International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 11 (2019) pp. 115-123
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Received: 5 Jan, 2019; Accepted: 10 March, 2019; Published: 7 May, 2019

A Machine Learning Approach to Arabic Native Language Identification

Seifeddine Mechti, Lamia Hadrich Belguith

Faculty of Economics and management of Sfax, Tunisia
MIRACL Laboratory,Route de Tunis Km 10 B.P. 242 SFAX 302
mechtiseif@gmail.com
l.belguith@fsegs.rnu.tn

***Abstract*: Native Language Identification (NLI) is the task of identifying a writer's native language (L1) based only on their writings in a second language (the L2). This Paper presents a method for the identification of the learners of the Arabic language (ANLI). The contribution of our method revolves around the use of the standard deviation for the optimization of supervised learning. This technique explores a multitude of linguistic features extracted from the text. The feature selection stage allowed improving the results that outperformed those achieved by the best systems applied on the same corpus. The achieved accuracy outperformed that of the state-of-art (45% vs 41%), taking into account the limited data and the unavailability of accurate tools dedicated to the Arabic language**.

Keywords: Native language identification, standard deviation, machine learning**.**

I. Introduction

The rapid growth of the Internet and computer technology in the last two decades has resulted in an incredible increase of the online data amount. According to 'Internetlivestats1, in one second the Internet traffic2 is about 36,411 GB. This impressive amount of data -mostly of text type- is shared, published, and transferred in a free (and sometimes anonymous) way; in fact, an important portion of internet users are misrepresenting themselves while surfing on the Net; therefore, there is a need to deal with the data which sources are unknown.

Two main sectors are interested in knowing the potential source of data, first, the commercial sector where information such as age, sex, nationality, and native language about customers is of a high value for marketing intelligence, second, the security sector that bears the burden of protecting the Internet from crimes such as plagiarism and identity theft, etc. Therefore, the research community promotes researchers to discover and develop effective methods and techniques in related fields such as plagiarism detection and author profiling.

This work was achieved in the context of author profiling as a vibrant field of research that cares about the detection of author background characteristics, e.g. the age, gender, native language and even personality of the individual who produced the text. In particular, we focused on identifying the native language as one of these characteristics. It falls in the crossroads of text classification and Computational Linguistics, where linguistic patterns from texts are used to predict the author's first language.

II. Related works

Native Language Identification tackles the problem of determining the native language of an author based on a text that the author has written in a second language [1] This definition highlights the fact that native language identification (NLI) is the science of automatically identifying the first language (L1) of an unknown author on the basis of their production patterns in the target language (L2). It is based on the fact that people unconsciously use patterns from their L1 when they express ideas in L2, even after years of learning and exposure. This problem typically falls at the crossroads of text classification and Computational Linguistics where linguistic patterns of the text are used to predict the author's first language.

While NLI is a new line of inquiry, the technical problems of NLI are often solved by resorting to machine learning algorithms where a number of patterns is extracted from multilingual corpora and used in generating classification models able to predict the L1 of a new text.

Although research in the text classification field offered a wide range of machine-learning algorithms [2] to experiments in NLI studies, which include Support Vector Machine (SVM), Decision Tree algorithm (DT), Naive Bayes algorithm (NB), it is argued that the choice of the right features is by far more important than the choice of classification techniques [3][4]. Therefore, our analysis relied on feature types to distinguish between different approaches.

When examining the previous studies, a multiplicity of information on different types has been explored individually and combined in different ways in order to determine the common transferred patterns of authors that share the same native language. The feature type sets involve, among others, three main classes, namely lexical, syntactic and n-gram features.

A. Lexical features

The earliest and most traditional features are the lexical ones where texts are viewed as a sequence of sentences composed of words and punctuation marks.

The Lexical features include character-based, vocabulary-richness, and word-length-frequency features. Seeing that the choice of the appropriate words and vocabulary used by authors in their writing is influenced by their mother tongue, the lexical-dependent features are considered as good indicators of language transfer. Such features proved their usefulness in the classification context, e.g. in automatic Arabic author attribution [5], in Arabic email-author profiling [6], and in native language prediction [7] where experimenting various types of features collected from Web-based texts showed the obvious efficiency of such lexical features (use of just 64 features) compared to other features.

Since the most available data contain topic bias, the lexical features were avoided in many studies to get around the topic influence. However, an important discovery was noted: Brooke and Hirst [8] argued that topic bias affects not only lexical features, but also non-lexical ones, and that avoiding lexical features was pointless. Regarding this point, they presented an optimistic lexical-based approach enabling their system to reach an accuracy of 95.2 %. Despite this positive achievement, researchers are still exploring other feature types such as information related to syntax.

B. Syntactic features

In linguistics, syntax –the study of phrase and sentence structures [9]- covers such features as the way words are constructed, the way the endings of words change according to context (inflection), the classification of words into parts of speech (nouns, verbs, pronouns, etc.), and the way parts of speech are connected together.

