MAD Style: Multivalent Authorship Detection (MAD) Topic Models for Stylometric Analysis

David Dohan, Charles Marsh, Shubhro Saha, Max Simchowitz

May 13, 2014

Abstract

We present the *MAD Topic Model*, which uses syntactic and stylometric n-gram features (e.g., part-of-speech tags, meter) to extract the distinctive qualities of author style. MAD fits separate topic models to each of these n-gram vocabularies and combines the models through a multi-class logistic regression classifier. MAD breaks stylistic features into topics over vocabularies, creating a compact representation of stylistic tendency among different authors. We test MAD on several real world corpora using a variety of n-gram features, including part-of-speech, syllable stress, and sequences of word lengths. All relevant code, including the topic model, can be found at github.com/dmrd/mad_topic_model.

1 Introduction

In the *authorship detection* problem, one is first given a set of documents labeled (by author) on which to train, and then asked to identify authors of anonymized text snippets [22]. Accurate author classification is typically used for such tasks as plagiarism identification [20, 22]. However, the approaches used typically fail to produce interpretable descriptions of author styles. By approaching the authorship detection problem from a topic model perspective, we provide a multivalent sLDA algorithm that both classifies anonymized text and creates a compact representation of author style.

2 Literature Review

Existing methods for author classification focus on careful feature engineering, incorporating synonym use, part-of-speech tags, and sentence structure [21]. Classification then amounts to feeding these features through generic classifiers, most often SVMs, Logistic Regression, and Random Forests. More sophisticated methods employ feature transformations (see the Writeprints method proposed by Abbasi and Chen [1]). While these techniques are very powerful for classification, they fail to produce interpretable descriptions of each author's probabilistic tendencies.

In this paper, we use probabilistic modeling to extract better insight into the factors that differentiate literary style. Our starting point is the Latent Dirichlet Allocation topic model [5], which describes topics as categorial distributions over words, and describes documents as proportions of topics. Topic models have become extremely popular in the field of "Digital Humanities", and scholars in the humanities are beginning to use LDA to understand massive document corpora [3]. Here, we adapt topic modeling to extract stylistic insight. For stylometric analysis, we regard stylistic tendencies as topics in the sense that they correspond to categorical distributions over stylistic features.

sLDA [4] extends LDA to the supervised setting, where per-document topic proportions are linked to a generalized linear model response [14]. The original sLDA focused on Poisson and Gaussian responses, and [25] presents an approximate inference algorithm for a softmax response with applications to image annotation. Rosen-Zvi et al. [19] incorporate authorship into the LDA framework by assigning topic distributions on a per-author, rather than per topic basis. This method neglects the possibility that author writing style may vary over documents, and is best suited to cases when documents share many authors. Thus, this paper will follow Wang et al. [25] and extend the softmax response LDA to incorporate separate classes of stylometric features (e.g., meter, etymology, and part-of-speech).

3 Data

To collect data for training and testing, we wrote scrapers for Project Gutenberg, Quora, and the Nassau Weekly. We selected these three data sources for their diversity in topic, language, and length. For example, Project Gutenberg features lengthy narrative texts while Quora features shorter comments in colloquial language. With Nassau Weekly we see a mix: modern prose in a mix of narrative and editorial styles. Because of this diversity, these corpora provide ample training and testing data for our models.

We implemented our scrapers in Python. The Project Gutenberg dataset features excerpts from fiction books by five authors. The Quora dataset features about 1600 comments from roughly 100 popular Quora users. The users were selected based on online reports for "most followed" users on the network. Because Quora is a question-answer web site, this content is mostly informative in nature. Depending on the thoroughness of a user's answer, the length can vary from a single word to several paragraphs.

The Nassau Weekly is a student-run humor/culture newspaper. Our dataset features over 550 articles from about 200 authors. The content in this dataset is largely narrative or editorial in nature, and tend to be several paragraphs in length. Interestingly, authors for the publication tend to write in vastly different tones across articles because of the unique, cultish nature of the newspaper. The challenge for our authorship models is to detect consistent features in this dataset across articles by the same author.

