Gender Prediction Using Browsing History

Chapter *in* Advances in Intelligent Systems and Computing · January 2014

DOI: 10.1007/978-3-319-02741-8_24

2 authors:

Do Viet Phuong
Tru?ng Đ?i H?c Nông Lâm TP. H? Chí Minh
2 PUBLICATIONS **7** CITATIONS
SEE PROFILE

Tu Minh Phuong
Posts and Telecommunications Institute of T…
54 PUBLICATIONS **328** CITATIONS
SEE PROFILE

Gender Prediction Using Browsing History

Do Viet Phuong[1] and Tu Minh Phuong[2]

[1] R&D Lab, Vietnam Communication Corporation, Hanoi, Vietnam
phuongdoviet@hotmail.com
[2] Department of Computer Science,
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
phuongtm@ptit.edu.vn

Abstract. Demographic attributes such as gender and age of Internet users provide important information for marketing, personalization, and user behavior research. This paper addresses the problem of predicting users' gender based on browsing history. We employ a classification-based approach to the problem and investigate a number of features derived from browsing log data. We show that high-level content features such as topics or categories are very predictive of gender and combining such features with features derived from access times and browsing patterns leads to significant improvements in prediction accuracy. We empirically verified the effectiveness of the method on real datasets from Vietnamese online media. The method substantially outperformed a baseline, and achieved a macro-averaged F1 score of 0.805. Experimental results also demonstrate the effectiveness of combining different feature types: a combination of features achieved 12% improvement of F1 score over the best performing individual feature type.

Keywords: Gender prediction, browsing history, classification

1 Introduction

The effectiveness of many web applications such as online marketing, recommendation, search engines largely depends on their ability to provide personalized services. An example of such personalization is in behavioral targeting. Behavioral targeting is a technique for online advertising that helps advertisers to match advertisements to proper users based on the user behaviors when using Internet. According to a recent study [21], behavior-based ads have gained a significant business traffic improvement over simple web ads. These systems rely heavily on prior search queries, locations, and demographic attributes to provide personalized targeting.

For personalized services, many user features have been explored, among them demographic attributes such as gender and age have been shown to be of great value. In many cases, however, such information is not explicitly available or is incomplete because users are not willing to provide. Prediction of these features is an alternative way to get the missing information, which has generated significant research interest and practical values. Previous studies on demographic prediction have focused on analyzing blogs or reviews to predict the gender and age of their authors [8,15] . However, a major part of Internet users do not write blogs and reviews, thus limiting the applicability of

such methods. Another approach is to predict user gender and age from web page click-through log data. The main advantage of this approach is that such data is available for most Internet users, who browse pages for news, music, videos, products, etc. [10,11] . To predict gender and age from browsing data, a common strategy is to treat webpages and their elements (such as words, categories, and hyperlinks) as hidden attributes, and use these attributes to propagate demographic information between users.

In this paper, we present a method for predicting user gender from browsing history data. We cast the problem as an obvious binary classification problem. We extract different features from browsing log data and use these features to represent users. In [10], Hu and colleges noted that training a classifier in the user side using low-level textual features from visited webpages may not provide accurate predictions. In this study we show that by using high-level content features, namely topics and categories of webpages, and combining these features with access times and order of viewing pages, the user centered classification approach is quite competitive and can achieve state-of-the-art performance. We conducted experiments on a dataset collected from Vietnamese web-sites to verify the effectiveness of the proposed method. The results show that combining different feature types leads to significant improvement over baseline algorithms. The method achieved a macro-average F1 score of 0.805.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the proposed method in detail. Section 4 present experimental study and results. Section 5 concludes the paper.

2 Related work

There are two main groups of methods for predicting demographic attributes. The first group relies on textual data users have written such as reviews, blogs, comments, tweets, emails. Methods of this group focus on the analysis of writing style or speaking style of the user and its association with demographic attributes. Herring and Paolillo studied the prediction of gender from traditional, well written documents [8]. Herring et al. [9], and Yan et al. [23] investigated the identification of gender from blogs. Nowsom *et al*. used n-gram features of the user's post to the forums with 1,400 posts of 47 men and 450 women [14]. Burger *el al*, predicted gender of Twitter users by analyzing content of tweets and additional features [2]. They use several types of features including n-grams, name and descriptions. Another work that used other features, in addition to the text content of blogs, was proposed by Rosenthal et al. [19]. Otterbacher combined writing style, words, and metadata for predicting gender of movie review users [15]. Recently, researchers started to investigate the identification of gender for Twitter users [1,16]. In addition to writing style, speaking style has also been exploited to infer user gender from conversational data [6,7].

