# Postprint: Feature Selection Methods for Non-Standard Text in Authorship Identification

**Authors:** Guo Xu, Qi Ruihua

**Date:** 2017-11-08T00:00:00+00:00

## Abstract

【Purpose】 To extract features from non-standard text for identifying online text authorship. 【Method】 We propose two feature extraction methods for non-standard text: utilizing a non-standard text similarity M defined on the basis of the Jaccard coefficient, and utilizing the frequency of occurrence of non-standard text within the text. 【Results】 The identification accuracy of the two features reaches 85.1% and 80.2%, respectively. After incorporating these two features, the identification accuracy of traditional classifiers based on statistical value features improves by 5.8% and 4%, respectively. 【Limitations】 Only the lexical-level non-standardness of online text is considered, without investigating higher-level characteristics such as syntactic and structural levels. 【Conclusion】 The feature extraction methods proposed in this paper can effectively extract non-standard text features and contribute to improving the identification accuracy of authorship identification systems.

## Full Text

## Preamble

ChinaXiv Collaborative Journal, Issue 276, 2016, No. 11
Feature Selection Methods for Non-standard Text in Author Identification
Guo Xu, Qi Ruihua
(School of Software, Dalian University of Foreign Languages, Dalian 116044)

## Abstract

[**Objective**] This paper aims to extract features from non-standard text to identify the authorship of online texts. [**Methods**] We propose two methods for feature extraction from non-standard text: first, utilizing a non-standard text similarity metric M defined based on the Jaccard coefficient; second, employing the frequency of non-standard text occurrences in the corpus. [**Results**]

The identification accuracy of the two features reached 85.1% and 80.2% respectively. When incorporated into traditional classifiers based on statistical features, the overall recognition accuracy improved by 5.8% and 4% respectively. [**Limitations**] The study only considered lexical-level non-standardization in online texts, without investigating higher-level characteristics such as syntactic or structural features. [**Conclusion**] The proposed feature extraction methods can effectively capture non-standard text features and contribute to improved accuracy in authorship identification systems.

**Keywords:** Author identification; Non-standard text; Network text; Text similarity
**Classification:** TP391.1; G353

## 1. Introduction

Authorship identification represents an important direction in natural language processing that has consistently attracted significant attention. With the rise of social networks such as WeChat and Weibo, coupled with the advent of the big data era, the demand for accurate authentication of online text authorship has become increasingly urgent for both ethical and information security considerations. For example, in public opinion monitoring, it is crucial to determine whether malicious messages originate from the same author, or to identify the authors of spam emails. However, one persistent issue that negatively impacts the effectiveness of various authorship identification techniques is "non-standard text writing."

Traditional authorship identification systems achieve high accuracy primarily when dealing with well-written, standardized samples. When confronted with non-standard writing, the recognition accuracy of these systems drops substantially. Even when some systems acknowledge the issue of "non-standard writing," they typically resort to simple exclusion or normalization without deeper processing [1-2]. In reality, non-standard textual expressions often represent a concentrated manifestation of an author's unique writing style. Therefore, this paper attempts to extract features from non-standard text for authorship identification. This approach to feature extraction is closely tied to the online application environment and can complement existing typical features such as statistical features and multi-level features, as non-standard text features specifically target samples with low recognition rates due to non-standard writing.

## 2. Related Research on Non-standard Text

Non-standard text refers to text produced by authors' non-standard or erroneous writing behaviors, primarily arising from spelling errors, colloquialisms, filler words, internet slang, emoticons, abbreviations, and slang [3]. In English writing, examples such as "lol," "soooo," "1ist," and "CU" all constitute non-standard text. While rarely appearing in formal publications, non-standard text is ubiquitous in online texts such as blogs. Consequently, most research on

online texts touches upon non-standard text to some extent, with main contributions falling into four categories:

First, some studies acknowledge non-standard text as a characteristic of online texts but do not conduct specific research on it. Although many current studies recognize that non-standardization negatively impacts research results, their systems demonstrate sufficient robustness to achieve good performance without specifically addressing non-standard text. For example, Nie et al. mentioned that spelling errors, slang, and abbreviations might affect classification accuracy in their research on hypertext question-answering systems [4].

Second, some studies specifically experiment with non-standard text and achieve good results. For instance, Chen et al. proposed a semantic topic extraction method for online texts based on Baidu Baike, arguing that non-standardization makes online texts difficult to mine. Because Baidu Baike contains rich entries that include even non-standard internet terms, their method achieved good results on both standard and non-standard texts [5].