4 Feature Extraction

We incorporated six different stylometric features, each of which was composed into n-grams of varying sizes before being fed into the model:

1. Part-of-Speech (POS) tags (e.g., 'Noun' for the word "apple"). The Penn-Treebank tag set was used, and tagging was performed using a Maximum Entropy approach [18].

2. Etymological tags (e.g., 'Old English' for the word "great"), a relatively novel feature that captures the 'formality' of the writing style. Etymological information was scraped from *Webster's* Dictionary [17]. As etymology is inherently root-based, words absent

from the dataset were first stemmatized using the method of Porter [16] and lemmatized using the WordNet method of Fellbaum [7]. If either of these roots were present in the dictionary, their corresponding etymological tag was returned. Else, the entry with minimum Levenshtein distance [12] was used instead.

3. Syllables-per-word (i.e., '3' for "continue"). Syllables were extracted from the CMU Pronouncing Dictionary [11]. As with etymology, words absent from the dictionary were looked up by minimizing Levenshtein distance with the present keys.
4. Syllable counts, i.e., the total number of syllables between pieces of punctuation.
5. Word counts, i.e., the total number of words between pieces of punctuation.

On top of these primitives, we also developed an algorithm to extract meter, which is outlined in Section A of the Appendix. In total, this composed six stylometric features. For each document, we extracted these features and generated the relevant 2-, 3-, and 4-grams (apart from meter, for which only 8-grams were produced, as described in the Appendix).

For illustrative purposes, Table 1 presents several common stylistic n-grams and corresponding examples drawn from a Quora post by Yishan Wong, the CEO of Reddit. Notice that our feature extraction heuristics correctly identify "Bitcoin" as a two-syllable word (despite the fact that it is not in the CMU Pronouncing Dictionary). Similarly, the etymology of "stated" (Latin) was deduced by looking up its root, "state"; as was the etymology of "aims" (Old French) by looking up its root, "aim".

Type	n-gram	Matching Text
Etymology	(AS, OE, L, OF)	"... the key stated aims..."
Syllable	(1, 1, 2, 1)	"...fact that Bitcoin is..."
Part-of-Speech	(DT, JJ, NN, IN)	"...the libertarian culture of..."
Word Counts	(7, –, 2, –)	"It has all the features of Bitcoin–technologically speaking– ..."

Table 1: Matching stylistic 4-grams from a Quora post.

5 Methods

To explore our data, we feed the extracted features as bags of n-grams to a novel LDA extension, the Multivalent Authorship Detection (MAD) Topic Model. The MAD Topic Model combines the sLDA algorithm presented in [25] and [4], with the Author Topic Model presented in [19], extending both of these models to account for multiple word types. For each word type t, MAD posits its own LDA topic model. Unlike conventional LDA, in which each document shares a common Dirichlet prior, MAD gives each author its own Dirichlet prior which can be optimized with coordinate descent. This differs from the Author Topic Model, which treats each author's oeuvre as one contiguous document.

Like sLDA, MAD has a multi-class regression parameter η; classes are drawn from softmax$(\eta^T \bar{z})$, where $\bar{z}$ are the average topic assignments for each work. The complete generative process is specified in Algorithm 1, and the graphical model is show in Figure 3. The key innovation in this model is that it is doubly supervised: first, each author has its own topic proportions, which enforce shared topics between its documents. And second, upon conditioning on the multi-class logistic regression η, the topic assignments z which contribute to correct classification are given a higher likelihood. Thus, one would expect that the *more salient* features are selected during inference. It is crucial to note that, during training, authorship is thereby treated as both a known label and a random variable. In the test stage, however, we marginalize over authors, as described in Section B of the Appendix.

