A Statistical Authorship Attribution Method Applied to the Buddhist Scriptures of Ārya Asaṅga

Introduction

Studying what has been written on the scriptures attributed to Asaṅga and Maitreya, one is easily tempted to speculate on using a statistical method to compare these texts stylistically. Reading some articles of the last decade on the subject of statistical authorship attribution methods, I have learnt that there are indeed methods that can be used fruitfully to decide between different possible authors for a text. Common N-Gram analysis (CNG), one of the most successful of these, has been applied experimentally also to texts written in non-Western languages. We could try to apply this method on the available digitalised texts of Asaṅga and Maitreya, and see if the results might point in one particular direction within the rather large variety of theories concerning the authorship of these scriptures.

The Data Set and Preprocessing

In researching authorship using statistical methods, we need texts in their original languages, as the translation process introduces too much bias to identify the author's style of writing. Many Sanskrit Buddhist texts have been digitalised in the last decade, among which are quite a number of the scriptures attributed to Asaṅga. A source for these digitalised texts is the GRETIL archive, the "Göttingen Register of Electronic Texts in Indian Languages", which is found on the World Wide Web at http://gretil.sub.uni-goettingen.de/gretil.htm The Unicode (UTF-8) encoded versions of the texts are the ones we need here. The available texts are summed up in Table 1 below.

Mnemonic	Title	GRETIL	Text Based on Edition	Size (chars)
AAa	Abhisamayālaṃkāra (a)	abhisamu.htm	[Stcherbatsky & Obermiller, 1929]	22900
AAb	Abhisamayālaṃkāra (b)	bsa026_u.htm	Tripathi, 1977 (Haribhadra)	22661
AS	Abhidharmasamuccaya	asabhs_u.htm	cf. Pradhan, 1950	163175
BBh	Bodhisattvabhūmi	bsa034_u.htm	Dutt, 1966	457701
Bhav	Bhavasaṃkrāntiṭīkā	bsa058_u.htm	Shastri, 1938	13479
MAV	Madhyāntavibhāṅgakārikā	bsa010_u.htm	Pandeya, 1971	9237
MSAa	Mahāyānasūtrālaṃkāraśāstra (a)	bsa030_u.htm	Bagchi, 1970, karika only	
MSAb	Mahāyānasūtrālaṃkāraśāstra (b)	asmahsuu.htm	Bagchi, 1960	268299
RGV	Ratnagotravibhāga	bsa073_u.htm	Johnston, 1950	115965
SBha	Śrāvakabhūmi (a)	srabhu_u.htm	Śrāvakabhūmi Study Group, 1998	
SBhb	Śrāvakabhūmi (b)	srabhusu.htm	Shukla, 1973,Śrāvakabhūmi	372851
Vaj	Vajracchedikāprajñāpāramitāsūtraśāstra	atrisatu.htm	Tucci, 1956	5875

Table 1

Before applying CNG, the texts should be preprocessed by what is called canonicising, that is removing all "unhistorical" elements, like titles or verse numbers. Canonicising could also involve unifying case, stripping punctuation, stripping numbers and normalising spaces. The text size in Table 1 is the size in UTF-8 characters after canonicising. Some experimenting with the raw GRETIL files shows that preprocessing is indeed necessary in order to obtain reliable results. The beginning of a preprocessed text as used in this article looks like

> tridharmaḥsaṃgrahaḥsaṃprayogo'nvayaścalakṣaṇeviniścayesatyadharmauprāptiḥsāṃkathyamevacakatikasmādu
> pādānaṃvyavasthānaṃcalakṣaṇamanukamārthadṛṣṭāntabhedājñeyāḥsamuccaye[...]

Not for all digitalised texts the canonicising is straightforward, because some are in pausa (without external sandhi). In some Sanskrit texts, missing lines are suppleted by lines in Tibetan. Having put these texts aside for convenience, this leaves us with 10 out of a total of 12 digitalised texts.

The Training Set

One or more texts are chosen as a training set, serving as a reference for the quantification of differences between the texts. Most suitable as a training text would be those for which the author is known beyond doubt, or (in our case) least doubtful.

They should be written by only one author, they should not be Indian style (sub-) commentaries, and they should be large enough (number of characters) to produce distinctive results. If possible they should have some variety, in terms of their subjects.

