Gender Identification of Blog Authors using Stylometic Analysis

Ajinkya Potdar, Akassh Mishra, Shaoli dutta
Computer Science Department
Stony Brook University
Stony Brook, New York
{apotdar, amishra, shdutta}@cs.stonybrook.edu

***Abstract*—Recent days have seen a rapid growth of blogs, thus their value as an important source of information is increasing. Bloggers use blogs as a medium to express their opinions, thoughts, expressions for daily events, products, services, etc. By analyzing blogs, we can identify the trend emerging in the bloggers' realm in near future. This trend is a source of valuable information for the business experts in deciding the marketing strategies to provide value-added services – blog search, blog tracking, etc. Gender classification of the authors of the blogs is one such study, which has many commercial applications. For example, this study can help in finding out which topics or products/services are most talked about by males and females, and what products and services are liked or disliked by men and women. This information is crucial for market intelligence because the information can be exploited in targeted advertising and product development. Existing systems for gender identification mainly use features such as words, word classes, and POS (part-of-speech) tagging, for classification learning. We use N-gram sequencing and POS sequencing for classification which yield better results than earlier systems in terms of accuracy.**

Keywords—PCFG; PosTagging; PosSequencing; F-Measure; ngram sequence

I. INTRODUCTION

The most informal and personal writings that people post on their own blog sites are the sources of the best personalized text. Blogging can be considered as one of the most important online activities. People tend to share their blogs with their friends and family members. The topics of blog posting cover almost everything, ranging from personal life, political opinions, recipes, product reviews, or even just random rants. Although some bloggers do not reveal their information on their blog sites, many don't make such information public. Therefore, authors' attribute classification such as their gender or age will have a significant effect in many commercial-based applications, such as targeted advertising and product development [2]. In blog search, this information can also help people and blogger to get information pertaining to a specific age group or gender, on any topic that they may be interested in. From a research perspective, blog author gender classification is an interesting problem. Blog posts are unstructured and short. They differ tremendously from formal texts, since they may have informal sentences, grammar errors, slang words and phrases, and wrong spellings. These characteristics of blog posts may complicate any classification or categorization attempts [2].

The goal of our project is to identify author gender of blogs coming from a wide variety of sources. We are interested in knowing how well we can tackle this problem, what methods and features are most effective in this task.

Gender classification problem can be treated as a binary classification problem. That is, given two classes {male, female} the blog will belong to either one of them. To design this hypothesis, we will design a set of features that remain constant for large number of blogs written by the same gender.

We use the N-Gram sequences and POS-sequences for modeling and show that they perform better than N-Gram models and POS-Tagged Models.

II. RELATED WORK

A lot of research has been done till now on gender classification of blogs. These research have used word classes (Schler et al., 2006), POS tag features, content word classes (Arga-mon et al., 2007), POS n-grams together with content words (Koppel et al. 2002), personality types for gender classification (Nowson et al., 2005). POS n-grams was used in (Koppelet et al., 2002; Argamon et al., 2007).

We are using variation of n-grams i.e. n-gram sequences for the purpose of gender classification. We have also used POS sequences [1] of length 3, 5, 7 for gender identification.

III. DATA SET

Previously, a lot of work has been done by focusing on English literature, newswire articles and British Natural Corpus (BNC) (e.g. Argamon et al. (2003)). Recent studies are based on informal writings such as web blogs [1] (e.g., Mukherjee and Liu (2010)). We are using the same dataset as used by Mukherjee and Liu (2010) [1]. Dataset is taken from the following website "http://www.cs.uic.edu/~liub/FBS/blog-genderdataset.rar". This dataset was collected from many blog hosting sites and blog search engines, e.g. blogger.com, technorati.com, etc.

We have considered 800 female blogs and 800 male blogs for our experiments. This dataset has been preprocessed by removing all the stop words.

IV. Model selection and nested cross validation

We have considered 640 blogs as training and 160 blogs as test data. We use five-fold nested cross validation. Here, we split our data into 5 folds. In each step, we use 4 folds as training data and the remaining as the test data. We run the model selection algorithm for each of the folds to get five models and select the best model.

