Multilingual Author Profiling from SMS

Deepanshu Gaur[1], Meghna Ayyar[2], Ashutosh Kumar Singh[3], and Rajiv Ratn Shah[4]

[1] Maharaja Agrasen Institute of Technology,New Delhi 110086, INDIA
deepanshugaur1998@gmail.com
[2] Indraprastha Institute of Information Technology, Delhi 110020, INDIA
leomi7ayyar@gmail.com
[3] Delhi Technological University, Delhi 110042, INDIA
ashu0788@gmail.com
[4] Indraprastha Institute of Information Technology, Delhi 110020, INDIA
rajivratn@iiitd.ac.in

Abstract. This paper presents our solution for the Author Profiling task in the FIRE challenge 2018. This task mainly focuses on finding the age and gender of people from South Asian countries such as India, Pakistan, Nepal, and Bangladesh from their short messaging services (SMS). Since most of these people use a combination of languages such as Hindi, English and Roman Urdu (i.e., multilingual text) on social media platforms such as WhatsApp, Facebook, and Twitter, they also follow the same practice in SMS while communicating. Thus, we aim to perform author profiling by identifying gender and age of people by analyzing their multilingual SMS. In this paper, we classify the gender of a person into male or female categories. Moreover, we classify age into the following three age groups: (i) 15-19, (ii) 20-24, and (iii) above 25. After preprocessing steps including tokenization and normalization, we provide the results of an experiment with several machine learning models like SVM, Random Forest, and Naive Bayes. Experimental results show that Naive Bayes provides competitive results when used with bilingual dictionary for translation and count vectorizer for feature extraction.

Keywords: Multilingual Corpus · Naive Bayes · SMS · Author Profiling

1 Introduction

The ubiquitous availability of smartphones and affordable network infrastructure has helped social media in increasing its popularity dramatically, clearly seen in Fig.1 taken from [26]. Often people share their thoughts, ideas, opinion, feedback, and sentiments on social media platforms such as blogging websites, Twitter, WhatsApp, YouTube, and Facebook. Social media has helped in bridging the gap between different communities. Often many people use multilingual text on social media because (i) they feel more comfortable to express themselves in local languages, (ii) they have some affinity towards their native languages, (iii) they want to increase their reachability to even those people who cannot understand

English well, and (iv) probably, they themselves might not be very good at expressing their thoughts via English. Social media platforms understand this need for using multilingual text in communication. Thus, they allow users to choose their preferred language from a list of languages that are used around the world.

Authorship Profiling(AP) task, which is predicting the authorship of a text only by extracting the linguistic and stylistic features, has a number of potential applications. For instance, it has been a pertinent task to the intelligence and security agencies that monitor author information. Moreover, author details can be of utmost importance in cyber forensics for identifying fake profiles, fraud messages and social harassment etc. For companies focused on marketing, this information will be definitely helpful in providing clear understanding of the target audience, solely by using blogs and reviews as a source.

Much work has been done on author profiling in languages such as English and other European languages like German, French, Spanish etc. However, a very limited research attention has been paid to South Asian languages such as Roman Urdu, Hindi etc. For example, in Hindi we say "आपसे मिलकर अच्छा लगा" which we can write using the English script as "Aapse milkar acha laga" which translates to "Nice to meet you". While, in Urdu [15], we say "آپ سے ملکر خوشی ہوئی" which we can write with the English alphabets (Roman Urdu) as "Aapse milkar Khushi hui" which translates to "Pleased to meet you". Translations like these coupled with analysis of the patterns seen in the common text can prove beneficial in profiling the South Asian users too.