Unfortunately we have in our collection of digitalised texts only a few examples where the author is more or less undisputed. The most important of which will be the MAV, which is commonly attributed to Maitreya or Maitreyanātha. The MAV is not a commentary, which is positive, but it is not a very long text. On the other hand the AS is invariably attributed to Asaṅga himself. It is a bit longer than the MAV. For Asaṅga as an author, the best training text will therefore be the AS.

Moreover, for our purpose here Maitreya and Maitreyanātha can be considered the same person, or entity. We will therefore use the name Maitreyanātha from now on for this person or entity, as this seems to be most accurate.

Event Frequencies

The most reliable authorship attribution methods seem to be those based on n-grams, like CNG. Many of the methods available are only suitable for texts in Western languages, or only for the English language. CNG does not have this restriction. In this method, all sequences of n characters are generated from the text, or preferably a set of texts, and their frequences counted. The list of n-grams with their frequencies constitute a more or less reliable author profile (histogram), which is then compared to other profiles. A profile of 6-grams with their absolute frequencies can be represented like

alakṣa 98
sthāna 96
kandha 96
vasthā 95
[...]

The profile is cut off at a certain limit (culling), leaving only the most relevant (e.g. most frequent) events. Then a procedure is applied to compare the profile generated from the training text with the other profiles, of the texts of which the author is unknown. One of the ways of comparing is calculating distances between the profiles, and evaluating these.

Vlado Kešelj (2003) “revived” the use of n-grams for authorship attribution, inspired by an earlier study.[1] In his article on n-grams, Kešelj uses a simple procedure to calculate a distance between two profiles. Besides English and Middle English texts, trials were done with Greek and Chinese, for which the results were only slightly less accurate than for the English texts. In a later trial this approach proved to be successful (Juola 2008).[2]

Some of the more succesful approaches using CNG employ part-of-speech-tagging (POS), for which an automated Sanskrit POS-tagger is needed. Such POS-taggers for Sanskrit are becoming avaiblable in the past few years, but as it is they are not easily adaptable for use in this particular investigation. Methods using word counts are not suitable, as word separation may not have been applied consistently in the ancient Sanskrit manuscripts. Therefore in our case, that is dealing with Sanskrit texts, the best way to go is using a method employing n-grams based on simple character n-grams.

Calculating Distance

I have prepared a PHP routine (see appendix) for generating two profiles from two UTF-8 encoded texts and calculating their distance following the description of Kešelj (2004).[3] I have adjusted the distance calculation so, that whenever an n-gram is present in only one of the profiles, no distance term is added. This provides a numerically different, but slightly more realistic result. The formula for distance is basically unchanged

$$\sum_{x \in D_1 \cap D_2} (2 \,.\, (f_1(x) - f_2(x)) \,/\, (f_1(x) + f_2(x)))^2$$

In this formula, x is an n-gram in profiles D_1 and D_2, and $f_1(x)$ and $f_2(x)$ their respective relative frequencies, that is absolute frequencies divided by the total n-gram count.

In Kešelj's trials, the results with n-grams are most accurate when the number n is around 6. For n<=2 the results are unreliable. For our purpose we will calculate the distances to a training text profile for 3 <= n <= 8. The most succesful limit (L) for culling proves to be around 2000 n-grams, so we will also follow Kešelj there.

The distances calculated are summed up in Table 2.

3-grams		4-grams		5-grams	
AS	0,0	AS	0,0	AS	0,0
SBhb	295,3	SBhb	58,2	SBhb	37,2
BBh	326,1	BBh	73,3	BBh	61,1
MSAb	343,8	MSAb	87,0	RGV	61,4
RGV	402,7	RGV	100,8	MSAb	77,6
AAa	714,6	AAb	353,5	AAa	241,4
AAb	722,8	AAa	361,4	Bhav	255,5
Bhav	727,2	Vaj	408,3	AAb	273,9
Vaj	744,9	Bhav	421,7	Vaj	278,1
MAV	779,7	MAV	425,9	MAV	330,8

