Authorship Attribution through Punctuation n-grams and Averaged Combination of SVM

Notebook for PAN at CLEF 2019

Carolina Martín-del-Campo-Rodríguez[1], Daniel Alejandro Pérez Alvarez[1], Christian Efraín Maldonado Sifuentes[1], Grigori Sidorov[1], Ildar Batyrshin[1], and Alexander Gelbukh[1]

Instituto Politécnico Nacional (IPN),
Center for Computing Research (CIC), Mexico City, Mexico
cm.del.cr@gmail.com, daperezalvarez@gmail.com, chrismaldonado@gmail.com, sidorov@cic.ipn.mx, batyr1@gmail.com, gelbukh@gelbukh.com

Abstract This work explores the exploitation of pre-processing, feature extraction and the averaged combination of Support Vector Machines (SVM) outputs for the open-set Cross-Domain Authorship Attribution task. The use of punctuation n-grams as a feature representation of a document is introduced for the Authorship Attribution in combination with traditional character n-grams. Starting from different feature representations of a document, several SVM are trained to represent the probability of membership for a certain author to latter obtain an average of all the SVM results. This approach managed to obtain 0.642 with the Macro F1-score for the PAN 2019 contest of open-set Cross-Domain Authorship Attribution.

1 Introduction

The problem of authorship attribution in cross-domain conditions is defined when documents of known authors that come from different writing domains (different genres or themes) are used to gather information that enables the classification of documents of unknown authorship from a list of possible candidates. In the case that no candidate matches the style of an unattributed document it is possible that the actual author was not included within the candidate list, such case is known as an open-set attribution problem.

The 2019 edition of PAN [2] focuses in an open-set Cross-Domain Authorship Attribution in fanfiction. Fanfiction is a literature work in which a fan seeks to imitate as much as possible the writhing style of an admired author, and where a fandom is referred as the genre or original work of a certain writer. In this edition of PAN a set of

documents are provided from known fans writing in several fandoms, to then require the classification of documents from unknown authors writing in a single fandom, being possible that the author of the document is not part of the previous set of known writers [4].

2 Related work

Within the main factors to reach an improvement in Cross-Domain Authorship Attribution, the pre-processing stage has been identified has a key tool to increase the effectiveness of the classifiers, and therefore, it becomes a principal concern to the development of this work. In 2017 Markov et al. [6] shows that the elimination of topic-dependent information from texts allows to improve the performance of authorship attribution classifiers. By the replacement of digits, punctuation marks splitting, and the replacement of named entities before the extraction of character n-grams the score of correct assignments rise for cross-domain authorship attribution. Besides these findings, it is also identified that the appropriate selection of the dimensionality of the representation of character n-grams is a crucial feature in pre-processing for the cross-domain task.

The authors of [1] included a character n-gram model of variable length in which non-diacritics were distorted focusing in punctuation and other non-alphabetical symbols to represent the structure of the text. On the other hand [5] experimented with text representation based purely on punctuation n-grams for the task of native language identification. In [3] syntactic n-grams are proposed, that is n-grams in non-linear manner. This type of n-grams allows using syntactic information within the automatic text processing methods related to classification or clustering.

3 Method

3.1 Features extraction

The following are the principal features that were considered for the development of our approach:

- Character n-grams.
- Pure-punctuation n-grams.
- Typed character n-grams.
- Bag of Words (BoW).

As follows, we will describe the pure-punctuation n-grams, typed character n-grams. The overall procedure described in this section is summarized in Figure 1.

n-grams based on punctuation (pure-punctuation n-grams) The style of an author can be determined, to some extent, through the use of punctuation. We determined that beyond the counting of punctuation an important factor is the way that the author uses it. So, we proposed to extract n-grams based only in these. We considered as punctuation

all those characters that are not letters, numbers or spaces from the training corpus plus the characters obtained from the library string.punctuation of python 3 [1]. All the characters different to these were removed, obtaining for each text a representation only based in punctuation.

So, considering the text in (1), the punctuation representation is: **'.,,'.,–.**. After, we obtained the character n-grams for each new text representation.

Table 1: Fragment of text extracted from the training corpus

't speak to anyone. I saw her at the funeral, and she said a few words, but that's it. I went to see her afterwards, to pick up your stuff – they let me have it after the forensic team did their thing.

