Not All Character N-grams Are Created Equal: A Study in Authorship Attribution

Upendra Sapkota and **Steven Bethard**
The University of Alabama at Birmingham
1300 University Boulevard
Birmingham, AL 35294, USA
{upendra,bethard}@cis.uab.edu

Manuel Montes-y-Gómez
Instituto Nacional de Astrofísica
Optica y Electrónica
Puebla, Mexico
mmontesg@ccc.inaoep.mx

Thamar Solorio
University of Houston
4800 Calhoun Rd
Houston, TX, 77004, USA
solorio@cs.uh.edu

Abstract

Character n-grams have been identified as the most successful feature in both single-domain and cross-domain Authorship Attribution (AA), but the reasons for their discriminative value were not fully understood. We identify subgroups of character n-grams that correspond to linguistic aspects commonly claimed to be covered by these features: morpho-syntax, thematic content and style. We evaluate the predictiveness of each of these groups in two AA settings: a single domain setting and a cross-domain setting where multiple topics are present. We demonstrate that character n-grams that capture information about affixes and punctuation account for almost all of the power of character n-grams as features. Our study contributes new insights into the use of n-grams for future AA work and other classification tasks.

1 Introduction

Authorship Attribution (AA) tackles the problem of determining who, among a set of authors, wrote the document at hand. AA has relevant applications ranging from plagiarism detection (Stamatatos, 2011) to Forensic Linguistics, such as identifying authorship of threatening emails or malicious code. Applied areas such as law and journalism can also benefit from authorship attribution, where identifying the true author of a piece of text (such as a ransom note) may help save lives or catch the offenders.

We know from state of the art research in AA that the length of the documents and the number of potential candidate authors have an important effect on the accuracy of AA approaches (Moore, 2001; Luyckx and Daelemans, 2008; Luyckx and Daelemans, 2010). We can also point out the most common features that have been used successfully in AA work, including: bag-of-words (Madigan et al., 2005; Stamatatos, 2006), stylistic features (Zheng et al., 2006; Stamatatos et al., 2000), and word and character level n-grams (Kjell et al., 1994; Keselj et al., 2003; Peng et al., 2003; Juola, 2006).

The utility of bag-of-words features is well understood: they effectively capture correlations between authors and topics (Madigan et al., 2005; Kaster et al., 2005). The discriminative value of these features is thus directly related to the level of content divergence among authors and among train and test sets.

The utility of stylistic features is also well understood: they model author preferences for the use of punctuation marks, emoticons, white spaces, and other traces of writing style. Such preferences are less influenced by topic, and directly reflect some of the unique writing patterns of an author.

Character n-grams are the single most successful feature in authorship attribution (Koppel et al., 2009; Frantzeskou et al., 2007; Koppel et al., 2011), but the reason for their success is not well understood. One hypothesis is that character n-grams carry a little bit of everything: lexical content, syntactic content, and even style by means of punctuation and white spaces (Koppel et al., 2011). While this argument seems plausible, it falls short of a rigorous explanation.

In this paper, we investigate what in the make-up

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 93–102,
Denver, Colorado, May 31 – June 5, 2015.

of these small units of text makes them so powerful. Our goal is two-fold: on the one hand we want to have a principled understanding of character n-grams that will inform their use as features for AA and other tasks; on the other hand we want to make AA approaches more accessible to non-experts so that, for example, they could be acceptable pieces of evidence in criminal cases.

The research questions we aim to answer are:

- *Are all character n-grams equally important?* For example, are the prefix of '**the**re', the suffix of 'brea**the**' and the whole word '**the**' all equivalent? More generally, are character n-grams that capture morpho-syntactic information, thematic information and style information equally important?

- *Are the character n-grams that are most important for single-domain settings also the most important for cross-domain settings?* Which character n-grams are more like bag-of-words features (which tend to track topics), and which are more like stylistic features (which tend to track authors)?

- *Do different classifiers agree on the importance of the different types of character n-grams?* Are some character n-grams consistently the best regardless of the learning algorithm?

- *Are some types of character n-grams irrelevant in AA tasks?* Are there categories of character n-grams that we can exclude and get similar (or better) performance than using all n-grams? If there are, are they the same for both single-domain and cross-domain AA settings?