This feature type was first investigated by Wong and Dras[10]. It has been proposed that this feature consists of more topic-independent and unconscious markers of the native language for the simple reason that they are rather dependent on the target language than on the textual content. Thus, it is considered to be more reliable than lexical patterns. However, its use requires the availability of some kind of Natural Language Processing tool, like a Part-of-Speech Tagger and a Syntactic Parser.

C. N-gram features

N-gram features are widely used in authorship analysis; these features are defined by relying on the two previous feature types in order to extract additional information by going beyond what can be found when examining features individually.

The use of n-grams has many advantages:

- N-grams are easy to compute and are extracted in a systematic way in contrast with the syntactic and lexical features that need specific tools such as dictionaries, taggers and parsers.
- Spelling and orthography Error Tolerance: n-grams perform well in data that contain errors.
- N-grams are language independent: in the case of Chinese and similar languages where no spaces between words are left, the use of character and word n-grams proved its efficiency.

III. Relevant studies

NLI, the focus of our study, is typically modelled as a sub-class of text classification, more specifically, text author profiling. To date, this topic has attracted the interest of several papers and research projects. In this section, we described the most important studies found in the literature classified by L2 while taking into account the corpora, the features, the used classifiers and the obtained results.

A. English native language identification

English, the first world language, has been widely studied in Second-Language Acquisition SLA and Natural Language Processing NLP. Therefore, since [11], the first NLI study, most of the works dealing with automatic native language prediction have focused on English.

In their study, Shlomo et al. [12] dealt with 'native language' among other dimensions including age, gender and personality in their authorship-profiling approach using the Bayesian Multinomial Regression (BMR) as a learning algorithm. Because of the unavailability of a comprehensive corpus, they resorted to the use of a separate corpus to accomplish their studies. They used the International Corpus of Learners of English ICLE3 for the sub-task of language-dimension identification. Their cross-validation tests were carried out on each of the following feature types separately (stylistic features only, content features only) and on both of them. The results of their studies showed that the content-based feature is slightly useful in the gender-dimension identification. Indeed, when combined with the style-based feature, it gave a classification accuracy of 76% for gender and 77.7 % for age. In contrast, the classifiers that were learned using only style features provided an accuracy rate of 63% for personality dimensions.

Concerning the language dimension, an outperformance of 82.3% was achieved using only the content features which reflects the preference for a specific word usage among speakers of different languages. However, such results were discussed by the researchers themselves who noted that they might be infected by possible topic bias. For this reason, the majority of the subsequent studies avoided the use of content features.

Wong and Dras [13] conducted their experiments on the same data (i.e. ICLE). They integrated three common syntactic types of error made by English learners. These mistakes are related to subject-verb disagreement, noun-number disagreement and misuse of determiners with the lexical features, function words, character n-grams and POS n-grams, used in previous approaches. They achieved an accuracy of 73.71% with all features combined. The results showed the usefulness of these error types in the NLI task. This fact was further investigated by E.Kochmar[14], who suggested the use of character n-grams errors.

Although most English NLI studies were conducted on ICLE, other corpora have been used in parallel. Serhiy and Detmar [1] conducted a research on the second version of the ICLE corpus and three other corpora (NOCE4, USE5 and HKUST6 in this case). They used different Support Vector Machines (SVMs) as classifiers. The researchers defined features based on recurring n-grams of all accruing lengths. Three classes of recurring n-grams are defined in their work, (viz.): one word-based n-gram and two generalizations of the first class (POS- and OpenClass-POS- based n-grams). They conducted experiments based on random samples from ICLE in a single-corpus evaluation and a cross-corpus7 evaluation. The word-based n-gram was proved to be the best performing class with a high accuracy reaching 89.71%. The results also demonstrated that the pattern learned on ICLE generalized well across corpora and gave an accuracy of 88%.

In another study, [15], Xiao et al. used a large longitudinal data to identify the native language of the learners of a second language. They focused mainly on a corpus collected from Cambridge University to investigate the different proficiency levels. The authors used an accurate learning machine based on Support Vector Machines SVM. Both syntactic and lexical features were tested separately and combined in this experiment. The results showed that lexical features outperformed the syntactic ones when tested individually at all levels. The same findings were achieved when combining the features, especially at an advanced level.

In [16], S. Nisioi investigated the proficiency of the different features for the task of native-language identification benefiting from the EF Cambridge Open Language error-annotated Database. Nisioi's objective was firstly to analyse the different features used for automatic text classification in the frame of NLI. Secondly, he intended to highlight the important learner's linguistic background role in the learning process. Then, he used it to distinguish the native country of the people who share the same mother tongue. The analyzed features in this study covered topic-independent features (function words, POS n-grams, anaphoric shell nouns and annotated errors) and other characteristics that are dependent on topic (character n-grams and positional token frequencies). His experiments demonstrated that anaphoric shell nouns8 and positional token frequencies contributed to achieve the best accuracy. In fact, a topic-independent feature combination reached an accuracy of 93.75. For the sensitive topic feature characters, 4-grams achieved the highest accuracy with about 99% across the corpora. He carried out the linguistic background analysis to explain some mis-classifications of native country for some native languages.

