Representation of Target Classes for Text Classification - AMRITA_CEN_NLP@RusProfiling PAN 2017

Barathi Ganesh HB, Reshma U, Anand Kumar M and Soman KP
Center for Computational Engineering and Networking
Amrita University
Coimbatore, India
barathiganesh.hb@gmail.com,reshma.anata@gmail.com,m_anandkumar@cb.amrita.edu,soman_kp@amrita.edu

ABSTRACT

This working note describes the system we used while participating in RusProfiling PAN 2017 shared task. The objective of the task is to identify the gender trait of the author from the author's text written in the Russian Language. Taking this as a binary text classification problem, we have experimented to develop a representation scheme for target classes (called class vectors) from the texts belonging to the corresponding target classes. These class vectors are computed from the traditional representation methods available in Vector Space Models and Vector Space Models of Semantics. Followed by the representation, Support Vector Machine with a linear kernel is used to perform the final classification. For this task, genre independent corpus is provided by the RusProfiling PAN 2017 shared task organizers. This proposed model attains almost equal performance across all the genre available in the test corpus.

CCS CONCEPTS

• **Computing methodologies → Natural language processing**; **Feature selection**;

KEYWORDS

Author Profiling, Class Representation, Russian Language, Text Representation, Text Classification, Vector Space Models, Vector Space Models of Semantics

1 INTRODUCTION

Prediction of author's traits (gender, age, native language and personality traits) from their texts is known as author profiling and its applications in targeted internet advertising, forensic science and consumer behaviour analysis induces the researchers[1], [2], [3] and the industries[4] to develop a reliable author profiling systems. The growth of digital text shared in social media (Facebook, Twitter) feeds the researchers with the required corpus to develop the author profiling systems and its related shared tasks to build the state of art systems [4–6].

Unlike other text classification problems which identify the context, here more than identifying the context shared by the author, identifying the style used to share the content by the author is more relevant [1, 2]. The general text classification works by stacking the text representation followed by the feature learning and classification. The text representation is a pivotal task that has a direct proportion with the performance of the system.

By considering the above discussed points, we have experimented to develop a representation scheme for target classes, which is termed in this paper as class vectors. On successive representation of texts using the methods available in Vector Space Models (Document-Term Matrix and Term Frequency - Inverse Document Frequency) and Vector Space Models of Semantics (Document-Term Matrix with Singular Value Decomposition and Term Frequency-Inverse Document Matrix with Singular Value Decomposition), we summed the text vectors in the respective Target classes to form the class vectors. Later, the variation between the class vectors and the text vectors are computed through the distance and correlation measures. These measures are taken as the features and fed to the Support Vector Machine with a linear kernel to make the final prediction. The experimented model s given in Figure 1.

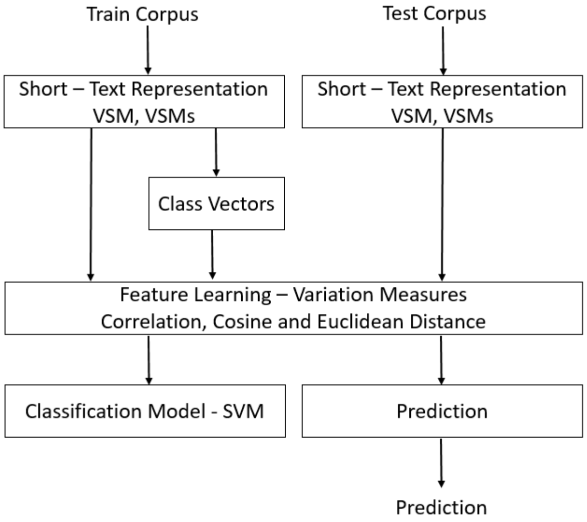
Train Corpus
Test Corpus
Short – Text Representation VSM, VSMs
Short – Text Representation VSM, VSMs
Class Vectors
Feature Learning – Variation Measures Correlation, Cosine and Euclidean Distance
Classification Model - SVM
Prediction
Prediction

Figure 1: Experimented Model

The objective of the task is detailed in Section 2; statistics about the corpus are given in Section 3; underlying components for computing class vectors, feature learning, and classification method are explained in Section 4; cross-validation reports, results reported by the shared task organizers and observations about the results are detailed in Section 5.

