

Classifying *Second Life* Player Gender Using Chat Data

Antonio Ricciardi
Stanford University
Computer Science Department
Stanford, CA 94305
aricciardi@stanford.edu

ABSTRACT

The goal of this study was to predict the genders of players of the online game *Second Life* using linguistic patterns from their chat data. This was accomplished using a rich set of stylistic features combined with various machine learning models. This project builds upon a previous study done at Stanford's Virtual Human Interaction Lab in which very few linguistic features were used. Results show that adding the new features significantly improves prediction accuracy.

1. INTRODUCTION

Last year a study done at Stanford's Virtual Human Interaction Lab tracked a group of 76 participants for an average of 6 hours per week for 6 weeks as they played the online game *Second Life*. During this time the researchers gathered an enormous amount of data, including the players' movements, actions and chat messages. The participants were also required to fill out behavioral and demographic questionnaires containing such information as gender, race, social openness, and self-esteem.

The goal of their study was to predict the personal information gathered from the questionnaires using a set of features extracted from the *Second Life* data. This would provide insight into how certain behaviors in virtual worlds are influenced by a participant's gender, race, and personal qualities. The features included total time spent typing, number of teleports, area traveled, and many others. However, the only linguistic features used in the task of gender prediction were the average number of words used per message and the total number of distinct

words used. Using this set of features they were able to achieve reasonably good prediction accuracies (see section 6 for old and new accuracies).

This project built upon that earlier study by expanding the set of features to include many new stylistic traits of the players' chat behavior. I also experimented with several new machine learning models better suited to natural language processing tasks than those used in the previous study.

2. PREVIOUS WORK

I approached the problem of predicting a player's gender as a matter of text classification, where a document is the cumulative chat data for a single *Second Life* player. While genre classification and author attribution have been studied heavily in literature [4], the idea of classifying texts according to author gender is much less prevalent.

Koppel et. al. [1] showed that many techniques used in topic categorization and author attribution can also be applied to classifying texts according to author gender. However, the corpus used in this study was composed of long narratives exhibiting complex sentence structure. The features appropriate for such a corpus differ significantly from those appropriate for chat data (see section 3).

Kucukyilmaz et. al. [2] applied the idea of gender prediction to chat data collected from the Heaven BBS chat server. This study provided a starting point in terms of feature ideas and machine learning models appropriate to the task of *Second Life* player classification. However, there are

several differences between this study and my own. First, the chat messages in the Heaven BBS data are written in Turkish, whereas the *Second Life* data is in English. More importantly, the domains of the datasets are significantly different; the Heaven BBS server is devoted entirely to peer-to-peer communication, whereas *Second Life* is a large virtual world with unique social norms, slang, and topics of conversation.

3. DATASET

3.1 Differences from Typical Corpora

The chat data collected from the *Second Life* players differs significantly from corpora typically used for text categorization and author attribution problems. First, the chat data for any particular player is a collection of many small conversations, rather than a coherent sequence of sentences related to the same topic. Moreover, even a sequence of chat messages from the same conversation will often be discontinuous, since only one side of the conversation is recorded. Consequently, the linguistic features used must be extremely local, rather than relying on the larger context in which words were typed (see section 4 for examples).

Another important characteristic of chat data is the use of punctuation marks and individual characters. Often players will use special sequences of punctuation marks to express various emotions, such as smiley faces (e.g., “:-)”), and hearts (“<3”). Moreover, players will often overuse certain characters to show emphasis, forming an “emoticon” (e.g., “heyyyy”) [2]. Thus, limiting the features to the word-level (as is typical in text categorization) would forfeit potentially useful information. Specific features related to special characters are discussed in section 4.

In this study, all the chat messages for each player are concatenated to form a document, for a total of 76 instances. The dataset differs significantly from those of typical author attribution tasks in that documents from the same category are written by different people. Stylistic

similarities are much less consistent across large classes of authors than they are for documents written by the same author [1]. However, as evidenced by the results (see section 6), using stylistic features to classify players into larger groups still works reasonably well.

