CENG-539 Natural Language Processing

Term Project Report

By
Harun Reşit Zafer
50050819

Lecturer: Asst. Prof Zeynep Orhan

13 January 2010

Fatih University

Department of Computer Engineering

An Author Prediction Experiment for Turkish

Abstract

In this experiment, newspaper articles of 10 Turkish columnist have been analyzed in terms of different attributes such 1-gram root, 1-gram stem 1-gram word frequency, word count in sentence, letter count in word or word count in paragraph. 2-gram and 3-gram analyze of text also has been accomplished but due to their large size and need for vast amount of memory they couldn't be used for classification. Naïve Bayes, Naïve Bayes Multinomial, Sequential Minimal Optimization are the classifiers that have been used to classify texts according to their authors. Experiment implemented on WEKA data mining tool.

1-Introduction

Authorship analysis or **Authorship attribution** can be defined as identifying the author or determining some characteristics of the author of a given text by using natural language processing methods.

Matching a sequence of text and author according to attributes extracted from texts and known attributes of authors is called **author prediction**.

Author prediction is a sub-category of **text categorization** which is also a sub-category of **document classification**.

For English language there have been a lot of studies and accomplishments for last 35 years.[1] For Turkish similar studies are very rare.

2-Problem

For this experiment, 10 different Turkish columnists which usually write under same genre, politics, has been chosen from several newspapers. 100 articles of each author are collected from the internet sites of the newspapers they write for. These articles have been used as learning data. After this data analyzed and properties of each author learnt, it was expected to be able to identify the author of a new article among 10 authors. It was also expected that if the given article does not belong to any author of 10, this is noticed.

Preferring such a corpus has some advantages such as ease of collecting data. Since each text is written in a newspaper article and they are mostly about politics genre, this doesn't affect the results of experiment. Another advantage is avoiding time dependency. All articles are written within 12-14 months and the articles to be classified are also written recently. So time also doesn't -or almost not- affect the results of our experiment.

With these advantages a more accurate measurement for effectiveness of our method was implemented.

Selected Authors for this experiment and their newspapers can be seen from Table 1

Table 1

Author	Newspaper
Ahmet Turan Alkan	Zaman Gazetesi
Nedim Hazar	Zaman Gazetesi
Engin Ardıç	Sabah Gazetesi
Haşmet Babaoğlu	Sabah Gazetesi
Hıncal Uluç	Sabah Gazetesi
Perihan Mağden	Radikal Gazetesi
Yasemin Çongar	Taraf Gazetesi
Gülay Göktürk	Bugün Gazetesi
Ahmet Altan	Star Gazetesi
Dücane Cündioğlu	Yenişafak Gazetesi

3-Method

3.1. Preprocessing

In this project different features of given texts will be determined and analyzed. At first **Turkish Text Frequency Analyzer [4]** of **Fatih University NLP Group** is used to analyze the articles. The analyzed features can be seen from Table 2

Table 2

Feature	Description
1-gram Root Frequency	Frequency of root words of all words in the text
1-gram Stem Frequency	Frequency of stems of all words in the text
1-gram Word Frequency	Frequency of all words in the text
2-gram Root Frequency	Frequency of roots of 2-grams in the text
2-gram Root Group Frequency	Frequency of root groups of 2-grams in the text. Ex: ben-gel and gel-ben will be accepted as same.
2-gram Stem Group Frequency	Frequency of stem groups of 2-grams in the text.
2-gram Stem Frequency	Frequency of stems of 2-grams in the text
2-gram Word Frequency	Frequency of 2-grams in the text
3-gram Root Frequency	Frequency of root groups of 3-grams in the text. Ex: ben-ev-gel and gel-ev-ben or ev-ben-gel will be accepted as same.
3-gram Root Group Frequency	Frequency of roots of 3-grams in the text
3-gram Stem Group Frequency	Frequency of stem groups of 3-grams in the text
3-gram Stem Frequency	Frequency of stems of 2-grams in the text
3-gram Word	Frequency of 3-grams in the text

Frequency	
Word Frequency by Letter Count	Frequencies of words according to their lengths.
Word Frequency by Syllable Count	Frequencies of words according to their syllable count
Sentence Frequency by Word Count	Frequencies of sentences according to their word count. In other words sentence length.
Paragraph Frequency by Word Count	Frequencies of sentences according to their word count. In other words paragraph length.
Word Structure Frequency by Phoneme Type	Frequency of each word structure such as CVC or VCVC
Word Language Frequency	Frequency of each language that words originally belong to.