In [7], the researchers were interested in the identification of authors' native languages based on their writing on the Web. Their proposed method was based on automatic classification using various types of features (namely Lexical, Syntactic, Structural and Content features) collected from Web-based texts written by native and nonnative authors as a source corpus. To achieve this goal, the researchers compared three different classification techniques (the C4.5 decision tree, the support vector machine and Naïve Bayes). Their experiments showed the obvious efficiency of the lexical and the content-specific features compared to the rest of the features. Concerning the learning algorithm, the SVM outperformed the Naïve Bayes and C4.5 significantly with a satisfactory accuracy of 70% to 80%.

B. Non-English Native Language Detection

Apart from English, many other languages have attracted researchers' attention in the recent years in order to assess the applicability of NLI techniques to such languages. In this context, Malmasi and Drass [17] and Lan and Hayato [18] addressed the Chinese language. del Río Gayo et al. [28] addressed the Portuguese language.

These research studies can be considered as the first works presenting an expansion of the NLI application to non-English data. Based on a feature set that involves part-of-speech tags, n-grams, function words and context-free grammar production rules, their system revealed that the use of all the combined features outperformed the use of individual ones with an accuracy of 70.61%.

In the second work, Lan and Hayato were the first to use skip-grams as features in the NLI problem combined with the traditional lexical features based on the Jinan Chinese Learner Corpus JCLC. Because the skip-gram features' number or dimension grows enormously, they accept only n-grams that occur in more than ten essays as informative features. Unlike most of the NLI studies which have adopted term frequency (TF) or term frequency– inverse document frequency (TF-IDF). The plus point of this study is that a special attention is paid to assigning an effective weight to each feature. They adopted the BM25 term-weighting method [19]. Their proposed system reached a higher performance with 75% accuracy using hierarchical linear SVM classifiers.

Furthermore, the Finnish and the Norwegian languages have also been addressed in [20] and [21]. The key objective of these two studies was to determine if the NLI techniques previously applied to L2 English can be effective to other different languages. Data were collected from Finnish and Norwegian learners (Advanced Learner Finnish LAS2; Andre spraks korpus, Second Language Corpus ASK). Their results indicated a promising evidence of the applicability of NLI techniques to other languages.

C. Arabic native language identification

Arabic was currently perceived as a critical and strategically useful language. However, Malmasi and Dras's work [22] was the unique study dealing with this language in the NLI field. An interesting study might be relevant to the enrich the field. Their aim was to investigate the usefulness of the syntactic features, mainly CFG production rules, Arabic function words and Part-of-speech n-grams. They used a supervised multi-class classification approach. As a result, their experiments proved to be successfully applied to Arabic NLI. Added to that, it is notable that combining the features led to a reasonable accuracy of about 41%, which was 10% lower compared with their previous study of English using the same set of features. This was due to the fact that the morphological and syntactic richness of Arabic is significantly different from that of English, on the one hand, and to the small size of dataset used in the learning phase, on the other.

D. NLI Shared Task

The growing interest in the NLI field reflected by a number of papers that have been published motivated research groups to organize shared Tasks [23] (to our knowledge, this is the first and the only shared task). The main goal of the task was to further unify the community and help the field progress by providing a competitive environment for systems to be directly compared.

29 teams from different countries participated in this task and 24 teams were elected to write papers describing their systems. These 24 teams competed across three different subtasks. The same test set of data was used for each task. Only the training data changed from one task to another. The teams developed systems trained on Data compiled from the TOEFIL11 corpus only, from External corpora, and from both, respectively in the closed-task, open1-task and open2-task.

The teams were free to choose the convenient learner methods and features. Based on the report of [23], it is observed that the most common features were word, character and POS n-gram features.
Unsurprisingly, the ‘Support Vector Machines’ was the most used among other machine-learning algorithms.

IV. Proposed method

In this study, we addressed the prediction of Arabic learners’ native language as inspired by [22]’s work. In order to come up with an optimal classification model, our focus was on the feature-selection step, which was not given great importance in most of the previous works. Optimality refers to the reduction of features and performance improvement.

Our proposed method, as shown in Figure 1, is divided into three steps. Firstly, in the pre-processing step, we prepared the text to be used in the next step. Then, in the feature-extraction phase, we extracted the set of features that seem to be useful for Ll learners’ background discrimination. Finally, we applied a classification algorithm to generate the classification model. Obviously, the last two steps are supported by a sub-step of feature selection.