Third, some studies normalize non-standard text into standard forms. For example, Zhang et al. used "proofreading dictionaries," "domain noun dictionaries," and "internet sentiment dictionaries" to normalize non-standard text and annotate sentiment orientation in their experiments on extracting opinion sentences [6]. Similarly, Dehkharghani et al. converted special symbols and emoticons into corresponding sentiments when classifying sentiment in Twitter texts [7].

Fourth, some studies extract features from non-standard text. While the first three approaches mitigate the negative effects of non-standard text to some extent, they fail to fully utilize it. In fact, in certain research domains such as authorship identification, non-standard text often provides excellent discriminative power. For example, some authors habitually use the internet slang "CU" to mean "See You," while others add multiple "o"s after "So" to express stronger emotion. Therefore, finding effective methods to extract features from non-standard text for authorship identification can not only eliminate the negative impact of non-standard text but also effectively improve recognition accuracy, as demonstrated by Iqbal et al. who used lexical spelling errors and syntactic error features for email authorship identification [8].

## 3. Definitions Related to Non-standard Text

We define words in a word list as standard text. Using a lookup table approach, we identify words in the text that are not included in the word list and are not named entities, numbers, URLs, etc., as non-standard text. We also define non-standard degree N and non-standard text similarity M.

The non-standard degree N represents the degree of non-standardization in a text, calculated as:

$$N = \frac{S_n}{S}$$

where $S_n$ is the number of non-standard words in the sample, and $S$ is the total number of words in the sample.

The non-standard text similarity M represents the similarity between texts. Traditional text similarity calculation methods mostly first obtain keywords from text using algorithms such as TF-IDF, converting a text into a vector composed of keywords, then use cosine similarity or Jaccard coefficient to represent similarity [9]. When calculating similarity M, we use non-standard text contained in the text as keywords and represent similarity through a modified Jaccard coefficient. The Jaccard coefficient equals the intersection of two sets divided by their union. Our similarity calculation follows this principle, but considering that our application is not simply to judge whether two texts are similar, but to compare the similarity of one text with multiple texts to determine which text it is more similar to, this horizontal comparison should satisfy three conditions:

1. As the number of "shared non-standard text" between two texts increases, the M value should increase.
2. As the frequency of a particular "shared non-standard text" increases, the M value should increase, but not excessively, to avoid one frequently occurring "shared non-standard text" making the M value too large. This is because when comparing cases where there are two different "shared non-standard texts" each appearing once versus one "shared non-standard text" appearing twice, we expect the former to have a higher M value.
3. It should avoid bias caused by too many or too few non-standard texts in a particular text.

Accordingly, we define the non-standard text similarity M between text a and text b as:

$$M = \frac{\sum_{i=1}^{n} \ln(P_{ai} + 1) \times \ln(P_{bi} + 1)}{S_a + S_b}$$

where $n$ is the number of different types of "shared non-standard text" between the two texts, $P_{ai}$ is the count of the $i$-th type of "shared non-standard text" in text a, $P_{bi}$ is the count in text b, $S_a$ is the total number of non-standard words in text a, and $S_b$ is the total number in text b.

In formula (2), summing all "shared non-standard text" satisfies condition (1). Taking the natural logarithm of the count of each type of "shared non-standard text" plus 1 satisfies condition (2), making the relationship between one frequently occurring "shared non-standard text" and multiple single-occurrence ones equal to $2n-1$—that is, one "shared non-standard text" appearing $2n-1$ times yields the same M value as $n$ different "shared non-standard texts" each

appearing once. Finally, dividing by the total number of non-standard words in both samples satisfies condition (3).

## 4. Acquisition of Non-standard Text

We extracted non-standard text from English blogs using data primarily from two corpora. The first is the authorship corpus constructed by Schler et al. [10-11], containing 681,288 blog posts from 19,320 authors on Blogger (https://blogger.com/). The second is the Moby word list containing approximately 350,000 words, constructed by Ward [12].

After expanding the Moby word list, we ultimately defined a word list containing 377,121 words as standard text. Based on this, we extracted non-standard text from 517,643 blog posts by 18,828 authors from the authorship corpus. The specific process was as follows:

1. Text preprocessing: Using the natural language processing toolkit developed by Stanford University's NLP Group [13], we tokenized and lemmatized the corpus.
2. Named entity removal: Using the same toolkit [13], we identified and removed named entities from the corpus.
3. Initial non-standard text extraction: Using the lookup table method, we counted words not appearing in the word list and their frequencies as preliminary non-standard text.
4. Non-standard text filtering: We removed stop words from the preliminary non-standard text. The stop word list is shown in Table 1 .