3.2 Preprocessing

Some of the participants engaged in very few chat conversations over the course of the 6-week tracking period, whereas others chatted extremely often. Consequently, the number of chat messages sent differed in orders of magnitude between some participants, ranging from 50 to 2500 (with an average of about 400). Moreover, there were significantly more males than females in the dataset (51 versus 25). In the original *Second Life* study, all 76 instances were used in the prediction task. For consistency, I did not remove any of the instances when comparing old and new results. However, I also report the prediction accuracy after removing participants with fewer than 200 chat messages and performing random undersampling to even out the numbers of males and females (see section 6). In addition, to further account for the wide range of document sizes, many of my features are scaled by the size of a player document. Specifically, all word counts are normalized by the total number of words and all character counts by the total number of characters.

Many of the features in this project involved counting instances of specific words. In order to ensure that two identical words were not differentiated based on capitalization and surrounding punctuation, all capitalization and special characters were removed before counting the words in each message. However, since capitalization and special characters may also provide useful features, the original message is preserved as well.

4. FEATURES

The process of generating features from a chat document is done in two phases. First, roughly 10,000 features are extracted from each

document in the training set. These features are then filtered using standard feature selection methods, reducing the size to less than 300. A comprehensive list of all the new features (in order of chi-squared scores) is provided in the appendix.

4.1 Feature Engineering

The only linguistic features used in the original *Second Life* study for predicting gender were the average number of words used per message and the total number of distinct words used. Without the additional features provided by non-linguistic data, this was not enough information to predict a player's gender with reasonable accuracy. Below I describe how I extended the feature set to significantly improve the results.

4.1.1.1 Stop Words

Function word frequencies have long been used in author attribution tasks as features of style [4]. However, it is common practice in chat messaging to drop certain function words, either by absorbing them into contractions or omitting them altogether. Some examples are:

- "see you guys later"
 - omits "I will"
- "time to ride"
 - omits "it is"
- "oh sweet"
 - omits "it is"
- "whats up"
 - absorbs "is" into "what"

In addition, certain chat slang words, such as "lol" (short for "Laugh Out Loud") are used commonly enough to be considered stop words, though they are largely non-existent in datasets from other domains.

For these reasons, my initial attempts to hard-code a list of typical function words for which to count frequencies resulted in poor performance. Many such words were not useful, and many potentially useful words were left out. My solution was to create a domain-specific set of

stop words by taking the union of the top 100 words for each gender.

This set ended up containing many surprising words. From the ranked list of features in the appendix, we see the following stop-word features in the top 10:

4. frequency_of_the_word_lot
 - Males tended to use the phrase "a lot" much more often than females.
5. frequency_of_the_word_get
 - Males tended to say phrases such as "how do you get" when asking for help.
7. frequency_of_the_word_ass
 - Profanity words were much more common among males than females.
8. frequency_of_the_word_ohh
 - See the emoticon section below.

For the reasons described above, such stop-word frequencies were extremely useful for distinguishing between genders.

4.1.1.2 Smileys

In this dataset, smileys, hearts, and other special sequences of characters were quite rare. Consequently, I initially tried combining the counts of all such sequences into a single feature for each chat document. However, the difference between genders was not significant enough to make this feature useful. After examining the data further, however, it became apparent that certain smileys were used more commonly by one gender than the other. In particular, females tended to use the smileys ":" and ":p" much more often than males. Thus, my final set of features treated smileys individually rather than grouping them all together, resulting in such features as:

- frequency_of_the_word_:p
- frequency_of_the_ngram_hi_:

These features proved to be much more useful than the original combined feature.

4.1.1.3 Emoticons

As with smileys, I initially attempted to count the total number of emoticons used by each participant, rather than counting them individually. It seemed unreasonable to treat each emoticon separately, since players tend to use varying numbers of extra characters, even for the same word (e.g., “ahh”, “ahhh”, “ahhhh”), resulting in very sparse counts.