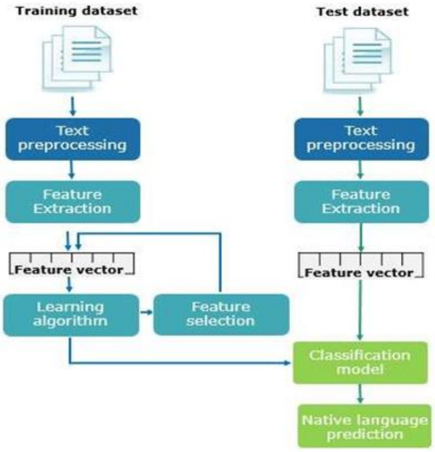


Figure 1. Proposed method NLI process; the blue arrows indicate the training phase and the green arrows indicate the test phase

After that, we ensured that our system is optimal enough to be used in the new L1 text prediction.

A. *Text pre-processing*

The aim of text pre-processing is to prepare the training data for the processing stages as our training data are formed of Arabic texts written by non-Arab learners. The texts are initially handwritten before they are transcribed into computerized form. The analysis of these texts reveals that they include inappropriate characters, words and marks like those found in the example below. Moreover, some notes are inserted into the bodies of the texts when they are transcribed:

For example, the note “كلمة غير معروفة” (indefinite word) is added to indicate that a word or phrase is unclear and cannot be recognized.
ركبنا الحافلة #كلمة غير معروفة#. (We got on the bus #Unknown word #)
The note “معلومة شخصية محذوفة” (Personal information Deleted) is added to indicate that some personal data concerning the text’s author was deleted (e.g. learner’s name, contacts, etc.). In this case, neglecting or removing these notes can influence the structure of the whole sentence; thus, we deal with each set of notes separately by replacing them with the most convenient words in order to preserve the sentence structure as much as possible. However, in the case of inappropriate characters or marks, the solution is to remove them. Once the text pre-processing phase is achieved (i.e. the corpus data are transformed into usable data), the texts are ready for the next phase where features will be extracted from them.

B. *Feature extraction*

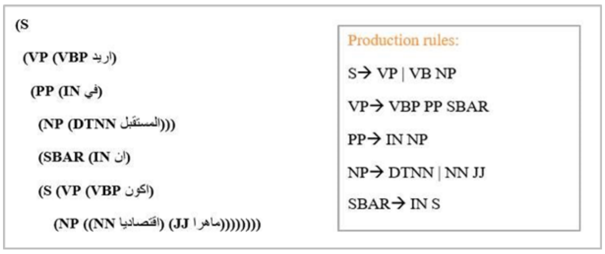
In this study, we explored three syntactic feature types, namely ‘function word’, ‘Part of speech n-grams’ and ‘Context-free grammar production rule’. In this way, we generated three sets of features for each text. For each individual feature, we measured the frequency (TF) with which it appears.

Function words refer to context and topic-independent words used differently by learners to obtain coherent sentences. Here, we adopted 411 common Arabic function words regrouped into seventeen types (classes). Below are examples of the Arabic function words listed by classes.

Type	Examples
Linking words	علاوة على(furthermore),برغم (despite),حيث أن (whereas), etc.
Conjunctions	أو ,)ro(بل(but/ rather), و(and) etc.
snoitisoperP	من(from),إلى (to), في (in),على (on),etc.
…	…

Table 1. Examples of Arabic function words

Part of speech n-grams: These are representative features that highlight the words’ linguistic category. We applied the tagger to assign the grammatical category for each word. Context-free grammar production rules: This term refers to rewriting rules that describe both the syntactic class of the ‘words’ and ‘sentences’ structures. Figure 2 represents an example of production rules extracted from the corresponding parse tree of a given sentence.



```
(S
 (VP (VBP اريد)
  (PP (IN في)
   (NP (DTNN المستقبل)))
   (SBAR (IN ان)
   (S (VP (VBP اكون)
     (NP ((NN اقتصاديا) (JJ ماهرا)))))))
```

Production rules:
S→ VP | VB NP
VP→ VBP PP SBAR
PP→ IN NP
NP→ DTNN | NN JJ
SBAR→ IN S

Figure 2. A constituent parse tree along with the extracted context-free grammar production rules

C. *Feature selection*

The main contribution of the current study is the importance given to the feature-selection phase. Let us recall that we had a set of M documents, each of which is represented by N features and in which N largely exceeds M. Our proposed strategy consisted in using the standard deviation measure (std) to determine the most useful features to discriminate between the classes.

Our feature-selection technique was inspired from the study of [24]. In this study, Alireza et al. assume that the standard deviation can be effectively used as a feature-selection criterion in the context of sentiment classification9. To prove their hypothesis, they conducted a comparative study on two popular feature-selection techniques, information gain (IG) and chi-square (CHI), in comparison with their proposed technique, which is based on standard deviation. They report that their proposed technique was as accurate as (and sometimes outperforms) the two other techniques.

1) Standard deviation

Standard deviation is a statistical measure that indicates the spread and variability of a set of values (or data) around its average (or mean). A high Standard deviation shows that the values are widely spread above and below their mean (Figure 3a), and a low standard deviation shows that the values are clustered closely around the mean (Figure 3b).

