Taxonomy and Evaluation of Markers for Computational Stylistics

Foaad Khosmood[1] **and Robert Levinson**[1]
[1]Department of Computer Science, University of California Santa Cruz, USA

Abstract - *Currently, stylistic analysis of natural language texts is achieved through a wide variety of techniques containing many different algorithms, feature sets and collection methods. Most machine-learning methods rely on feature extraction to model the text and perform classification. But what are the best features for making style based distinctions? While many researchers have developed particular collections of style features – called style markers – no definitive list exists. In this paper we present an organized collection of such style markers with performance data on a diverse set of texts. We show that for each training document, one or more markers exist that can distinguish it from others, providing a basis for a weighted, combined set of markers that outperform any of the individual ones. We examine and categorize 502 style markers, both individually and as a set, and evaluate their performance on several English language text collections.*

Keywords: Computational Stylistics, Style Processing, Natural Language Processing, Machine Learning, Computational Linguistics, Artificial Intelligence

1 Introduction

Researchers have been performing stylistic analysis on text corpora for millennia [1] beginning (in the Western tradition) with authorship attribution efforts in ancient Greece [2]. Nontraditional authorship attribution with internal evidence (only the text itself) is one of the most common applications for stylistics.

Computational methods have been in use in the past decades and have been instrumental in solving several high profile authorship disputes [3]. While these cases and other success stories have shown the power of computational stylistics, the methods used, particularly the style marker selection, have almost always tended to be in support of a traditional theory held by researchers who utilized them [4]. Lack of uniformity in processes, lack of standardization in methods as well as selections of convenience have been among the problems cited in the field [5][6].

We aim to automate a process of statistically deriving the best combination and parameterization of style markers for a given problem. The first step in that process is production of a superset of style markers with detailed definitions to serve as an extensible marker library. We organize these markers into a style marker taxonomy. The superset of markers can be applied to particular problems and relevant markers would be distinguished based on performance.

To find an initial set of markers, we examined many collections in the literature and adopted style markers that were used by the respective authors that could be formalized.

We demonstrate the utility of the taxonomy by applying its markers to our reference corpus and evaluating the markers on attribution performance. We are able to report the top performing markers across eight different English language authorship attribution problems.

In the following sections, we first describe the reference corpus and reasons for choosing it (section 2). In section 3, we state our assumptions with this specific study. In section 4, we describe the taxonomy hierarchy in depth and list current taxonomy marker entries. In Section 5, we run a number of experiments designed to find one or more document discriminators and evaluate the markers in both individual and combined fashion. The results are discussed in section 6. Conclusions are made in section 7.

2 Reference corpus

The Adhoc Authorship Attribution Contest (AAAC) was held in 2004 with many researchers participating [7][8]. The AAAC corpus is particularly well suited for our marker evaluation task for several reasons. The most important reason is that the corpus has purposefully provided significant level of diversity in its many problems.

Documents of different problems differ with each other in genres and text types, as well as document sizes and training/test size ratios, but remain highly uniform within each problem. The corpus is created and prepared by contest organizers and is available to anyone in its original form, allowing for ease of repeatability. The formatting is machine friendly and in fact, has already been used as an example corpus bundled with the software package JGAAP [9]. The texts are used exactly as distributed in plain text files without the need for any further preparation.

The AAAC corpus is divided into multiple problems. Each problem consists of a set of unlabeled test documents and a set of labeled documents, associated with an author. Usually multiple labeled documents exist per author that can serve collectively as a training corpus, allowing for cross-validation. However, problem H only has one training and one test document for each author.

The types of writing in the problems themselves are diverse. They include student short essays in American English (problems A and B), novels (problem C and G), plays (problems D and E), letters (problem F) and speech transcripts (problem H). Problems I through M are in French, Serbian-Slavonic, Latin and Dutch respectively and are not used in our study mainly because many of the markers used are English specific.

Participants in the AAAC utilized many algorithms each depending on a relatively small set of features extracted from the contest texts [7][8][9]. Each algorithm/feature set/parameter set can be thought of as a "recipe" for authorship attribution. The composition of the recipes, as well as the procedure to apply them however was entirely the work of individual participants based on their own hypotheses.