Our study shows that using the default bag-of-words representation of char n-grams results in collapsing sequences of characters that correspond to different linguistic aspects, and that this yields suboptimal prediction performance. We further show that we can boost accuracy by loosing some categories of n-grams. Char n-grams closely related to thematic content can be completely removed without loss of accuracy, even in cases where the train and test sets have the same topics represented, a counter-intuitive argument. Given the wide spread use of char n-grams in text classification tasks, our findings have significant implications for future work in related areas.

2 Categories of Character N-grams

To answer our research questions and explore the value of character n-grams in authorship attribution, we propose to separate character n-grams into ten distinct categories. Unlike previous AA work where all character n-grams were combined into a single bag-of-n-grams, we evaluate each category separately to understand its behavior and effectiveness in AA tasks. These categories are related to the three linguistic aspects hypothesized to be represented by character n-grams: morpho-syntax (as represented by affix-like n-grams), thematic content (as represented by word-like n-grams) and style (as represented by punctuation-based n-grams). We refer to these three aspects as super categories (SC).

The following sections describe the different types of n-grams. We use the sentence in Table 1 as a running example for the classes and in Table 2 we show the resulting n-grams in that sentence. For ease of understanding, we replace spaces in n-grams with underscores (_).

The actors wanted to see if the pact seemed like an old-fashioned one.

Table 1: Example sentence to demonstrate the selection of different n-gram categories.

2.1 Affix n-grams

Character n-grams are generally too short to represent any deep syntax, but some of them can reflect morphology to some degree. In particular, we consider the following affix-like features by looking at n-grams that begin or end a word:

prefix A character n-gram that covers the first n characters of a word that is at least $n + 1$ characters long.

suffix A character n-gram that covers the last n characters of a word that is at least $n + 1$ characters long.

space-prefix A character n-gram that begins with a space.

SC	Category	Character n-grams
affix	*prefix*	**act** wan pac **see** lik fas
	suffix	ors ted **act** med ike ned
	space-prefix	_ac _wa _to _se _if _th _pa _li _an _ol _on
	space-suffix	he_ rs_ ed_ to_ ee_ if_ ct_ ke_ an_
word	*whole-word*	The **see** the old **one**
	mid-word	cto tor ant nte eem eme ash shi hio ion **one**
	multi-word	e_a s_w d_t o_s e_i f_t e_p t_s d_l n_o d_o
punct	*beg-punct*	-fa
	mid-punct	d-f
	end-punct	ld- ne.

Table 2: Example of the n-gram categories (n = 3) for the sentence in Table 1. The first column represents the super category (SC). The n-grams that appear in more than one category are in bold.

space-suffix A character n-gram that ends with a space.

2.2 Word n-grams

While character n-grams are often too short to capture entire words, some types can capture partial words and other word-relevant tokens. We consider the following such features:

whole-word A character n-gram that covers all characters of a word that is exactly n characters long.

mid-word A character n-gram that covers n characters of a word that is at least $n + 2$ characters long, and that covers neither the first nor the last character of the word.

multi-word N-grams that span multiple words, identified by the presence of a space in the middle of the n-gram.

2.3 Punctuation n-grams

The main stylistic choices that character n-grams can capture are the author's preferences for particular patterns of punctuation. The following features characterize punctuation by its location in the n-gram.

beg-punct A character n-gram whose first character is punctuation, but middle characters are not.

mid-punct A character n-gram with at least one punctuation character that is neither the first nor the last character.

end-punct A character n-gram whose last character is punctuation, but middle characters are not.

The above ten categories are intended to be disjoint, so that a character n-gram belongs to exactly one of the categories. For n-grams that contain both spaces and punctuation, we first categorize by punctuation and then by spaces. For example, 'e,_' is assigned to the *mid-punct* category, not the *space-suffix* category.

We have observed that in our data almost 80% of the n-grams in the *punct-beg* and *punct-mid* categories contain a space. This tight coupling of punctuation and spaces is due to the rules of English orthography: most punctuation marks require a space following them. The 20% of n-grams that have punctuation but no spaces correspond mostly to the exceptions to this rule: quotation marks, mid-word hyphens, etc. An interesting experiment for future work would be to split out these two types of punctuation into separate feature categories.

3 Datasets

We consider two corpora, a single-domain corpus, where there is only one topic that all authors are writing about, and a multi-domain corpus, where there are multiple different topics. The latter allows us to test the generalization of AA models, by testing them on a different topic from that used for training.