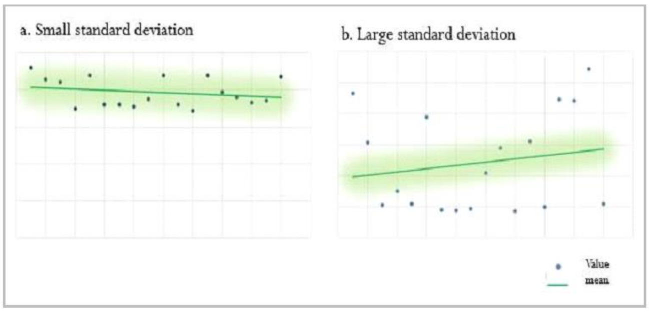


Figure 3. Example of two different dispersions of values around mean

For a given set of values: $x1, x2, \ldots, x$n, where n is the number of values, the standard deviation of this set is calculated in two steps as follows:

Step 1: Determine the mean value $\bar{x}$by adding all the values together and dividing that total by the number of values (n) as:

$$\bar{x} = \frac{\sum x}{n} \qquad (1)$$

Step 2: Calculate the sum of square of the subtraction of the mean from each set of values. Then Standard deviation (δ) is the square root of that sum divided by n:

$$\delta = \sqrt{\frac{\sum |x-\bar{x}|^2}{n}} \qquad (2)$$

2) Feature selection Using Standard Deviation

The idea was to use standard deviation to select features that contribute most to the classification. We calculated the standard deviation for each feature and sorted them in an ascending order as described in Algorithm 1.

Algorithm 1: Calculation of the Standard deviation of the features

```
Input: feature set, term-document matrix weight
Output: set of features sorted by standard deviation
Begin
(1) Calculating the mean for each feature
(2) Calculating the Standard deviation for each feature
(3) Sorting the feature's sets of values in ascending order
(4) Return the Sorted set
End.
```

After that, we muted out features that have the lower standard deviations in order to select only the most useful features. This was done because a lower standard deviation means that the feature values are located closely around the mean, which is not efficient to discriminate between classes. Then, we trained our model using the new subset of features. We replicated the process until no features were excluded without sacrificing accuracy. Later on, we trained the final model with the selected features. The process is described in the following informal algorithm:

Algorithm 2 : Feature selection Using Standard Deviation

```
Input: sorted feature set, a given classification algorithm
(classifier) and desired number of features to exclude in
each step (p)
Output: subset of most confident features
Begin
(1) Apply classifier using the full set
(2) Update feature set by removing the p first features
(3) Evaluate the new set by applying classifier using the
new subset
(4) If (stop criterion not verified) return to (2)
(5) Return the new set
End.
```

The algorithm starts with the full feature set and, for each step, the "p" worst features (in terms of Standard deviation) are excluded from the set. The number of removed features p is determined dynamically at the beginning of the algorithm (P<M where M is the size of the feature set). Then, the new feature set is evaluated by applying a given classification algorithm in order to compare the performance of the new set with the previous set. The process is run repeatedly making sure that no loss in prediction performance occurred (the stop criterion is not verified).

D. Classification model

Once the feature sets are extracted (in the feature extraction phase) and selected (in the feature selection phase), they would be used to train the final model by applying a learning algorithm. The output would be a classification model that is able to predict the native language for the response to new data. We compared the three most popular machine-learning algorithms (Support vector machines, Decision trees and Naive Bayes) [25] [24] [23] [14] and we used the one that outperformed the others.

V. Experiments

Native Language	Number of Texts	Number of Words
Chinese	76	11073
Urdu[1]	64	12341
Malay[2]	46	6686
French	44	5942
Fulani[3]	36	5571
English	35	5774
Yoruba[4]	28	4794
Total	329	52181

Table 2. L1 Distribution by number of texts and words

Several series of experiments were carried out on the test corpus. These experiments were validated and evaluated by the following techniques.

In machine learning, it is common to use validation methods to assess the generalizability of classification algorithms and methods to unknown test examples. K-fold cross validation is one of the most popular cross validation techniques. It consists in splitting the data into k subsets; one subset acts as the validation data and the rest act as training data. The validation process is then repeated k times. This technique becomes the de-facto standard of reporting NLI results; therefore, we reported our experimental results under K-fold cross-validation, with k=10.

Since our training dataset is roughly unbalanced, the use of different performance measures can be a useful solution to overcome the problem of imbalanced data in order to evaluate the classification model. Thus, to evaluate the performance of our method, we employed three metrics commonly used in data-mining evaluation: accuracy, precision and recall. Note that we report our result based on the accuracy measure; this is because this metric was commonly used in previous works.

VI. Results and Interpretations

We ran two sets of experiments in order to evaluate the performance of our suggested method. The first set experiments aimed to assess the performance of the learning algorithms (classifiers) and consequently we chose the most performing one to be used in the next set of experiments. The second set of experiment was dedicated to evaluate the contribution of our features set in different configurations: individually, together, with and without passing by the selection process.