For our experiment to evaluate markers, we used eight English language problems from the AAAC corpus: problems A, B, C, D, E, F, G and H [7].

3 Assumptions

The fundamental assumption in any feature extraction is that the extracted data is an approximate and representative model of the underlying text, in other words, "style" can be modeled by various markers and associated statistics [8][10].

In order to accomplish the necessary document comparisons, we must ensure that markers remain as universal as possible. Thus, we avoid introducing markers that may not be applicable to some texts, derived from situation-specific corpus comparisons (for example, common word frequencies) or may be direct reflections of the size of a corpus [11].

In general, we can enumerate the following assumptions about the corpora we are considering for stylistics work:

1. A corpus is a collection of documents or just one document.
2. Each document is divided into one or more paragraphs.
3. Each paragraph is divided into sentences (not necessarily a well-formed linguistic definition of sentence).
4. Each sentence consists of words.
5. Each word consists of characters.

Our standard unit of comparison is the corpus. Since a corpus can be a single document for our purposes, we can compare one chapter, one page or even one paragraph (corpus of a single document, single paragraph) against books and collected works, provided our markers are size-independent.

Precise tokenization routines are necessary in order to further specify a uniform way of extracting each of the above units [6]. We hope to standardize these extraction methods as well as the markers themselves. Some of them (for example word tokenizers) can be considered parameters for marker extraction routines as we will describe below. Otherwise, extraction routines will have to be available for examination to make repeatability possible.

4 Taxonomy of style markers

We have developed a taxonomic hierarchy based on previous observations of markers [4][5][8][9][11] [12][13][14]. This hierarchy consists of the following hierarchical elements: *Categories*, *Families*, *Markers*, *Parameters* and *Statistics*. An example illustration for category "Lenghts" is given in Fig. 1 below.

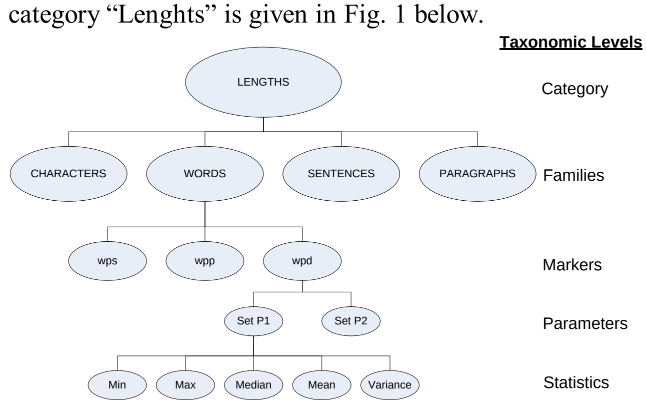

Fig. 1. Partial taxonomic hierarchy for category "Lengths"

4.1 Categories

These are high level themes for the marker collections and subsume many of the marker concepts we encountered in the literature. Examples: lengths, words, n-grams, readability and complexity.

4.2 Families

Families are middle level divisions describing the type of marker in the collection. Examples: characters, words.

4.3 Markers

Markers are base level stylistic events whose presence we are measuring. Example: characters per word, paragraphs per document.

4.4 Parameters

Parameters could be thought of as further sub-division of markers, i.e. variation of markers that may be used simultaneously. In most techniques the parameters correspond to canonizers or pre-processing phase routines which operate on the entire corpus, resulting in the same "parameters" for every marker, as the corpus is modified prior to extraction of any features. This is the way JGAAP[9] handles them and it functions efficiently since only one event set (feature) is being considered at a time.

To accommodate ensemble methods, we must allow for the possibility of using multiple markers with different pre-processing parameters each. Hence, we support separate individual parameterization of each marker. Examples of *parameters* are: unify capitalization, unify numerics (replace with token), exclude common words.

4.5 Statistics

Statistics are metrics used for summarizing marker statistics for the purposes of classification and machine learning. While categories, families, markers and parameters mainly describe a feature event, *statistics* specify how to extract numerical data from the said events. *Statistics* control the footprint of each marker (called marker-instance) in the final feature matrix. For example a *statistics* function for the marker "characters per word" (list of all word lengths in a corpus) could return any of the following:

- A large array of word lengths over the entire corpus (perhaps hundreds of marker instances)
- A single value, the average of all word lengths (1 marker instance)
- A 5-tuple of floats denoting minimum, maximum, mean, median and variance for the array (5 marker instances)

4.6 Types of categories

Categories are the highest level distinctions of the taxonomy. Table I outlines the implemented categories and the families associated with each. Neither the categories nor any of their sub-structure is meant to be a permanent statement on marker style markers or their organization.