[1] http://pan.webis.de/clef17/pan17-web/author-profiling.html
[2] http://nlp.amrita.edu:8080/INLI/Test.html
[3] https://sites.google.com/site/nlisharedtask/home
[4] https://personality-insights-livedemo.mybluemix.net/

2 TASK DESCRIPTION

Given the training corpus which consists of author's text tagged with the author's gender information, the objective is to predict the gender information for the author's text available in the test corpus[5]. In the below given equation $tag \, \epsilon \, \{male, female\}$ and n represents the total number of author's text in the corpus.

$$train\_corpus = \{(text_1, tag_1), (text_2, tag_1), ..., (text_n, tag_n)\} \quad (1)$$

$$test\_corpus = \{text_1, text_2, ..., text\_n\} \quad (2)$$

3 CORPUS STATISTICS

For this task, the corpus has been provided by the RusProfiling PAN 2017 shared task organizers [3]. The number of author's text (n) in the corpus is given in Table 1. In the given corpus Tweets are taken as author's text for training (**Train**), offline texts (picture descriptions, letter to a friend etc.) from RusPersonality Corpus taken for **Test1**, comments from Facebook are taken for **Test2**, tweets from Twitter taken for **Test3**, author's text from product and service online reviews taken for **Test4** and author's text from Gender imitation corpus taken for **Test5** (women imitating men and the other way around).

4 REPRESENTATION

Text representation is the task of transforming the unstructured texts into its equivalent numerical representation. On successive representation, further mathematical computation will be applied to it.

4.1 Author Representation

4.1.1 Vector Space Models. Document-Term Matrix (DTM) is a basic representation method, which accounts the count of the unique word's present in the document [7]. The reweighing scheme introduced along with the DTM to handle the uninformative words through the inverse document frequency in Term Frequency-Inverse Document Frequency Matrix (TF-IDF) [7]. These both methods are from Vector Space Models (VSM) and represented as follows,

$$D^{dtm}_{n\times m} = dtm(train\_corpus \mid test\_corpus) \quad (3)$$

$$D^{tfidf}_{n\times m} = tfidf(train\_corpus \mid test\_corpus) \quad (4)$$

In the above equation, n represents the number of author's text and m represents the number of unique words in the vocabulary. Unique words from the train and test corpus are used to build the common vocabulary.

4.1.2 Vector Space Models of Semantics. Singular Value Decomposition applied to the matrix from VSM to get the semantic representation of the author's text. Applying matrix factorization on top of the matrix from the VSM is known as Vector Space Models of Semantics (VSMs) [7]. This is represented as,

$$U^{dtm}_{n\times n} \Sigma_{n\times m} V^T_{m\times m} = svd\left(D^{dtm}_{n\times m}\right) \quad (5)$$

$$U^{tfidf}_{n\times n} \Sigma_{n\times m} V^T_{m\times m} = svd\left(D^{tfidf}_{n\times m}\right) \quad (6)$$

In the above equation, U represents the basis vector representation of the author's text and it is used while computing class vectors, σ represents the singular values (significance of topics) in the descending order and V represents the basis vector representation of the words in the vocabulary. In this work, we have not performed the dimensionality reduction.

4.2 Target Class Representation

On the successive computation of $D^{dtm}_{n\times m}, D^{tfidf}_{n\times m}, U^{dtm}_{n\times n}$ and $U^{tfidf}_{n\times n}$, the class vectors are computed by summing the respective vectors of the author's text belongs to the classes male and female. It is given as follows for computing class vectors from VSM,

$$C^{male}_{1\times m} = \sum_{i=1}^{n} vsm\_representation[i, :] \ if \ tag_i = male \quad (7)$$

$$C^{female}_{1\times m} = \sum_{i=1}^{n} vsm\_representation[i, :] \ if \ tag_i = female \quad (8)$$

$$vsm\_representation \, \epsilon \left\{D^{dtm}_{n\times m}, D^{tfidf}_{n\times m}\right\} \quad (9)$$

Similarly, class vectors from VSMs are computed as follows,

$$C^{male}_{1\times n} = \sum_{i=1}^{n} vsms\_representation[i, :] \ if \ tag_i = male \quad (10)$$

$$C^{female}_{1\times n} = \sum_{i=1}^{n} vsms\_representation[i, :] \ if \ tag_i = female \quad (11)$$

$$vsms\_representation \, \epsilon \left\{U^{dtm}_{n\times n}, U^{tfidf}_{n\times n}\right\} \quad (12)$$

4.3 Feature Learning

The variation between the classes and the author's texts are computed by measuring the distance and correlation between the class vectors (male and female) and the vector representation of the author's text. We have considered correlation, cosine distance and Euclidean distance for measuring variation. This is given as follows,

$$F_{n\times 6} = feature\_learn\left(representation, C^{male}, C^{female}\right) \quad (13)$$

$$representation \, \epsilon \left\{D^{dtm}_{n\times m}, D^{tfidf}_{n\times m}, U^{dtm}_{n\times n}, U^{tfidf}_{n\times n}\right\} \quad (14)$$

$$C^{male} \, \epsilon \left\{C^{male}_{1\times m}, C^{male}_{1\times n}\right\} \quad (15)$$

$$C^{female} \, \epsilon \left\{C^{female}_{1\times m}, C^{female}_{1\times n}\right\} \quad (16)$$

5 EXPERIMENTS AND OBSERVATIONS

From the given corpus, the author's texts are represented as the matrix as shown in Section 4.1. For DTM and TF-IDF representation, the CountVectorizer[6] and TfidfVectorizer[7] modules from scikit-learn python library are used. SVD applied on this computed matrices and the basis matrices (U) alone are kept for further processing. SVD performed by using the numpy python library[8].