A. *First set of experiments*

Our method relies on the classifier choice. We compared the performance of the three classifiers SVM, the Naive Bayes and the Decision Table using all features. Figure 4 displays the classification accuracy for each classifier.

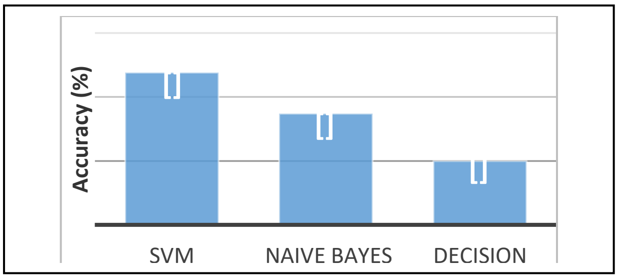


Figure. 4. Variation of accuracy according to the selected classifier

The results revealed clearly that the SVM significantly outperformed both of the Naive Bayes and the Decision Table classifiers consistently with previous studies, this one presents another substantiation of the SVM superiority in text classification problems. Therefore, we used this classifier in the rest of our experiments.

B. *Second set of experiments*

We conducted multiple 10-fold cross-validation experiments to test our features both separately and in combination. Table 3 summarizes the full classification accuracies of the different sets of features both with and without using our proposed feature selection step.

Features	Without feature selection		With feature selection	
	Number of features	Accuracy (%)	Number of features	Accuracy (%)
Production rules	1124	30.5	106	36,5
Function Words	17	31.0	11	31,0
POS unigrams	33	30.0	16	34,9
POS bigrams	594	35.4	145	38,0
POS trigrams[5]	580	29.0	347	29.0
Combined	2348	41.9	278	45.0

Table 3. Number of features and their accuracy

Based on the experimental results described in the table above, we found that removing the lowest deviation features in terms of standard enhanced the prediction capability of our solution. Indeed, applying our selection algorithm enabled our system to obtain a gain in accuracy ranging from 2.6% (case of POS bigrams) to 3.1% (case of combined features), as well as a gain in terms of memory space: we managed to bring the size

[1] The national language and lingua franca of Pakistan

[2] The Malaysian language

[3] Non-tonal language spoken in 20 countries of West and Central Africa

[4] One of the five most spoken languages in Nigeria

[5] POS-Trigrams are excluded from the classification when we combine features.

of the feature vector down to 10 times less than the size of the initial vector, from 2348 to 278. In the following, we detailed the results of individual and combined features after they were selected.

C. Individual features

For individual features, the production rules as well as the function words have shown their ability to distinguish L1 learners with 36.5% accuracy for production rules and 31% for function words.

Unsparingly, POS n-grams consistently with those given in earlier studies outperformed the other syntactic features. The best accuracy, 38%, was reached with n=2. We note here that though POS trigrams gave a reasonable accuracy of 29%; it turned out that combining trigrams POS with other features did not give better results. It rather underperformed the global performance. This can be explained by the fact that these trigrams represent redundant information compared to the other feature sets. Thus, we excluded it when we used features together.

D. Combined features

We combined 278 features distributed as follows: 16 unigrams, 145 bigrams, 11 classes of function words and 106 production rules. This set allowed achieving a better classification result of 45 %, for Arabic NLI which highlights the importance of the feature-selection step.

The confusion matrix in Table 4 shows the distribution of correctly-classified as well as misclassified samples for the different native languages. A combination of function words, POS and production rules were used as classification features.

	classified as						
L1	Chinese	Urdu	Malay	French	Fulani	English	Yoruba
Chinese	56	14	3	3	-	-	-
Urdu	13	40	6	5	-	-	-
Malay	9	11	21	2	2	1	-
French	8	16	6	13	1	-	-
Fulani	3	12	7	7	7	-	-
English	10	12	3	3	-	7	-
Yoruba	8	10	3	4	-	-	4

Table 4. Confusion matrix based on all features combined

The performance of the different native languages is slightly spaced. In fact, the experimental results show that we were able to detect more Asiatic learners of Arabic than European learners. For Chinese and Urdu authors, we obtained an accuracy rate of approximately 80% while this percentage was 36% for French authors and 30% for English writers.

In addition, we found out that the most mis-predicted samples are labelled as the Chinese or the Urdu samples. compared to the other class, it is probably because Chinese and Urdu are over-represented in terms of sample number in the training set, which is attributed to the idea of unbalanced training data and its impact on the effectiveness of the classification model.

Consequent to the two points above it was proven that Asian languages are effectively distinguished in the context of Arabic NLI. On the other hand, the two closely-related European languages are often mi-classified as Asian. African languages are the hardest to distinguish and represent the highest error rate. Especially for Yoruba, only one of seven texts was correctly classified. This may be because of the deficiency of the training data allocated to it.

VII. Comparison with Malmasi and Drass's results

Malmasi and Drass [22] developed the first and the only NLI method that addressed the Arabic language. They found that the best accuracy is obtained using a combination of syntactic features similar to the one used in the current study. Our results, which can be directly compared with theirs, do outperform those they reported by around 5% up in accuracy as shown in Figure 5.

