Pacific Association for Computational Linguistics

N-GRAM-BASED AUTHOR PROFILES FOR AUTHORSHIP ATTRIBUTION

VLADO KEŠELJ[†] FUCHUN PENG[‡] NICK CERCONE[†] CALVIN THOMAS[†]
[†]*Faculty of Computing Science, Dalhousie University, Canada*
{vlado, nick, thomas}@cs.dal.ca
[‡]*School of Computer Science, University of Waterloo, Canada*
f3peng@cs.uwaterloo.ca

We present a novel method for computer-assisted authorship attribution based on character-level n-gram author profiles, which is motivated by an almost-forgotten, pioneering method in 1976. The existing approaches to automated authorship attribution implicitly build author profiles as vectors of feature weights, as language models, or similar. Our approach is based on byte-level n-grams, it is language independent, and the generated author profiles are limited in size. The effectiveness of the approach and language independence are demonstrated in experiments performed on English, Greek, and Chinese data. The accuracy of the results is at the level of the current state of the art approaches or higher in some cases.

Key words: Authorship attribution, character n-grams, text categorization

1. INTRODUCTION

Automated authorship attribution is the problem of identifying the author of an anonymous text, or text whose authorship is in doubt [Love2002]. A famous example is the *Federalist Papers*, of which twelve are claimed to have been written both by Alexander Hamilton and James Madison [Holmes and Forsyth1995]. Recently, vast repositories of electronic text have become available on the Internet, making the problem of managing large text collections increasingly important. Plagiarism detection is another application area for automated authorship attribution. Automated text categorization is a useful way to organize a large document collection by imposing a desired categorization scheme. For example, categorizing documents by their author is an important case that has become increasingly useful, but also increasingly difficult in the age of web-documents that can be easily copied, translated and edited. Author attribution is becoming an important application in web information management, and is beginning to play a role in areas such as information retrieval, information extraction and question answering.

There are different subtasks of text classification and they can be divided into topic-based and non-topic-based classification. The traditional text classification is topic-based and a typical example is news classification. Recently, there has been an increasing activity in the area of non-topic classification as well, e.g., in sub-tasks such as

1. genre classification [Finn and Kushmerick2003], [E. Stamatatos and Kokkinakis2000],
2. sentiment classification,
3. spam identification,
4. language and encoding identification, and
5. authorship attribution and plagiarism detection [Khmelev and Teahan2003].

Many algorithms have been invented for assessing the authorship of a given text. These algorithms rely on the fact that authors use linguistic devices at every level—semantic, syntactic, lexicographic, orthographic and morphological [Ephratt1997]—to produce their text.

[1]This work is supported by NSERC.

Typically, such devices are applied unconsciously by the author, and thus provide a useful basis for unambiguously determining authorship. The most common approach to determining authorship is to use stylistic analysis that proceeds in two steps: first, specific *style markers* are extracted, and second, a *classification procedure* is applied to the resulting description. These methods are usually based on calculating lexical measures that represent the richness of the author's vocabulary and the frequency of common word use [Stamatatos et al.2001]. Style marker extraction is usually accomplished by some form of non-trivial NLP analysis, such as tagging, parsing and morphological analysis. A classifier is then constructed, usually by first performing a non-trivial feature selection step that employs mutual information or Chi-square testing to determine relevant features.

However, there are several disadvantages of this standard approach. First, techniques used for style marker extraction are almost always language dependent, and in fact differ dramatically from language to language. For example, an English parser usually cannot be applied to German or Chinese. Second, feature selection is not a trivial process, and usually involves setting thresholds to eliminate uninformative features [Scott and Matwin1999]. These decisions can be extremely subtle, because although rare features contribute less signal than common features, they can still have an important cumulative effect [Aizawa2001]. Third, current authorship attribution systems invariably perform their analysis at the *word level*. However, although word level analysis seems to be intuitive, it ignores the fact that morphological features can also play an important role, and moreover that many Asian languages such as Chinese and Japanese do not have word boundaries explicitly identified in text. In fact, word segmentation itself is a difficult problem in Asian languages, which creates an extra level of difficulty in coping with the errors this process introduces. Additionally, the number of authors is small in all reported experiments, so the size of author-specific information is not an issue. If the number of authors, or classes in general, is large, we have to set a limit on the author-specific information, i.e., on the author profile.