The model is fit using variational inference [24] which we address in detail in the Appendix (stochastic variational inference is supported as well [9]). After fitting the model, we extract per author distributions over topics and a total corpus distribution over topics. To classify documents, we extract topic assignments by applying LDA with the model parameters fit during training, and feed the topic assignments to the logistic regression classifier. Again, discussion is left to Section B of the Appendix.

6 Evaluation

Our implementation of MAD extended Wang's sLDA implementation by 1000 lines of C++, so preliminary testing focused on establishing the algorithm's correctness. First, we ensured that our model likelihood increases during variational inference. Next, we simulated documents according to our generative process, and verified that we could classify such documents with 100% accuracy, provided that each document had a fairly distinct distribution over topics. MAD's performance on this artificial data can be seen in Figure 5.

For each of the three corpora, we tested the classification accuracy sLDA against an 80/20 train/test split. We restricted our tests to authors above a certain threshold of documents, with this threshold varying by corpus. As a benchmark, we tested sLDA against Random Forest, Logistic Regression, and SVMs, applied to the n-grams representations of each document. In each test, sLDA significantly outperformed random guessing, but underperformed compared to other benchmark methods. Figure 2 suggests that the style topics contained very even distributions across n-gram terms. Hence,the average per-word topic assignments may not have been a meaningful input to the softmax classifier (see Appendix C). Full results can be seen in Figure 4 on Page 8.

7 Exploration

While MAD did not perform particularly well for classification, it's true usefulness is its exploratory capabilities: by creating topic models over vocabularies of n-gram stylistic features, MAD helps uncover the hidden stylometric structure of authors' writing styles.

To visualize the generated topic models, we adapted the Termite tool [6], which allowed us to see the marginal distributions of the various n-grams (for a given word type, such as part-of-speech tag) as well as the distributions of the vocabularies across the topics.

As an example of MAD's exploratory power, consider the 'runs of syllables' topic model for the Gutenberg dataset, which contains long-form documents from famous authors, such as Jane Austen's *Pride and Prejudice* and Mark Twain's *Huckleberry Finn*. The Termite representation of the topic model can be seen in Figure 1 on Page 6.

MAD assigns each author its own distribution over topics. In this case, the most heavily-used topic for Jane Austen was Topic 0; distinctive features for this topic (i.e., those that do not appear in any other topic) are of the form (1, 3, 1, 1) and (3, 1, 1, 1)–that is, runs of one-syllable words broken up by a three-syllable word. Meanwhile, the most heavily-used topic for Mark Twain was Topic 27, with distinctive features of the form (1, 1, 1, ‘?’) and (1, 1, ‘?’, 2)–that is, questions composed of mono-syllabic phrases.

These patterns can be found quite easily in the writings of the two authors. For example, consider the famous first sentence of *Pride and Prejudice*: “It is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife.” This sentence contains the (1, 3, 1, 1) and (3, 1, 1, 1) patterns with the phrase “in pos-ses-sion of a wife”. In another famous sentence, “Pride relates more to our opinion of ourselves, vanity to what we would have others think of us.”, we again see the (3, 1, 1, 1) pattern with “van-it-y to what we”.

Meanwhile, the mono-syllabic question-based structure of Twain is prevalent in the Southern drawls of his main characters. For example, in Line 153, we have: “Who is you? Whar is you?”, which contains the (1, 1, 1, ‘?’) pattern twice in immediate succession. Similarly, on Line 256, we have: “Do ´bout him?”. Notice that the character’s accent transforms the two-syllable “about” to the mono-syllabic “’bout”, again demonstrating the topic model’s ability to capture the structure of Twain’s dialogue. These results are presented in Table 2.

Further analysis could be performed on and across the topic models for part-of-speech tags, etymology, etc. However, even with this small example, the usefulness of MAD for exploratory and explanatory purposes is immediately evident.