The first collection is the CCAT topic class, a subset of the Reuters Corpus Volume 1 (Lewis et al., 2004). Although this collection was not gathered for the goal of doing authorship attribution studies, previous work has reported results for AA with 10 and 50 authors (Stamatatos, 2008; Plakias and Stamatatos, 2008; Escalante et al., 2011). We refer to these as CCAT_10 and CCAT_50, respectively. Both CCAT_10 and CCAT_50 belong to CCAT category (about corporate/industrial news) and are balanced across authors, with 100 documents sampled for each author. Manual inspection of the dataset revealed that some of the authors in this collection consistently used signatures at the end of documents. Also, we noticed some writers use quotations a lot. Con-

Corpus	#authors	#docs /author/topic	#sentences /doc	#words /doc
CCAT_10	10	100	19	425
CCAT_50	50	100	19	415
Guardian1	13	13	53	1034
Guardian2	13	65	10	207

Table 3: Some statistics about the datasets.

sidering these parts of text for measuring the frequencies of character n-grams is not a good idea because signatures provide direct clues about the authorship of document and quotations do not reflect the author's writing style. Therefore, to clean up the CCAT collection, we preprocessed it to remove signatures and quotations from each document. Since the CCAT collection contains documents belonging to only corporate/industrial topic category, this will be our single-domain collection.

The other collection consists of texts published in The Guardian daily newspaper written by 13 authors in four different topics (Stamatatos, 2013). This dataset contains opinion articles on the topics: World, U.K., Society, and Politics. Following prior work, to make the collection balanced across authors, we choose at most ten documents per author for each of the four topics. We refer to this corpus as Guardian1. We also consider a variation of this corpus that makes it more challenging but that more closely matches realistic scenarios of forensic investigation that deal with short texts such as tweets, SMS, and emails. We chunk each of the documents by sentence boundaries into five new short documents. We refer to this corpus as Guardian2.

Table 3 shows some of the statistics of the CCAT and Guardian corpora and Table 4 presents some of the top character n-grams for each category (taken from an author in the Guardian data, but the top n-grams look qualitatively similar for other authors).

4 Experimental Settings

We performed various experiments using different categories of character n-grams. We chose n=3 since our preliminary experiments found character 3-grams to be more effective than other higher level character n-grams. For each category, we considered only those 3-grams that occur at least five times in the training documents.

The performance of different authorship attribution models was measured in terms of accuracy. In the single-domain CCAT experiments, accuracy was measured using the train/test partition of prior work. In the cross-domain Guardian experiments, accuracy was measured by considering all 12 possible pairings of the 4 topics, treating one topic as training data and the other as testing data, and averaging accuracy over these 12 scenarios. This ensured that in the cross-domain experiments, the topics of the training data were always different from that of the test data.

SC	**Category**	**N-grams**				
affix	*prefix*	tha	the	wit	con	hav
	suffix	ing	hat	ion	ent	ers
	space-prefix	_th	_of	_to	_an	_in
	space-suffix	he_	of_	to_	ed_	ng_
word	*whole-word*	the	and	for	was	not
	mid-word	tio	ati	iti	men	ent
	multi-word	e_t	s_a	t_t	s_t	n_t
punct	*beg-punct*	._T	's_	,_t	,_a	._I
	mid-punct	s,_	e,_	s._	e's	y's
	end-punct	es,	on.	on,	es.	er,

Table 4: Top character 3-grams in each category for author 'Catherine Bennet' in the cross-domain training data.

We trained support vector machine (SVM) classifiers using the Weka implementation (Witten and Frank, 2005) with default parameters. We also ran some comparative experiments with the Weka implementation of naive Bayes classifiers and the LibSVM implementation of SVMs. In the results below, when performance of a single classifier is presented, it is the result of Weka's SVM, which generally gave the best performance. When performance of other classifiers are presented, the classifiers are explicitly indicated.

5 Experimental Results and Evaluation

In this section, we present various results on authorship attribution tasks using both single as well as cross-domain datasets. We will explore character n-grams in depth and try to understand why they are so effective in discriminating authors.