As given in Section 4.2, the class vectors are computed from the author's text vectors. As given in Section 4.3, the feature matrix

[5] http://en.rusprofilinglab.ru/rusprofiling-at-pan/

[6] http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

[7] http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

[8] https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.linalg.svd.html

Corpus	Tweets	Offline	Facebook	Tweets	Reviews	Imitation
Total # authors	600	370	228	400	776	94

Table 1: Corpus Statistics

Corpus/ Run	Train CV	Offline	Facebook	Tweets	Reviews	Imitation
Run1	0.69	0.45	0.49	0.45	0.50	0.45
Run2	1.0	0.49	0.54	0.50	0.50	0.50
Run3	1.0	0.47	0.51	0.46	0.50	0.40
Run4	1.0	0.50	0.52	0.50	0.52	0.50
Run5	0.66	0.50	0.49	0.50	0.50	0.50

Table 2: Results

computed by measuring the variation between the author's text vectors and class vectors. The scipy python library used to compute the cosine distance, Euclidean distance, and correlation measures.

The computed feature matrix along with the target classes fed to SVM with a linear kernel to perform the final classification. SVM from the scikit-learn python library[9] used for classification. In order to observe the training performance, we have computed the 10-fold cross validation score and given in Table 2 (Train CV). This is given as follows,

$$Accuracy = \frac{correctly\ predicted\ short\ texts}{total\ \#\ short\ texts} \tag{17}$$

$$Train\ CV = \frac{\sum_{i=1}^{10} Accuracy_i}{10} \tag{18}$$

Similar to the training corpus, the feature matrix computed for the five test corpus given by the shared task organizers and prediction for the author's texts are found by using the model built in the training period. We have totally submitted the five runs and those details are given below,

- Run1: TFIDF -> Class Vectors -> Feature Learning -> Classification
- Run2: DTM -> Class Vectors -> Feature Learning -> Classification
- Run3: TFIDF -> SVD -> Class Vectors -> Feature Learning -> Classification
- Run4: DTM -> SVD -> Class Vectors -> Feature Learning -> Classification
- Run5: DTM -> Classification

The results reported by the shared task organizers for the submitted five runs are given in Table 2. Out of five runs, Run2 performed better and attained 51.45% as the accuracy measure for the concatenated test corpus. Run5 attained 47.06% as accuracy measure for the concatenated test corpus and this ensures that class embedding enhanced the accuracy by 4%.

6 CONCLUSION

The given train corpus and the test corpus are represented as the author's text vectors by using methods available in Vector Space Model and Vectors Space Models of Semantics. Class vectors are computed from the author's text vectors and variation between the class vector and the author's text vectors are taken as the features to perform the Support Vector Machine based classification. The preliminary results showed that class vectors based classification improve the accuracy by nearly 4% for the final concatenated test corpus. There has been a large performance variation between the cross-validation score and the score against test corpus. Hence our future work will be focused more on reducing this margin and computing the class vectors though distributed representation methods.

REFERENCES

[1] Barathi Ganesh H. B., M. Anand Kumar, and K. P. Soman. 2016. Statistical Semantics in Context Space : Amrita_CEN@Author Profiling. (2016), 881–889.

[2] Barathi Ganesh H. B., M. Anand Kumar, and K. P. Soman. 2017. Vector Space Model as Cognitive Space for Text Classification. *CoRR* abs/1708.06068 (2017). arXiv:1708.06068 http://arxiv.org/abs/1708.06068

[3] Tatiana Litvinova, Francisco Rangel, Paolo Rosso, Pavel Seredin, and Olga Litvinova. [n. d.]. Overview of the RUSProfiling PAN at FIRE Track on Cross-genre Gender Identification in Russian. In *Notebook Papers of FIRE 2017, FIRE-2017, Bangalore, India, December 8-10, CEUR Workshop Proceedings.*

[4] Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian. 2017. A Report on the 2017 Native Language Identification Shared Task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications.* 62–75.

[5] Francisco Rangel, Paolo Rosso, Martin Potthast, and Benno Stein. 2017. Overview of the 5th author profiling task at pan 2017: Gender and language variety identification in twitter. *Working Notes Papers of the CLEF* (2017).

[6] Francisco Rangel, Paolo Rosso, Ben Verhoeven, Walter Daelemans, Martin Potthast, and Benno Stein. 2016. Overview of the 4th author profiling task at PAN 2016: cross-genre evaluations. *Working Notes Papers of the CLEF* (2016).

[7] Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research* 37 (2010), 141–188.

[9] http://scikit-learn.org/stable/modules/svm.html