Surprisingly, I found that counting each emoticon individually produced better results than combining them all into a single feature. Players from both genders tended to create emoticons from a wide variety of words, with the total number of emoticons not differing significantly between genders. However, females tended to be much more consistent about the number of extra trailing characters applied to a word. Consequently, the following features were very useful for identifying females:

- frequency_of_the_word_ohh
- frequency_of_the_word_heyyyy

4.1.1.4 Character Counts

Due to the nature of chat data (see section 3), I suspected that character-based features would be much more useful in this task than they typically are in author attribution problems. Consequently, I kept counts of all letters and digits, as well as most punctuation marks. Surprisingly, very few of the character counts showed noticeable differences between genders. Moreover, those that did, specifically “e”, “m”, and “)”, can all be explained at the word level. In particular, the list of top words for males contained many more e’s and m’s than those for females (e.g., “me”, “get”, “hey”) while females tended to use the smiley “:)” much more often than males. Thus, character-based features were redundant and often resulted in overfitting rather than improving results.

4.1.1.5 Contractions

Initially, I did not include any special features for contractions, and instead treated them as normal words. However, after observing the top words

for each gender, I noticed that many contractions were appearing with unexpectedly low frequencies. This was due to the fact that some players would use apostrophes, whereas others would not, resulting in split counts for each contraction. In particular, the tokens “its” and “it’s” are used as a contraction for “it is” with nearly identical frequencies in the dataset, even though omitting the apostrophe may create confusion with the possessive pronoun “its”.

To address this issue, I added together counts for each contraction with and without the apostrophe. This noticeably increased the number of useful features involving contractions, such as “uses_its_more_often_than_get” (see Word Frequency Comparisons below for an explanation of what this feature means).

4.1.1.6 Word Frequency Comparisons

Stop word counts are useful features when a word is used relatively frequently compared to all other words in the vocabulary. However, such features do not explicitly capture differences in frequencies between two particular words. After adding features for stop word frequencies, I noticed that the ordering of the top words for each chat document often correlated with the participant’s gender. For example, males would often use the words “hey” and “nice” much more frequently than “hi” and “cool”, respectively, whereas the opposite was true for females. To capture such information, I added a binary feature for each pair of stop words which equals 1 whenever the first word occurs more frequently than the second (and 0 otherwise).

Adding these features greatly increased prediction accuracy, as evidenced by the fact that this is the most common feature type in the appendix. In particular, the feature “uses_thanks_more_often_than_get” had the highest chi-squared score. This was due to the fact that males tended to use the word “get” much more often than females when asking for help (e.g., “how do you get...”), yet both genders tended to say “thanks” after receiving help. Since the word “get” is used in other contexts as well, its total

frequency was much higher than “thanks” for many males.

In addition to comparing frequencies for stop words, I also added features for comparing how often each contraction appeared with and without an apostrophe. Players tend to be consistent about whether they use apostrophes for particular contractions, and it seemed plausible that such patterns could persist for a particular gender as well. In general, however, contraction-comparison features did not improve results significantly.

4.1.1.7 Slang

In addition to stop words, I initially included features for a hard-coded set of slang words and abbreviations common to the chat domain. Examples of such words include “lol”, (“Laugh Out Loud”), “brb” (“Be Right Back”), and “afk” (“Away From Keyboard”). Unfortunately, “lol” was the only such term which showed any significant difference across genders, and it was already a stop word.

However, when I then added features for an additional set of slang words, these specific to *Second Life*, I began to see some slight improvement. In particular, I found that females tended to use the term “sl” (“Second Life”) more frequently than males.

4.1.1.8 N-Grams

One set of stylistic features commonly used in author attribution tasks (as well as author gender prediction [1]) is part-of-speech N-gram frequencies. However, after adding POS tags to each word and counting POS bigrams and trigrams, I found that most of the instances of any particular N-gram tended to come from the same phrase. Consequently, I removed the POS tags and simply counted bigrams and trigrams for specific sequences of words. This removed a lot of noise caused by unrelated singleton phrases that happened to share the same POS tags.

Using features for specific N-gram phrases significantly improved the results. In particular, by adding <S> and </S> tokens the beginning

and end of each message, respectively, I was able to model occurrences of particular words starting or ending a message. Some examples of useful features include:

- frequency_of_the_ngram_<S>_omg_</S>
 - Common to females.
- frequency_of_the_ngram_hi_:)
 - Common to females.
- frequency_of_the_ngram_what_up
 - Common to males.