Author	Primary Topic	Distinctive n-gram	Matching Text
Mark Twain	Topic 0	(1, 1, 1, ‘?’)	“Who is you?”, “Do ’bout him?”
Jane Austen	Topic 27	(3, 1, 1, 1)	“pos-ses-sion of a wife”, “van-it-y to what we”

Table 2: Distinctive n-grams for ‘runs of syllables’.

8 Conclusions

It turns out that the MAD topic is far more useful for exploratory purposes than as a classifier. Initially, we hypothesized that reducing dimensionality from word counts to topics was responsible for this inferior performance. However, Logistic Regression performs roughly as well even after applying PCA (setting the number of principal components equal to the total number of topics). Given the strong performance of our model on artificially data, we should reconsider the extent to which topic models (in general) and the Dirichlet distribution (in particular) capture the statistical properties of natural language; this skepticism is reinforced by the relative similarity across topics, as seen in Figures 1 and 2, which hamper MAD’s performance as a classifier, but actually supports its ability to capture the common structure of writing for exploratory purposes. In Section C of the Appendix, we suggest future improvements to our model based on the Pitman-Yor Process [23]. Nevertheless, we hope that the MAD Topic Model can still serve the digital humanities community as a tool for exploring corpora of large documents.

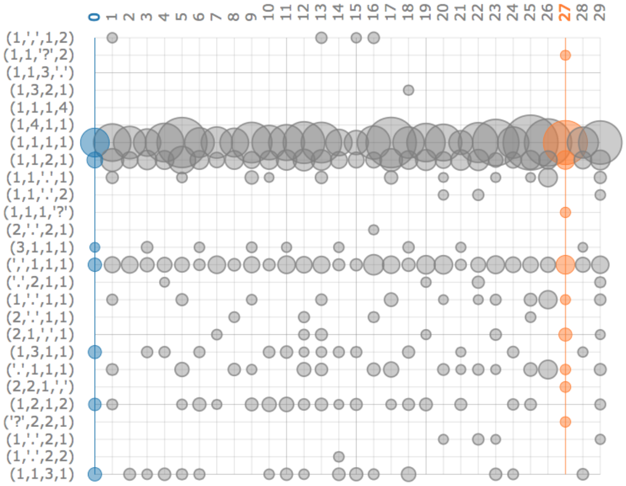

Figure 1: Termite visualization for 'runs of syllables' 4-grams on the Gutenberg corpus. The x-axis indicates topic indices, while the y-axis indicate n-grams. Size of circle corresponds to n-gram importance within a topic. Austen's primary topic (0) is highlighted in blue, with Twain's (27) in orange.

.

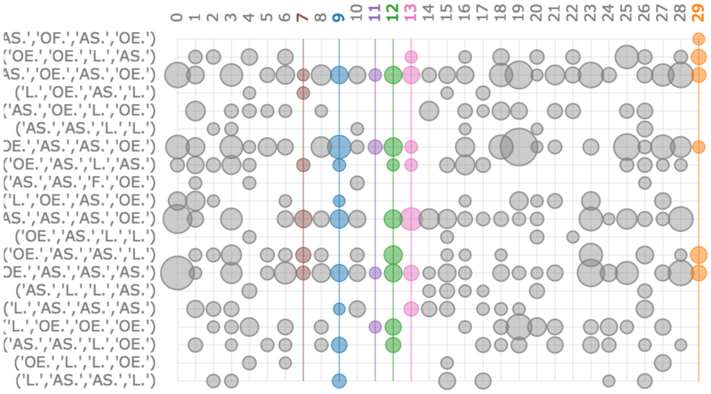

Figure 2: Termite visualization for etymology 4-grams on the Gutenberg corpus. The top 10 most common 4-grams were removed as they were very common among all authors. Topics 7 and 13 account for 40% of Austen's writing, while 9, 11, 12, and 29 account for 40% of Twain's. The two are quite similar, reflecting that the vast majority of English users rely on Old English (OE) and Anglo Saxon (AS) words (some scholars do not distinguish between these etymologies). This similarity also reflects Austen and Twain's colloquial tones (OE and AS words are regarded as more casual). When the authors choose to use "fancier" latinate (L) words, Austen tends to use them consecutively, whereas Twain's more casual diction separates L words with OE and AS. Austen (more-so than Twain) enjoys using triplets of AS words, reflecting her longer phrases.