In this paper, we propose a simple method that avoids each of these problems. Our approach is based on building a byte-level n-gram author profile of an author's writing. Hence, we do not use any language-dependent information, not even the information about space character used for word separation, the new line character, information about uppercase and lowercase letters, and similar. The profile is essentially a relatively small set of frequent n-grams. Two important operations are:

1. choosing the optimal set of n-grams to be included in the profile, and
2. calculating the similarity between two profiles.

The approach does not depend on a specific language, and it does not require segmentation for languages such as Chinese or Thai. There is no any text preprocessing or higher level processing, so we avoid the necessity for use of taggers, parsers, feature selection, or other language-dependent and non-trivial NLP tools. The small profile size is not important only for efficiency reasons, but it is also a natural mechanism for over-fitting control.

2. RELATED WORK

In [Ephratt1997], a traditional approach, based on linguistic clues, to authorship attribution is presented. A similar approach is recently reported in [E. Stamatatos and Kokkinakis1999] and [E. Stamatatos and Kokkinakis2000]. This work is inspired by these two papers, and our experiments were performed on the Greek data sets used in those papers.

The use of n-gram probability distribution and n-gram models in NLP is a relatively simple idea, but it has been found to be effective in many applications. For example, character level n-gram language models can be easily applied to any language, and even non-language sequences such as DNA and music. Character level n-gram models are widely used in text compression—e.g., the PPM model [T. Bell and Witten1990]—and have recently been found to be effective in text mining problems as well [I. Witten and Teahan1999]. Text categorization with n-gram models has also been attempted by [Cavnar and Trenkle1994]. In the domain of language independent text categorization, [C. Apté and Weiss1994] have used word-based language modeling techniques for both English and German. However, their techniques do not apply to Asian languages where word segmentation remains a significant problem.

Our approach is based on the character n-gram distribution. In [Fuchun et al.2003], it has been shown that the state of the art performance in authorship attribution can achieved by building n-gram language models of the text produced by an author, and these models serve the role of author profiles. The standard perplexity measure is used as the similarity measure between two profiles. In this paper, we apply an alternative approach, using a different method, and building author profiles of a small size.

To motivate the specific similarity function that was used in this paper, let us revive some early published work on the problem. In [Bennett1976], some pioneer methods for authorship attribution were discussed. In a chapter about the use of computers for language processing, a range of problems from some early ideas about language modeling to cryptography, language evolution and authorship attribution, are discussed and tackled using character-level n-grams. Specifically, for authorship attribution (i.e., author identification—as called in the book), the bigram letter statistic was used. Two texts are compared for the same authorship, using the similarity formula (i.e., dissimilarity, more precisely):

$$\sum_{I,J}[M(I,J)-E(I,J)]\cdot[N(I,J)-E(I,J)], \tag{1}$$

where I and J are indices over range $\{1,2,\ldots,26\}$, i.e. all letters, $M(I,J)$ and $N(I,J)$ are normalized character bigram frequencies for one and the other author, and $E(I,J)$ is the same normalized frequency for "standard English." As the bigram frequencies of "standard English" are obviously language-dependent parameters, another dissimilarity measure is given:

$$\sum_{I,J}[M(I,J)-N(I,J)]^2 \tag{2}$$

The results of an experiment using "statistically significant samples of works"[2] by the authors Hemingway, Poe, Baldwin, Joyce, Shakespeare, Cummings, Washington, and Lincoln, are reported, and, in summary, they demonstrate 100% accuracy of the method.

3. ALGORITHM

In our approach, we attempted to revive the algorithm proposed in [Bennett1976], and evaluate how well it performs on the newer data and adapted to newer machines. Hence we use the equation (2). The first equation (1) requires a model of "standard English," which is avoided as a language-dependent parameter. The equation (2) is based on the bigram letter frequencies. If we use 26 letters, then the number of parameters needed for this approach is

[2]We put this in quotes since we are not really sure what precisely the author of [Bennett1976] meant by this phrase.

Algorithm 1 Profile Dissimilarity(profile$_1$, profile$_2$)

```
1: sum ← 0
2: for all n-grams x contained in profile1 or profile2 do
3:    let f1 and f2 be frequencies of x in profile1
      and profile2 (zero if they are not included)
4:    add square of the normalized difference
      of f1 and f2 to sum:
      sum ← sum + (2 · (f1 − f2)/(f1 + f2))^2
5: Return − sum
```

$26^2 = 676$, which is considered a small profile for our purpose. In order to keep the profile small when larger n-grams are used, we define an author profile to be a set of L the most frequent n-grams with their normalized frequencies. So, an author profile is simply a set of L pairs $\{(x_1, f_1), (x_2, f_2), \ldots (x_L, f_L)\}$, of the most frequent n-grams and their normalized frequencies, generated from training data.