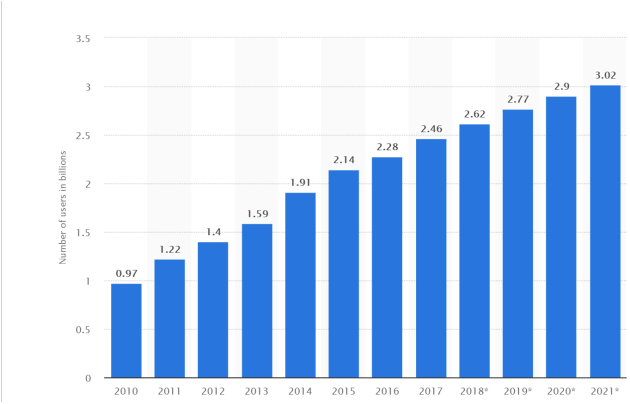

Fig. 1. Increasing number of social media users

The article [26] shows a worldwide increase in the number of people who use mobiles and smartphones to connect to the social media platforms. The following statistics reported in the article [1] is very interesting but at the same time very disturbing too. A huge number of user profiles on Facebook are reported fake. This necessitates that, appropriate action should be taken to address the increasing number of the fake profile. It is therefore essential to develop auto-

matic tools and techniques for the detection of fake profiles from different types of texts like Facebook posts/comments, SMS texts, Twitter tweets, blogs posts etc. AP is one technique that acts as a starting step for the detection of fake user profiles.

The rest of the paper is organized as follows. Sections 2 and 3 describe related work and methodology, respectively. Evaluation is presented in Section 4. Finally, Section 5 concludes the paper.

2 Related Work

The findings of Martine et al. [13] suggested that in author profiling tasks done before, the texts used were mostly monolingual as is the case provided by BNC [4], where they used only parts of the British National Corpus for their work. Pardo et al. [18] provided an overview about author profiling, its uses, and how the traits apart from gender and age, like personality type etc. could also be used as author aspects. There has been a growing interest in including a multitude of dialects and languages for author profiling, with PAN including different languages in their shared tasks. Different dialects that were explored are Portuguese varieties Zampieri et al. [28, 29], Castro et al. [2], English varieties Lui, Cook [10], Romanian dialects Ciobanu, Dinu [3] and Chinese varieties Xu et al. [27].

For features to be used for the task, simple n-gram models or using word n-grams have been observed to be effective in getting better results than character n-grams as proved by Maharajan et al. [11]. The major drawbacks of Maharajan et al. [11] approach was that they analyzed only monolingual text. Estival et al. [5] showed how author profiling tasks could be implemented for texts pertaining to emails also. But both these studies did not include the use of multilingual text, which is the most preferred way of communication among most of the people. Rangel et al. [16] pointed out that stylistic features could be useful in learning demographic traits. However, this suffered from a shortcoming that the results of stylistic features did not generalize for other traits like gender and thus have been ranked lower in preference while performing classification of gender for profiling. Many papers Zampieri et al. [28, 29], Castro et al. [2], Lui, Cook [10], Ciobanu, Dinu [3], Xu et al. [27] have focused so far on using monolingual text only, which encouraged us and many others to pursue further research in multilingual author profiling.

Gonzalez et al. [7] showed that POS and n-grams have been useful for the extraction of stylistic features. Unlike the previously mentioned papers, Gonzalez et al. [7] have used multilingual text for their study. Multilingual author profiling done by Bayot, Goncalves [1] have shown that word vectors outperform TF-IDF when used with SVM and obtained best results on a dataset containing both Spanish and English text.

Research in the area of author profiling for less explored languages played a huge role in motivating different researchers to pursue their study further in such languages and one such study Kapociute et al. [9] on the Lithuanian literary

texts shed light on some results that were significant. They found that it was easy to determine gender and age when using literary texts than with parliamentary scripts. Rangel et al. [17] on PAN-AP 2016 task focused on cross-genre evaluation. The participants for the task were given different genres for training, early bird and testing, and the subcorpus for this contained three different languages, English, Spanish and Dutch. The final result was that most of the teams had scored below the baseline for Dutch but were significantly better for English and Spanish. This points to the fact that the profiling task itself heavily depends on the nature of the language being considered. For the social media corpus that was provided in the same task, they concluded that there is no such impact of the cross-genre evaluation on language such as English. However, for a language such as Spanish, there is a much greater impact on doing this joint evaluation for age identification task but not so much for gender classification. Another study by Rangel et al. [12] on cross-genre evaluation described some notable approaches in the field of cross-genre in author profiling. The results obtained varied for different languages with different models.

