International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 03 | Mar-2018 www.irjet.net p-ISSN: 2395-0072

Automatic Language Identification using Hybrid Approach and Classification Algorithms

Ajinkya Gadgil[1], Swaraj Joshi[2], Pranit Katwe[3], Prasad Kshatriya[4]

[1,2,3,4] *Students, Department of Computer Engineering, K.K.Wagh Institute of Engineering Education and Research, Nashik-422003, Maharashtra, India.*

---***---

Abstract - *In this paper, we implement the working of an uninterrupted classification algorithm for automatic language identification. The method introduced in this paper identify the language without user interaction during processing. In this process, we implement an efficient algorithm which contains Naive Bayesian classification and n-gram text processing algorithm. This method does not require any prior information (number of classes, initial partition) and is able to quickly process a large amount of data and results can be visualized. We address the problem of detecting documents that contain text from more than one language. This project also includes language preprocessing like special characters removal, suffix removal and token generation. Naive Bayesian classification is used for classification of texts in multiple languages. In this project, we use languages like Hindi, English, Gujarati, Sanskrit and other foreign languages. We can extend the project by including language translation, document summarization and sentiment analysis.*

Keywords: **Language identification, N-Gram, Multilingual text, Classification, Natural Language Processing (NLP)**.

1. INTRODUCTION

Currently, large number of methods in Natural Language Processing (NLP) projects are getting the advantage of text processing algorithm to improve their performance in text processing application to help stop the search space, for example, work of identifying several demographic properties of documents from a written text, we can identify the structure of the language. The method of identifying the native language depends on the manner of speaking and writing of the second language and is borrowed from Second Language Acquisition (SLA), which is known as language transfer. The method of language transfer says that the first language (L1) influences the way that a second language (L2) is learned (Ahn, 2011; Tsur and Rappoport, 2007). According to this theory, if we learn to identify what is being transferred from one language to another, then it is possible to identify the native language of an author given a text written in L2. For instance, a Korean native speaker can be identified by the errors in the use of articles 'a', 'an' and 'the' in his English writings due to the lack of similar function words in Korean. As we see, error identification is very common in automatic approaches; however, a previous analysis and understanding of linguistic markers are often required in such approaches.

R and D Department in recent years has given a lot of interest to text data processing and especially to multilingual textual data[1], this is for several reasons: a growing collection of networked and universally distributed data, the development of communication infrastructure and the Internet, the increase in the number of people connected to the global network and whose mother tongue is not English[2]. This has created a need to organize and process huge volumes of data. The manual processing of these data (expert, or knowledge-based systems) is very costly in time and personnel, they are inflexible and generalization to other areas are virtually impossible. So we try to develop automatic methods as *Automatic Language Identification using Hybrid Approach and Classification Algorithms.*

2. METHODOLOGICAL APPROACH

In this part, we describe our methodology with the use of approach based on n-grams[3][4] to group similar documents together. This combination will be examined in several experiments using the Naive Bayesian classification[7], Support Vector Machines (SVM)[8] and Logistic Regression[9] as similarity measures for several values of n.

The language detection of a written text is probably one of the most basic tasks in natural language processing (NLP). For any language-dependent processing of an unknown text, the first thing is to know in which language the text is written. The approach we have chosen to implement is pretty straightforward. The idea is that any language has a unique set of character (co-)occurrences. Our method depends on the three attributes of language that include character set, N-gram[5] and word list[3].

2.1. Character Set

Some languages have a very specific Character set (e.g. Marathi, Sanskrit); while for others, some characters give a good hint of what languages come in question (e.g. Hindi, Gujarati).