Table I. Marker categories

Category	Description	Families
Lengths	Counts and sizes of text features such as sentences and paragraphs.	Characters, words, sentences, paragraphs, syllables, numerics, vowels, punctuation, symbols.
Words	Counts of many categories of words, word frequencies and word lists.	Most frequent words, least frequent words, parts of speech, misspellings, word lists (dictionaries).
N-grams	Counts sequence-based counts of characters, words and parts-of-speech.	Characters, words, parts of speech
Read-ability	Measures presence of well-known readability, sentence complexity and phrases recognized as cliché or poor communication.	Readability (Coleman-Liau, Kinkaid, Flesch Reading, Fog index, ARI, Lix) and Complexity (syntactic depth, parser phrase counts), GNU Diction rules (cliché, rewrite, run-on sentence, superfluous language).
Semantics	Measures having to do with meaning and word senses.	Word Net synset size, synset depth and distance.

4.7 Marker extraction

Due to space restrictions, we are not able to list and discuss every *marker*, *parameter* and *statistic*. We provide a full listing of every marker mentioned in this paper in Table IV. In this section, we choose one category, *lengths*, to demonstrate marker diversity, extraction and values.

The category name "lengths" is meant as in the length of an array. Markers in this category are about counting the occurrences of some distinct feature in terms of another [13], like "words in a sentence" or "vowels in a paragraph." Each of the length markers is an array of integers denoting counts of the phenomenon for which they are named.

For example, let us assume we have a corpus with 2 documents. The first document has 5 paragraphs and the second has 3 paragraphs. The number of paragraphs is determined in accordance with parameter specifications of the marker (4^{th} level of taxonomy hierarchy) which are inputs into the paragraph tokenization routine. For instance, we may want to count a title as a paragraph, or we may not.

The marker output “paragraphs per document” (ppd) is given by an array of two integers representing document length in terms of paragraphs:

ppd (corpus, parameters) = (5,3)

Let us now further assume that the first document’s 5 paragraphs consist of 2,5,9,2 and 3 sentences respectively, and the second document’s 3 paragraphs consists of 12,3, and 4 sentences respectively, consistent with sentence level parameters. Thus a “sentences per paragraph” (spp) marker is given by an array of 8 integers representing paragraph lengths in terms of sentences:

spp(corpus, parameters) = (2,5,9,2,3,12,3,4)

To compare the markers of two different corpora, which likely have different length arrays, we use statistics, the lowest level of the hierarchy. For example, a *statistics* routine that returns maximum and mean of an array can be used to convert the above *spp* marker array into a tuple of floats.

stats(spp(corpus, parameters)) = (12.0,4.0)

These two values are the final contributions of the *spp* marker to the overall feature matrix of the corpus.

4.8 Other categories

The *words* category, unlike *lengths*, is not concerned with the size or number of all words, but rather the frequency of particular selection words within a corpus or document.

Markers in the “most frequent” family of category *words* are arrays of X decreasing floating point values. One strategy is for X to be chosen as a large number resulting in a large array of floats per marker, and then to summarize that array using statistics resulting in only a few floats per marker. Another strategy is to choose a small number for X, such as 5 or 10 and do not use statistical summaries.

The values in the “least frequent” family are hapax legomena, dis legomena, etc. They are straight integer counts rather than frequencies or ratios. 5-legomena, for example is the number of words in the corpus that occur exactly 5 times each. Part of speech ratios are single floating point values each, thus there is no need for statistics in this family.

The *dictionary* family consists of frequencies of words in a given list L. The size of list L is unspecified and therefore could contain thousands of entries. Statistics may or may not be used on these. “Top100F” family on the other hand is actual frequencies of words, and thus is made up of 100 floating point values.

A popular technique used for stylistics is n-gram analysis [15]. N-grams consist of N successive occurrences of a token type (could be a character, word, part-of-speech or even sentence types). These markers have been successfully used in many authorship attribution problems.