Most of the studies in the field of author profiling Zampieri et al. [28, 29], Castro et al. [2], Lui, Cook [10], Ciobanu, Dinu [3], Xu et al. [27] has been restricted to the European languages, Middle East, Japanese, and Chinese. However, a limited work has been done on South Asian languages such as Roman Urdu, Hindi, and other regional languages of the South Asian countries.

3 Methodology

The problem at hand is to use multilingual text collected from SMS to perform author profiling by classifying gender and age traits. The following sections describe in detail the various steps that have been performed to achieve the results. Section 3.2 describes the pre-processing steps, followed by the Section 3.3 which discusses feature extraction techniques applied. Finally, Section 3.4 deals with model selection to achieve the best results.

3.1 System Architecture

The complete architecture of the proposed system has been summarized in Fig. 2. The pipeline starts with the collection of data and its subsequent preprocessing. This is followed by the process of feature extraction which has been described in detail in Section 3.3. Once the features were extracted as vectors, we experimented with various models and selected the one which performs best by employing cross validation and then testing the model by predicting labels on the unseen data.

3.2 Data Preprocessing

Since the data given to us is raw, noisy and also prone to more errors, it cannot be directly used for analysis. It is necessary to perform some preprocessing to

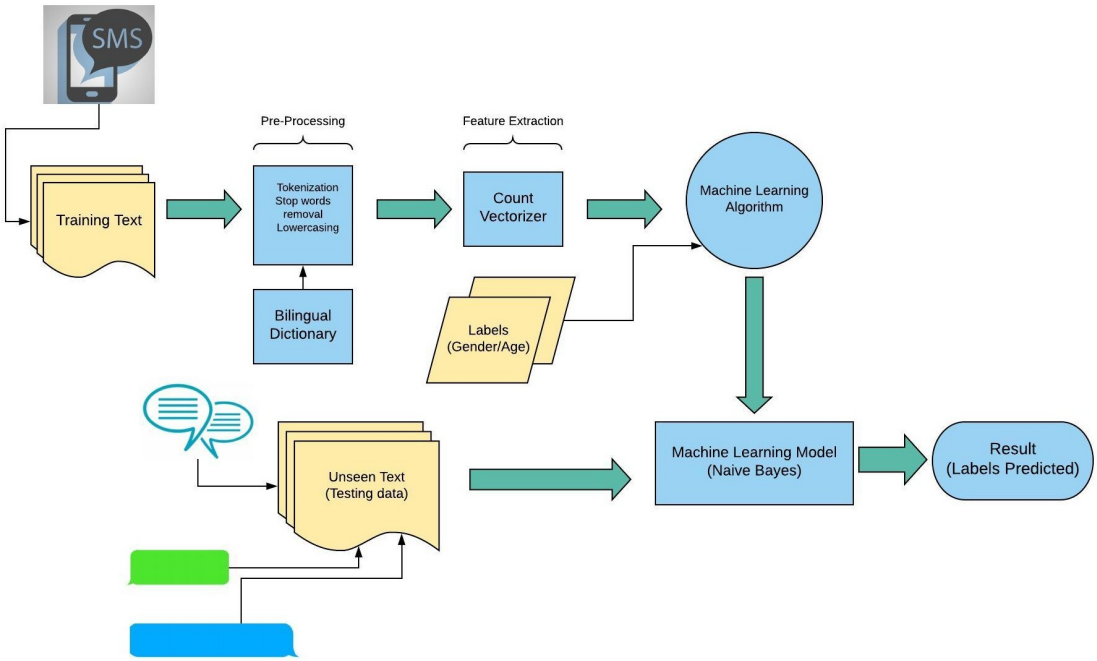

Fig. 2. System architecture of our proposed solution

make the data more suitable so that we can extract some information from it by feeding it to our model. The following preprocessing steps were performed:

