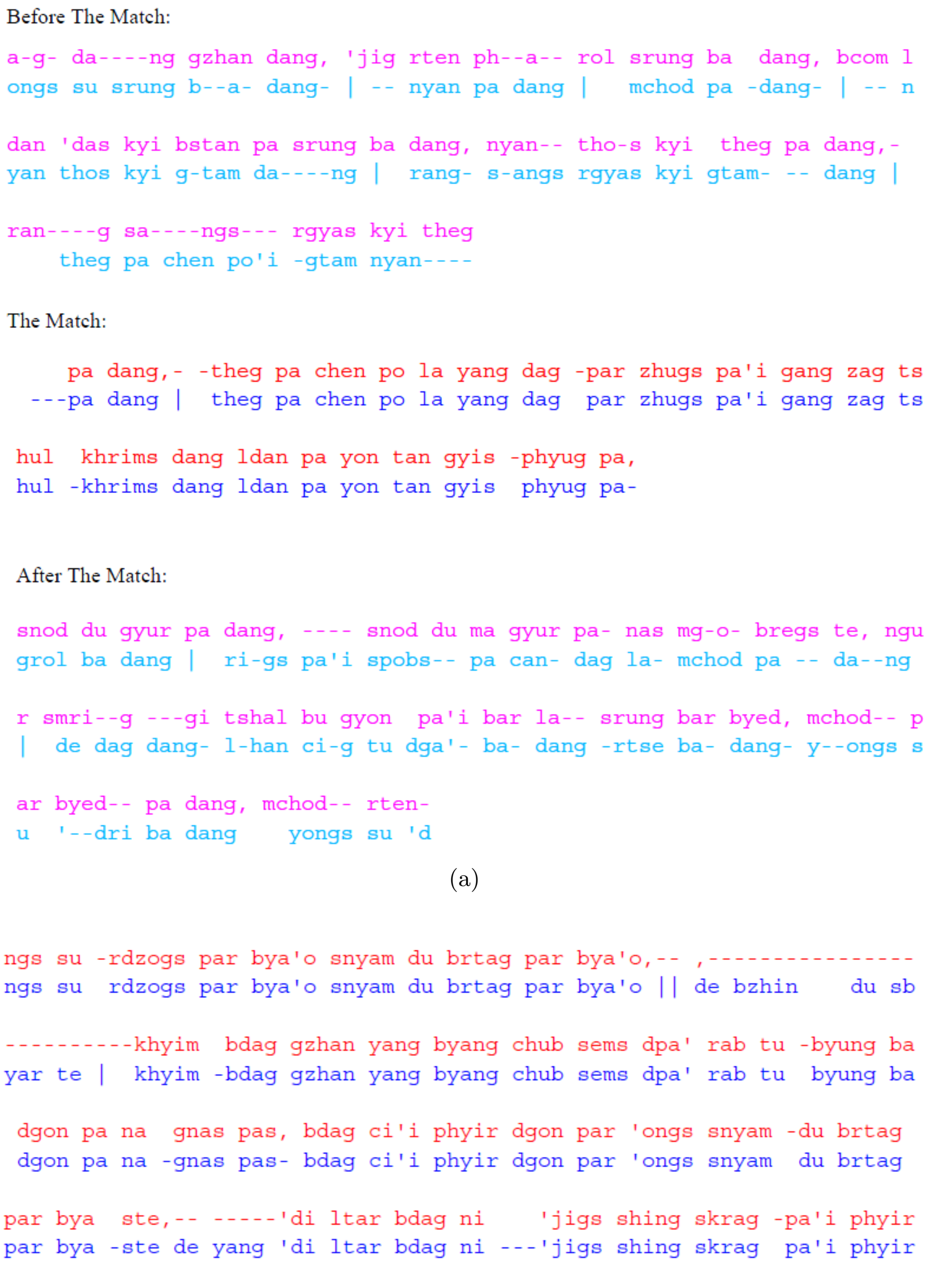
Fig. 1. Histogram depicting the count (y-axis) of matching texts of a certain length (x-axis) [the length in Sūtrasamuccaya]. As can be seen, while most of the matches are for texts of up to 200 characters (median=61), there are also matches for texts of a few thousand characters. Note that some texts (including long ones) have multiple matches.

marked texts. Many of the omissions and substitutions were not surprising to the language speakers, such as *cing* ⇝ *zhing*, *bting* ⇝ *gding*, and *po wang* ⇝ *pi bang*. Some of the variations seem to be simple typos, either by the scribe of the original or simply during the digital transliteration. Typical transcription errors include substitutions between b and p, or ng and d, which appear similar in Tibetan. In some cases, making the distinction whether a variant stems from an accidental typo or is in fact a substantive variant is not clear. For example, *gtor* means to scatter and *'thor*, which was correctly matched is some of the quotations, means to be scattered. Table 1 exhibits the "confusion matrix", i.e., the most common letter substitutions between the two texts.