4.1.1.9 Spelling Corrections

In a chat conversation, a conscientious speller will often correct a spelling mistake by sending an additional message with the corrected word preceded or followed by an asterisk. Interestingly, such behavior appears to be much more common for females than for males in the *Second Life* dataset. Examples include:

- "its because I have a lot of extra flech"
"*flesh"
- "i accidentally hit wuit"
"quit"
"*"

Such occurrences are captured by the feature frequency_of_the_word_*

4.2 Feature Filtering

Passing the entire set of extracted features to a learning model gives very poor results due to overfitting. In order to reduce the number of features to a reasonable number, I used two standard techniques for feature selection: chi-squared analysis and mutual information.

My implementation of these feature selection techniques is based on Manning, et. al. [3]. To compute the chi-squared score for a feature, I use the equation:

$$X^2(t) = \frac{(N_{11} + N_{10} + N_{01} + N_{00})(N_{11}N_{00} - N_{10}N_{01})^2}{(N_{11} + N_{01})(N_{11} + N_{10})(N_{10} + N_{00})(N_{01} + N_{00})}$$

In this equation, N_{ij} is the number of documents for which $e_t = i$ and $e_c = j$, where $e_t = 1$ if the

feature matches a given document (and 0 otherwise) and $e_c = 1$ if the document is in the male class (and 0 if female).

For mutual information, I use the equation:

$$I(t) = \frac{N_{11}}{N} \log_2 \left(\frac{NN_{11}}{N_1 N_1} \right) + \frac{N_{11}}{N} \log_2 \left(\frac{NN_{11}}{N_1 N_1} \right) + \frac{N_{11}}{N} \log_2 \left(\frac{NN_{11}}{N_1 N_1} \right) + \frac{N_{11}}{N} \log_2 \left(\frac{NN_{11}}{N_1 N_1} \right)$$

Note that these equations require features to be binary (either matching a document or not). Consequently, I binarized each real-valued feature before computing its chi-squared or mutual information score. This was accomplished by choosing the split value for each feature which maximized information gain, then assigning 1 or 0 to the feature if its value fell above or below that split point, respectively.

After trying both features selection techniques, I found I had better results using chi-squared scores. The final set of features was computed by removing all features whose scores fell below a certain threshold.

5. MODELS

In addition to engineering a new set of linguistic features for the *Second Life* chat data, I tried out several new machine learning models for making the gender predictions. In the original study, predictions were made using an alternating decision tree. However, after adding the new linguistic features, I found that results improved when using models more conventional for natural language processing tasks.

5.1 Naïve Bayes

I implemented a Naïve Bayes classifier which, for each gender, accumulates a count of the total value of each feature for participants of that gender. That is:

$$Count(f, C) = \sum_{d \in C} f(d),$$

where f is a feature and C is the set of documents for a particular gender. I then normalize the feature counts for each gender c to produce a probability distribution $p_c(f)$. Finally, the classifier predicts the gender for a document d as:

$$\operatorname{argmax}_c \left(\sum_f \log(p_c(f)) \times f(d) \right)$$

Note that since all possible features are known beforehand (and have non-zero probabilities), there is no need to reserve probability mass for previously unseen features.

5.2 k -NN

Another common model for text classification tasks is the k -nearest neighbors model, which predicts the class for a document using a majority vote of the k nearest training documents in vector space. I implemented this model by transforming each document into a vector of feature values. I then compute the cosine similarity between two document vectors as:

$$\operatorname{dist}(d_1, d_2) = \frac{\sum_f (f(d_1) \times f(d_2))}{\sqrt{\sum_f (f(d_1))^2} \sqrt{\sum_f (f(d_2))^2}}$$

While varying the value of k did not affect accuracy significantly, I found I got the best results with $k = 5$.

One important feature of both my Naïve Bayes and k -NN implementations is that the value of each feature was scaled to be between 0 and 1. This was particularly important for k -NN, since cosine similarity treats each dimension equally. Initially I did not scale any features, and I found that features with large values (such the average number of words per message) would often overshadow features with smaller values (such as normalized character counts). Scaling the feature values solved this problem, significantly improving prediction accuracy.

5.3 Others

In addition to Naïve Bayes and k -NN, I experimented with several other machine learning algorithms implemented in the open-source Weka software [5], including logistic regression and SMO. See section 6 for a comparison of results using the various models.