WEKA (Waikato Environment for Knowledge Analysis) is a popular suite of machine learning software written in Java, developed at the University of Waikato. WEKA is free software available under the GNU General Public License. [5]

WEKA data mining tool is used to classification in our experiment. WEKA application uses **ARFF** files. Turkish Text Frequency Analyzer (**TTFA**) converted the data analyzed to ARFF format.

Before this experiment TTFA wasn't capable of producing ARFF files from the analyzed text. This feature has been added to the application with during the experiment period with serious effort and support of **Atakan Kurt.**

After trying to analyze total 1000 article, we saw that there is a lot of words that TTFA didn't include. Especially proper names didn't exist in our database. After preparing 7000 Turkish names and adding them into our database, other words were added manually.

ARFF production feature of TTFA has been tested many times during this process. Because of the missing and erroneous fields in the database of TTFA, ARRF files produced were corrupted. That is why we worked on our database and debug the erroneous data and filled the missing fields.

Another problem that we faced during this process was memory overflow. Then we decided to reduce the number of articles to 30 for each author which is 300 totally.

3.2. Classification Algorithms

Naïve Bayes: A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem (from Bayesian statistics) with strong (naive)

independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model". [6]

Naïve BAyes Multinomial: A different version of Bayes rule which runs faster. The core equation for this classifier: P[Ci|D] = (P[D|Ci] x P[Ci]) / P[D] (Bayes rule) where Ci is class i and D is a document .[7]

Sequential Minimal Optimization: In mathematical optimization and machine learning, Sequential Minimal Optimization (SMO) is an algorithm for solving large quadratic programming (QP) optimization problems, widely used for the training of support vector machines. First developed by John C. Platt in 1999,[1] SMO breaks up large QP problems into a series of smallest possible QP problems, which are then solved analytically.[8]

3.3. Using WEKA

After we managed to get well-formed ARFF files with fewer amount of test data, we tried to produce them for the 1000 articles collected. We saw that 1000 articles were too much that it caused memory overflow and huge ARFF files more than 200 MB. Then we decided to reduce number of articles to 30 for each author. We used 90% percent (270) of these articles as training data and 10% percent (30) as test data.

ARFF files for 2-Gram and 3-gram attributes were still too big that WEKA didn't handled these datasets without at least **3GB** memory. That's why we couldn't manage to use these datasets.

4-Experiment Results

In this experiment training data contained 27 articles of each author as training data and 3 articles of each as testing data. In total 300 articles were divided as 270 to 30.

Here is a sample confusion matrix of 1-gram root dataset with Naïve bayes:

```
=== Confusion Matrix ===

 a b c d e f g h i j   <-- classified as
 2 0 0 0 0 0 1 0 0 0 | a = 05_Uluc
 0 3 0 0 0 0 0 0 0 0 | b = 01_Alkan
 0 0 3 0 0 0 0 0 0 0 | c = 09_Cundioglu
 0 0 0 3 0 0 0 0 0 0 | d = 02_Hazar
 0 0 0 0 3 0 0 0 0 0 | e = 06_Magden
 0 0 0 0 0 3 0 0 0 0 | f = 08_Gokturk
 0 0 0 0 0 0 3 0 0 0 | g = 07_Congar
 1 0 0 0 0 0 0 2 0 0 | h = 03_Ardic
 1 0 0 1 0 0 0 1 0 0 | i = 04_Babaoglu
 0 0 0 0 0 1 1 0 0 1 | j = 10_Altan
```

As you see from the table we have tried to classify 3 articles of each author.