5.1 Which n-gram Categories are Most Author-Discriminative?

After breaking character n-grams into ten disjoint categories, we empirically illustrate what categories are

	affix				word			punct		
Dataset	*prefix*	*suffix*	*space-prefix*	*space-suffix*	*multi-word*	*whole-word*	*mid-word*	*beg-punct*	*mid-punct*	*end-punct*
CCAT_10	**74.6**	**71.0**	**71.2**	66.0	65.8	48.0	**70.0**	60.2	35.4	56.2
CCAT_50	**61.9**	**59.6**	**57.0**	51.0	51.2	35.4	**61.0**	39.7	12.4	36.5

(a) Single Domain

	affix				word			punct		
Dataset	*prefix*	*suffix*	*space-prefix*	*space-suffix*	*multi-word*	*whole-word*	*mid-word*	*beg-punct*	*mid-punct*	*end-punct*
Guardian1	**41.6**	36.7	**41.9**	38.1	32.2	38.1	37.8	**43.5**	**46.1**	37.3
Guardian2	**31.0**	26.9	**29.7**	27.0	23.2	26.8	27.2	**33.6**	**33.5**	24.5

(b) Cross-Domain

Table 5: Accuracy of AA classifiers trained on each of the character n-gram categories. The top four accuracies for each dataset are in bold.

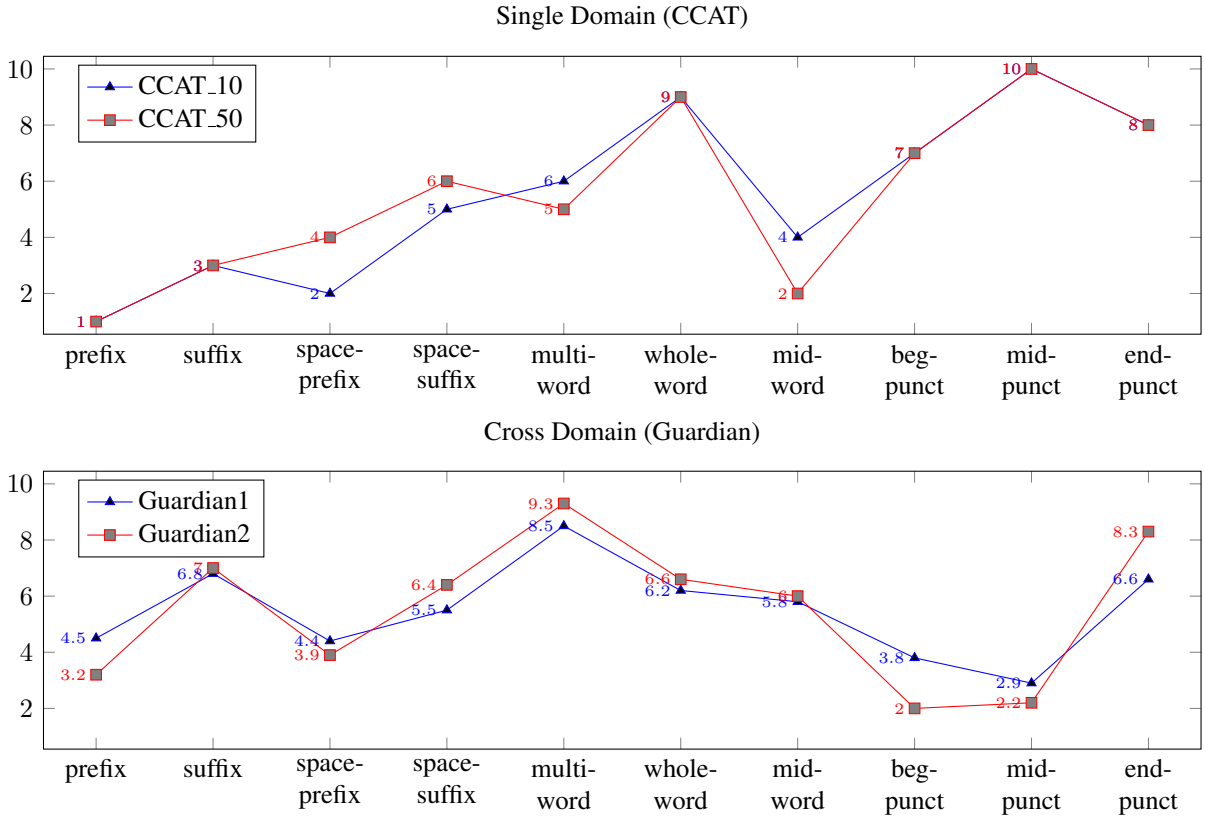

Figure 1: Average rank of the performance of each n-gram category on the single-domain CCAT tasks (top) and the cross-domain Guardian tasks (bottom).

most discriminative. Table 5 shows the accuracy of each type of n-gram for each of the different corpora.

Table 5(a) shows that the top four categories for single-domain AA are: *prefix*, *suffix*, *space-prefix*, and *mid-word*. These four categories have the best performance on both CCAT_10 and CCAT_50. In contrast, Table 5(b) shows that the top four categories for cross-domain AA are: *prefix*, *space-prefix*, *beg-punct*, and *mid-punct*.