6. RESULTS

Table 1 summarizes the results for 4 classification models using differing numbers of features. The results are also plotted in Figures 1 and 2 (see Appendix A). The prediction accuracies shown represent leave-one-out cross-validation scores. I report my scores this way for two reasons: first, the data set was extremely small, and using separate training and test sets would have deprived the classifiers of enough data to learn a useful model for each gender. Secondly, the original *Second Life* study also reported leave-one-out cross-validation scores, so it is easy to compare results this way.

As shown in Table 1, prediction accuracies are highest when using a chi-squared threshold of 3 or 6, producing 238 or 157 total features, respectively. This makes sense, as many of the features with scores lower than 3 are redundant with other features, causing overfitting. Conversely, increasing the threshold beyond 6 filters out too many features and leads to underfitting.

The results show a considerable improvement over the scores achieved by the original *Second Life* study. Previously, the best score achieved, with both linguistic and non-linguistic features,

was 0.7763 (using all 76 instances). Using a chi-squared threshold of 3, I achieved an average accuracy of over 0.9, peaking at 0.9737 for Naïve Bayes. Furthermore, my classifiers were trained only on linguistic features, ignoring all non-chat data.

7. ACKNOWLEDGMENTS

I would like to thank to Maria Jabon, Nick Yee, Helen Harris, and Jeremy Bailenson for their work on the original *Second Life* study at Stanford’s Virtual Human Interaction Lab, and for providing the data to make this project possible.

8. REFERENCES

- [1] Koppel, M., S. Argamon, and A. R. Shimoni, Automatically Categorizing Written Texts by Author Gender. *Literary and Linguistic Computing*, Vol. 17, No. 4, p.401-412 (2002). <http://llc.oxfordjournals.org/cgi/content/abstract/17/4/401>.
- [2] Kucukyilmaz, T., B. B. Cambazoglu, C. Aykanat, and F. Can. *Chat Mining for Gender Prediction*. *Lecture Notes in Computer Science*, Springer, Vol. 4243, p.274-283 (2006). <http://research.yahoo.com/pub/2217>.