5 Discussion

Two well-known and studied works were chosen as a trial case for our first experiment, namely, transliterated texts of the two Buddhist works that were translated into Tibetan in the 8th century. Due to the large number of shared citations found in these works, they made for a good trial case for algorithmically locating matches. But because these are two different anthologies (i.e., not two versions of the one and the same work), they are different enough to provide sufficiently many instances of approximate matches and discrepancies.

Besides the value of the citations themselves, in the absence of a critical edition of one of the works, statistics regarding the types of variations (particularly in the usage of particles) hint at the nature of the editing done by



Before The Match:

```
a-g- da----ng gzhan dang, 'jig rten ph--a-- rol srung ba  dang, bcom l
ongs su srung b--a- dang- | -- nyan pa dang |   mchod pa -dang- | -- n

dan 'das kyi bstan pa srung ba dang, nyan-- tho-s kyi  theg pa dang,-
yan thos kyi g-tam da----ng |  rang- s-angs rgyas kyi gtam- -- dang |

ran----g sa----ngs--- rgyas kyi theg
    theg pa chen po'i -gtam nyan----
```

The Match:

```
    pa dang,- -theg pa chen po la yang dag -par zhugs pa'i gang zag ts
  ---pa dang |  theg pa chen po la yang dag  par zhugs pa'i gang zag ts

hul  khrims dang ldan pa yon tan gyis -phyug pa,
hul -khrims dang ldan pa yon tan gyis  phyug pa-
```

After The Match:

```
snod du gyur pa dang, ---- snod du ma gyur pa- nas mg-o- bregs te, ngu
grol ba dang |  ri-gs pa'i spobs-- pa can- dag la- mchod pa -- da--ng

r smri--g ---gi tshal bu gyon  pa'i bar la-- srung bar byed, mchod-- p
|  de dag dang- l-han ci-g tu dga'- ba- dang -rtse ba- dang- y--ongs s

ar byed-- pa dang, mchod-- rten-
u  '--dri ba dang    yongs su 'd
```

(a)

```
ngs su -rdzogs par bya'o snyam du brtag par bya'o,-- ,----------------
ngs su  rdzogs par bya'o snyam du brtag par bya'o || de bzhin    du sb

----------khyim  bdag gzhan yang byang chub sems dpa' rab tu -byung ba
yar te |  khyim -bdag gzhan yang byang chub sems dpa' rab tu  byung ba

 dgon pa na  gnas pas, bdag ci'i phyir dgon par 'ongs snyam -du brtag
 dgon pa na -gnas pas- bdag ci'i phyir dgon par 'ongs snyam  du brtag

par bya  ste,-- -----'di ltar bdag ni    'jigs shing skrag -pa'i phyir
par bya -ste de yang 'di ltar bdag ni ---'jigs shing skrag  pa'i phyir
```

(b)

Fig. 2. Examples of quotations found. (a) The quotations with the 90th highest score, including its context, before and after. Line breaks have no significance. (Vertical bars serve as commas.) Two hues depict the two texts, red/pink for the Sūtrasamuccaya and blue/magenta for the Śikṣāmuccaya. (b) Part of the quotation with the fourth highest score, which is relatively clean except for some omissions.

the 11th-century writers. Through the statistics, scholars may be able to learn about some stylistic differences and editorial practices. Continuing this line of work, we may even have a case where through a careful analysis of the differences one could become aware of some forgotten philosophical differences and developments.

It has been argued before that the so-called "revisions" often involve only very minor and unimportant changes, and indeed some of the revisers had often been accused by some Tibetans of plagiarism. The difficulty, however, is that in most cases we only have access to the revised version(s), and thus cannot compare the revision with the initial translation. Our alignment and statistical tools can help scholars trace and re-evaluate this phenomenon.