Accuracy of Correctly Classified Instances can be seen from Table 3

Table 3

Data Set	Naïve Bayes	Naïve Bayes M.	SMO	Data Size (MB)
1-gram root	**77%**	**80%**	**93%**	**4.8**
1-gram stem	**83%**	**80%**	**100%**	**7.6**
1-gram word	**80%**	**83%**	**90%**	**26.7**
Letter count in word	**40%**	**37%**	**33%**	**0.023**
Syllable count in word	**27%**	**27%**	**17%**	**0.014**
Word Count in Paragraph	**53%**	**30%**	**30%**	**0.087**
Word count in Sentence	**43%**	**37%**	**27%**	**0.044**
Language of word	**37%**	**27%**	**30%**	**0.026**
Word Structure	**30%**	**40%**	**37%**	**0.216**
Merge of WCP, WCS, LCW	**53%**	**63%**	**50%**	**0.148**
2-gram Root				**56**
2-gram Root Group				**51**
2-gram Stem				**63**
2-gram Stem Group				**60**
2-gram Word				**76**
3-gram Root				**76**
3-gram Root Group				**73**
3-gram Stem				**77**
3-gram Stem Group				**75**
3-gram Word				**80**

From the table we can say that **1-gram stem** is the best dataset to use. Besides it gave the most accurate results, its size is feasible to use this method in real world applications.

SMO algorithm worked best with all 1-gram data sets and it also consumes less memory. One important disadvantage of SMO is, its taking remarkably more time than others.

Naïve Bayes Multinomial usually gave better results than Naïve Bayes. It also runs faster that Naïve Bayes. With better datasets, I believe that this classifiers can give satisfying accurate results.

Small data sets such as Letter count in word, word count in sentence or word structure didn't give satisfying results. That's why best three of them were selected and merged. After that better result were taken.

2-grams and 3-grams weren't feasible to use for classification. These dataset can only be used after elimination of most attributes.

5-Conclusion and Future Work

According to results of this experiment and the problems have been **experienced** during this period, it is believed that more work on stylistic

features such as word length or paragraph length must be done because they produce much smaller data for classification algorithms. As it is mentioned above by merging some of these features better results have been achieved. With true combinations much better results can be gained and therefore they can be used in real world applications.

Another major advantage of this small datasets is that they don't include the actual data. Instead they include just some numbers which offers a better solution in terms of **privacy**.

After seeing that WEKA can not handle our 2-gram and 3-gram files because of JVM heap size, it also has been tried to write a little Java application that eliminates useless features in these datasets. Unfortunately there weren't enough time left.

6-References and Resources

1. Amasyalı, M.F., Diri, B.: Automatic Turkish Text Categorization in Terms of Author. Genre and Gender, NLDB, Klagenfurt, Austria, 221–226 (2006)
2. Tayfun Kucukyilmaz, B. Barla Cambazoglu, Cevdet Aykanat, and Fazli Can, Chat Mining for Gender Prediction, Information Processing and Management: an International Journal, Volume 44 , Issue 4 (July 2008)
3. http://en.wikipedia.org/wiki/Document_classification, last visited 12.01.2010
4. OKTAY M., KURT A., KARA M., An Extendible Frequency Analysis Tool for Turkish, "Türkçe İçin Bir Sıklık Analizi Programı", 38th International Congress of Asian and North African Studies (ICANAS 38), ANKARA/TÜRKİYE, Sep. 2007.
5. http://en.wikipedia.org/wiki/Weka_%28machine_learning%29, last visited 12.01.2010
6. http://en.wikipedia.org/wiki/Naive_Bayes_classifier, last visited 12.01.2010
7. http://weka.sourceforge.net/doc/weka/classifiers/bayes/NaiveBayesMultinomial.html, last visited 12.01.2010
8. http://en.wikipedia.org/wiki/Sequential_Minimal_Optimization, last visited 12.01.2010