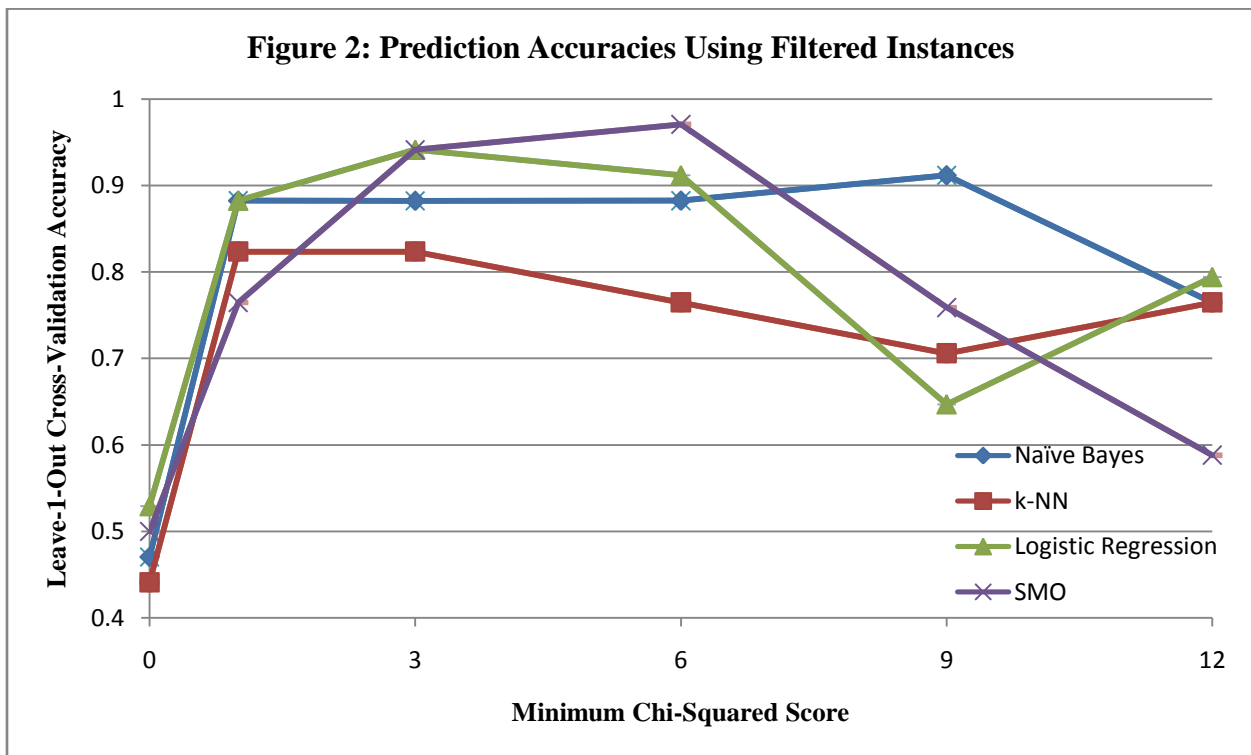
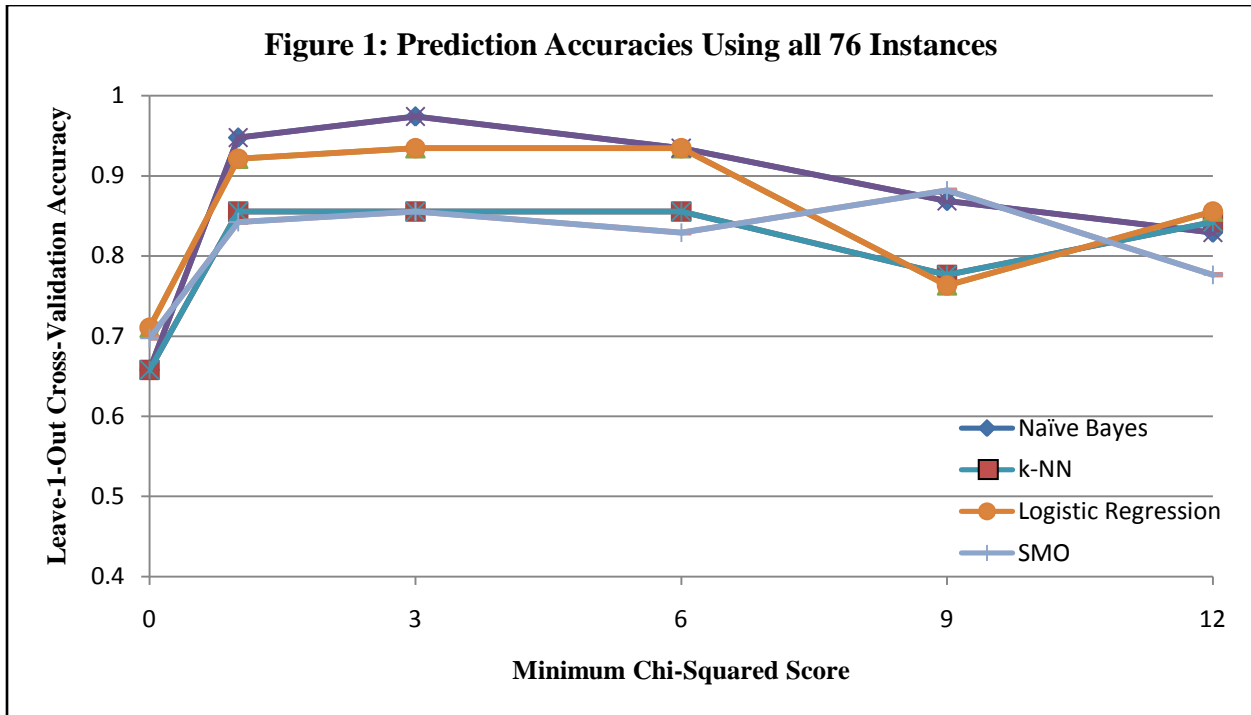
Chi-Squared Threshold	0	1	3	6	9	12
Number of Features	13,748	275	238	157	28	10
Accuracies Using All 76 Instances (51 Males, 25 Females):						
Naïve Bayes	0.6579	0.9474	0.9737	0.9342	0.8684	0.8289
k-NN	0.6579	0.8552	0.8553	0.8553	0.7763	0.8421
Logistic Regression	0.7105	0.9211	0.9342	0.9342	0.7632	0.8553
SMO	0.6974	0.8421	0.8553	0.8289	0.8816	0.7763
Accuracies After Filtering Instances (17 Males, 17 Females):						
Naïve Bayes	0.4706	0.8824	0.8823	0.8825	0.9118	0.7647
k-NN	0.4412	0.8235	0.8235	0.7647	0.7059	0.7647
Logistic Regression	0.5294	0.8824	0.9412	0.9118	0.6471	0.7941
SMO	0.5	0.7647	0.9412	0.9706	0.75882	0.5882