Table 1: Stop Word List
- Single punctuation marks (e.g., ,.?)
- Multiple consecutive punctuation marks (e.g., ???, ***, ^_^) are not considered stop words
- Contractions like "i'm", "n't"
- Alphanumeric combinations like "12m", "123"
- URLs like "http://blogger.com"
- Email addresses like "123@163.com"
- Hyphenated words like "T-shirt"
- File names like "123.jpg"
- Technical terms like "website", "homepage"
- Non-English text such as Chinese or Korean characters

We ultimately obtained 193,028 types of non-standard text, totaling 1,365,942 occurrences. Among them, 182,549 types (1,043,210 occurrences) consisted entirely of letters. Non-standard text appearing more than once accounted for 65,529 types (1,238,443 occurrences), while those appearing more than 100 times accounted for 1,121 types (809,685 occurrences). Ninety-nine percent of non-standard text appeared no more than 60 times. The top 10 most frequent non-standard texts and the top 10 most frequent letter-only non-standard texts are shown in Table 2 .

In terms of non-standard degree, the overall degree for all texts was 0.0109. When calculated by author, the average non-standard degree was 0.01282 with a standard deviation of 0.017, ranging from a maximum of 0.2268 to a minimum of 0. Among these, 521 authors (2.78% of the total) had a non-standard degree of 0 (i.e., wrote completely standard text). When calculated by blog post, the average non-standard degree was 0.0155 with a standard deviation of 0.041, ranging from a maximum of 1 to a minimum of 0. Among these, 293,254 blog posts (56.65% of the total) had a non-standard degree greater than 0 (i.e., contained non-standard text).

Thus, the authorship corpus contains substantial non-standard text suitable for further experiments, while also demonstrating the prevalence of non-standard writing in online texts.

## 5. Experiments

### 5.1 Classification Experiments Based on Basic Statistical Features

To evaluate the effectiveness of non-standard text features, we conducted comparative experiments using 18 basic statistical features for authorship identification: character count, digit count, lowercase letter count, uppercase letter count, word count, distinct word count, punctuation count, distinct punctuation count, words longer than 4 characters, average word length, hapax legomena count, dis legomena count, words appearing more than twice, sentence count, average sentence length (in words), average sentence length (in characters), longest sentence word count, and shortest sentence word count.

Experiments on basic statistical features were conducted using the data mining software WEKA [14], employing four classifiers: Bayes Network, Support Vector Machine (SVM), Neural Networks, and Bagging. The Bayes Network classifier was based on the Tree Augmented Naive Bayes (TAN) algorithm. The SVM classifier used the Sequential Minimal Optimization (SMO) algorithm with a Polynomial Kernel. The Neural Network classifier employed backpropagation with a learning rate of 0.3, momentum of 0.2, and a maximum of 500 iterations. The Bagging classifier used decision trees as base models with 10 iterations [14].

We randomly selected 10 groups of 6 authors each from the authorship corpus and performed 10-fold cross-validation. The average results are shown in Table 3 .

Based on these results and considering both accuracy and computation time, we selected the Bagging classifier for subsequent experiments.

### 5.2 Classification Experiments Based on Non-standard Text Similarity M

This classification experiment used non-standard text similarity M as the feature. Notably, when calculating M, we treated similar non-standard forms such

as “sooooo” and “soooo,” or “!!!” and “!!!!” as the same type, using a stem-like reduction method to merge them before calculation (i.e., both “sooooo” and “soooo” were converted to “sooo”).

We employed two classification algorithms: K-nearest neighbor Bayes and a unification algorithm. The K-nearest neighbor Bayes algorithm calculates the similarity M between an unknown sample and every known sample, identifies the K samples with the highest M values, uses the proportion of each author class among these K samples as class-conditional density, and finally applies Bayes' theorem to determine the unknown sample's class. The unification algorithm merges all known samples of each class into a single large sample, calculates the similarity M between the unknown sample and each merged class sample, and assigns the unknown sample to the class with the highest M value.