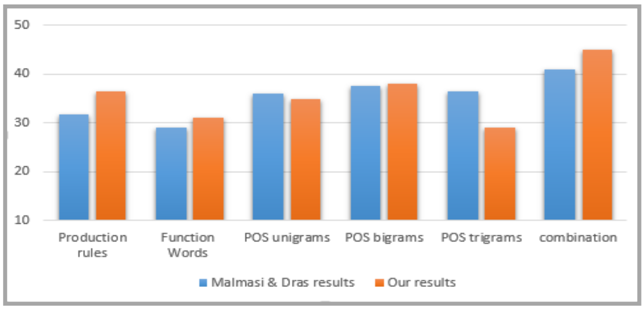


Fig. 5 Performance of our features compared with Malmasi and Drass's ones

The comparison results confirm, on the one hand, the significance of the particular attention that we paid to the choice of the feature sets; this includes the choice of function words and the validation of production rules. In addition, the comparison results confirm the importance of our feature-selection technique. Unlike Malmasi and Drass's study in which they did not state the use of any feature selection technique, ours can overcome the high-dimensional feature problem and improve the overall performance.

VIII. Conclusion

In this research work, we investigated the efficacy of language transfer to identify the first language of non-native Arabic speakers based on their texts written in Arabic. In particular, we focused on the syntax-related transfer. For this purpose, we presented a supervised method for an Arabic NLI task based on syntactic features extracted automatically from texts written by non-Arabic learners.

Essentially, our method consisted of three steps where the input is a set of texts and the output is a classification model able to predict the class of an unseen text: we started by pre-processing the text; in this step, we dealt with the inappropriate characters, words and marks by removing or replacing them depending on the case. Then the texts were passed to the next step where syntactic feature types were extracted. Therefore, the initial set was transformed into a space vector representation at the final stage. The new text representation was used as input for a machine-learning algorithm that served to build the classification model.

We found out that the features space was higher compared to the number of samples. Indeed, it exceeded two thousand when we used all the features together. We assumed that many of them were redundant and noninformative. Based on this hypothesis, we proposed an algorithm using a statistical metric (standard deviation) that enabled us to select the non-useful features.

To accomplish the task, we used the second version of the ALC corpus. We included the seven top native languages: three Asian languages (Chinese, Urdu and Malay), two European ones (French and English) and two African ones (Fulani and Yoruba). All in all, we performed our experiments using 329 texts including an average of 160 words per text. It is worth pointing out that our results are promising; we outperformed the state-of-the-art accuracy (45% vs 41%), given the issues that we faced in this study concerning the limited data and the unavailability of accurate tools dedicated to the Arabic language. Currently, our approach is based on a static-learning model where the used corpus for training and testing is the ALC. Therefore, in our future work, we are planning to address this issue by developing a new Arabic learner corpus that can be used to evaluate the generalizability of our method and more broadly to serve linguistic and computational research areas.

Furthermore, the ALC texts analysis showed that learners made several errors of different types (orthography, morphology, syntax, semantics, etc.) when they expressed their ideas [27]. The exploitation of the errors represents a prospective task. Indeed, these errors reflect one of the main aspects of language transfer resulting from the difference between the learner's native language and that of Arabic.

Finally, to allow a better author detection, we are thinking of going beyond the native language and considering the detection of the age, the gender and the geographical background of the author, and above all, the detection of his/ her personality.

References

[1] F adeeba, S hussain. "Native language identification in very short utterances using bidirectional long short-term memory network". *IEEE ACCESS* 7: 17098-17110. 2019.

[2] S Malmasi, M Dras: "native language identification with classifier stacking and ensembles". *computational linguistics* 44(3). 2018.

[3] K. Ala and M. Kavi, "Significance of Syntactic Features for Word Sense Disambiguation" *Advances in Natural Language Processing*, Lecture Notes in Computer Science, Springer Berlin Heidelberg, p. 340-348, 2004.

[4] A. Charles, "Automatically Detecting Authors' Native Language, *Doctoral dissertation Naval Postgraduate School, Monterey*", California, 120 pages, 2011.

[5] O. Siham and S. Halim, "Authorship Attribution of Short Historical Arabic Texts Based on Lexical Features" *In proceedings ofCyber-Enabled Distributed Computing and Knowledge Discovery*, p. 144-147, Washington, USA, 2013.

[6] E. Dominique, G. Tanja and H. Ben, "an author profiling tool with application to Arabic emails*" In Proceedings of the Australasian Language Technology Workshop*, p. 21-30, Melbourne, Australia, 2007.

[7] T. Parham , K. Cemal and R. Leila, "Author's Native Language Identification from Web Based texts" *Computer and Communication Engineering*,vol. 1, n. 1,p. 47-50, May 2012.