.

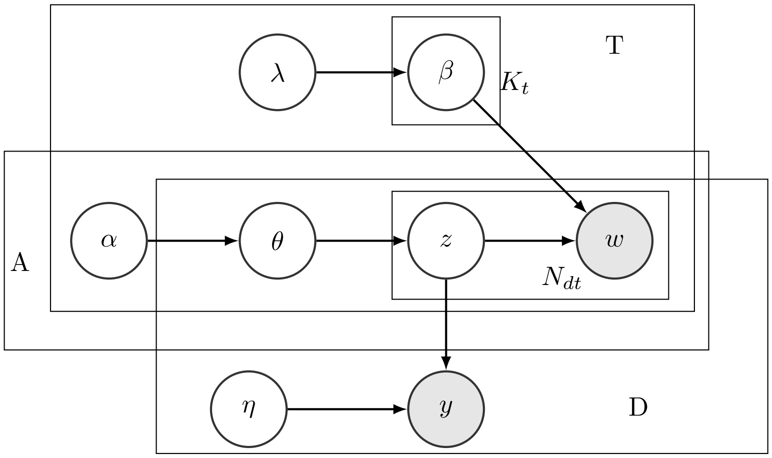

Figure 3: Graphical Model for the MAD Topic Model. α and λ are Dirichlet parameters governing distributions of topics and per topic distributions over words respectively. θ, z, β and w are multinomial distributions for per-document topics, latent topic assignments, per words topics, and observed word. y is a categorical GLM response, with canonical link function and parameter η. A is the number of authors, D the number of documents, T the number of word types, N_{dt} the number of words of type t in document d, and K_t the number of topics over words of type t.

Algorithm 1 The complete generative process for the MAD Topic Model.

```
procedure MAD Generative Process
    $T$ word types, with $K_t$ topics. $D$ documents, labelled into $A =$ classes, with $T$ separate word counts for each of the $T$ word types. Softmax parameter $\eta_t \in \mathbb{R}^{(A-1)\times\sum_{t=1}^{T} K_t}$, Dirichlet priors $\{\alpha_{at}\}$ and $\{\lambda_t\}$
    for Each Word Type $t$ do
        Fix vocabulary dirichlet $\lambda_t$. Draw $K_t$ topics $\beta_{tk} \sim \text{Dirichlet}(\lambda_t)$
    end for
    for Each Author $a$ do
        for Each Word Type $t$ do
            Fix author topic proportions $\alpha_{at}$
        end for
        for For each document $d$ written by author $a$ do
            Draw topic proportions $\theta_{dt}$, topics assignments $z_{dtn}$, words $w_{dtn} \sim \text{LDA}(\alpha_{at}, \beta_t)$.
            Draw document label $\sim (\text{softmax}(\sum_t \overline{z}_{dt}^T \eta_t))$,where $\overline{z}_{dt}$ are average topic assignments
        end for
    end for
end procedure
```