For both single-domain and cross-domain AA, *prefix* and *space-prefix* are strong features, and are generally better than the suffix features, perhaps because authors have more control over prefixes in English, while suffixes are often obligatory for grammatical reasons. For cross-domain AA, *beg-punct* and *mid-punct* are the top features, likely because an author's

use of punctuation is consistent even when the topic changes. For single-domain AA, *mid-word* was also a good feature, probably because it captured lexical information that correlates with authors' preferences towards writing about specific topics.

Figure 1 shows an alternate view of these results, graphing the rank of each n-gram type. For computing the rank, the accuracies of the ten different n-gram type classifiers are sorted in decreasing order and ranked from 1 to 10 respectively with ties getting the same rank. For the Guardian corpora, the average rank of each n-gram category was computed by averaging its rank across the 12 possible test/train cross-domain combinations. In both of the single-domain CCAT corpora, the classifier based on *prefix* n-grams had the top accuracy (rank 1), and the classifier based on *mid-punct* had the worst accuracy (rank 10). In both of the cross-domain Guardian corpora, on the other hand, *mid-punct* was among the top-ranked n-gram categories. This suggests that punctuation features generalize the best across topic, but if AA is more of a topic classification task (as in the single-domain CCAT corpora), then punctuation adds little over other features that more directly capture the topic.

Since our cross-domain datasets are small, we performed a small number of planned comparisons using a two-tailed t-test over the accuracies on the Guardian1 and Guardian2 corpora. We found that in both corpora, the best punctuation category (*punct-mid*) is better than the best word category (*whole-word*) with $p < 0.001$. In the Guardian2 corpus, the best affix category (*space-prefix*) is also better than the best word category (*whole-word*) with $p < 0.05$, but this does not hold in the Guardian1 corpus ($p = 0.14$). Also, we observed that in both Guardian1 and Guardian2 datasets, both *punct-mid* and *space-prefix* are better than *multi-word* ($p < 0.01$).

Overall, we see that **affix** n-grams are generally effective in both single-domain and cross-domain settings, **punctuation** n-grams are effective in cross-domain settings, and *mid-word* is the only effective **word** n-gram, and only in the single-domain setting.

5.2 Do Different Classifiers Agree on the Importance of Different n-gram Types?

The previous experiments have shown, for example, that *prefix* n-grams are universally predictive in AA tasks, that *mid-word* n-grams are good predictors in single-domain settings, and that *beg-punct* n-grams are good predictors in cross-domain settings. But are these facts about the n-gram types themselves, or are these results only true for the specific SVM classifiers we trained?

Comparison	CCAT	Guardian
Weka SVM vs LibSVM	0.93	0.81
Weka SVM vs Naive Bayes	0.73	0.57
LibSVM vs Naive Bayes	0.77	0.44

Table 6: Spearman's rank correlation coefficient (ρ) for each pair of classifiers on the single-domain (CCAT) and cross-domain (Guardian) settings.

To see whether certain types of n-grams are fundamentally good or bad, regardless of the classifier, we compare performance of the different n-gram types for three classifiers: Weka SVM classifiers (as used in our other experiments), LibSVM classifiers and Weka's naive Bayes classifiers[1]. Figure 2 shows the n-gram category rankings for all these classifiers[2] for both the single-domain CCAT and the cross-domain Guardian settings.

Across the different classifiers, the pattern of feature rankings are similar. Table 6 shows the Spearman's rank correlation coefficient (ρ) for the per-n-gram-type accuracies of each pair of classifiers. We observe fairly high correlations, with ρ above 0.70 for all single-domain pairings, and between 0.44 and 0.81 for cross-domain pairings.

As in Section 5.1, *prefix* and *space-prefix* are among the most predictive n-gram types. In the single-domain settings, we again see that *suffix* and *mid-word* are also highly predictive, while in the cross-domain settings, we again see that *beg-punct* and *mid-punct* are highly predictive. These results all confirm that some types of n-grams are fundamentally more predictive than others, and our results are not specific to the particular type of classifier used.

[1] Weka SVM and LibSVM are both support vector machine classifiers, but Weka uses Platt's sequential minimal optimization algorithm while LibSVM uses working set selection with second order information. The result is that they achieve different performance on our AA tasks.

[2] We also tried a decision tree classifier, C4.5 (J48) from WEKA, and it produced similar patterns (not shown).