The formula (2) gives equal weight to frequency differences of all n-grams included in a profile. This may be justified for bigrams that were used in [Bennett1976], because all of them were reasonably frequent and the sparse data problem is not an issue. However, with larger n-grams the frequency varies more and more, so if we used this absolute difference measure the more frequent n-grams would be emphasized more because the absolute differences in their frequencies are larger. In order to "normalize" these differences, we divide them by the average frequency for a given n-gram. Thus for example, the difference of 0.1 for an n-gram with frequencies 0.9 and 0.8 in two profiles will be less weighted than the same difference for an n-gram with frequencies 0.2 and 0.1. We obtain the following formula:

$$\sum_{n \in \text{profile}} \left(\frac{f_1(n) - f_2(n)}{\frac{f_1(n)+f_2(n)}{2}} \right)^2 = \sum_{n \in \text{profile}} \left(\frac{2 \cdot (f_1(n) - f_2(n))}{f_1(n) + f_2(n)} \right)^2 \tag{3}$$

where $f_1(n)$ and $f_2(n)$ are frequencies of an n-gram n in the author and the document profile.

Algorithm 1 gives the algorithm for calculating the dissimilarity between two profiles. Given two profiles the algorithm returns a positive number, which is a measure of dissimilarity. For identical texts, and more generally, for texts that have identical L most frequent n-grams, the dissimilarity is 0. Using this measure and a set of author profiles, we can easily assign a text to an author by generating a text profile and assigning the text to the category with which the calculated dissimilarity is minimal.

An interesting question is whether we can interpret the dissimilarity score in an absolute way, i.e., whether there is a dissimilarity threshold which separates profiles of the same author from other authors, independently of any specific set of author profiles. As we will see, we did find such a threshold in one data set.

4. EXPERIMENTAL RESULTS

In this section, we present experimental results. No preprocessing is done on texts, and we use simple byte n-grams, treating texts simply as byte sequences. The Perl package `Text::Ngrams` [Kešelj2003] is used to produce n-gram tables.

4.1. English data set

In the first pilot experiment, we have used the data given in Table 1. Two books are

	Author Name	Text size word character
A0	Emily Bronte	13394 / 116051
A1	Edgar Rice Burroughs	18575 / 142706
A2	Lewis Carroll	7806 / 56462
A3	John Cleland	8009 / 84759
A4	Charles Dickens	19890 /164279
A5	H. Ryder Haggard	13191 / 79675
A6	Washington Irving	1176 / 11811
A7	William Shakespeare	1838 / 7643

TABLE 1. Authors appearing in the English data set

included for three authors (Caroll, Burroughs, and Dickens), and one book for the other three. A profile is generated from each book and its similarity with all other books is measured. Since there are three authors with more than one book (two books), the accuracy is measured on attribution of those six books when compared to other eight books, which can be described as a 2-fold cross-evaluation method. In a pilot experiment with n-gram sizes up to 15, we obtained results shown in Table 2. Processing n-grams of larger size that 10 is slow and given the results shown in Table 2, we decided not to perform further experiments on n-grams sizes larger than 10.

2	3	4	5	7	10	15
0.67	0.83	1	0.83	1	1	0.83

TABLE 2. Pilot experiment on English

For n-gram sizes 7 and 10 with profile size 1000, there exist an absolute threshold, 0.1950 and 0.2880 respectively, that separates dissimilarity of same-author texts from texts written by different authors, which answers our previous question whether such threshold exist.

The results of a more extensive set of experiments are shown in Table 3. The highest accuracy of **100%** is reached for several n-gram and profile sizes.

The method is very successful on this data set. It is interesting to see that the maximal precision of 100% is achieved with 1-grams (simple byte frequencies), and profile size 20. We assume that this is accidental due to a relatively small set of texts. We can be more confident about a cluster of 100% accuracies of profiles with n-gram size 4–8 and profile size 500–3000.

In further experiments, we compare this method with some other results. The same data sets are used and the same experimental 'train-and-test' procedure: The data is divided into a training set and a testing set.