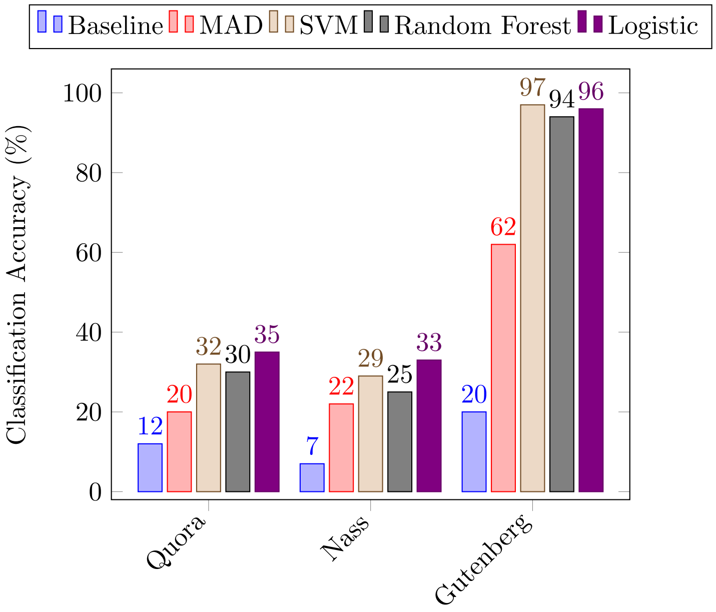

Figure 4: MAD outperforms the baseline, but does comparatively worse than other methods.

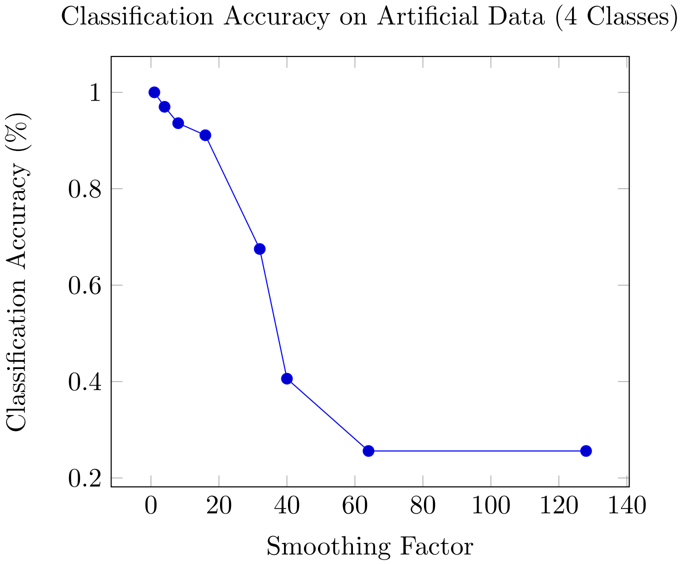

Figure 5: MAD's classification accuracy with a 50/50 test/train split using artificial data generated by Algorithm 1. The per-author topic Dirichlet parameter vectors α have entries of the form $\alpha_i = \beta_i + c$, where $\beta_i \sim \text{Unif}(0, 1)$ and c is a smoothing factor. If $x \sim \text{Dirichlet}(\alpha)$, then $\mathbb{E}[x|\alpha] = \frac{c+\beta_i}{nc+\sum_i \beta_i}$, which tends to $1/n$ as $c \to \infty$ (i.e., increasing values of c make it harder to tell per-author distributions apart).

References

[1] A. Abbasi and H. Chen. Writeprints: A Stylometric Approach to Identity-Level Identification and Similarity Detection in Cyberspace. *ACM Transactions on Information Systems (TOIS)*, 26(2):7, 2008.

[2] G. Andrew and J. Gao. Scalable Training of L1-Regularized Log-Linear Models. In *Proceedings of the 24th international conference on Machine learning*, pages 33–40. ACM, 2007.

[3] D. M. Blei. Topic Modeling and Digital Humanities. *Journal of Digital Humanities*, 2(1):8–11, 2012.

[4] D. M. Blei and J. D. McAuliffe. Supervised Topic Models. In *NIPS*, volume 7, pages 121–128, 2007.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] J. Chuang, C. D. Manning, and J. Heer. Termite: Visualization Techniques for Assessing Textual Topic Models. In *Advanced Visual Interfaces*, 2012.

[7] C. Fellbaum. *WordNet: An Electronic Lexical Database.* MIT Press, Cambridge, MA, USA, 1998.