We are using Naïve Bayes classifier for gender classification and using Kernel estimator parameter. Kernel estimator is a kernel density estimator. It uses one Gaussian kernel per observed data value. We have fine tuned kernel estimator and supervised discretion parameter for modeling.

V. Feature Selection and Modeling

A. Baseline

We will be using N-Gram and POS-tagged models as baselines to compare with our approaches: N-Gram sequencing and POS sequencing.

B. N-Gram Models

We expect that n-gram language models will be effective in learning shallow lexico-syntactic patterns of gender specific language styles. We are using unigrams, bigrams, and trigrams as features. We are tokenizing our blog instances using n-gram tokenizer provided by weka jar. Here, we are considering each of the blogs as an instance for the classifier. Before training our blog instances, we use StringToWordVector class for tf-idf normalization, removing stop words, etc. We convert each of the blog instances in arff format where attribute contains features and classes, and data contains feature encodings.

1) *Unigram*

$p(w_n) = C(w_n) / (\text{no. of words in the corpus})$.

2) *Bigram*

$p(w_n) = C(w_n \mid w_{n-1}) / C(w_{n-1})$

3) *Trigram*

$p(w_n) = C(w_n \mid w_{n-2}\, w_{n-1}) / C(w_{n-2}\, w_{n-1})$

C. POS Tagging

We have used the Stanford POS tagger for Parts-of-speech tagging and encoded these POS tags as unigram, bigram and trigram features for structural analysis of blogs.

Next, we are using intrinsic evaluation to evaluate our model on the test data.

D. N-Gram Sequences

We have used N-gram sequences of length 3, 5 as features for feature encoding. This is a variation of N-grams where we use all n-grams ranging from length 1 to N.

E. POS Sequences

We have used POS sequences of length 3, 5, 7 as features for feature encoding. This is a variation of POS N-grams where we use all n-grams ranging from length 1 to N.

VI . Experimental Analysis & Results

Table 1 shows the summary of all experiments in terms of accuracy, precision and F-measure for all the features that we have used for modeling our blog data.

A. Experiment 1 - Unigram, Bigram, Trigram as features

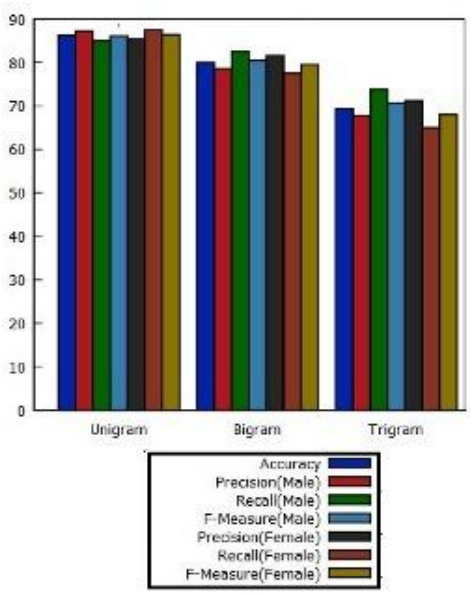

Fig. 1 N-Gram Model Results

We experimented with Unigram, Bigram, and Trigram language models and Naive Bayes used as classifier. Figure 1 shows the comparison between the three models with respect to accuracy, precision, recall and F-measure. We can see unigram model (86% accuracy), Bigram model (80% accuracy) and Trigram model (69.375% accuracy) has performed better than bigram and trigram models.