Profile size	N-gram size									
	1	2	3	4	5	6	7	8	9	10
20	1	0.67	0.67	0.67	0.5	0.83	0.67	0.67	0.67	0.67
50	0.67	0.67	0.83	0.67	0.83	0.83	0.83	0.67	0.67	0.67
100	0.5	0.67	1	1	0.83	0.83	0.83	0.83	0.67	0.83
200	0.5	0.83	0.83	0.83	1	0.83	0.83	1	0.83	0.83
500	0.5	0.83	0.83	1	0.83	1	1	0.83	0.83	0.83
1000	0.5	0.67	0.83	0.83	0.83	1	1	0.83	0.83	0.83
1500	0.5	0.33	0.83	1	1	1	1	1	0.83	0.83
2000	0.5	0.33	0.83	1	1	1	1	1	0.83	0.83
3000	0.5	0.33	0.83	0.83	1	1	1	1	0.83	0.83
4000	0.5	0.33	0.83	0.83	0.83	0.83	0.83	0.83	0.83	0.83
5000	0.5	0.33	0.83	0.83	0.83	0.83	0.83	0.83	0.83	0.83

TABLE 3. Accuracy for English

4.2. Greek data sets

We have experimented with two Greek data sets, A and B, used in the studies of [E. Stamatatos and Kokkinakis1999] and [E. Stamatatos and Kokkinakis2000]. Both sets were originally downloaded from the web site of the Modern Greek Weekly Newspaper TO BHMA. Each of the two data sets consists of 200 singly-authored documents written by 10 different authors, with 20 different documents written by each author. In our experiments we used 10 of each authors' documents as training data and 10 as test instances. The specific authors that appear are shown in Table 4. The main difference between the two sets is that the documents in group A are written by journalists on a variety of topics, including news reports, editorials, etc., whereas the documents in group B are written by scholars on topics in science, history, culture, etc. The result is that the documents in group A are more heterogeneous in their style, whereas the documents in group B are more homogeneous owing to the more rigid strictures of academic writing [E. Stamatatos and Kokkinakis2000].

The accuracy for the set GreekA is given in Table 5. The best reported accuracy in [E. Stamatatos and Kokkinakis2000] is 72% for this data set, and in the best reported accuracy in [Fuchun et al.2003] is 73%. We can see that this method compares favorably for certain configuration of parameters, achieving the best accuracy of 85%. Actually, it achieves an accuracy better than previously reported for all profile sizes ≥ 1000 and n-gram size ≥ 3.

The results for the data set Greek B are given in Table 6. For this data set, the best previously reported accuracies were 70% in [E. Stamatatos and Kokkinakis2000], and 0.89 in [Fuchun et al.2003]. Again, this method gives a better best accuracy of 97%. There is a large cluster of parameter values for which the accuracy is better than previously reported.

The data set Greek Bp (B plus) contains the same testing documents as Greek B. The difference is only in a larger training set (it is doubled). The results for the data set Greek Bp are given in Table 7. For this data set, the best previously reported accuracies were 87% in [E. Stamatatos and Kokkinakis2000], and 0.92 in [Fuchun et al.2003]. This method gives a better best accuracy of 97%.

Data set		Author Name	Train size (characters)
A	A0	G. Bitros	47868
	A1	K. Chalbatzakis	71889
	A2	G. Lakopoulos	77549
	A3	T. Lianos	45766
	A4	N. Marakis	59785
	A5	D. Mitropoulos	70210
	A6	D. Nikolakopoulos	75316
	A7	N. Nikolaou	51025
	A8	D. Psychogios	35886
	A9	R. Someritis	50816
B	B0	S. Alaxiotis	77295
	B1	G. Babiniotis	75965
	B2	G. Dertilis	66810
	B3	C. Kiosse	102204
	B4	A. Liakos	89519
	B5	D. Maronitis	36665
	B6	M. Ploritis	72469
	B7	T. Tasios	80267
	B8	K. Tsoukalas	104065
	B9	G. Vokos	64479

TABLE 4. Authors in two Greek data sets, A and B.

Profile size	N-gram size									
	1	2	3	4	5	6	7	8	9	10
20	0.37	0.41	0.46	0.33	0.32	0.37	0.32	0.25	0.32	0.39
50	0.36	0.4	0.46	0.43	0.43	0.4	0.44	0.49	0.45	0.4
100	0.42	0.5	0.51	0.56	0.47	0.51	0.5	0.53	0.46	0.47
200	0.41	0.59	0.72	0.62	0.58	0.53	0.59	0.56	0.58	0.55
500	0.41	0.72	0.72	0.66	0.71	0.71	0.69	0.69	0.64	0.62
1000	0.41	0.74	0.8	0.8	0.76	0.75	0.78	0.73	0.77	0.62
1500	0.41	0.68	0.82	0.77	0.78	0.77	0.74	0.78	0.78	0.67
2000	0.41	0.61	0.81	0.78	0.8	0.77	0.75	0.76	0.81	0.74
3000	0.41	0.61	0.83	0.81	0.78	0.77	0.81	0.81	0.79	0.77
4000	0.41	0.61	0.85	0.78	0.75	0.77	0.81	0.78	0.8	0.78
5000	0.41	0.61	0.82	0.79	0.75	0.8	0.78	0.8	0.8	0.79