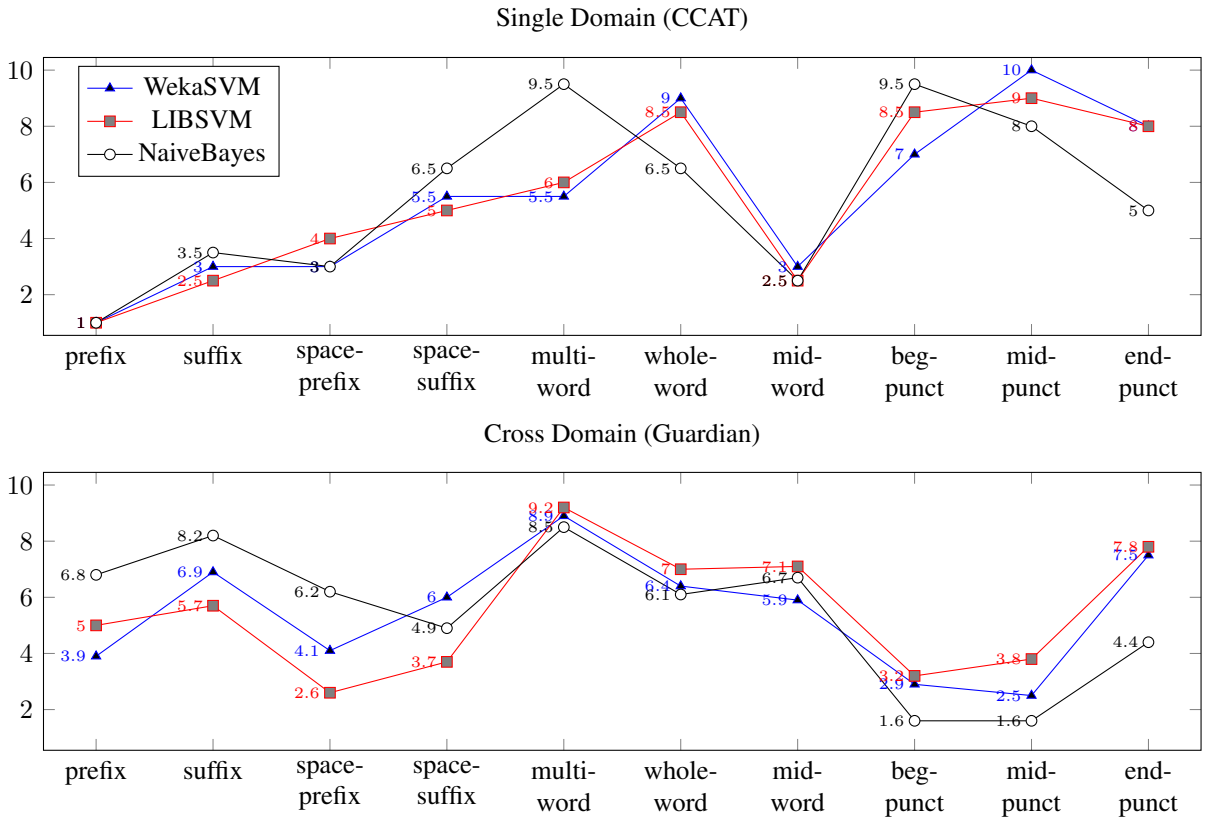

Figure 2: Average rank of the performance of each n-gram category across different types of classifiers on the single-domain CCAT task (top) and the cross-domain Guardian task (bottom).

5.3 Are Some Character N-grams Irrelevant?

In the previous sections, we have seen that some types of character n-grams are more predictive than others - **affix** n-grams performed well in both single domain and cross-domain settings and **punctuation** n-grams performed well in cross-domain settings. In general, **word** n-grams were not as predictive as other types of n-grams (with the one exception being *mid-word* n-grams in the single domain setting). Given this poor performance of **word** n-grams, a natural question is: could we exclude these features entirely and achieve similar performance?

Our goal then is to compare a model trained on **affix** n-grams and **punct** n-grams against a model trained on "all" n-grams. We consider two definitions of "all":

all-untyped The traditional approach to extracting n-grams where n-gram types are ignored (e.g., 'the' as a whole word is no different from 'the' in the middle of a word)

all-typed The approach discussed in this paper, where n-grams of different types are distinguished (equivalent to the set of all **affix+punct+word** n-grams).