Since similarity M represents similarity in terms of non-standard text, our classifiers have no effect on samples without non-standard text (i.e., samples with non-standard degree N equal to 0) or on unknown samples sharing no non-standard text with any known sample (i.e., all similarity M values equal 0). Therefore, the experiments are only effective on samples beyond these two categories, with sample validity rate defined as the ratio of effective samples to all samples.

We designed corresponding classifiers based on these algorithms and selected 6 authors with varying numbers of texts (ranging from many to few) totaling 1,652 samples from the authorship corpus for 10-fold cross-validation. The detailed experimental data are shown in Table 4 and Table 5 . Table 4 records basic sample information, while Table 5 shows the experimental results of using the two classifiers to identify effective texts, verifying classification performance across 15 combinations of 2 authors, 15 combinations of 4 authors, and 1 combination of all 6 authors.

To more accurately verify the identification effectiveness of similarity M, we randomly extracted 30 groups of data (10 groups each of 6 authors, 4 authors, and 2 authors) from the authorship corpus for 10-fold cross-validation. The results are shown in Table 6 .

These experiments reveal that classifiers based on similarity M have a sample validity rate of approximately 40%, meaning they cannot identify about 60% of samples—most of which are well-written texts or texts without shared non-standard features. Like most classification algorithms, both classifiers perform best on 2-class problems, with accuracy decreasing as the number of classes increases. Additionally, because the unification algorithm utilizes more sample information during decision-making, its identification performance is generally better than K-nearest neighbor Bayes, though the latter is faster and more scalable, as it doesn't require recalculating similarities between old samples when new samples are added.

For effective samples, the classifier based on non-standard text similarity M achieves an average accuracy about 13% higher than the basic statistical features

classifier. As the number of author classes increases, the accuracy of the latter declines more rapidly. This makes the advantage of the similarity M-based classifier more pronounced with more classes. Although low sample validity is an inherent limitation of non-standard text features, the ability to clearly distinguish between identifiable and unidentifiable samples makes them easy to combine with other classifiers.

### 5.3 Classification Experiments Based on Non-standard Text Frequency

In addition to using similarity M, we also used the frequency of non-standard text occurrences as features, representing each sample as a vector $(N_1, N_2, N_3, ..., N_n)$ where the dimension $n$ is determined by the types of non-standard text in the known samples.

We first conducted classification experiments on the 6 authors from Section 5.2 using the WEKA software with a Bagging classifier based on decision trees, comparing its performance with basic statistical features. The results are shown in Table 7 . The sample validity rates of classifiers using non-standard text frequency and similarity M are essentially consistent because both target the same types of non-standard text, with minor differences arising from the merging of similar forms like “sooooo” and “soooo” in the latter. In terms of identification accuracy, the frequency-based approach is slightly lower than the similarity M approach, but its feature format is more standard and applicable to most classification algorithms. Moreover, under the same classifier, the accuracy of the frequency-based classifier is significantly higher than that of basic statistical features.

### 5.4 Combining Non-standard Text Features with Basic Statistical Features

There are two methods to overcome the insufficient sample validity of non-standard text features: first, concatenating non-standard text features with other features to form a new feature vector; second, first classifying valid samples with a non-standard text feature-based classifier, then classifying the remaining invalid samples with another classifier. Based on these methods, we combined non-standard text features with basic statistical features for authorship identification.

Following the first method, we concatenated non-standard text features with basic statistical features to form a new feature vector and used the Bagging classifier to perform 10-fold cross-validation on the 6 authors from Section 5.2. The results are shown in Table 8 .

Following the second method, we first used the unification algorithm-based classifier with similarity M to classify samples, then used the Bagging classifier with basic statistical features to classify the invalid samples that the first classifier

could not identify. The 10-fold cross-validation results on the 6 authors are shown in Table 9 .

To further verify the effectiveness of non-standard text features, we conducted additional 10-fold cross-validation experiments on the 30 randomly extracted groups from Section 5.2. The results are shown in Table 10 .

These experiments demonstrate that introducing non-standard text features significantly improves identification accuracy, with a maximum improvement of 5.8%. The second method proves more effective and stable because it better leverages the characteristic of non-standard text features to clearly distinguish between identifiable and unidentifiable samples.

## 6. Conclusion

Addressing the prevalence of non-standard writing in online texts, this paper proposes methods for extracting features from non-standard text to identify authorship. We designed and completed author identification experiments using both similarity M and non-standard text frequency as features. The results demonstrate that non-standard text features can effectively identify authors. The proposed feature models are closely connected to the online application environment and explore a novel angle for feature extraction that differs from most existing models, making them easy to combine with or complement other feature models.