Typed character n-grams In [8] the typed character n-grams were introduced, basically these are subgroups of character n-grams. These subgroups are call super categories (SC). Each of these SC are divided in different categories:

- Affix n-grams: Consider morphosyntactic aspects. This SC capture morphology to some extent. It is divided in prefix, suffix, space-prefix and space-suffix.
- Word n-grams: Consider thematic content aspects. This SC capture partial words and other word-relevant tokens (whole-word, mid-word, multi-word).
- Punctuation n-grams (typed-punctuation n-grams[2]): Consider style aspects. This SC capture patterns of punctuation (beg-punct, mid-punct, end-punct).

The features obtained were filtered considering the document frequency (*df*), a term is ignored if the *df* is lower than a threshold (*th*). This means that if a term appear in less than *th* documents, it will be ignored (not considered for the vectorization). For the features weightening the tf-idf was applied. The vectorization was made with the library *scikit-learn*[3], using the function *TfidfVectorizer*. This function allows us to do the vectorization, the filter based on *df* and the features weightening at the same time.

3.2 Evaluation of features with SVM

SVM was the selected algorithm to resolve the task of open-class Authorship Attribution. We used the same configuration as in the baseline, applying *CalibratedClassifierCV* to get the beloging probabilities to the classes per document. For each feature representation (5 different representations) we trained different SVM's and got the belonging probability model (unknown document, class) for each representation.

[1] https://docs.python.org/3/

[2] to avoid confusion with the pure-punctuation n-gramas proposed, we named the SC punctuation n-grams as typed-punctuation n-grams

[3] https://scikit-learn.org/stable/

3.3 Probability models point-to-point average

Having the probability models an average point to point was made (averaged probability model), this is the idea behind the VotingClassifier with a soft voting approach. Weighted of the probabilities was discard to avoid a posible overfitting.

With the averaged probability model, for each unknown document was considered the following: the probabilities of class belonging was sorted (from highest to lowest), the difference (*diff*) of the two highest values was taken, if *diff* was smaller than a threshold, it was considered that the document was not written for any of the candidates, otherwise, the unknown document is assigned to the class with the highest probability.

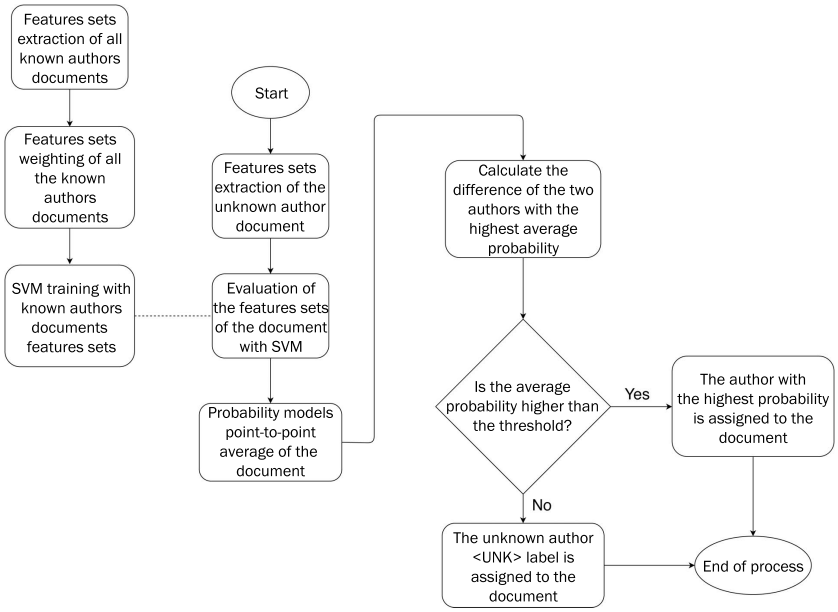

Figure 1. Flow chart of the methodology applied to the open-set cross-fandom attribution PAN 2019.

4 Experiments

For each feature type different experiments were made, related to the size on n-grams, variation of n was made from 1 to 10. Also, concatenation of the characteristics (for type) was made with a variation from 1 to 8.

For each characteristic type, different values of *df* was considered, variations from 1 to 5 were made to determine, for type of feature, witch was the best filter to consider.

The weightening of the features was done with tf-idf. Two different methods were considered to obtain it: the one implemented in *gensim*[4] TfidfModel and the other one with *scikit-learn* TfidfVectorizer (that applies a normalization, by the Euclidean norm, after the weightening). Considering that for the use of TfidfModel is necesary to convert the data type in corpus type, the facility of TfidfVectorizer for the use of filters and weightening, and preliminary tests, TfidfVectorizer was selected to get the weightening.

Table (2) shows the features considered in the final configuration of our approach.