6-grams		7-grams		8-grams	
AS	0,0	AS	0,0	AS	0,0
SBhb	19,5	SBhb	18,8	SBhb	21,2
RGV	48,2	BBh	38,9	BBh	31,8
BBh	55,3	RGV	44,9	RGV	38,0
MSAb	61,4	MSAb	50,0	MSAb	44,5
Bhav	140,8	Bhav	93,3	Bhav	48,5
AAb	152,3	Vaj	112,8	Vaj	61,4
Vaj	160,5	MAV	116,0	AAb	67,3
AAa	173,6	AAb	130,3	AAa	68,0
MAV	179,4	AAa	131,4	MAV	71,4

Table 2

Evaluating Clustering

To categorize the texts according to their distance to the training text, we will make use of the k-NN algorithm. In the k-NN algorithm, for example for k=2 we connect every element with its two nearest neighbours. In Table 3 we can see that k-NN with k=2 and k=3 results in a consistent structure of two clusters. This is the case for all n we investigated. For k=1 the groups differ with different n. For k>3 all texts are in the same group.

6-grams		1-NN	2-NN	3-NN
AS	0,0	AS	AS	AS
SBhb	19,5	SBhb	SBhb	SBhb
RGV	48,2	RGV	RGV	RGV
BBh	55,3	BBh	BBh	BBh
MSAb	61,4	MSAb	MSAb	MSAb
Bhav	140,8	Bhav	Bhav	Bhav
AAb	152,3	AAb	AAb	AAb
Vaj	160,5	Vaj	Vaj	Vaj
AAa	173,6	AAa	AAa	AAa
MAV	179,4	MAV	MAV	MAV

Table 3

We can set up some 2D graphs visualising the distances, for 5- and 6-grams, 3- and 4-grams, and 7- and 8-grams, see the Figures 1-3 respectively.