[8] B. Julian and H. Graeme, "Robust, Lexicalized Native Language Identification". *In proceedings ofthe24th International Conference on Computational Linguistics*, p. 391408, Mumbai, India, 2012.

[9] B. Julian et H. Graeme, "native language detection with 'cheap' learner corpora,". *Conference of Learner Corpus Research(LCR2011*), Louvain-la-Neuve, Belgium, 2011.

[10] S. Wong et M. Dras, "Exploiting Parse Structures for Native Language Identification,". .*In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, Scotland, UK, 2011.

[11] K. Moshe, S. Jonathan and Z. Kfir, "Automatically determining an anonymous author's Native Language" *In proceedings of International Conference on Intelligence and Security Informatics*, p. 209-217, Berlin, Germany, 2005.

[12] A. Shlomo, K. Moshe, P. James and S. Jonathan, "Automatically Profiling the Author of an Anonymous Text". *Communications of the ACM*, vol. 52, n. 2, p. 119-123, 2009.

[13] S. Wong and M. Dras, "Contrastive Analysis and Native Language Identification". *In proceedings of the Australasian Language Technology Association Workshop*,p. 53-61, Sydney, Australia, 2009.

[14] E. Kochmar, "Identification of a Writer's Native Language by Error Analysis*", Master thesis,University of Cambridge*, 46 pages, 2011.

[15] J. Xiao, G. Yufan, G. Jeroen, A. Dora, S. Lin and K. Anna, "Native Language Identification Using Large, Longitudinal Data*". In proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*, p. 3309-3312, Reykjavik , Iceland, 2014.

[16] S. Nisioi, "Feature analysis for native language identification". *In proceedings of International Conference on Computational Linguistics and Intelligent Text Processing*, p. 644-657, Cairo, Egypt, 2015.

[17] S. Malmasi and M. Drass, "Chinese Native Language Identification". *In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, p. 95-99, Gothenburg, Sweden, 2014.

[18] W. Lan and Y. Hayato, "Robust Chinese Native Language Identification with Skipgram". *Thesis, Hangzhou*, 80 pages, China, 2015.

[19] W. Lan, T. Masahiro and Y. Hayato, "What is your Mother Tongue?: Improving Chinese native language identification by cleaning noisy data and adopting BM25". *In proceedings of International Conference on Big Data Analysis*, p. 1-6, Hangzhou, China, 2016.

[20] M. Shervin and D. Mark, "Finnish Native Language Identification". *In Proceedings of Australasian*

Language Technology Association Workshop, p. 139-144, Sydney, Australia, 2014.

[21] M. Shervin, D. Mark and T. Irina, "Norwegian Native Language Identification". *In proceedings of Recent Advances in Natural Language Processing*, p. 404-412, Hissar, Bulgaria, 2015.

[22] S. Malmasi and M. Dras, "Arabic Native Language Identification*". In proceedings of the Arabic Natural Language Processing Workshop*, p. 180-186, Doha, Qatar, 2014.

[23] T. Joel, B. Daniel and C. Aoife, "A Report on *the First Native Language Identification Shared Task". In proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, p. 48–57, Atlanta, USA, 2013.

[24] Y. Alireza, I. Roliana, H. Haza and H. Mohammad Sadegh, ."Feature Reduction Using Standard Deviation with Different Subsets Selection in Sentiment Analysis". In proceedings of Intelligent Information and Database System, p. 33–41, Bangkok, Thailand, 2014.

[25] A. Charu and Z. Cheng Xiang, "A survey of text classification algorithms". Mining Text Data, Springer US, p. 163-222, March 2012.

[26] A. Alfaifi, E. Atwell and H. Ibraheem, «Arabic learner corpus (ALC) v2:A New Written and Spoken Corpus of Arabic Learners» *In proceedings Learner Corpus Studies in Asia and the World*, p. 77-89, Kobe , Japan, 2014.

[27] A. Ghazi, F. Reem, F. Anna and F. Eileen, "Annotating an Arabic Learner Corpus for Error". *In proceedings of the International Conference on Language Resources and Evaluation"*, p. 1347-1350, Marrakech, Morocco, 2008.

[28] I del Río Gayo, M Zampieri, S Malmasi. "A Portuguese Native Language Identification dataset". *BEA@NAACL-HLT* : 291-296. 2018.

Authors Biographies

Seifeddine mechti obtained her thesis in computer science in 2018 from the Tunis Higher Institute of Management. he works on plagiarism detection and machine learning for classfication of documents.

he currently have more than 13 papers published in international conferences. In 2018, he became chairman of the organizing committee of the international language processing and knowledge management conference (LPKM).

Lamia Hadrich Belguith is a Professor of Computer Science at Faculty of Economics and Management of Sfax (FSEGS) - University of Sfax (Tunisia). She teaches at the department of computer Science of FSEGS since 1992. She is Head of **Arabic Natural language Processing Research Group** (ANLP-RG) of Multimedia, InfoRmation systems and Advanced Computing Laboratory (MIRACL).