The most important difference between n-grams and other markers is that n-grams are dependent on a meta parameter, N which must be specified to instantiate n-gram routines. Character trigram statistics, for example, could be completely different than bigram ones. To make the taxonomy simpler, we represent n-grams with their generic “n” parameter.

Another important set of indicators are readability indices [16]. Several well-known formulas exist and are reportedly in use to assist essay graders for standardized tests. The particular indices chosen (ARI, Coleman-Liau, Flesch-Kinkaid, Flesch reading ease, Gunning-Fog and SMOG indices) were the ones described in the GNU Style manual, and we refer the reader there for full explanation of each formula [16]. All outputs are single integer or single floating point values.

The *complexity* family markers have array-length outputs similar to those markers in category “Lengths”, and thus could be summarized with statistics. “Syntactic depth” is the deepest level of a parse tree achieved by parsing the single sentence in question. Phrase count is a simple count of independent phrases as tagged by the parser, regardless of level or nested status.

The markers in the *semantics* category are mainly derived from Word Net synsets [17] and associated numerical properties with the ontology.

5 Evaluation of markers

One of the principle uses for having a comprehensive taxonomy is that it allows us to experimentally evaluate the markers against particular corpora.

We extract 502 marker instances (marker-parameter-statistics) from all training document collections. This results in a single set of markers for all same-labeled documents within a problem, as well as individual sets for each document. These 502 cover most of the taxonomic markers we presented above with the notable exception of sentence-level complexity statistics (phrase count and syntactic depth).

We used all of the “N-gram” markers with N <=5 and number of top n-grams = 300 for characters, words and parts of speech, meaning the statistics are derived only from the top 300 n-gram events. There was no unification of white space, elimination of numbers and symbols or capitalization for n-gram based markers.

For all word counts, we used alphabetic words only, although for sentence and paragraph lengths, we used all space separated tokens. For dictionary based operations (stop words, top 1000 English words and top 100 non-stop words) we used a uniform capitalization.

For all array based markers, we used standard *statistics* which reduced all arrays to a 5-tuple of minimum, maximum, mean, median and variance. For single value markers (such as xLegomena, readability indices) as well as for top 100 non stop word frequencies, we did not use any derivative statistics, only the raw frequencies.

We used Python with NLTK tools [18] to extract most of the markers. Part of speech tagging was done by the Stanford Tagger [19], and the LinkGrammar parser [20] was used for some phrase level semantic statistics.

Specifically, we ran two experiments: For the first experiment, we considered each of the 502 marker instances un-weighted, in isolation. We used nearest neighbor with a simple Euclidean distance formula and performed attribution on the problem set based on finding the labeled corpus with the minimum distance.

For the second experiment, we combined all 502 markers using a weight vector. We trained the vector with a fixed number of maximum cycles (200800) for each problem. The resulting weight vectors were used to classify the documents.

6 Experimental results

The eight AAAC problems (A-H) have 187 labeled documents total representing the work of over 50 authors. We classify each document with the given choices in the corresponding problem and thus we derive an "absolute" performance value for each marker.

A problem that has a large number of documents could really dominate the evaluation of the markers if marker performance is measured by absolute number of correct document attributions. Thus, we also calculate the results in terms of problem-relative and marker-relative correct attributions.

The problem-relative results (called adjusted relative performance) are displayed in Fig. 2, along with the absolute performance for each marker. The Problem-relative performance numbers are calculated by considering the percentage of correct attributions per problem, regardless of the how many documents it has.

For example if a problem had only 2 training documents, and only one of them was correctly attributed by marker X, then it would have 50% problem-relative correct attribution for X. If the problem had 10 training documents and still only one was correctly attributed, then X would have 10% attribution score. If one marker was able to correctly attribute every document in the problem it would achieve 100% problem-relative score. Thus, the relative value eliminates the size of the problem as a factor.

Every attribution problem does not have the same degree of difficulty. We are also interested in marker performance relative to other markers per problem. To accomplish this, we rank all the markers from the best performing to the worst for each problem attribution, and calculate difference from the mean in units of standard deviations (z-scores) for each marker, for each problem. This allows us to normalize for the problem difficulty when comparing markers to each other. Table II ranks the top 15 markers and their average z-score performance across all eight problems. For reference, the marker's relative and absolute performances are also listed.