Methods of the second group are based on browsing behaviors to predict demographic attributes of the user. Pages or multimedia content the user has visited are used to propagate demographic information from users for which such information is known to other users viewing the same content. Filippova studied the detection of user gender from both user comments on videos and the relationship of the people watching the same video [5]. Hu *et al*. used data from browsing log data of users whose gender is

known to infer the demographic characteristics associated with elements of webpages the users visited [10]. They then used these demographic characteristics to predict the gender and age of users visiting the same webpages. Kabbur *et al*. followed a similar approach: they used machine learning techniques to predict demographic attributes associated with different elements of web-sites such as words, HTML elements, hyperlinks, but not requiring information from web-sites' users [11].

Our method presented here belongs to the second group of methods in that it relies on browsing behaviors of users. However, in contrast to methods presented in [10] and [11], we do not learn demographic attributes of web-sites but instead train a classifier in the user side directly with appropriately chosen features.

3 Methods

3.1 General framework

We assume that we are given a set of users for which we know the gender. Also, for each of those users as well as users whose gender is unknown we are given the user's browsing history in forms of a sequence of visited webpages $(p_1, \ldots, p_n)$ along with times when the pages were visited $(t_1, \ldots, t_n)$ (number *n* of visited pages can be different from user to user). Note that such a sequence can be derived from multiple browsing sessions of the user by concatenating the browsing sequences of all sessions. The problem is to predict the gender for new users, whose gender is unknown.

We cast the problem of predicting user gender as a binary classification problem with two class labels: male and female. From the sequence of accessed pages we derive a set of features to represent the user. We choose *Support Vector Machines* (SVM) as the classification algorithm due to its superior performance in a number of applications. In the next sections we will describe features for use with SVM in detail.

3.2 Features

An important step in classification-based methods is to select features to represent users. A straightforward way is to represent each user by the word content of all the pages the user has visited. However, as noted by Hu *et al*. [10] and confirmed by our experiments, using only words as features gives poor prediction accuracy. Thus, for accurate predictions, we need to consider other sets of features.

In this study, we use the content of webpages visited to represent users. However, instead of using word features directly, we generate several types of higher-level content based features, namely the category to which a page belongs to and the main topic of the page. We also augment the content-based features by considering the access times and the order of pages (or more precisely their categories) visited. In this section, we describe each type of features in detail.

Category-based features. Most news websites organize their pages in categories, for examples sports, political, entertainments. Many other types of web pages such as forums, e-commerce pages are also classified into categories for convenient browsing. It

is often observed that men and women are biased toward different categories [3]. For example, sport category has more male readers whereas fashion category has more female readers. To utilize this observation in predicting user gender, we first determine the categories of pages a user has visited and create category based features.

Mapping webpages to standard categories.

In practice, each web-site uses its own collection of categories which may be different from those of other websites. Thus, we need to define a standard unified collection of categories and map pages from different publishers into this standard collection. In this work, we use the category scheme provided by Wada (`http://wada.vn`) for this purpose. Wada is a search engine and web directory specially developed for Vietnamese webpages and currently is the best maintained web directory in Vietnam. Wada's directory has two levels: the first level consists of 12 categories which are further divided into 126 subcategories in the second level. Table 1 shows the categories provided by Wada web directory. Due to space limit, the table shows only subcategories from "Sports" as an example of second level subcategories.