TABLE 5. Accuracy for Greek (set A)

4.3. Chinese data set

We used the same Chinese data set as in [Fuchun et al.2003]. Due to a large number of Chinese characters, the number of different n-grams generated from the Chinese texts

Profile size	N-gram size									
	1	2	3	4	5	6	7	8	9	10
20	0.41	0.49	0.45	0.49	0.43	0.44	0.4	0.38	0.35	0.35
50	0.58	0.54	0.55	0.53	0.65	0.56	0.54	0.54	0.54	0.48
100	0.61	0.67	0.72	0.71	0.67	0.65	0.63	0.62	0.63	0.58
200	0.6	0.74	0.81	0.81	0.74	0.78	0.71	0.68	0.75	0.7
500	0.6	0.86	0.87	0.79	0.81	0.83	0.79	0.8	0.81	0.82
1000	0.6	0.81	0.84	0.85	0.83	0.87	0.84	0.87	0.82	0.84
1500	0.6	0.68	0.92	0.91	0.86	0.91	0.89	0.86	0.83	0.85
2000	0.6	0.65	0.94	0.91	0.9	0.93	0.89	0.88	0.88	0.88
3000	0.6	0.65	0.93	0.93	0.95	0.88	0.89	0.9	0.88	0.86
4000	0.6	0.65	0.93	0.94	0.93	0.9	0.87	0.89	0.88	0.83
5000	0.6	0.65	0.92	0.97	0.93	0.92	0.87	0.88	0.85	0.82

Table 6. Accuracy for Greek (set B)

Profile size	N-gram size									
	1	2	3	4	5	6	7	8	9	10
20	0.49	0.44	0.53	0.53	0.47	0.44	0.47	0.48	0.46	0.43
50	0.65	0.56	0.58	0.63	0.67	0.58	0.53	0.56	0.58	0.5
100	0.61	0.67	0.73	0.67	0.68	0.73	0.71	0.61	0.59	0.57
200	0.57	0.76	0.78	0.8	0.8	0.81	0.75	0.72	0.76	0.74
500	0.57	0.91	0.87	0.87	0.9	0.85	0.85	0.86	0.82	0.81
1000	0.57	0.85	0.93	0.9	0.87	0.89	0.88	0.85	0.87	0.83
1500	0.57	0.71	0.93	0.95	0.94	0.93	0.88	0.86	0.85	0.85
2000	0.57	0.66	0.97	0.97	0.96	0.93	0.89	0.87	0.87	0.84
3000	0.57	0.66	0.96	0.94	0.95	0.94	0.9	0.88	0.87	0.84
4000	0.57	0.66	0.97	0.95	0.94	0.91	0.91	0.88	0.88	0.84
5000	0.57	0.66	0.97	0.95	0.94	0.92	0.88	0.86	0.85	0.84

Table 7. Accuracy for Greek (set Bp)

was prohibitively large, even for small n. For this reason, we limited the total number of n-grams being counted to 200000. Otherwise, the same method is applied. The results are reported in Table 8. The best achieved accuracy is 0.89, which is worse than 0.94 reported in [Fuchun et al.2003]. The cause for a worse performance in this case may be that the restriction in n-gram counting to 200,000 is too strict, or the fact that we treat the text simply as a sequence of bytes instead of Chinese characters. Namely, the characters are 2-byte long, so 75% n-grams that we count may not be very useful because they include half-characters (i.e., all odd-length n-grams, and half of the even-length n-grams), and they are not sensible strings in Chinese.