However, the study has several limitations: First, it only addressed the "which author" problem in authorship identification, not the "whether it belongs to this author" problem. These differ in that the former can be a multi-class problem where unknown samples must belong to one of the known classes, involving horizontal comparison to determine the closest match; the latter is typically a binary classification problem requiring a threshold to determine if an unknown sample belongs to a known author. Second, the study only considered lexical-level non-standardization in online texts, without investigating higher-level characteristics such as syntactic or structural features [15]. In reality, non-standard writing at higher levels, such as non-standard word order or non-standard character usage, often better reflects an author's writing habits.

## References

[1] Abbasi A, Chen H. Applying Authorship Analysis to Extremist-group Web Forum Messages [J]. IEEE Intelligent Systems, 2005, 20(5): 67-75.

[2] Iqbal F, Binsalleeh H, Fung B C M, et al. A Unified Data Mining Solution for Authorship Analysis in Anonymous Textual Communications [J]. Information Sciences, 2013, 231(9): 98-112.

[3] Luo Changri, He Tingting. Characteristics of Internet Language and Its Emotional Meanings [J]. Journal of Wuhan University of Technology: Social

Sciences Edition, 2015, 28(2): 322-328.

[4] Nie L, Wang M, Gao Y, et al. Beyond Text QA: Multimedia Answer Generation by Harvesting Web Information [J]. IEEE Transactions on Multimedia, 2013, 15(2): 426-441.

[5] Chen Yewang, Wang Huazhen, Li Haibo, et al. Topic Extraction Method for Chinese Web Text Based on Baidu Baike and Text Classification [J]. Journal of Chinese Computer Systems, 2012, 33(12): 2605-2610.

[6] Zhang Wenwen, Wang Ting. Unsupervised Subjective Sentence Extraction for Non-Standard Texts [J]. Computer and Digital Engineering, 2013, 41(1): 64-68.

[7] Dehkharghani R, Mercan H, Javeed A, et al. Sentimental Causal Rule Discovery from Twitter [J]. Expert Systems with Applications, 2014, 41(10): 4950-4958.

[8] Iqbal F, Binsalleeh H, Fung B C M, et al. Mining Writeprints from Anonymous E-mails for Forensic Investigation [J]. Digital Investigation, 2010, 7(1): 56-64.

[9] Huang Chenghui, Yin Jian, Hou Fang. A Text Similarity Measurement Combining Word Semantic Information with TF-IDF Method [J]. Chinese Journal of Computers, 2011, 34(5): 856-864.

[10] Schler J, Koppel M, Argamon S, et al. Effects of Age and Gender on Blogging [C]. In: Proceedings of the 2006 AAAI Spring Symposium. 2006.

[11] Schler J, Koppel M, Argamon S, et al. The Blog Authorship Corpus [DS/OL]. [2014-05-28]. http://u.cs.biu.ac.il/~koppel/BlogCorpus.htm.

[12] Ward G. Moby Words [DS/OL]. [2016-06-24]. http://icon.shef.ac.uk/Moby/mwords.html.

[13] Manning C D, Surdeanu M, Bauer J, et al. The Stanford CoreNLP Natural Language Processing Toolkit [C]. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 2014.

[14] Witten I H, Frank E, Hall M A. Data Mining [M]. Beijing: China Machine Press, 2012.

[15] Qi Ruihua, Yang Deli, Guo Xu, et al. Blogger Identification Based on Multidimensional Stylistic Features [J]. Journal of the China Society for Scientific and Technical Information, 2015, 34(6): 628-634.

**Author Contributions:** Guo Xu: proposed the research plan, conducted experiments, and wrote the paper; Qi Ruihua: proposed research ideas and experimental procedures, revised the paper.

**Conflict of Interest Statement:** All authors declare no conflict of interest.

**Supporting Data:** Supporting data is self-archived by the authors, E-mail: guoxu@dlufl.edu.cn.

[1] Guo Xu, Qi Ruihua. nsw.xlsx. Non-standard text statistics table.
[2] Guo Xu, Qi Ruihua. nsm.xlsx. Non-standard text identification effectiveness statistics table.

**Received:** July 12, 2016
**Revised:** September 19, 2016

*Note: Figure translations are in progress. See original paper for figures.*