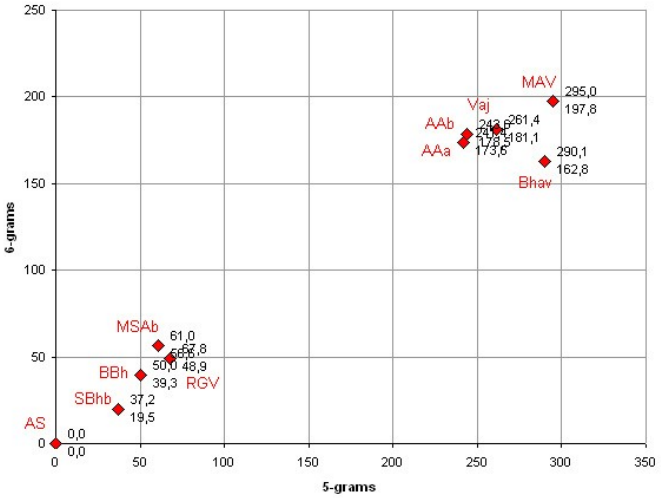

Figure 1

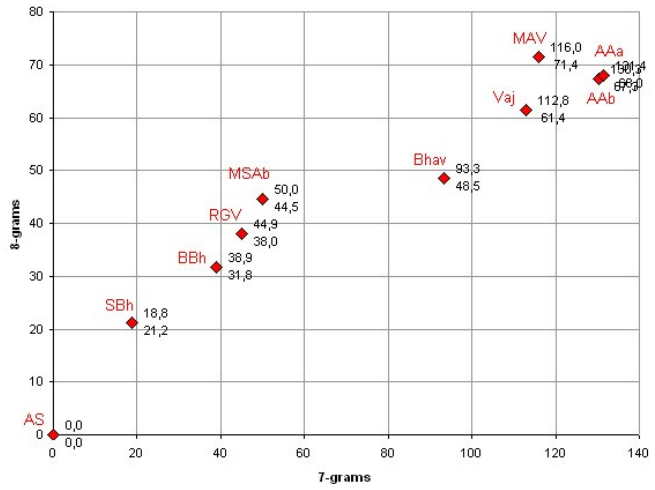

Figure 2

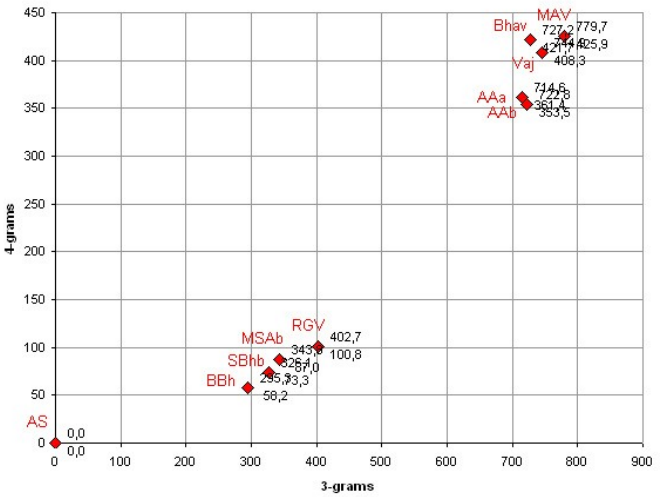

Figure 3

Interpretation and Observations

In Figure 1 we see two clearly distinguished clusters, one being closer to the AS, and the other to the MAV, which we will call clusters 1 and 2 respectively. In Figure 2 we see the same groups, keeping in mind that 3- and 4-grams (and 7- and 8-grams) would be less accurate than 5- and 6-grams. In Figure 3 again we see the same clusters, with cluster 1 more spread out across the diagonal axis.

We might interpret cluster 1 as the Maitreyanātha cluster, as it contains the MAV, and most of the texts in cluster 1 are most often attributed to Maitreyanātha, and cluster 2 as the Asaṅga cluster, containing the AS, YBh and other texts most often attributed to Asaṅga.

Results are summarized in Table 4, together with some of the traditions and modern views concerning the authorship of the scriptures under consideration. (without aspiration to completeness) Wherever a traditional or theoretical view does not correspond to the cluster separation we found here, the indicator is set in red.

Tohoku	3786	3841		4020	4021	4024	4049	4035-42
Taishō			1510	1604	1601	1611	1605	1579
	AA	**Bhav**	**Vaj**	**MSA**	**MAV**	**RGV**	**AS**	**YBh***
Cluster	2	2	2	1	2	1	1	1
Interpretation	M	M	M	A	M	A	A	A
Chinese tradition			**A/V**	A	M	A	A	**M**
Tibetan tradition	M	M		**M**	M	**M**	A	A
Huì Lì				**M**	M			A
mKhas Grub	M			**M**	M	**M**	A	A
Bu sTon	M			**M**	M	**M**	A	A
Ui	M		M	M/V or M/A	M			**M**
Tucci	**A**			M/A	**M/A**	M/A		
Obermiller	**A**			A	**A**	A		
Winternitz	M (karikas)		**A**	**M**				**M**
Lévi				A/A				
Frauwallner	M			**M**	M	**S**	A	
Ruegg	M	(M)	**A**	**M**	M	**M**	A	A

M= Maitreya(nātha), A=Asaṅga, V=Vasubandhu, S=Sāramati * only BBh and SBh tested

Table 4

Finally, we might sum up some observations.

1. Theories supported by our findings are those of Lévi (100%), Ui (75%), followed by Huì Lì, mKhas Grub and Bu sTon (all 67%). A theory unsupported is Winternitz' (75%).

   The Chinese and Tibetan traditions both correspond quite closely (71% and 67%) to the cluster separation we found. However in the Chinese tradition, the Vaj and the YBh are attributed to Asaṅga while they are part of our Maitreyanātha cluster. In the Tibetan tradition, the MSA and the RGV are attributed to Maitreyanātha while they are part of our Asaṅga cluster.

2. The fact that the two clusters are so clearly distinguished, suggests that there will be differences in vocabulary, style, background, which might be demonstrated using other, more direct methods. CNG, like any other authorship attribution method, does not by itself compare author specific traits.

The method can therefore also be used to compare genre, style, consistency, etc. The way we used it here, it is not possible to determine that the texts in one cluster are by the same author. Crucial is that we have only one text (AS) of which the author is undisputed.

In Figure 1, the MSA and the RGV are closest to each other, almost as close as the two versions of the AA, our AAa and AAb, which are textually almost identical, however this must be coincidental, as in Figures 2 and 3 they are much further apart. Also the outlaying position of the Bhav in Figure 3 is coincidental.