			Male			Female		
Approach	**Features** (Naïve Bayes)	**Accuracy** %	P	R	F	P	R	F
POS Tagging	Unigram	74.375	0.791	0.663	0.721	0.71	0.825	0.763
	Bigram	77.5	0.789	0.75	0.769	0.762	0.8	0.78
	Trigram	77.5	0.814	0.713	0.76	0.744	0.838	0.788
Text Categorization	Unigram	86.25	0.872	0.85	0.861	0.854	0.875	0.864
	Bigram	80	0.786	0.825	0.805	0.816	0.775	0.795
	Trigram	69.375	0.678	0.738	0.707	0.712	0.65	0.68
N-Gram Sequencing	N-Gram Sequence 3	91.875	0.904	0.938	0.92	0.935	0.9	0.917
	N-Gram Sequence 5	91.25	0.902	0.925	0.914	0.923	0.9	0.911
POS Sequence	POS Sequence 3	80.625	0.802	0.813	0.807	0.81	0.8	0.805
	POS Sequence 5	81.875	0.807	0.838	0.822	0.831	0.8	0.815
	POS Sequence 7	81.25	0.805	0.825	0.815	0.821	0.8	0.81

Table 1 Experimental Results

B. Experiment 2 - POS tags used as Features

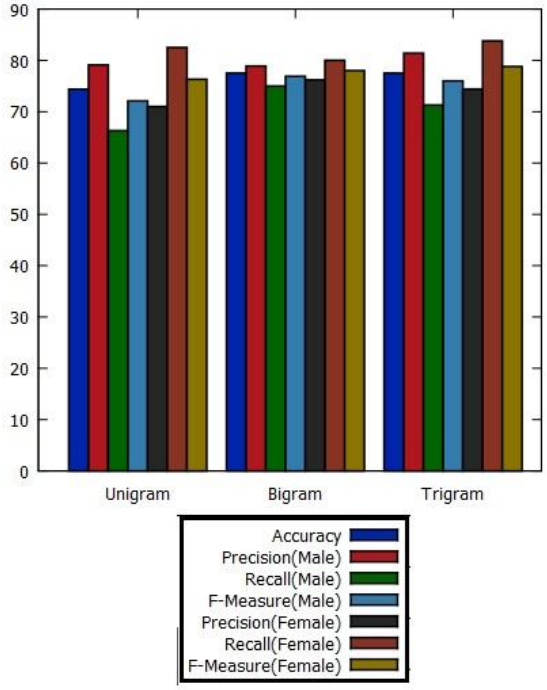

Fig. 2 POS Tag Model Result

We used Stanford POS tagger for tagging our blog data and then used unigram; bigram & trigram model over POS tagged data. Figure 2 shows the comparison between the three models with respect to accuracy, precision, recall and F-measure. We can see that bigram (77.5%) and trigram (77.5%) have performed better in terms of accuracy in comparison to unigram (74.375 %).

C. Experiment 3 - N-gram sequences used as Features

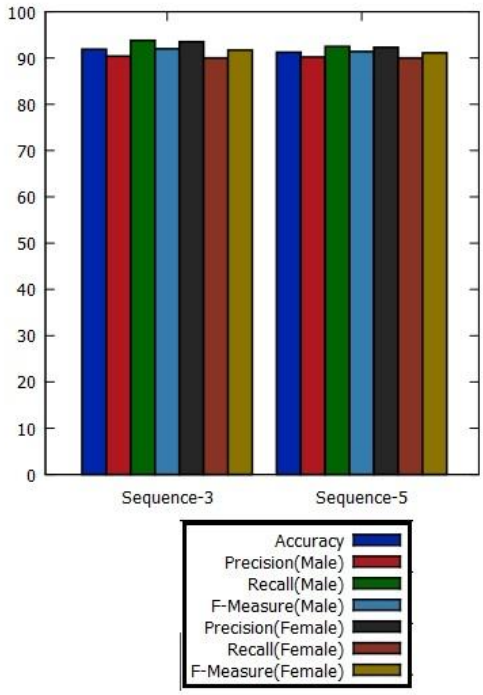

Fig. 3 N-Gram Sequencing

This is the new approach that we have used for feature encoding. It is the variation of N-gram models.
N-gram sequence includes all the sequences of length 1, 2, 3... N. So our feature vector includes all these sequences.
We consider only those sequences which have the frequency greater than some minimum frequency.