We compare these models trained on all the n-grams to our **affix+punct** model.

Table 7 shows this analysis. For either definition of "all", the model that discards all **word** features achieves performance as high or higher than the model with all of the features, and does so with only about two thirds of the features. This is not too surprising in the cross-domain Guardian tasks, where the **word** n-grams were among the worst features. On the single-domain CCAT tasks this result is more surprising, since we have discarded the *mid-word* n-grams, which was one of the best single-domain n-gram types. This indicates that whatever information *mid-word* is capturing it is also being captured in other ways via **affix** and **punct** n-grams. Of all 1024 possible combinations of features, we tried a

Dataset	all-untyped		all-typed		affix+punct	
	Acc	*N*	**Acc**	*N*	**Acc**	*N*
CCAT_10	77.8	*8245*	77.2	*9715*	**78.8**	*5474*
CCAT_50	69.2	*14461*	69.1	*17062*	**69.3**	*9966*
Guardian1	55.6	*5689*	53.6	*6966*	**57.0**	*3822*
Guardian2	45.9	*5687*	45.6	*6965*	**48.0**	*3820*

Table 7: Results of excluding **word** n-grams, compared to using all n-grams, either in the traditional approach (untyped n-grams) or in the approach of this paper (typed n-grams). Accuracy (Acc) and the number of features (*N* in italics) are reported for each classifier. The best accuracy for each dataset is in bold.

number of different combinations and were unable to identify one that outperformed **affix+punct**. Overall, this experiment gives compelling evidence that **affix** and **punct** n-grams are more important than **word** n-grams.

6 Analysis

We did a manual exploration of our datasets. In our cross-domain dataset, the character 3-gram 'sti' shows up as both *prefix* and *mid-word*. All 13 authors use 'sti' frequently as a *mid-word* n-gram in words such as in**sti**tution, exi**sti**ng, ju**sti**ce, and di**sti**nction. For example:

- The government's story is that the exi**sti**ng warheads might be deteriorating.
- For all the ju**sti**ce of many of his accusations, the result is occasionally as dreadful as his title suggests.

But only six authors use 'sti' as a *prefix*, in examples like:

- Their mission was to convince tourists that Britain was **sti**ll open for business.
- There aren't even any dead people on it, since by the very act of being dead and **sti**ll famous, they assert their long-term impact.

Thus 'sti' as a *prefix* is predictive of authorship even though 'sti' as a *mid-word* n-gram is not. Notably, under the traditional untyped bag-of-n-grams approach, both versions of 'sti' would have been treated the same, and this discriminative power would have been lost.

To use old-fashioned language, she is motherly - a plump, rosy-cheeked woman of Kent, whom nature seemed to have created to raise children.
To use old-fas ned lan age, she is mot rly - a plump, rosy-che ked woman of Kent, whom nature seemed to have cre ted to raise chi ren.

Table 8: Example sentence showing the opacity of each character. Darkness of character is determined by the number of categories it belongs to (lowest=lighter, highest=darkest color). Categories in **word** are discarded.

As already demonstrated in Section 5 that **affix+punct** features perform better than using all the features, we would like to use an example from our dataset to visualize the text when features in SC **word** are discarded. Out of seven categories in **affix** and **punct**, we computed in how many of them each character belongs to, three being the maximum possible value. Therefore, we show each character with different opacity level depending on number of categories it belongs to: zero will get white color (word related n-grams), one will get 33% black, two will get 67% black, and three will get 100% black. In Table 8, we show an example sentence before (first row of Table 8) and after (second row of Table 8) showing the opacity level of each character. It is clear that the darkest characters are those around the punctuation characters and those around spaces are second darkest, while the lightest (with 0% darkness) are the ones in the middle of long words. This gives us an idea about the characters in a text that are important for AA tasks.

7 Discussion

Various hypotheses have been put forth to explain the "black magic" (Kestemont, 2014) behind the success of character n-gram features in authorship attribution. Kestemont (2014) conjectured that their utility was in capturing function words and morphology. Koppel et al. (2009) suggested that they were capturing topic information in single domain settings, and style and syntactic information in cross-domain settings. Our study provides empirical evidence for testing these claims. We did indeed find that the ability of character n-grams to capture morphology is useful, as reflected in the high prediction performance of **af-**

fix n-grams in both single-domain and cross-domain settings. And we found that **word** n-grams (capturing topic information) were useful in single domain settings, while **punct** n-grams (capturing style information) were useful in cross-domain settings. We further found that **word** n-grams are unnecessary, even in single-domain settings. Models based only on **affix** and **punct** n-grams performed as well as models with all n-grams regardless of whether it was a single-domain or cross-domain authorship attribution task.