Table 1. The standard collection of 12 categories. Subcategories are shown only for "Sports"

Category
Thể thao (Sports); Xã hội (Social); Kinh tế – Kinh doanh (Economy - Business); Sức khỏe (Health) ; Khoa học – Giáo dục (Science - Education); Văn hóa – Nghệ thuật (Culture and Arts); Thời trang – Làm đẹp (Fashion – Beauty); Hi-Tech; Gia đình (Family); Ô tô – Xe máy (Cars – Motorcycles); Không gian sống (Living space); Giải trí – Du lịch (Entertainment)
Subcategories of *Thể thao* (Sports)
Bóng đá(Football); Bóng Chuyền(Volleyball); Võ thuật (Martial arts); Golf; Yoga; Billiards; Bóng bàn (Table tennis); Quần vợt (Tennis); Bơi lội (Swimming); Cầu lông (Badminton); Cờ vua (Chess); Đua xe (Racing); Mạo hiểm (Adventure); Others

We use the following procedure to map a webpage to one of standard categories/ subcategories:

- If the page has already been assigned a category/subcategory by the web-site it comes from, and the original category/subcategory is equivalent to one of standard categories/subcategories then the page is assigned the equivalent category/subcategory. Note that if a page is mapped to a standard subcategory then its category is the parent category of the mapped subcategory. Table 2 shows examples of categories from Vietnamese sites which can be mapped to their equivalent categories from the standard collection.
- If the page does not have its own category/subcategory or the original page category/subcategory does not have its equivalence from the standard category collection then we use an automated classifier to classify the page into one of standard (sub) categories. Specifically, we retrieved 694628 articles from (`http://www.soha.vn`). These articles have already been classified by Wada, thus we used them as training data. We used the TF-IDF scheme to represent articles and used a Naïve

Bayes classifier implementation from library Minorthird (`http://minorthird.sourgeforge.net`) for training and prediction.

Table 2. Examples of original (sub)categories and their equivalent standard (sub)categories

Standard category	Original category	Standard sub-category	Original subcategory
Sports	thethao.vnepxress.net/*	Football	*/bongdaplus.vn/*
	thethao24.tv/*	Football	*/tin-tuc/bong-da-quoc-te/*
	dantri.com.vn/the-thao/*	Tennis	*/tin-tuc/tennis/*
	kenh14.vn/sport/*	Tennis	*/tennis-dua-xe/*
Hi-Tech	dantri.com.vn/suc-manh-so/*	Computer	*/vi-tinh/*
	duylinh.vn/*	Computer	*/thiet-bi-vi-tinh/*
	www.ictnews.vn/*	Devices	*/dien-thoai/*
	http://vozforums.com/*	Devices	*/may-tinh-bang/*

Creating category-based features.

Using standard categories assigned to the pages a user has visited, we create category-based features for the user as follows. Assume the user has visited n pages denoted by $(p_1, \ldots, p_n)$ and their categories are $(c_1, \ldots, c_n)$, where c_i can be one of the 12 standard categories. We count the number of times each of the 12 standard categories occurs in $(c_1, \ldots, c_n)$ and put the counts together to form a vector of 12 elements. We normalize the vector so the elements sum to one and use the normalized vector as category-based features.

We apply a similar procedure to standard subcategories assigned to the pages to create additional 126 subcategory-based features for the user.

Topic-based features. The categories and subcategories used in the previous section are created manually and provide only a coarse-grained categorization of webpage contents. It is useful to consider other methods, preferably automated ones, to organize pages into categories at finer levels of granularity. Here, we adopt *latent Dirichlet allocation* (LDA) [1], a widely used topic modeling technique, to infer topics from textual content of web pages and use the topics to create additional features. We use features created this way to augment the feature set generated from categories as described in the previous section.

In LDA, a topic is a distribution over words from a fixed vocabulary, where highly probable words tend to co-occur frequently in documents about this topic. The main idea of topic modeling is to see documents as mixtures of topics. A topic model defines a topic structure for a collection of documents (or a corpus), and a stochastic procedure to generate documents according to the topic structure. Documents in a collection are generated using the following procedure. First, the model generates a set of topics, which will be shared by all the documents in the collection. Second, the model generates each document in the following two-step process. In the first step, the model randomly

selects a distribution θ over topics. Then, for each word of the document, the model randomly picks a topic according to θ and randomly draws a word from the word distribution of the selected topic. Since only the textual content of documents are observed while the other information such as topics, the proportion of topics in a given document, and topic assignment for each word are hidden, the central computational problem of LDA is to infer those structures from observed document content. This process is often known as learning topic model and can be performed by using approximate algorithms such as variational inference [1] or Gibbs sampling [22].