Table II. Marker performance relative to other markers

Marker #	**average z-score**	**rel. perf.**	**abs. perf.**
292	1.65467884	66.29%	55.08%
232	1.47054735	63.72%	52.41%
247	1.34549143	63.88%	49.20%
152	1.29048525	61.84%	50.27%
310	1.24059087	62.47%	49.20%
229	1.22346475	60.98%	47.59%
40	1.21858176	62.66%	44.39%
230	1.21157427	62.12%	48.13%
290	1.21157427	62.12%	48.13%
217	1.21012445	61.59%	48.66%
169	1.18920219	63.19%	44.92%
14	1.17039029	59.15%	44.92%
172	1.15087611	62.98%	44.39%
92	1.13307073	60.92%	48.13%
8	1.12271391	59.68%	45.45%
44	1.11259672	60.44%	45.99%
257	1.11227455	60.73%	47.06%
289	1.11002161	59.69%	48.66%
245	1.10662925	61.31%	44.39%
234	1.08100501	59.34%	47.59%

For the second experiment, we train the weight vector using a modified greedy first-choice Hill Climb algorithm. We also use the k Nearest Neighbor to perform the actual attribution. The results are summarized in Table III. Table III column 1 (left most) designates the problem from the AAAC corpus [4]. Each problem has a number of training documents and a number of possible authors (classes).

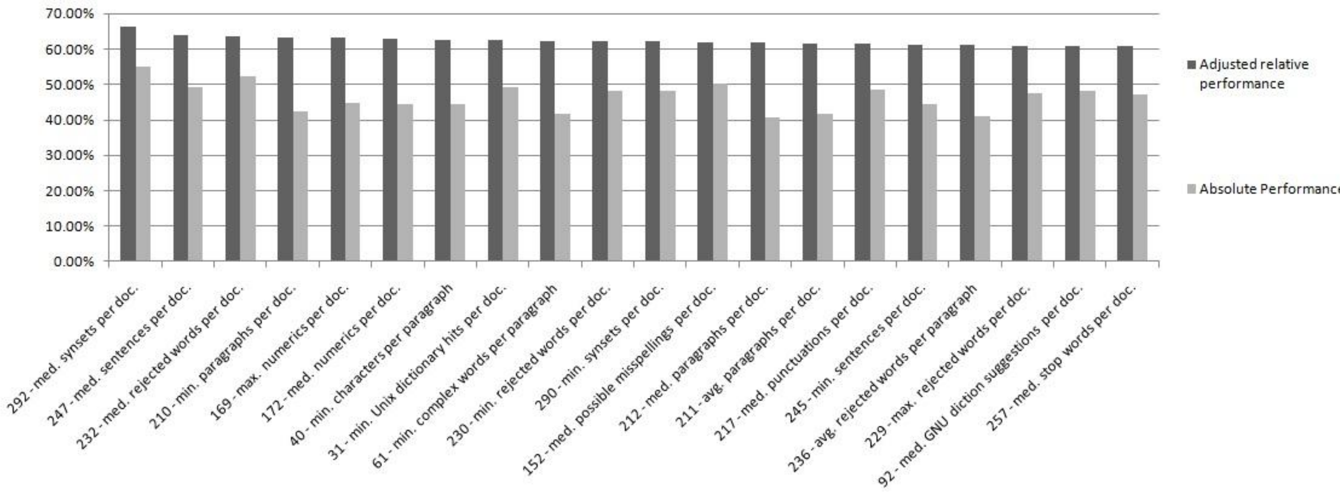

Figure 2. Top performing marker instances by adjusted relative performance.

Columns 2 and 3 of Table IV indicate the best performing individual markers on the particular AAAC problem. For example, the highest number of correct associations that could be performed with a single marker on problem A is 21 documents out of 39 possible or 53.84%. Marker *57* can perform 21 correct attributions and is thus the best independent marker (column 3 from left).