Here is the algorithm for selection of sequences:

Pseudo Code:

Variables/Parameters Set:
M = Maximum Continuous N-Gram sequence Threshold
F = A Defined Frequency of Pattern above which those patterns become acceptable.
D = Corpus.
L = List of N-Gram Sequences (currently null)
Loop 1:
For i = 1 to Length (D)
    For j = 1 to M
        Current pattern = sequence of length j
        Search for current pattern in List L

        If found is yes
            List L[Current Pattern].count++
        Else
            Add the Pattern to the List L
            List L[Current Pattern].count = 1
        Move to next sequence
Loop 2:
For k = 1 to Length(List L)
    If List L[k].count < F
        Delete List L[k] from List L

Figure 3 shows the results of different N-gram sequences when used as features in terms of accuracy, precision, recall, F-measure. Accuracy for length 3 sequence is almost similar to accuracies of length 5 sequences when used as features.
We can see N-gram sequences yield better accuracy than N-grams when used as features.

D. Experiment 4 - Part-of-speech tag sequences used as Features

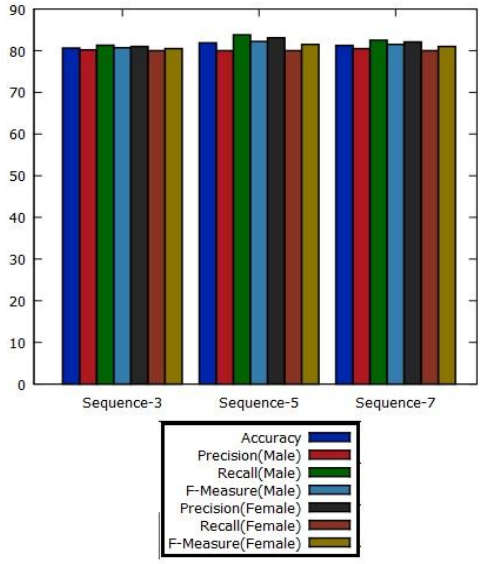

Fig. 4 POS Sequencing Result

This is variation of use of POS tags as features. Here we consider POS tag sequences of different lengths.
So when we say POS sequence of length 5 we consider all POS sequences of length 1, 2, 3, 4, 5 for features encoding.
Figure 4 shows the comparison of different POS sequences of different lengths when used for feature encoding. We can say that accuracy of classification increases initially as we keep on increasing length of sequences and after words it becomes constant.

VI. Analysis with Respect to Baseline

A. Text Categorization Trigram vs N-Gram Sequence 3

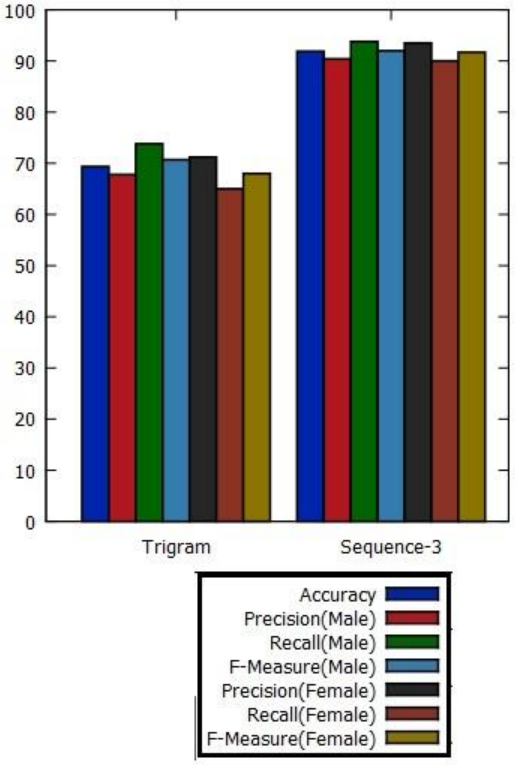

Fig. 5 Trigram Model vs N-Gram Sequence 3

We can see N-Gram sequencing yield better accuracy than N Grams when used as features. Accuracy of trigram model is 69.375% whereas for N-Gram sequences-3 model is 91.875%. Hence, we can conclude from the experimental result that N-gram sequencing is much more capable of capturing complex stylistic regularities of male and female authors than N-Gram Model.