We use the following procedure to assign topics for web pages accessed by a given user:

- First, we retrieve a set of web pages to form a corpus. We use LDA as the model and use the Gibbs sampling algorithm described in [22] to learn topic structures from the corpus. Note that this step is performed only once and the learned model will be used for all users.
- Next, for each web page accessed by the user, we use the topic model learned in the previous step to infer the topic proportion and topic assignment for words within this page.
- Finally, for each page, we use the topic with the highest probability from the topic distribution inferred in the previous step as the topic of the page.

This procedure maps each accessed page into one of K topics, where K is the prechosen topic number. For the given user, we count the number of times the user has accessed each of K topics. The vector of these counts is then normalized to form a distribution, i.e. the vector elements sum to one. We use the normalized count vector elements as topic-based features for the given user.

Time features. The next type of features to consider is time features. The motivation for using time features is based on the assumption that men and women have different time patterns when surfing web. For example, because women usually spend more time for preparing dinner than men, it can be expected that women access Internet less frequently then men at 6 pm – 8 pm time slot.

For each page a user has accessed, we record the access time. We use one-hour intervals to represent time, thus access time can get an integer value from [0,...,23] corresponding to 24 hours of a day. For the given user, we count the number of times the user has clicked a page in each of 24 time intervals, resulting in a vector with 24 elements. We normalize this vector so that the elements sum to one and used the normalized vector as time features.

Sequential features. Besides the actual pages (or more precisely, their categories) a user has viewed, we hypothesize that the order of viewing is also influenced by the user's gender. For example, men tend to change between categories more frequently while browsing than women do. To verify this hypothesis, we create features to represent the order of page categories a user has viewed and experimentally study the impact of such feature on the prediction accuracy.

Given a sequence of pages $(p_1, \ldots, p_n)$ a user has viewed, we first determine their categories $(c_1, \ldots, c_n)$ as described in one of previous sections. From this sequence, we extract all *k*-grams, where each *k*-gram is a subsequence of length *k* with *k* being a pre-specified parameter. Formally, a *k*-gram starting at position *i* is the subsequence $(c_i, c_{i+1}, \ldots, c_{i+k-1})$. Note that if the user's access history has been recorded from multiple sessions then we do not consider *k*-grams that cross session borders, i.e. only *k*-grams belonging to single sessions are counted.

From 12 standard categories, it is possible to generate 12^k *k*-grams. To reduce the complexity and exclude irrelevant k-grams, we use a feature selection technique based on mutual information to select *m k*-grams that have maximum correlation with gender classes. The mutual information between *k*-gram *S* and gender class *G* is computed as follows:

$$I(G,S) = \sum_{g \in G} \sum_{s \in S,\ s \neq 0} p(g,s) \log \left(\frac{p(g,s)}{p(g)p(s)} \right) \quad (1)$$

where $G = \{\text{male, female}\}$, and we only sum over users with nonzero values of *S*, i.e. users with *k*-gram *S* occurring in their access sequences.

We select *m* (*m* is a parameter of the method) *k*-grams with the highest values of mutual information to create so called *sequential features* as follows. For a given user, the number of times each of the *m k*-grams occurs in her browsing history is counted. This results in a vector of size *m*, which is then normalized to form a distribution. The elements of this vector are then used as sequential features for SVM.

3.3 Combining features

We experimented with different combinations of features. There are a number of ways to combine features of different types for SVM classifiers. For example, one can compute a separate kernel for each type of features and then take a (weighted) linear combination of kernels. Another method is to train a separate classifier for each type of features and use a Bayesian framework to combine the classifiers' output. In this study, we simply concatenate vectors of different feature types to form a unified feature vector.

We recall that feature vector of each type has been normalized so that the elements sum to one. This guarantees equal contributions of all feature types and allows avoiding the excessive influence of features with large values. We also normalize the final feature vector to have unit norm.