Table 2: Final configuration of features. [n, m] is concatenation of features in the range n to m

Features	**n**	**Document Frequency threshold**
character n-grams	[1, 6]	5
	6	2
	5	2
	4	5
	3	5
pure-punctuation n-grams	2	1
	3	1
	4	1
	5	1
	[1, 5]	2
typed character n-grams	1	4
bag of words	1	5
typed-punctuation n-grams	1	5

5 Results

The Macro F1-score was the measure used for the evaluation. The results obtained with our approach for the development corpus are shown in table 3. the system was executed in TIRA [7].

Table 4 shows the competition scores. Our approach **delcamporodriguez19** had a Macro F1-score equal to **0.642**. The best approach was the one proposed by mutten-thaler19 with a Macro F1-score of **0.690**, with a value of **0.048** superior to ours.

[4] https://radimrehurek.com/gensim/

Table 3: Result of our approach in the development corpus

Problem	Language	Macro F1-score
problem00001	english	0.800
problem00002	english	0.549
problem00003	english	0.649
problem00004	english	0.551
problem00005	english	0.572
problem00006	french	0.732
problem00007	french	0.680
problem00008	french	0.629
problem00009	french	0.732
problem00010	french	0.732
problem00011	italian	0.752
problem00012	italian	0.644
problem00013	italian	0.800
problem00014	italian	0.729
problem00015	italian	0.785
problem00016	spanish	0.843
problem00017	spanish	0.703
problem00018	spanish	0.816
problem00019	spanish	0.667
problem00020	spanish	0.582
Average		**0.697**

Table 4: Results in the competition corpus.

Contester	Macro F1-score
muttenthaler19	**0.690**
neri19	0.680
eleandrocustodio19	0.650
devries19	0.644
delcamporodriguez19	0.642
isbister19	0.622
johansson19	0.616
basile19	0.613
vanhalteren19	0.598
rahgouy19	0.580
gagala19	0.576
kipnis19	0.259

6 Conclusions

The application of several feature representations, and the inclusion of features based on punctuation represent a factor in the improvement of the classification of authorship in open-class Cross-Domain Authorship Atttribution. Besides from the pre-processing benefits presented in this work, the use of several SVM's probability models are applied to select the author of the fanfiction by an average of the outputs. This approach man-

aged to obtain 0.642 with the Macro F1-score for the PAN 2019 contest of open-class Cross-Domain Authorship Attribution in fanfiction.

References

1. Custódio, J.E., Paraboni, I.: Each-usp ensemble cross-domain authorship attribution. Working Notes Papers of the CLEF (2018)
2. Daelemans, W., Kestemont, M., Manjavancas, E., Potthast, M., Rangel, F., Rosso, P., Specht, G., Stamatatos, E., Stein, B., Tschuggnall, M., Wiegmann, M., Zangerle, E.: Overview of PAN 2019: Author Profiling, Celebrity Profiling, Cross-domain Authorship Attribution and Style Change Detection. In: Crestani, F., Braschler, M., Savoy, J., Rauber, A., Müller, H., Losada, D., Heinatz, G., Cappellato, L., Ferro, N. (eds.) Proceedings of the Tenth International Conference of the CLEF Association (CLEF 2019). Springer (Sep 2019)
3. Grigori, S.: Syntactic n-grams in computational linguistics. Springer (2019)
4. Kestemont, M., Stamatatos, E., Manjavacas, E., Daelemans, W., Potthast, M., Stein, B.: Overview of the Cross-domain Authorship Attribution Task at PAN 2019. In: Cappellato, L., Ferro, N., Losada, D., Müller, H. (eds.) CLEF 2019 Labs and Workshops, Notebook Papers. CEUR-WS.org (Sep 2019)
5. Markov, I.: Automatic Native Language Identification. Ph.D. thesis, Instituto Politecnico Nacional (2018)
6. Markov, I., Stamatatos, E., Sidorov, G.: Improving cross-topic authorship attribution: The role of pre-processing. In: Proceedings of the 18th International Conference on Computational Linguistics and Intelligent Text Processing. CICLing 2017, Springer (2017)
7. Potthast, M., Gollub, T., Wiegmann, M., Stein, B.: TIRA Integrated Research Architecture. In: Ferro, N., Peters, C. (eds.) Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF. Springer (2019)
8. Sapkota, U., Bethard, S., Montes-y Gómez, M., Solorio, T.: Not all character n-grams are created equal: A study in authorship attribution. In: Proceedings of the 2015 Annual Conference of the North American Chapter of the ACL: Human Language Technologies. pp. 93–102. NAACL-HLT'15, Association for Computational Linguistics (2015)