[8] D. Genzel, J. Uszkoreit, and F. Och. “Poetic” Statistical Machine Translation: Rhyme and Meter. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP ’10, pages 158–166, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[9] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley. Stochastic Variational Inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[10] M. Johnson, T. L. Griffiths, and S. Goldwater. Adaptor Grammars: A Framework for Specifying Compositional Nonparametric Bayesian Models. *Advances in neural information processing systems*, 19:641, 2007.

[11] K. Lenzo. Carnegie Mellon Pronouncing Dictionary, 2007.

[12] V. I. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Cybernetics and Control Theory*, 10(8), 1966.

[13] D. C. Liu and J. Nocedal. On the Limited Memory BFGS Method for Large Scale Optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[14] P. McCullagh. Generalized Linear Models. *European Journal of Operational Research*, 16(3):285–292, 1984.

[15] T. Minka. Estimating a Dirichlet Distribution, 2000.

[16] M. F. Porter. An Algorithm for Suffix Stripping. In K. Sparck Jones and P. Willett, editors, *Readings in Information Retrieval*, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[17] N. Porter, editor. *Webster's Revised Unabridged Dictionary*. Merriam-Webster, Inc., 1913.

[18] A. Ratnaparkhi. A Maximum Entropy Model for Part-of-Speech Tagging, 1996.

[19] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The Author-Topic Model for Authors and Documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494. AUAI Press, 2004.

[20] E. Stamatatos. Intrinsic Plagiarism Detection using Character n-gram Profiles. *threshold*, 2:1–500, 2009.

[21] E. Stamatatos. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556, 2009.

[22] B. Stein, M. Koppel, and E. Stamatatos. Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection. *SIGIR Forum*, 41(2):68–71, Dec. 2007.

[23] Y. W. Teh. A Hierarchical Bayesian Language Model Based on Pitman-Yor Processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 985–992. Association for Computational Linguistics, 2006.

[24] M. J. Wainwright and M. I. Jordan. Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

[25] C. Wang, D. Blei, and F.-F. Li. Simultaneous Image Classification and Annotation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1903–1910. IEEE, 2009.

[26] F. Wood, C. Archambeau, J. Gasthaus, L. James, and Y. W. Teh. A Stochastic Memoizer for Sequence Data. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1129–1136. ACM, 2009.

A Meter

Our algorithm for meter extraction is inspired by Genzel et al. [8]. While meter is traditionally a poetic quality (and is treated as such in [8]), we brought it to prose by focusing on classical meter styles (Iambic, Spondee, etc.), all of which are based on 2- or 3-syllable 'feet'. We viewed meter as a function of per-syllable stresses (e.g., '(0, 1, 0)' for "continue", where a 1 indicates stress) and specifically focused on the stress $(2 \times 3) = 6$-grams, appending these 6-grams with two additional bits: the first to indicate distance from the latest comma, and the second to indicate distance from the latest piece of punctuation, both taken modulo 2. Stresses were looked up in the CMU Pronouncing Dictionary [11] and again used a minimum-Levenshtein-distance heuristic for absent words. In total, this gave us meter 8-grams that captured positioning relative to the nearest comma or other piece of punctuation.

B Model Discussion

We adopt the following notation: let d be a document, $\mathcal{D}$ be the set of documents, and D the number of documents. In general, lower case letters will denote instances, calligraphic letters sets, and upper case letters cardinalities: t will correspond to word types, n to words, and k to topics, $a$author. We will use subscripts in the natural way: $\mathcal{D}_a$ is the set of documents written by a, K_t are the number of topics for type t.