Profile size	N-gram size									
	1	2	3	4	5	6	7	8	9	10
20	0.34	0.34	0.33	0.28	0.31	0.30	0.32	0.28	0.14	0.29
50	0.41	0.47	0.56	0.49	0.45	0.38	0.33	0.32	0.32	0.34
100	0.44	0.61	0.48	0.57	0.53	0.48	0.46	0.44	0.41	0.39
200	0.45	0.71	0.63	0.63	0.69	0.67	0.66	0.61	0.47	0.44
500	0.45	0.78	0.74	0.82	0.75	0.74	0.67	0.65	0.58	0.60
1000	0.45	0.82	0.83	0.85	0.80	0.78	0.74	0.74	0.68	0.61
1500	0.45	0.83	0.85	0.85	0.84	0.82	0.77	0.74	0.68	0.64
2000	0.45	0.83	0.86	0.88	0.85	0.83	0.78	0.75	0.71	0.67
3000	0.45	0.78	0.87	0.88	0.88	0.84	0.81	0.76	0.74	0.70
4000	0.45	0.73	0.85	0.88	0.89	0.87	0.83	0.78	0.76	0.73
5000	0.45	0.60	0.86	0.89	0.89	0.87	0.84	0.79	0.76	0.73

TABLE 8. Accuracy for Chinese

5. CONCLUSION

We have presented a new method to automated authorship attribution based on byte n-gram profiles. We have demonstrated our approach on three different languages and obtained a state of the art performance. In experiment on the Greek data sets, the results were uniformly better than previously reported. The approach relies on author profiles of restricted size and very simple algorithm.

REFERENCES

[Aizawa2001] A. Aizawa. 2001. Linguistic techniques to improve the performance of automatic text categorization. In *Proceedings 6th NLP Pac. Rim Symp. NLPRS-01.*

[Bennett1976] William Ralph Bennett. 1976. *Scientific and engineering problem-solving with the computer.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

[C. Apté and Weiss1994] F. Damerau C. Apté and S. Weiss. 1994. Toward language independent automated learning of text categorization models. In *Proceedings SIGIR-94.*

[Cavnar and Trenkle1994] W. Cavnar and J. Trenkle. 1994. N-gram-based text categorization. In *Proceedings SDAIR-94.*

[E. Stamatatos and Kokkinakis1999] N. Fakotakis E. Stamatatos and G. Kokkinakis. 1999. Automatic authorship attribution. In *Proceedings EACL-99.*

[E. Stamatatos and Kokkinakis2000] N. Fakotakis E. Stamatatos and G. Kokkinakis. 2000. Automatic text categorization in terms of genre and author. *Computational Linguistics*, 26(4):471–495.

[Ephratt1997] M. Ephratt. 1997. Authorship attribution - the case of lexical innovations. In *Proc. ACH-ALLC-97.*

[Finn and Kushmerick2003] Aidan Finn and Nicholas Kushmerick. 2003. Learning to classify documents according to genre. In *IJCAI-03 Workshop on Computational Approaches to Style Analysis and Synthesis.*

[Fuchun et al.2003] Peng Fuchun, Dale Schuurmans, Vlado Kešelj, and Shaojun Wang. 2003. Automated authorship attribution with character level language models. In *Proceedings of the 10th*

Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003), Budapest, Hungary, April 12–17.

[Holmes and Forsyth1995] D. Holmes and R. Forsyth. 1995. The federalist revisited: New directions in authorship attribution. *Literary and Linguistic Computing*, 10:111–127.

[I. Witten and Teahan1999] M. Mahoui I. Witten, Z. Bray and W. Teahan. 1999. Text mining: A new frontier for lossless compression. In *Proceedings of the IEEE Data Compression Conference (DCC)*.

[Kešelj2003] Vlado Kešelj. 2003. Perl package Text::Ngrams.
`http://www.cs.dal.ca/~vlado/srcperl/Ngrams` or
`http://search.cpan.org/author/VLADO/Text-Ngrams-0.03/Ngrams.pm`.

[Khmelev and Teahan2003] D. Khmelev and W. Teahan. 2003. A repetition based measure for verification of text collections and for text categorization. In *SIGIR'2003*, Toronto, Canada.

[Love2002] H. Love. 2002. *Attributing Authorship: An Introduction*. Cambridge University Press.

[Pang et al.2002] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
`http://citeseer.nj.nec.com/pang02thumbs.html`.

[Scott and Matwin1999] S. Scott and S. Matwin. 1999. Feature engineering for text classification. In *Proceedings ICML-99*.

[Stamatatos et al.2001] E. Stamatatos, N. Fakotakis, and G. Kokkinakis. 2001. Computer-based authorship attribution without lexical measures. *Computers and the Humanities*, 35:193–214.

[T. Bell and Witten1990] J. Cleary T. Bell and I. Witten. 1990. *Text Compression*. Prentice Hall.