1. **Tokenization:** Tokenization refers to the breaking down of the given text into individual words and symbols called as tokens. The words, phrases, symbols, and other characters present in a text are the basic characteristics of a document or in our case the SMS. Hence we use the NLTK library's word tokenizer to perform tokenization of the corpus. Consider an example for a tokenization of a text like "Natural language processing is fun" will give out different tokens such as 'Natural', 'language', 'processing', 'is' and 'fun'. People may use word tokenization or sentence tokenization depending upon their needs.
2. **Normalization:** Normalization is another type of preprocessing step which is performed to transform the data into a consistent form. It involves a number of techniques that ensure that the text is converted into a canonical form. We perform the following steps to normalize our corpus:
   - **Stopwords removal:** Many words like *the, is, are etc.* occur frequently in all the text. They do not convey any new meaning and are essentially used just to join words and sentences. They are not necessary to derive information from the text and hence are removed from our corpus by using the stopwords package from the NLTK library.

- **Punctuation removal:** Punctuation marks, similar to stopwords do not convey special meaning. Hence they are also discarded from the texts using NLTK library.
- **Text case conversion** We convert all the text to lower case before storing (using .lower() method in python). Capital letter and small letters of the same alphabets convey similar meaning and hence the conversion is necessary to ensure that they both are not treated as separate entities of the text under consideration. For instance, if we consider a text like "Apple is Red" since the previous text will also convey the same meaning as “apple is red” we may change the case of all the words to lower/upper so as to normalize the data.

3.3 Text Feature Extraction

After creating tokens, extracting features is an important task that needs to be performed before using our model so that we can predict from texts. Each token after creating vectors will act as a feature for our model.

For extracting relevant features from the text we have used two major feature extraction techniques namely TF-IDF [25] and count vectorizer [24]. TF-IDF gives a numerical value to the importance of each word in a particular text whereas count vectorizer gives us a document-term matrix which is then used as a feature for each document or in other words it counts the number of time a word is appearing in a text. The theory behind using count vectorizer is that the more frequent a word or token is the more central or important it might be to the text we used. It is an effective way to determine significant words in a text based on the number of times they are used. It is observed that word vectors perform the best in each of the classification tasks that we were given. Thus, we choose count vectorizer as our final text feature extraction technique.

3.4 Model Selection

We are given the task of classifying the author's gender and age from his/her text and thus this task falls into the category of supervised learning. For gender identification, we have binary classification task of predicting labels as either male or female. For the age identification, we have a multi-classification task whose goal is to predict age among three classes: 15-19, 20-24, 25 and above.

We use cross-validation with a ratio of 80:20 for the training and testing dataset. While considering different models a cheat sheet [23] provided by scikit learn is also considered. After analyzing the problem we conclude that the model to be used must fall into the category of supervised learning thus we choose models that are part of the same. Since the dataset given to us is not large we select Random Forest, Linear Support Vector Machines (SVM) and Logistic Regression. Also considering a fact that dataset given to us is in the form of text we opt for Naïve Bayes model as well. Following models from Scikit-Learn [19] library are selected :

1. Random Forest
2. Linear Support Vector Machines (SVM)
3. Naive Bayes
4. Logistic Regression

4 Evaluation

Section 4.1 gives a description of the data while Section 4.2 provides insight into the experimental setup used. Section 4.3 describes the evaluation metrics used and Section 4.4 gives details about the results[5] obtained and also provides some analysis of the same.

4.1 Dataset Description

Our dataset contains the multilingual text of Roman Urdu and English in different text files. We classify gender as one of the two given classes, male or female and similarly, for age, we used three categories: 15-19, 20-24, 25 and above.