Lastly, going beyond direct matches, it may be possible to identify two different texts that share similar passages, which are paraphrases, not citations. There have been reports in the past of at least one text that is found in an abridged form in another very important text of the tradition. In another example, a Tibetan text was said to be a combination of two canonical (that is, Indian) texts. In order to methodologically find such examples, Tibetan texts would need to be compared to the entire canon. The scholarly implications promise to be far reaching, as this would enable the discovery of the history and emergence of texts and scriptures, and allow for the estimation of the popularity of certain texts that are cited more often (and to determine in which circles these are cited). Moreover, since in the case of the Tibetan canonical texts, translated material is being used, the translation and editorial practices can also be explored.

Besides enlarging the corpus (and hopefully obtaining the coöperation of those who hold the rights to the digital texts and images), many algorithmic improvements suggest themselves and are planned: Transliterations are not always available, so we would like to apply the method directly to the Tibetan (or any other language). In many cases, texts have not yet been transcribed and all that may be available are poor-quality mechanical (OCR) transcriptions, which may still lend themselves to approximate matching. We will experiment with edit distances that charge less for morphological variations, phonetic shifts, and transcription (or OCR) errors than for more significant differences. Rather than unify adjacent matches a posteriori, the algorithm should be adapted to consider the density of errors in long matches, with no absolute upper bound. The alignment of variants should be done in a more scholarly fashion, indicating which words have been added or deleted. Having collected several variants of the same text, we will apply machine-learning tools to suggest which text is citing which, and to suggest an ur-text.