2.2. N-gram Algorithm

The term "n-gram"[6] was introduced in 1948. An n-gram may designate both: n-tuple of characters (n-gram character) or an n-tuple of words (n-gram words). This model does not represent documents by a vector of term's frequencies but by a vector of n-gram frequencies in the documents. An n-gram character is a sequence of n consecutive characters. For any document, all n-grams that can be generated are the result obtained by moving a window of n boxes in the text. This movement is made in stages; one stage corresponds to a character for n-grams of characters and a word for n-grams of words. Then we count the frequencies of n-grams found. In scientific literature, this term sometimes refers to sequences that are neither ordered nor straight, for example, a bigram can be composed of the first letter and second letter of a word; consider an n-gram as a set of unordered n words after performing the stemming and the removing of special characters. In our experiments, n-grams of characters are used. Thus an n-gram refers to a string of n consecutive characters. In this approach, we do not need to conduct any linguistic processing of the corpus. For a given document, as we already said, extracting all n-grams (usually n = (2, 3, 4, 5)) is the result obtained by moving a window of n boxes in the main text. This movement is made by steps of one character at a time, every step we take a "snapshot" and all these 'shots' constitute the set of all n-grams of the document. We cut the texts of the corpus based on the value of n chosen. We took n = 2, 3, 4 and 5.

2.3. Word List

The last source of disambiguation is the actually used words. Some languages (Marathi and Hindi) are almost identical in character set and also the occurrences of the specific n-grams. Still, different words are used in different frequencies.

2.4. Naive Bayes Classification

Naive Bayes classifier is an efficient and powerful algorithm for the classification task. Even if we are working on a data set with millions of records with some attributes, it is suggested to try Naive Bayes approach. Naive Bayes classifier gives good results when we use it for textual data processing such as Natural Language Processing. To learn the Naive Bayes classifier we need to understand the Bayes theorem. It works on conditional probability. Conditional probability is the probability that something will happen, given that something else has already occurred. Using the conditional probability, we can calculate the probability of an event using its prior knowledge.

Naive Bayes is a kind of classifier which uses the Bayes Theorem. It predicts membership probabilities for each class such as the probability that given record or data point belongs to a particular class. The class with the highest probability is considered as the most likely class. This is also known as Maximum A Posteriori (MAP).

Naive Bayes classifier assumes that all the features are unrelated to each other. Presence or absence of a feature does not influence the presence or absence of any other feature. We can use the example of Wikipedia for explaining the use of Naive Bayes classifier i.e.

A fruit may be considered to be an apple if it is red, round, and about 4″ in diameter. Even if these features depend on each other or upon the existence of the other features, a Naive Bayes classifier considers all of these properties to independently contribute to the probability that this fruit is an apple.

In real datasets, we test a hypothesis given multiple evidence (feature). So, calculations become complicated. To simplify the work, the feature independence approach is used to 'uncouple' multiple evidence and treat each as an independent one.

2.5. Language Identification

In this step, we classify input text document that matches the model of dominant language.

2.6. Equations

These are the equations for the n-gram model used in our methodology

- unigram: p(wi) (i.i.d. process)
- bigram: p(wi |wi−1) (Markov process)
- trigram: p(wi |wi−2, wi−1)

We can estimate n-gram probabilities by counting relative frequency on a training corpus. (This is maximum likelihood estimation.)

$$\hat{p}(w_a) = \frac{c(w_a)}{N}$$

$$\hat{p}(w_b|w_a) = \frac{c(w_a, w_b)}{\Sigma_{w_b} c(w_a, w_b)} \approx \frac{c(w_a, w_b)}{c(w_a)}$$

where N is the total number of words in the training set and c(·) denotes count of the word or word sequence in the training data.

Following equation is used for the Naive Bayes Classifier to calculate the conditional probability.

$$P(C \mid A) = \frac{P(A \mid C)P(C)}{P(A)}$$

- P(C) is the probability of hypothesis H being true. This is known as the prior probability.
- P(A) is the probability of the evidence (regardless of the hypothesis).
- P(A|C) is the probability of the evidence given that hypothesis is true.
- P(C|A) is the probability of the hypothesis given that the evidence is there

2.7. Working of Classification Algorithms

A Path Lab is performing a Test of disease say "D" with two results "Positive" & "Negative." They guarantee that their test result is 99% accurate: if you have the disease, they will give test positive 99% of the time. If you don't have the disease, they will test negative 99% of the time. If 3% of all the people have this disease and test gives "positive" result, what is the probability that you actually have the disease?

For solving the above problem, we will have to use conditional probability. Probability of people suffering from Disease D, P(D) = 0.03 = 3%. Probability that test gives "positive" result and patient have the disease, P(Pos | D) = 0.99 =99%.