3. Interestingly, all texts in which Maitreyanātha is explicitly stated as an author in a heading or closing formula, are part of the Maitreyanātha cluster. (AAa kṛtirmaitreyanāthasya; AAb āryamaitreyanāthaviracitam; Bhav maitreyanāthakṛtā, paṇḍitamaitreyanāthakṛtaḥ; MAV āryamaitreyapraṇītā) The Vaj's closing formula "kṛtir iyam āryāsaṅgapādānām iti" may point to the Chinese tradition. Nevertheless the Vaj is part of our Maitreyanātha cluster.

   The discussion whether Maitreyanātha is a mythological or historical figure is perhaps relevant here to the extent to which a mythological character can or cannot be considered the author of a text. Tāranātha's description of the process of inspired writing suggests that Maitreya dictated the texts to Asaṅga, memorizing them literally, and not for many years later entrusted them to the palmleaf. This way, Maitreya's style of writing could be different from Asaṅga's in another way than when Maitreya (-nātha) would be a human author. Otherwise, we will not be able to distinguish between the styles of Maitreyanātha and Asaṅga. Vice versa, if we conclude from the separation of the two clusters, that there are two different authors here, either we believe in a process of inspired writing as in Tāranātha's description, or we see Maitreyanātha as a human author.

4. The SBh and BBh containing most "śrāvakayānist elements" are part of the Asaṅga cluster. They are closest to the AS, which is also in many aspects a śrāvakayāna-style work. The SBh and BBh correspond to each other very closely, as might be expected, both being part of the larger YBh. ■

Notes

1. Kešelj, V., and others, N-gram-based author profiles for authorship attribution, in Proceedings of the Conference of the Pacific Association for Computational Linguistics, PACLING'03, pp. 255-264, Halifax, 2003
2. Juola, P., Authorship Attribution, in Foundations and Trends in Information Retrieval (R), Vol. 1, No. 3 (2006), pp. 233-334, [Pittsburg], 2008
3. Kešelj, V., and Cercone, N., CNG Method of Weighted Voting, in Ad-hoc Authorship Attribution Competition, Halifax, 2004

Appendix: PHP5 code

```
<?php

//
// Program:         ngrams6_.php
// Function:
// Date:            23 januari 2013
// Language:        PHP 5.x
// Author:          softwareATingmardeboerDOTnl
//

function generate($str, $l) {
    $ret = array();
    if ($l > 0) {
        $len = mb_strlen($str, "UTF-8");
        for ($i = 0; $i < $len; $i++) {
            $h = mb_substr($str, $i, $l, "UTF-8");
            if (mb_strlen($h, "UTF-8")==$l) {
```

```
                    $ret[] = $h;
                }
            }
        }
    return $ret;
}

function distance_($f1, $f2, $cnt1, $cnt2) {
    if (isset($f1) && isset($f2)) {
        $f2 = $f2 / $cnt2;
        $f1 = $f1 / $cnt1;
        $f2 =  2 * ($f1 - $f2) / ($f1 + $f2);
        $f2 = $f2 * $f2;
    }
    else {
        $f2 = 0;  // 4 acc. to Keselj
    }
    return $f2;
}

function distance($arr1, $arr2, $cnt1, $cnt2) {
    $arr3 = array_fill_keys(array_keys($arr1), "");
    $arr4 = array_fill_keys(array_keys($arr2), "");
    $arr = array_merge($arr3, $arr4);
    foreach ($arr as $key => $value) {
        $arr[$key] = distance_($arr1[$key], $arr2[$key], $cnt1, $cnt2);
    }
    ksort($arr);
    arsort($arr, SORT_NUMERIC);
    return $arr;
}

function profile(&$text, $n, $limit, &$cntall, $name) {

    // preprocess
    $text0 = str_replace("\'", "$", $text);

    $len = mb_strlen($text0, "UTF-8");

    // generate all n-grams
    $arr = generate($text0, $n);

    // create profile
    $arr = array_count_values($arr); // generate absolute frequencies
    arsort($arr, SORT_NUMERIC); // reverse sort by frequency
    $cntuniq = count($arr); // number of unique generated n-grams
    $cntall = array_sum($arr); // number of all generated n-grams
    $arr = array_slice($arr , 0, $limit, true);   // culling


[...]
```