Table 1. Description of training dataset

FileName	Gender	Age Group
author_id_001	male	25 and above
author_id_002	female	15-19
author_id_003	male	20-24
author_id_004	female	25 and above
author_id_005	male	15-19

There are 350 text files and each contains a SMS written in Roman Urdu or English. Some of the messages from the dataset looks something like : "Abhi tak to ni hai", "Check kr rahe hain ameer sahab" and "Aa ni rahe?". A separate CSV file is also provided which contains all the text files. The details have been summarized in the Table 1. Apart from the training corpus, we are also provided with the testing corpus to evaluate our final results that contained 150 text files. The task is, therefore, to predict the gender and age corresponding to AuthorId and also predict the gender and age individually.

4.2 Experimental Setup

This section describes the approaches and resources employed for model selection to obtain best results for the task.

[5] Source code is made available in downloads folder in repository at `https://bitbucket.org/deep1998/author-profiling-on-sms/downloads/`

1. **Resources Employed:** We use a bilingual dictionary [14] that contains 7189 commonly used words in Hindi and Urdu which can be used to translate Roman Urdu words into English.
2. **Model Selection Results:** Different models like SVM, Random Forest and Naive Bayes are tested on the data with different parameter combinations and the results are summarized in Table 2. The random forest model performed lower (accuracy=0.78 for gender and 0.55 for age) as compared to Naive Bayes in terms of accuracy with parameters : n_estimators=1000(for gender),n_estimators=1000(for age), criterion='gini', max _depth=None, min_samples _split=5, min_samples_leaf=1(for gender) and min_samples_leaf=1 (for age).

Table 2. Accuracy results of different models on the holdout set

Model	Gender	Age
Random Forest	0.78	0.55
SVM	0.70	0.52
Naive Bayes	**0.87**	**0.60**
Logistic regression	0.82	0.54

Next, we use SVM, which performs badly on the dataset and rank lowest on gender with accuracy = 0.70 with parameters C=1.0, kernel='rbf', degree=3, gamma='auto', coef0=0.0, shrinking=True, probability=False. Accuracy is 0.52 for age and thus perform badly as well on age classification task. Finally, we use logistic regression model with parameters: penalty='l2', dual=False, tol=0.0001, C=1.0(for gender), C=1.0(for age), fit _intercept=True, intercept _scaling=1 which gets an accuracy of 0.82 for gender and a value of 0.54 for age and both values are close to the results obtained by Naive Bayes model. Out of all the models used we get best results with the Naive Bayes classifier(accuracy of 0.87 and 0.60 for gender and age respectively), which is used with the parameters: alpha=1.0, fit_prior=True, class_prior=None. For both age and gender classification task we use the same Naive Bayes classification model to evaluate the performance.

4.3 Evaluation Measures

Accuracy [8] has been used as the metric for evaluating classification models in our experiments. Formally, accuracy has the following definition which says accuracy is the ratio between a total number of correct predictions N_c upon total number of predictions N_t :

$$\text{Accuracy} = \frac{N_c}{N_t}$$

4.4 Result and Analysis

We tested the accuracy of our Naive Bayes classifier for both the tasks on both the datasets : (i) Official dataset given for training (ii) Official test dataset (only for testing). The accuracy for the gender classification task was 0.87 and the accuracy for the age classification task was 0.60 on the training dataset(Table 2). The accuracy on the test dataset for the gender classification task was 0.75 and for the age classification task was 0.64 and jointly for both the accuracy was found to be 0.47(see Table 3).

Table 3. Accuracy on the official test corpus using Naive Bayes technique

Naive Bayes Model	
Task	Accuracy
Gender	**0.75**
Age	0.64
Jointly	0.47

The model performed its best on the gender classification task achieving competitive accuracy and gave a bit lower accuracy for the age classification task. By comparing the test results with training results, we found that the results were better on test dataset for age classification task than on training dataset which might signify that the model might generalize well for the age classification task as it performs well on unseen data. In contrast, this also points out that for gender classification the model could be overfitting on the training dataset which has lead to a drop in the accuracy of the testing dataset.