4 Experiments and results

4.1 Experimental setup

Data. We collected data for our experiments from users of different services offered by VCcorp (`http://vccorp.vn`). VCcorp is one of the biggest online media companies in Vietnam which operates several popular websites including news (`http://kenh14.vn`,`http://dantri.com.vn`), e-commerce websites (`http://enbac.com`,`http://rongbay.com`), social media, digital content (`http://soha.vn`). VCcorp is also an advertising network with online advertising system Admicro.

We used a set of users with known gender to create training and test sets for the experimented methods. The information about those users was collected via MingId (`http://id.ming.vn`). MingId is a *Single Sign On* system that serves as a unified account management service for entire VCcorp eco-system. A user with a MingId account can log in to any account-required service offered by VCcorp. MingId has more than two million registered users, among them about 200 thousands users log in regularly. MingId uses several rules to check if information provided by registered users is reliable. We removed users with unreliable account information, as detected by the system, and users who rarely use Internet (who visit less than 20 pages in a month), thus retained 150 thousands users for experiments. Among those users, 97 thousands are male and the remaining 53 thousands are female.

We collected browsing history of these users for a period of one month and used the collected data to create features as described in the previous sections. The same set of 694628 news articles from `http://www.soha.vn`, as used for training the Naïve Bayes classifier in section 3.2, was used for training LDA.

Evaluation metrics. We judged the performance of the proposed and a baseline method in terms of precision *p*, recall *r*, and F1 score. For each class (male, female), precision is defined as the number of correctly predicted cases divided by the number of all predictions of this class. Recall is defined as the number of correctly predicted cases divided by the number of all cases of this class. F1 is the harmonic mean of precision and recall:

$$F1 = \frac{2pr}{p+r}$$

We also report macro-averaged F1 score, which is the average of F1 scores of two classes: male and female.

Evaluation and settings. We used 10-fold cross-validation to evaluate the merits of each method under test. The set of all users was divided into 10 subsets with equal numbers of male and female users. One subset was held out as the test set and the remaining subsets were used as the training set. We repeated this procedure ten times for each subset held out as the test set and reported averaged results.

We used the implementation of SVM provided in LIBSVM library (`http://www.csie.ntu.edu.tw/~cjlin/libsvm/`) with RBF kernel. Two parameters *C* and γ were selected by running the tool grid.py provided with LIBSVM on a held-out dataset and then used for all experiments. For LDA, we used the Matlab topic modeling toolbox by Steyvers and Griffiths (`http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm`) for training and inference. Following [22], parameters α and β of LDA were set to 50/K and 0.01 respectively, where *K* is the number of topics. The selection and effect of *K* will be discussed in the next section. We used bi-grams for sequential features and selected 31 bi-grams with the highest mutual information values. The number of bi-grams was chosen using the same small held-out set that was used to select LDA parameters.

Comparison and baselines. We tested our method with different combinations of feature types and compared the method with the one proposed by Hu *et al*. [10] using the same cross-validation setting. Hu and colleges described a method consisting of two parts: First, they represented webpages by different content-based features and used support vector regression to predict demographic tendencies for each page. The predicted tendencies of webpages are then used to predict gender for users by aggregating over visited webpages. Second, they leveraged similarity among users and webpages to deal with data sparseness. When applying only the first part, they achieved a macro-averaged F1 score of 0.703. Adding the second part improved the F1 score up to 0.797.

Due to lack of data, we could not re-implement the second part. In our experiments, we re-implemented the first part of their method and used it for comparison.

4.2 Results

Effect of topic number for topic-based features. We first studied the effect of number of topics K on the performance of topic-based features. We experimented with different values of K ranging from 50 to 200. For each value of K we learned the topic structure and used only topic features to represent users. Fig. 1 shows the macro-averaged F1 scores of the method when using only topic-based features with different number of topics.

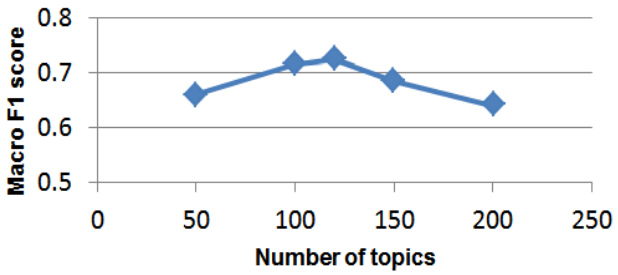

Fig. 1. F1 scores for different topic numbers when using only topic-based features

The results show that $K = 120$ gives the highest F1 score. Interestingly, this number of topic is very close to the number of subcategories in our standard category collection, although an inspection of important words in each topic shows that not all topics have subcategories with the same meaning. In all experiments reported in the following section, we used $K = 120$, which corresponds to 120 topic-based features.