Our findings on the value of selecting n-grams according to the linguistic aspect they represent may also be beneficial in other classification tasks where character n-grams are commonly used. Promising tasks are those related to the stylistic analysis of texts, such as native language identification, document similarity and plagiarism detection.

Morphologically speaking, English is a poor language. The fact that we identified significant differences in performance by selecting n-gram categories that are related to affixation in this poorly inflected language suggests that we may find even larger differences in performance in morphologically richer languages. We leave this research question for future work.

Acknowledgements

This research was partially supported by NSF awards 1462141 and 1254108. It was also supported in part by the CONACYT grant 134186 and the WIQ-EI IRSES project (grant no. 269180) within the FP 7 Marie Curie.

References

H. J. Escalante, T. Solorio, and M. Montes-y Gomez. 2011. Local histograms of character n-grams for authorship attribution. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 288–298, Portland, Oregon, USA, June. Association for Computational Linguistics.

G. Frantzeskou, E. Stamatatos, S. Gritzalis, and C. E. Chaski. 2007. Identifying authorship by byte-level n-grams: The source code author profile (SCAP) method. *Journal of Digital Evidence*, 6(1).

P. Juola. 2006. Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3):233–334, December.

A. Kaster, S. Siersdorfer, and G. Weikum. 2005. Combining text and linguistic document representations for authorship attribution. In *SIGIR Workshop: Stylistic Analysis of Text for Information Access (STYLE*, pages 27–35.

V. Keselj, F. Peng, N. Cercone, and C. Thomas. 2003. N-gram based author profiles for authorship attribution. In *Proceedings of the Pacific Association for Computational Linguistics*, pages 255–264.

M. Kestemont. 2014. Function words in authorship attribution. From black magic to theory? In *CLFL*, pages 59–66, Gothenburg, Sweden, April.

Bradley Kjell, W.Addison Woods, and Ophir Frieder. 1994. Discrimination of authorship using visualization. *Information Processing & Management*, 30(1):141 – 150.

M. Koppel, J. Schler, and S. Argamon. 2009. Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, 60(1):9–26.

M. Koppel, J. Schler, and S. Argamon. 2011. Authorship attribution in the wild. *Language Resources and Evaluation*, 45:83–94.

D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. 2004. RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397.

K. Luyckx and W. Daelemans. 2008. Authorship attribution and verification with many authors and limited data. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008)*, pages 513–520, Manchester, UK, August.

K. Luyckx and W. Daelemans. 2010. The effect of author set size and data size in authorship attribution. *Literary and Linguistic Computing*, pages 1–21, August.

D. Madigan, A. Genkin, S. Argamon, D. Fradkin, and L. Ye. 2005. Author identification on the large scale. In *Proceedings of CSNA/Interface 05*.

R. Moore. 2001. There's no data like more data (but when will enough be enough?). In *Proceedings of the IEEE International Workshop on Intelligent Signal Processing*, Budapest, Hungary.

F. Peng, D. Schuurmans, V. Keselj, and S. Wang. 2003. Language independent authorship attribution using character level language models. In *10th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, pages 267–274.

S. Plakias and E. Stamatatos. 2008. Tensor space models for authorship attribution. In *Proceedings of the 5th Hellenic Conference on Artificial Intelligence: Theories, Models and Applications*, volume 5138 of *LNCS*, pages 239–249, Syros, Greece.

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

E. Stamatatos. 2006. Authorship attribution based on feature set subspacing ensembles. *International Journal on Artificial Intelligence tools*, 15(5):823–838.

E. Stamatatos. 2008. Author identification: Using text sampling to handle the class imbalance problem. *Information Processing and Managemement*, 44:790–799.

E. Stamatatos. 2011. Plagiarism detection using stopword n-grams. *Journal of the American Society for Information Science and Technology*, 62(12):2512–2527.

E. Stamatatos. 2013. On the robustness of authorship attribution based on character n-gram features. *Journal of Law & Policy*, 21(2):421 – 439.

I. H. Witten and E. Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kauffmann, 2nd edition.

R. Zheng, Jiexun Li, Hsinchun Chen, and Zan Huang. 2006. A framework for authorship identification of online messages: Writing-style features and classification techniques. *J. Am. Soc. Inf. Sci. Technol.*, 57(3):378–393, February.