B.1 Parameter Fitting

It is well known that exact inference for LDA requires computing a prohibitive integral [5]. Instead, the posterior distribution $p(a, w, z, \theta | \alpha, \theta, \lambda, \eta)$ is approximated with a variational family: $q(\theta_{t,d} | \gamma_{t,d}) \prod_{(t,n) \in d} q(z_{dt,n} | \phi_{dt,n})$, indexed by parameters γ and ϕ. Here $\theta | \gamma \sim \text{Dirichlet}(\gamma)$ and $z | \phi \sim \text{Multi}(\phi)$, so that complete conditionals of θ and z under p are in the same family as their variational counterparts. Up to a constant independent of the variation parameters ϕ and γ, the KL divergence between p and q gives a lower bound on the posterior log likelihood. This is known as the ELBO, and (though non-convex), can be optimized with coordinate-wise gradient ascent. The parameters of the model–notably the per-author topics –can be fit using maximum likelihood methods. The updates follow [25] very closely, and in the interest of brevity, are omitted.

In each step, Wang's code reinitializes ϕ and γ at each iteration [25], and then optimizes a fixed point method to convergence before updating global parameters β and η. We also implement a method that keeps γ from the past iteration, and then then uses fixed point methods to optimizes ϕ. This seems to have slightly faster convergence, though it suffers a bit on classification accuracy. We also implement L_1 regularization with L-BFGS [13] and OWL-QN[1] [2].

Our code also implements a number of extensions to the algorithm in [25]. First, we allow for Maximum Likelihood Estimation of the per-author and global Dirichlet parameters (optimizing the variational lower bound). Both L-BFGS and Fixed Point [15] are supported.

[1] BFGS is not guaranteed to work with L_1 because the objective is not smooth, though it is still found to work well with L_1 penalties in practice. OWL-QN was implemented but not thoroughly tested and has some external package dependencies

We also use a method which gives us the expected topic assignments for authors, if all the authors documents are treated as one combined text: that is, for each word type t, $Pr(\text{topic} = k) = \epsilon + \sum_{d \in \mathcal{D}_a} \phi_{nd,t}$, where ϵ is a smoothing parameter.

Finally, our code extends the original sLDA implementation by including support for stochastic variational inference (SVI) [9]. On each run, a mini-batch of documents are sampled, the local parameters are computed, and then the global parameters are updated using a noisy estimation of the gradient via lines 10 and 11 of Figure 6 in [9]. Using the same mini-batch, we obtain a noisy estimate of the gradient of the variational objective with respect to η, and optimize. Every few iterations, we run a non-stochastic step to speed up convergence. This method converges very slowly: we hypothesize this is because the optimization of η relies on an approximation, second order expansion (see [25]), and hence sampling may not produce unbiased estimates. We only tested the algorithm with uniform sampling, but the code is written to accommodate non-uniform sampling as well (for example, one might want a mini-batch of documents for each author)

B.2 Classification

Our model implements two classification algorithms: the first methods assigns each with LDA (using a global (not per-author) prior over topics). For each document d, we then feed the average topic assignments $\bar{z}_t = \sum_{n \in D} \phi_{dn,t}$ into our softmax classifier and rank the potential authors for d using the softmax likelihood. We then output perplexity, accuracy, and recall at 2 and 3. The second classification method supported computes the total document likelihood under each author (using the per-author topic prior), and then chooses the author which maximizes this likelihood. In practice, the first method yields better performance, and also runs more quickly.

B.3 Smoothing

With large vocabularies, we may encounter terms in the test set that were not present in the training set. To this end, we implemented a vocabulary smoothing factor (see Section 5.4 in [5]) that is equivalent to placing a Dirichlet prior on vocabulary distributions. We also fixed a segmentation fault in Chong Wang's original code that occurred when new words in the test set were encountered.

C The Pitman-Yor Process

The Hierarchical Pitman-Yor (HPY) process obeys the power law frequencies observed from English language text [23], wheres as LDA induces more even proportions over words than one would see in natural language. Due to this insight, the HPY process underlies various various Bayesian natural language models, including Adaptor Grammars [10] and the Sequence Memoizer [26]. Thus, a supervised Pitman Yor Scheme might overcome some of the weakness of sLDA and the MAD. However, we choose not to use Pitman-Yor models because they are generally fit with MCMC [23], which would prove too slow given the size of our document corpora, and limited computational resources.