Performance of individual feature types. In the next experiment, we compared the performance of each feature type when used individually. We used features of each type as input for SVM and reported the recalls, precisions, and F1 scores averaged over 10 folds in table 3.

Table 3. Performance of individual feature types. The numbers of feature are in parentheses

Type of feature	Gender	p	r	F1	Macro F1
Sequential features (31)	Male	0.6	0.561	0.58	0.595
	Female	0.62	0.61	0.61	
Time features (24)	Male	0.684	0.667	0.68	0.665
	Female	0.661	0.64	0.65	
Category-based features (12)	Male	0.66	0.67	0.66	0.66
	Female	0.67	0.653	0.66	
Category-based and sub category-based features (12+126)	Male	0.72	0.71	0.71	0.7
	Female	0.7	0.68	0.69	
Topic-based features (120)	Male	0.73	0.708	0.72	0.725
	Female	0.74	0.726	0.73	

As shown in the table, sequential features gave the lowest F1 scores (macro F1 is only 0.595) while the highest macro-averaged F1 score of 0.725 was achieved with 120 topic-based features. The relatively low F1 score of sequential features show that the order of page views is not very relevant to user gender. Simple time features performed surprisingly well, achieving macro F1 of 0.665, higher than F1 score of 0.66 achieved when using 12 features based on categories. This result supports the intuition that male and female users browse the Web at different times. It is interesting to note that topic-based features, which have been generated automatically by LDA, resulted in more accurate predictions than features from manually created categories (0.725 vs 0.7 macro F1 scores).

Combining features and comparison with method by Hu et al. In the next experiment, we measured the performance of different combinations of feature types and compared with that of method by Hu *et al.* (without the second part), which serves as the baseline in this experiment. Since topic-based features have shown the best performance in previous experiments, we used them as the base features and gradually added features of other types. The results are summarized in table 4. The results show that our method outperformed the baseline in term of F1 score when using any combination of topic–based and other features. It is also observed that adding more features resulted in accuracy improvement. The best macro F1 score of 0.805 was achieved when combining all the four types of features or when using just topic–based, sequential, and time features. This is 12% improvement over the best individual feature type and 15% improvement over the baseline. Note that, Hu *et al*. achieved F1 score of 0.703 in their dataset when using only the first part of their method which is the same as implemented in our baseline. When adding the second part (smoothing by leveraging similarity among users and webpages), they achieved F1 score of 0.797. In our dataset, the baseline achieved F1 score of 0.695 (very close to their reported 0.703), while our method achieved the F1 score of 0.805. We note that it is not possible to compare the proposed method and full implementation of method by Hu *et al*. in this way and the reported results just show how our method improves over the baseline.

Table 4. Performance of different combinations of feature types

Type of feature	Gender	p	r	F1	Macro F1
Baseline (first part of method by Hu *et al*. [10])	Male	0.68	0.72	0.7	0.695
	Female	0.67	0.71	0.69	
Topic-based + Time	Male	0.762	0.758	0.76	0.75
	Female	0.74	0.736	0.74	
Topic-based + Sequential	Male	0.747	0.72	0.73	0.735
	Female	0.737	0.74	0.74	
Topic-based + Time + Sequential	Male	0.821	0.8	0.81	0.805
	Female	0.81	0.79	0.8	
All features	Male	0.82	0.807	0.81	0.805
	Female	0.81	0.8	0.8	

5 Conclusion

We have proposed a method for predicting the gender of Internet users based on browsing behavior. From browsing log data of users the method extracts several types of features including high-level content features and time features. The features are used as input for SVM based classification. Experimental studies with real data show that our method outperformed a baseline method by a large margin. The experimental results also show the usefulness of combining different types of features for improved prediction accuracy. Although we experimented only with gender prediction, the method can be extended to predict other demographic information such as age.