Columns 4 and 5 do the same for the combined approach. Column 4 lists the best attribution results and column 5 lists the top weighted markers in order. We have listed all markers whose weights have grown as a result of training (starting from uniform weight). Therefore, all additional attributions beyond what the top individual marker can do, is due to combining the particular markers in column 5. For problems D, E, G and H, the weighted approach does not add to performance because there exists at least one marker for each of these problems that can by itself categorize 100% of the documents correctly.

For problem H where only three documents needed to be classified, 351 different markers can achieve 100% classification. Another 58 markers can correctly attribute 2 out of the three documents.

Please see the glossary in Table IV for description of the marker instances mentioned in this paper. A full description of all markers used in this study can be found in [21].

7 Conclusions and future work

We have developed an extensible taxonomy of 502 style marker instances in a hierarchical classification. Although many more markers could be added to this collection, either through additional novel markers and extraction routines or production of more variations of the existing markers.

We have conducted experiments to determine the effectiveness of these marker instances.

As can be seen in Fig. 2, and Table II, marker *#292* (median of Word Net synset counts per document) is the best performing marker according to all three individual marker performance measurements (absolute, problem-relative and marker-relative).

Table III. Individual versus combined (weighted) marker performance results

AAAC Problem (docs / authors)	**Best ind. marker results (%)**	**Best ind. mark-ers**	**Best comb. results (%)**	**Top weighted marker instances**
A (39/13)	53.84	*57*	94.87	*92,72,396,57,59, 436,226,298,410, 479,484,244,333, 262,369,288,8*
B (38/13)	55.27	*57*	94.74	*176,39,494,2,13, 228,11,47,19,412 ,33,12,411,417, 14,9,8,459,0,7*
C (17/5)	76.47	*247*	100.00	*79,8,0*
D (12/3)	100.00	*210,211 ,212*	-	-
E (12/3)	100.00	*210,211 ,212*	-	-
F (60/9)	61.67	*14,19*	88.33	*14,19,301*
G (6/2)	100.00	*40,456, 480*	-	-
H (3/3)	100.00	*many**	-	-

**351 markers could distinguish between all three documents in problem H.*

Interestingly, marker *#292* is not highly weighted in any of the combined marker results. Marker *#57* (median of complex words per document) is the highest weighted marker for problems A and B in the combined exercise.

The attribution performance for problems A and B reaches 21 out of 39 and 38 respectively[7]. The training sets for A and B consist of short essays on various topics written by students. For problems with smaller number of documents such as D,E,G and H, there is usually at least one marker that can do perfect classification on the training docs. Thus, the combined method cannot exceed it.

In this paper, we verified two important hypotheses: First, given a large set of individual markers, a few can be found that perform well on a particular set of corpora. Experiment 1 and Table V and VI show the best performing markers in our corpus set. Second, combining markers linearly using a weight vector often performs better than any individual marker on the same data set. However, for smaller sets, combining multiple markers may not be necessary.

We agree with [6] that many problems in authorship attribution studies remain, including lack of standardization and specificity of markers and methods in the literature. Given the severe space limitations in scientific publications, we believe an important step to address some of these problems is to codify markers and extraction routines, complete with full parameterization. This should at least allow the chance to make meaningful and precise comparisons via referencing.

Ongoing work includes further extending the taxonomy and making marker references more descriptive and uniform. We also are applying classification methods based on our markers to different corpora and are maintaining an extensive database of our findings.