Table 4 summarizes the final results of other teams that participated in the FIRE'18 task. We were ranked 4th in the competition with a joint accuracy of 0.47. Highest accuracy was achieved by the first team whose joint accuracy was 0.57. The lowest joint accuracy achieved in the competition was 0.23. The baseline scores for joint accuracy was 0.32 and our results like above this base limit, showing that the model makes progress from the current state of the art and perform efficiently.

5 Conclusion and Future Work

To conclude our study, we say that we have presented our approach to an intriguing task of FIRE'18, the Multilingual Author Profiling on SMS. Previous research work [21, 20, 22] have also been taken into consideration while deciding the best approach for the task. Detailed preprocessing techniques with their proper methodologies as well as other approaches such as using bilingual dictionaries have been discussed in a lucid manner. Lastly, we have presented the results achieved on the official FIRE 18-Multilingual Author Profiling on SMS test set. The best results for the gender and age classification tasks using accuracy as

Table 4. Final results of all teams in the competition

Teams	Gender	Age	Joint
Sharmila Devi et al.	0.87	0.65	0.57
D. Thenmozhi et al.	0.85	0.63	0.52
Ali Nemati	0.83	0.60	0.49
Deepanshu Gaur et al.	***0.75***	***0.64***	***0.47***
Dijana Kosmajac et al.	0.74	0.59	0.43
Òscar Garibo	0.77	0.57	0.43
Ramsha Imran et al.	0.73	0.53	0.38
Asmara Safdar et al.	0.69	0.53	0.35
Abdul Sittar et al.	0.55	0.37	0.23

an evaluation measure were achieved with values 0.75 and 0.64 respectively and for jointly value was 0.47. The results of the competition made our team stand at fourth position globally in terms of joint accuracy, **second** in age classification and fourth for gender classification in the FIRE'18.

We conclude from our study that, to identify age and gender from a multilingual corpus, using a bilingual dictionary can be an efficient method to translate Roman Urdu to English as supported by Fatima et al. [6]. Though using dictionary is an efficient way but some drawbacks of using dictionary has also been observed. For instance, if words in the dictionary are not translated properly it may affect the results. Moreover, different writing styles in messages nowadays can cause a problem in understanding meaning of the sentences and may affect the results. For example, people may write the word *Please* as *Plz* or *Pls* and word like *Good* as *gud* and in many other ways as well.

We aim to extend the model by making it more efficient by using different techniques we did not explore such as using other features like POS or n-grams, combined with the ones we already tried. So far the text contained two languages but in the future, it would be beneficial to include more South Asian languages as they are relatively less explored and contain potential to be very useful.

References

1. Bayot, R., Gonçalves, T.: Multilingual author profiling using word embedding averages and svms. In: Software, Knowledge, Information Management & Applications (SKIMA), 2016 10th International Conference on. pp. 382–386. IEEE (2016)
2. Castro, D.W., Souza, E., Vitório, D., Santos, D., Oliveira, A.L.: Smoothed n-gram based models for tweet language identification: A case study of the brazilian and european portuguese national varieties. Applied Soft Computing **61**, 1160–1172 (2017)
3. Ciobanu, A.M., Dinu, L.P.: A computational perspective on the romanian dialects. In: LREC (2016)
4. Corpus, B.N.: Bnc home page, `http://www.natcorp.ox.ac.uk/`
5. Estival, D., Gaustad, T., Pham, S.B., Radford, W., Hutchinson, B.: Author profiling for english emails. In: Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics. pp. 263–272 (2007)