Acknowledgments. Financial support for this work was provided by FPT Software. We thank VCcorp for providing data.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I. (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research 3(2003) page 993-1022.
2. Burger, J. D., J. Henderson, G. Kim & G. Zarrella (2011). Discriminating gender on Twitter. In Proc. of EMNLP-11, pp. 1301–1309.
3. Computerworld Report: Men Want Facts, Women Seek Personal Connections on Web,`http://www.computerworld.com/s/article/107391/Study_Men_want_facts_women_seek_personal_connections_on_Web` .
4. Ellist, D. (2009). Social (distributed) language modeling, clustering and dialectometry. In Proc. Of TextGraphs at ACL-IJCNLP-09, pp. 1–4.
5. Filippova, K. (2012), User demographics and language in an implicit social network .Proceedings of EMNLP-CoNLL '12 Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning Pages 1478-1488
6. Garera, N. & D. Yarowsky (2009). Modeling latent biographic attributes in conversational genres. In Proc. of ACL-IJCNLP-09, pp. 710–718.

7. Gillick, D. (2010). Can conversational word usage be used to predict speaker demographics? In Proceedings of Interspeech, Makuhari, Japan, 2010.
8. Herring, S. C. & J. C. Paolillo (2010). Gender and genre variation in weblogs. Journal of Sociolinguistics, 10(4):710-718.
9. Herring, S. C., L. A. Scheidt, S. Bonus & E. Wright(2004). Bridging the gap: A genre analysis of weblogs. In HICSS-04.
10. Hu, J., Zeng, H.J., Li, H., Niu, C., Chen, Z. (2007). Demographic prediction based on user's browsing behavior. Proceedings of the 16th international conference on World Wide Web. Pages 151-160.
11. Kabbur, S., Han, E.H., Karypis, G., (2010). Content-based methods for predicting web-site demographic attributes. Proceedings of ICDM 2010.
12. MacKinnon, I. & R. Warren (2006). Age and geographic inferences of the LiveJournal social network. In Statistical Network Analysis: Models, Issues, and New Directions Workshop at ICML-2006, Pittsburgh, PA, 29 June, 2006.
13. Mulac, A., D. R. Seibold & J. R. Farris (2000). Female and male managers' and professionals' criticism giving: Differences in language use and effects. Journal of Language and Social Psychology,19(4):389–415.
14. Nowson, S. & J. Oberlander (2006). The identity of bloggers: Openness and gender in personal weblogs. In Proceedings of the AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs, Stanford, CA, 27-29 March 2006, pp.163–167.
15. Otterbacher, J. 2010. Inferring Gender of Movie Reviewers: Exploiting Writing Style, Content and Metadata. In Proceedings of CIKM 2010.
16. Pennachiotti, M., and Popescu, A.M, (2011). A machine learning approach to Twitter user classification. Proceedings of AAAI 2011.
17. Phuong, D.V., Phuong, T.M. (2012). A keyword-topic model for contextual advertising. Proceedings of SoICT 2012.
18. Popescu, A. & G. Grefenstette (2010). Mining user home location and gender from Flickr tags. In Proc. of ICWSM-10, pp. 1873–1876.
19. Rosenthal, S. & K. McKeown (2011). Age prediction in blogs: A study of style, content, and online behavior in pre- and post-social media generations. In Proc. of ACL-11, pp. 763–772.
20. Schler, J., M. Koppel, S. Argamon & J. Pennebaker (2006). Effects of age and gender on blogging. In Proceedings of the AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs, Stanford, CA, 27-29 March 2006, pp. 199–205.
21. Search Engine Watch Journal, Behavioral Targeting and Contextual Advertising, http://www.searchenginejournal.com/?p=836
22. Steyvers. M., Smyth, P., Rosen-Zvi, M., Griffiths T. (2004). Probabilistic author-topic models for information discovery. In Processing KDD'04 (ACM New York, NY, USA)
23. Yan, X. & L. Yan (2006). Gender classification of weblogs authors. In Proceedings of the AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs, Stanford, CA, 27-29 March 2006, pp. 228–230.