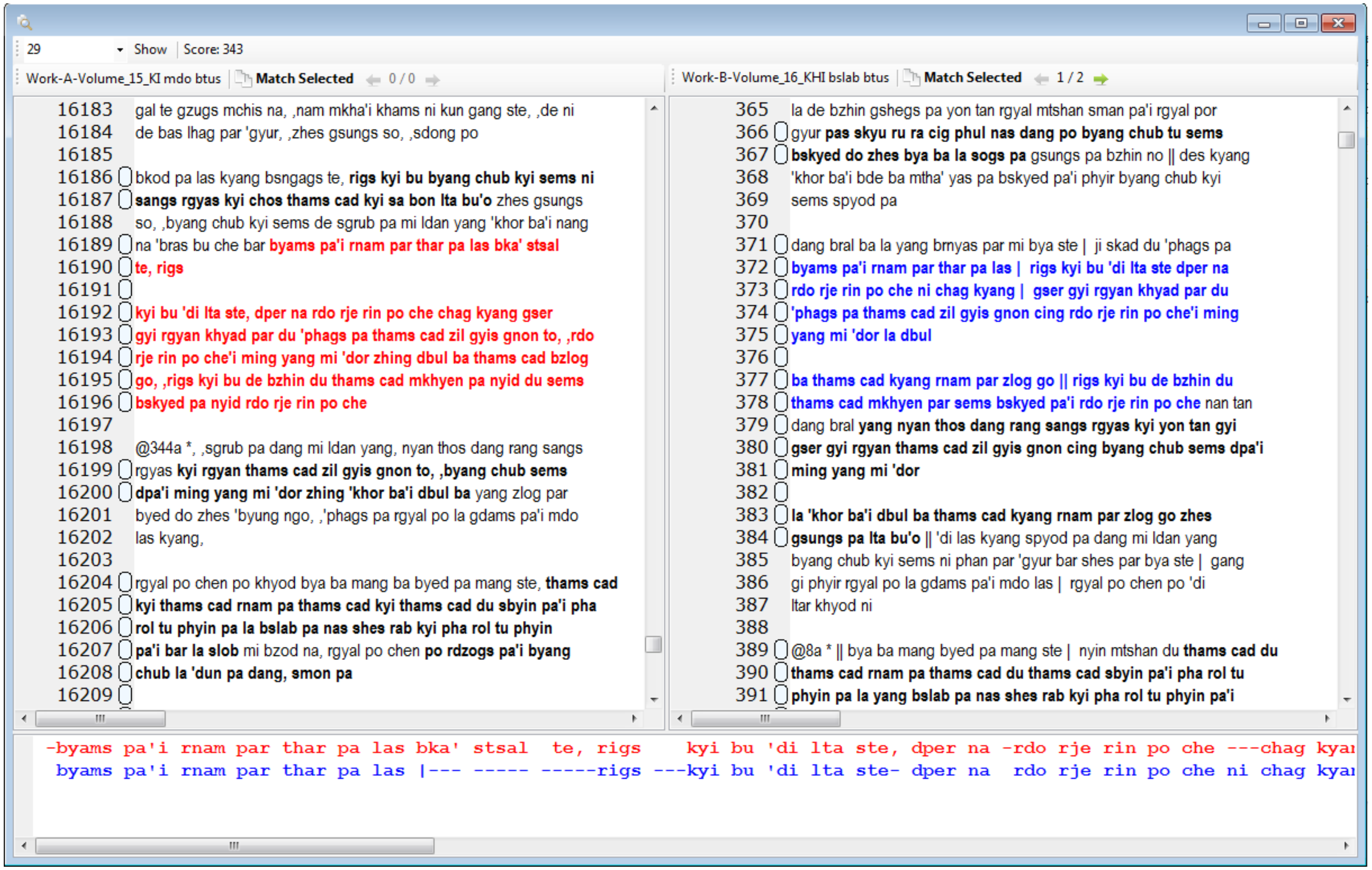
References

1. Barsky, M., Stege, U., Thomo, A., Upton, C.: A graph approach to the threshold all-against-all substring matching problem. ACM Journal of Experimental Algorithmics **12** (2008)

2. Prasad, A.S., Rao, S.: Citation matching in Sanskrit corpora using local alignment. In: Jha, G. (ed.): Sanskrit Computational Linguistics. Lecture Notes in Computer Science, Vol. 6465. Springer Berlin, Heidelberg (2010) 124–136

Table 1. Substitution counts between the Sūtrasamuccaya text (rows) and the Śikṣāmuccaya (columns).

	a	b	c	d	e	g	h	i	j	k	l	m	n	o	p	r	s	t	u	v	w	y	z
a	23685	6	1	21	24	52	4	39	1	1	20	4	15	58	3	30	14	6	18	0	0	5	1
b	6	7678	6	11	2	33	3	1	0	16	9	5	12	2	73	6	17	6	0	0	0	4	3
c	0	8	2842	4	0	0	5	0	3	0	4	0	2	0	11	6	5	3	8	0	0	4	1
d	30	18	2	9371	2	42	12	4	2	5	15	10	55	3	12	16	35	10	8	0	0	29	0
e	25	1	0	4	4635	16	2	33	0	0	2	15	6	19	0	0	16	2	8	0	0	1	0
g	33	21	5	48	10	15099	4	19	0	8	23	13	27	13	3	15	42	5	12	0	0	11	2
h	8	4	6	10	4	9	4965	1	0	0	3	0	19	5	1	2	36	13	1	0	0	10	5
i	54	2	0	8	25	40	0	6084	0	0	4	7	15	23	0	30	31	2	25	0	0	0	0
j	1	0	0	0	1	1	3	0	448	0	0	4	3	0	0	1	1	3	0	0	0	0	0
k	1	11	2	2	0	6	0	0	0	1549	5	0	4	0	2	2	3	3	3	0	0	5	3
l	6	10	1	16	1	4	1	2	2	4	3038	6	14	0	10	12	9	5	0	0	0	14	1
m	18	5	0	8	5	25	1	0	0	3	8	4653	8	14	5	8	13	1	3	0	0	5	0
n	58	11	5	51	4	26	7	19	1	1	18	12	11613	3	49	6	38	7	6	0	0	16	0
o	59	0	0	2	17	9	2	15	0	0	1	1	6	5077	1	8	3	0	11	0	0	5	0
p	23	92	15	29	0	3	2	1	0	1	25	8	59	0	8473	3	4	2	1	0	0	2	14
r	4	4	2	12	1	10	3	3	0	1	11	8	3	2	0	7194	37	9	3	0	0	2	2
s	11	6	5	28	19	84	56	24	0	1	11	7	27	6	11	48	10151	6	6	0	0	2	3
t	23	0	0	15	1	16	3	2	0	2	12	1	7	0	3	11	3	2267	0	0	0	16	1
u	68	0	0	1	25	48	1	26	0	0	0	1	8	10	0	5	15	0	3936	0	0	4	2
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0
w	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0
y	13	2	2	23	0	8	12	4	2	6	6	4	7	0	8	3	11	19	6	0	0	6719	2
z	18	0	4	3	0	6	1	0	1	2	4	3	0	0	1	1	10	3	0	0	0	6	2582