Table IIII. Glossary of markers discussed in this paper

Code	**Marker instance**
0	Automated Readability Index (ARI)
2	Flesch-Kinkaid Grade index (FKG)
7	ratio of adjectives to all words
8	ratio of adverbs to all words
9	ratio of other words (other than noun, verb, adjective and adverb) to all words
11	ratio of plural nouns to all nouns
12	ratio of past tense verbs to all verbs
13	SMOG index
14	frequency of most common character
19	max. character bigram frequency
31	avg. frequency of top 300 character 4-grams
33	variance of top 300 character 4-grams
39	max. characters per paragraph
40	min. characters per paragraph
44	max. characters per sentence
47	med. characters per sentence
57	med. complex words per document
59	max. complex words per paragraph
61	avg. complex words per paragraph
72	med. number of unique GNU Diction rules applicable per document
79	max. GNU Diction new sentence suggestions
92	med. total number of diction suggestions per document
152	med. number of possible misspellings per document
169	max. numerics per document
172	med. numerics per document
176	avg. numerics per paragraph
210	min. paragraphs per document
211	avg. paragraphs per document
212	med. paragraphs per document
217	med. punctuations per document
226	avg. punctuations per sentence
228	variance of punctuations per sentence
229	max. rejected words per document
230	min. rejected words per document
232	med. rejected words per document
234	max. rejected words per paragraph
236	avg. rejected words per paragraph
244	max. sentences per document
245	min. sentences per document
257	med. stop words per document
262	med. stop words per paragraph
288	variance of syllables per word
290	min. synsets per document
292	med. synsets per document
298	variance of synsets per paragraph
301	avg. synsets per sentence
310	min. Unix dictionary hits
333	variance of vowels per sentence
369	max. words per paragraph
396	frequency of the word "go"
410	frequency of the word "come"
411	frequency of the word "made"
412	frequency of the word "may"
417	frequency of the word "little"
436	frequency of the word "right"
456	frequency of the word "must"
459	frequency of the word "turn"
479	frequency of the word "animal"
480	frequency of the word "house"
484	max. freq. of the most frequent English word
494	ratio of total hapax legomena to all words

8 References

[1] J. Middleton Murray "The Problem of Style," Oxford University Press, London, 1922, p. 77.

[2] Harold Love, "Attributing Authorship: An Introduction", Cambridge University Press, 2002, pp 14-16

[3] David I. Holmes, "The Evolution of Stylometry in Humanities Scholarship", Literary and Linguistic Computing Volume 13 Issue 3 September 1998.

[4] S. Fish, "What is stylistics and why are they saying such terrible things about it?" in Essays in Modern Stylistics, edited by DC Freeman, Routledge, 1981, pp 53-66.

[5] J. Rudman, "The State of Authorship Attribution Studies: Some Problems and Solutions," Computers and Humanities 31: 351-365, 1998.

[6] J. Rudman "Cherry Picking in Nontraditional Authorship Attribution Studies," CHANCE, vol. 16, No. 2, pp 26-32, 2003.

[7] P. Juola, "Ad-hoc Authorship Attribution Contest" ACH/ALLC 2004, Gothenberg, Sweden, 2004.

[8] P. Juola, Authorship Attribution, NOW Publishers, 200

[9] P. Juola, et. al. JGAAP, a Java-based,modular, program for textual analysis, text categorization, and authorship attribution. http://www.mathcs.duq.edu/~fa05ryan/wiki/index.php/Main_Page , 2009.

[10] J. Walpole "Style as Option," College Composition and Communication, vol. 31, No. 2, Recent Work in Rhetoric: Discourse Theory, Invention, Arrangement, Style, Audience, (May, 1980), pp. 205-212, 1980.

[11] M. Eder, "Does Size Matter? Authorship Attribution, Small Samples, Big Problem," Digital Humanities, London, UK, 2010.

[12] C . DiMarco, "Stylistic Choice in Machine Translation," AMAT, 1994.

[13] F. Khosmod and R. Levinson, "Toward Unification of Attribution Processes and Techniques", ICMLC, Dalian, China, August 2006.

[14] F. Khosmood and R. Levinson, "Toward automated stylistic transformation of natural language text," Digital Humanities, Washington, D.C. , 2009.

[15] V. Kesselj et. Al "N gram-based Author Profiles for Authorship Attribution." Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING'03, Dalhousie University, Halifax, Nova Scotia, Canada, August. 2003.

[16] M. Haardt, GNU *diction(1)* PDF manual, accompanying *diction* version 1.11. (http://www.gnu.org/software/diction/diction.html)

[17] WordNet at Princeton University Cognitive Science Library, http://wordnet.princeton.edu , 2009.

[18] NLTK, The Natural Language Tool Kit, project home page: http://www.nltk.org/, 2009.

[19] Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of HLT-NAACL 2003*, pp. 252-259.

[20] D. Grinberg, J. Lafferty and D. Sleator, A robust parsing algorithm for link grammars. Carnegie Mellon University Computer Science technical report CMU-CS-95-125, and Proceedings of the Fourth International Workshop on Parsing Technologies, Prague, September, 1995.

[21] 502 Markers used in this paper are listed on the authors' website: http://style.soe.ucsc.edu/node/101 , 2011.