B. POS Tagging Trigram vs POS Sequence 3

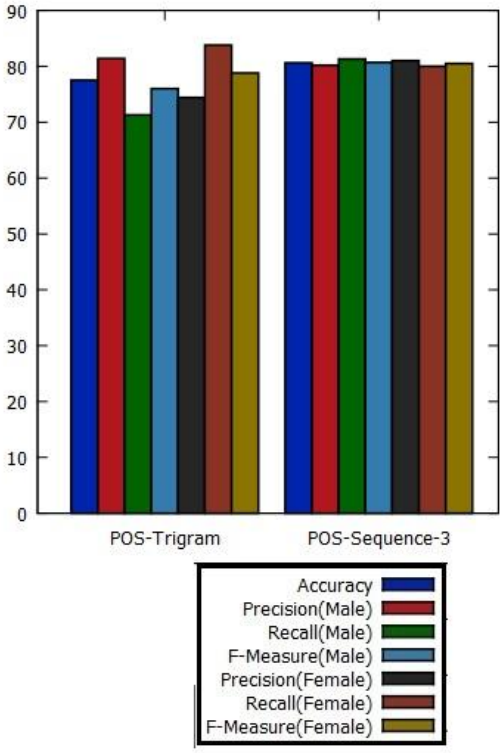

Fig. 6 POS Trigram vs POS Sequence 3

We can see POS sequences yield better accuracy than POS tags when used as features. Accuracy of POS-Trigram model is 77.5% whereas for POS sequences-3 model is 80.625%. Similarly, we can conclude that POS sequencing is much more capable of capturing complex stylistic regularities of male and female authors than POS Tagging.

VII. Conclusion & Future Work

We studied the problem of gender classification. We performed our experiments on real-life web blogs with Naive Bayes classifier with N-gram, Part-of-speech tags, N-gram sequences, Part-of-speech tag sequences for feature encodings. Even though lot of research has been done on gender based classification of blog authors which involved using N-grams, Gender Preferential Features, stylistic features, our approach of N-gram sequences performs better in terms of accuracy. POS Sequence pattern and N-Gram Sequencing are successful in capturing complex stylistic regularities of male and female authors.

We will also be using deep syntactical features. N-gram and POS tags are not able to classify when there is the same set of words, only rearranged. Such differences in sentence structures can be captured using deep syntactic features. A probabilistic context-free grammar (PCFG) captures the deep syntactic regularities, as against the shallow n-gram based models which are based on the shallow lexico-syntactic patterns.

A. The approach we will be using to perform analysis is as follows:

1) Generating the parse tree of the blog using the Stanford PCFG parser.
2) The parse tree is converted to the set of corresponding production rules.
3) Now we consider these rules as unigram features.
4) We consider rules up to a certain level for this analysis and keep on increasing the levels until we get the most accurate results. (fine-tuning)

References

[1] Arjun Mukherjee and Bing Liu, 2010, "Improving gender classification of blog authors", Conference on Empirical Methods in Natural Language Processing, EMNLP '10, pages 207–217, Stroudsburg, PA, USA.

[2] Susan C. Herring and John C. Paolillo, 2006, "Gender and genre variations in weblogs", Journal of Sociolinguistics, Vol. 10, No. 4., pages 439 –459.

[3] Na Cheng, R. Chandramouli and K.P. Subbalakshmi, 2011, "Author Gender Identification from Text Documents", Journal of Digital Investigation, Elsevier.

[4] Claudia peersman, walter daelemans, leona van vaerenbergh, "predicting age and gender in online social networks", 3rd international workshop on search and mining user-generated contents, 2011, pages 37-44.

[5] Ruchita Sarawgi, Kailash Gajulapalli, and Yejin Choi, 2011, "Gender Attribution: Tracing Stylometric Evidence Beyond Topic and Genre". CoNLL 11, Proceedings of the Fifteenth Conference on Computational Natural Language Learning, Pages 78-86.