29 Show | Score: 343

Work-A-Volume_15_KI mdo btus | Match Selected 0 / 0

16183 gal te gzugs mchis na, ,nam mkha'i khams ni kun gang ste, ,de ni
16184 de bas lhag par 'gyur, ,zhes gsungs so, ,sdong po
16185
16186 □ bkod pa las kyang bsngags te, **rigs kyi bu byang chub kyi sems ni**
16187 □ **sangs rgyas kyi chos thams cad kyi sa bon lta bu'o** zhes gsungs
16188 so, ,byang chub kyi sems de sgrub pa mi ldan yang 'khor ba'i nang
16189 □ na 'bras bu che bar **byams pa'i rnam par thar pa las bka' stsal**
16190 □ **te, rigs**
16191 □
16192 □ **kyi bu 'di lta ste, dper na rdo rje rin po che chag kyang gser**
16193 □ **gyi rgyan khyad par du 'phags pa thams cad zil gyis gnon to, ,rdo**
16194 □ **rje rin po che'i ming yang mi 'dor zhing dbul ba thams cad bzlog**
16195 □ **go, ,rigs kyi bu de bzhin du thams cad mkhyen pa nyid du sems**
16196 □ **bskyed pa nyid rdo rje rin po che**
16197
16198 @344a *, ,sgrub pa dang mi ldan yang, nyan thos dang rang sangs
16199 □ rgyas **kyi rgyan thams cad zil gyis gnon to, ,byang chub sems**
16200 □ **dpa'i ming yang mi 'dor zhing 'khor ba'i dbul ba** yang zlog par
16201 byed do zhes 'byung ngo, ,'phags pa rgyal po la gdams pa'i mdo
16202 las kyang,
16203
16204 □ rgyal po chen po khyod bya ba mang ba byed pa mang ste, **thams cad**
16205 □ **kyi thams cad rnam pa thams cad kyi thams cad du sbyin pa'i pha**
16206 □ **rol tu phyin pa la bslab pa nas shes rab kyi pha rol tu phyin**
16207 □ **pa'i bar la slob** mi bzod na, rgyal po chen **po rdzogs pa'i byang**
16208 □ **chub la 'dun pa dang, smon pa**
16209 □

Work-B-Volume_16_KHI bslab btus | Match Selected 1 / 2

365 la de bzhin gshegs pa yon tan rgyal mtshan sman pa'i rgyal por
366 □ gyur **pas skyu ru ra cig phul nas dang po byang chub tu sems**
367 □ **bskyed do zhes bya ba la sogs pa** gsungs pa bzhin no || des kyang
368 'khor ba'i bde ba mtha' yas pa bskyed pa'i phyir byang chub kyi
369 sems spyod pa
370
371 □ dang bral ba la yang brnyas par mi bya ste | ji skad du 'phags pa
372 □ **byams pa'i rnam par thar pa las | rigs kyi bu 'di lta ste dper na**
373 □ **rdo rje rin po che ni chag kyang | gser gyi rgyan khyad par du**
374 □ **'phags pa thams cad zil gyis gnon cing rdo rje rin po che'i ming**
375 □ **yang mi 'dor la dbul**
376 □
377 □ **ba thams cad kyang rnam par zlog go || rigs kyi bu de bzhin du**
378 □ **thams cad mkhyen par sems bskyed pa'i rdo rje rin po che** nan tan
379 □ dang bral **yang nyan thos dang rang sangs rgyas kyi yon tan gyi**
380 □ **gser gyi rgyan thams cad zil gyis gnon cing byang chub sems dpa'i**
381 □ **ming yang mi 'dor**
382 □
383 □ **la 'khor ba'i dbul ba thams cad kyang rnam par zlog go zhes**
384 □ **gsungs pa lta bu'o** || 'di las kyang spyod pa dang mi ldan yang
385 byang chub kyi sems ni phan par 'gyur bar shes par bya ste | gang
386 gi phyir rgyal po la gdams pa'i mdo las | rgyal po chen po 'di
387 ltar khyod ni
388
389 □ @8a * || bya ba mang byed pa mang ste | nyin mtshan du **thams cad du**
390 □ **thams cad rnam pa thams cad du thams cad sbyin pa'i pha rol tu**
391 □ **phyin pa la yang bslab pa nas shes rab kyi pha rol tu phyin pa'i**

```
-byams pa'i rnam par thar pa las bka' stsal  te, rigs    kyi bu 'di lta ste, dper na -rdo rje rin po che ---chag kya
 byams pa'i rnam par thar pa las |--- ----- -----rigs ---kyi bu 'di lta ste- dper na  rdo rje rin po che ni chag kya
```

Fig. 3. A screenshot of the user interface, with the two matching texts are displayed side by side. The text regions for which matching texts exist are emboldened. The first out of two texts that match the blue text are shown in red. This match has a score of 343 and is ranked 19th out of all matches. The panel at the bottom of the screen displays the two texts aligned character by character.