6. Fatima, M., Hasan, K., Anwar, S., Nawab, R.M.A.: Multilingual author profiling on facebook. Information Processing & Management **53**(4), 886–904 (2017)
7. González-Gallardo, C.E., Torres-Moreno, J.M., Rendón, A.M., Sierra, G.: Efficient social network multilingual classification using character, pos n-grams and dynamic normalization. arXiv preprint arXiv:1702.06467 (2017)
8. Google: Machine learning crash course, `https://developers.google.com/machine-learning/crash-course/classification/accuracy`
9. Kapočiūtė-Dzikienė, J., Utka, A., Šarkutė, L.: Authorship attribution and author profiling of lithuanian literary texts. In: The 5th Workshop on Balto-Slavic Natural Language Processing. pp. 96–105 (2015)
10. Lui, M., Cook, P.: Classifying english documents by national dialect. In: Proceedings of the Australasian Language Technology Association Workshop 2013 (ALTA 2013). pp. 5–15 (2013)
11. Maharjan, S., Shrestha, P., Solorio, T.: A simple approach to author profiling in mapreduce. In: CLEF (Working Notes). pp. 1121–1128 (2014)
12. Markov, I., Gómez-Adorno, H., Sidorov, G., Gelbukh, A.F.: Adapting cross-genre author profiling to language and corpus. In: CLEF (2016)
13. Martinc, M., Škrjanec, I., Zupan, K., Pollak, S.: Pan 2017: Author profiling-gender and language variety prediction. Cappellato et al.[13] (2017)
14. Mathur, P., Shah, R., Sawhney, R., Mahata, D.: Detecting offensive tweets in hindi-english code-switched language. In: Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media. pp. 18–26. Association for Computational Linguistics (2018), `http://aclweb.org/anthology/W18-3504`
15. Omniglot: Omniglot urdu phrases, `https://www.omniglot.com/language/phrases/urdu.php`
16. Rangel, F., Rosso, P.: Use of language and author profiling: Identification of gender and age. Natural Language Processing and Cognitive Science **177** (2013)
17. Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., Stein, B.: Overview of the 4th author profiling task at pan 2016: cross-genre evaluations. In: Working Notes Papers of the CLEF 2016 Evaluation Labs. CEUR Workshop Proceedings/Balog, Krisztian [edit.]; et al. pp. 750–784 (2016)
18. Rangel Pardo, F.M., Celli, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd author profiling task at pan 2015. In: CLEF 2015 Evaluation Labs and Workshop Working Notes Papers. pp. 1–8 (2015)
19. Scikit: Scikit learn home page, `http://scikit-learn.org/stable/`
20. Shah, R., Zimmermann, R.: Multimodal analysis of user-generated multimedia content. Springer (2017)
21. Shaikh, A.D., Jain, M., Rawat, M., Shah, R.R., Kumar, M.: Improving accuracy of sms based faq retrieval system. In: Multilingual Information Access in South Asian Languages, pp. 142–156. Springer (2013)
22. Shaikh, A.D., Shah, R.R., Shaikh, R.: Sms based faq retrieval for hindi, english and malayalam. In: Post-Proceedings of the 4th and 5th Workshops of the Forum for Information Retrieval Evaluation. p. 9. ACM (2013)
23. sklearn: cheat-sheet, `http://scikit-learn.org/stable/tutorial/machine_learning_map/index.html`
24. sklearn: Count-vectorizer, `http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html`
25. sklearn: tf-idf, `http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html`
26. Statista: Statistics page, `https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/`

27. Xu, F., Wang, M., Li, M.: Sentence-level dialects identification in the greater china region. arXiv preprint arXiv:1701.01908 (2017)
28. Zampieri, M., Gebre, B.G.: Automatic identification of language varieties: The case of portuguese. In: KONVENS2012-The 11th Conference on Natural Language Processing. pp. 233–237. Österreichischen Gesellschaft für Artificial Intelligende (ÖGAI) (2012)
29. Zampieri, M., Malmasi, S., Sulea, O.M., Dinu, L.P.: A computational approach to the study of portuguese newspapers published in macau. In: Proceedings of Workshop on Natural Language Processing Meets Journalism (NLPMJ). pp. 47–51 (2016)