Probability of people not suffering from Disease D, P(~D) = 0.97 = 97%. The probability that test gives "positive" result and patient does have the disease, P(Pos | ~D) = 0.01 =1%.

For calculating the probability that the patient actually has the disease i.e, P(D | Pos) we will use Bayes theorem:

$$P(D\ |Pos) = \frac{P(Pos\ |D)\ *\ P(D)}{P(Pos)}$$

We have all the values of numerator but we need to calculate P(Pos):

P(Pos) = P(D | pos) + P(~D | pos)
= P(pos|D)*P(D) + P(pos|~D)*P(~D)
= 0.99 * 0.03 + 0.01 * 0.97
= 0.0297 + 0.0097
= 0.0394

Let's calculate,

P(D | Pos) = (P(Pos | D) * P(D)) / P(Pos)
= (0.99 * 0.03) / 0.0394
= 0.753807107

So, Approximately 75% chances are there that the patient is actually suffering from the disease.

2.8. Results and Evaluation

In this section, we present and evaluate the obtained results. We evaluate our method using three classification algorithms.

- Naive Bayesian classification[7]
- Support Vector Machines (SVM)[8]
- Logistic Regression[9]

Following diagram show the comparison between this three methods.

Distance	NBC				SVM				LR			
n-gram	2	3	4	5	2	3	4	5	2	3	4	5
Time for extracting n-grams (second)	244	204	204	152	244	204	204	152	244	204	204	152
Time required to Classify (second)	49	272	403	400	54	130	285	263	28	65	99	61
F-measure (%)	51.71	44.35	47.45	51.81	48.65	44.09	40.63	49.58	44.39	40.28	40.4	46.95

3. CONCLUSIONS

The work presented in this paper shows that it is possible to identify the language automatically in an unsupervised manner. It aims to enhance the unsupervised methods and techniques applied for classification of a text document to identify the language in a multilingual corpus. Furthermore, we can extend our project to translate the data from one language to another language.

REFERENCES

[1] Marco Lui, Jey Han Lau and Timothy Baldwin, "Automatic Detection and Language Identification of Multilingual Documents", Department of Computing and Information Systems, The University of Melbourne, NICTA Victoria Research Laboratory, Department of Philosophy, King's College London, 2014

[2] Pavol ZAVARSKY, Yoshiki MIKAMI, Shota WADA, "Language and encoding scheme identification of extremely large sets of multilingual text documents", Department of Management and Information Sciences, Nagaoka University of Technology, 1603-1 Kamitomioka, 940-2188 Nagaoka, Japan

[3] IOANNIS KANARIS, KONSTANTINOS KANARIS, IOANNIS HOUVARDAS, and EFSTATHIOS STAMATATOS, "WORDS VS. CHARACTER N-GRAMS FOR ANTI-SPAM FILTERING", Dept. of Information

and Communication Systems Eng., University of the Aegean, Karlovassi, Samos - 83200, Greece, 2006

[4] Bashir Ahmed, Sung-Hyuk Cha, and Charles Tappert, “Language Identification from Text Using N-gram Based Cumulative Frequency Addition”, Proceedings of Student/Faculty Research Day, CSIS, Pace University, May 7th, 2004

[5] Rahmoun A, Elberrichi Z. Experimenting N-Grams in Text Categorization. International Arab Journal of Information Technology, 2007, Vol 4, N°.4, pp. 377-385.

[6] ABDELMALEK AMINE, MICHEL SIMONET, ZAKARIA ELBERRICHI, “On Automatic Language Identification International Journal of Computer Science and Applications, ©Technomathematics Research Foundation Vol. 7, No. 1, pp. 94 – 107, 2010. (references)

[7] Yuguang Huang, Lei Li, “NAIVE BAYES CLASSIFICATION ALGORITHM BASED ON SMALL SAMPLE SET”, Beijing University of Posts and Telecommunications, Beijing, China, 2011

[8] Thorsten Joachims, “Text Categorization with Support Vector Machines: Learning with Many Relevant Features”, Universit□at Dortmund, Informatik LS8, Baroper Str. 301, 44221 Dortmund, Germany

[9] Daniel Jurafsky & James H. Martin, “Speech and Language Processing, Chapter 7 - Logistic Regression”, Copyright c 2016, Draft of August 7, 2017.