Table 1: Leave-1-out cross-validation accuracies after filtering out features using different minimum chi-squared scores. Accuracies are shown before and after filtering out instances from the training set that did not contain at least 200 chat messages and then randomly removing male instances until there were an equal number from each gender.

- [3] Manning, C. D., P. Raghavan and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press (2008). p.471-495 (2000).
<http://www.mitpressjournals.org/doi/abs/10.1162/089120100750105920>
- [4] Stamatatos, E., N. Fakotakis, and G. Kokkinakis. Automatic Text Categorization in Terms of Genre and Author. *Computational Linguistics*, Vol. 26, No. 4,
- [5] Witten, I. H. and E. Frank. "Data Mining: Practical machine learning tools and techniques", 2nd Ed., Morgan Kaufmann, San Francisco (2005).

APPENDIX:

A. Results



B. List of Features

1. uses_thanks_more_often_than_get
2. frequency_of_ascii_character_101
3. uses_get_more_often_than_there
4. frequency_of_the_word_lot
5. frequency_of_the_word_get
6. uses_like_more_often_than_me
7. frequency_of_the_word_ass
8. frequency_of_the_word_ohh
9. frequency_of_the_ngram_<S>_ohh
10. uses_me_more_often_than_there
11. frequency_of_the_ngram_there_a
12. uses_get_more_often_than_from
13. frequency_of_the_word_talked
14. frequency_of_the_ngram_like_my
15. frequency_of_the_word_wanna
16. frequency_of_the_word_ride
17. uses_up_more_often_than_really
18. frequency_of_the_word_guy
19. uses_oh_more_often_than_got
20. uses_thanks_more_often_than_me
21. frequency_of_the_word_room
22. uses_got_more_often_than_was
23. uses_get_more_often_than_out
24. uses_really_more_often_than_can
25. frequency_of_the_word_car
26. uses_get_more_often_than_know
27. frequency_of_the_ngram_what_up
28. uses_my_more_often_than_get
29. uses_get_more_often_than_about
30. uses_thanks_more_often_than_this
31. frequency_of_the_ngram_people_here
32. frequency_of_the_ngram_omg_</S>
33. frequency_of_the_word_lived
34. frequency_of_the_ngram_a_lot_of
35. frequency_of_the_word_ugh
36. frequency_of_the_ngram_<S>_omg_</S>
37. frequency_of_the_word_discovered
38. frequency_of_the_ngram_lot_of
39. frequency_of_the_word_sunday
40. frequency_of_the_word_chatting
41. uses_got_more_often_than_about
42. uses_right_more_often_than_about
43. uses_time_more_often_than_u
44. frequency_of_the_ngram_soon_</S>
45. frequency_of_the_word_turn
46. frequency_of_the_ngram_<S>_hold
47. frequency_of_the_ngram_<S>_what_up
48. uses_thanks_more_often_than_just
49. uses_time_more_often_than_look
50. frequency_of_the_word_perfect
51. frequency_of_the_word_weather
52. frequency_of_the_word_tomorrow
53. frequency_of_the_word_thinking
54. frequency_of_the_word_worry
55. frequency_of_the_word_21
56. uses_not_more_often_than_up
57. uses_get_more_often_than_here
58. uses_time_more_often_than_at
59. frequency_of_the_word_4
60. uses_up_more_often_than_was
61. uses_thanks_more_often_than_my
62. uses_but_more_often_than_get
63. uses_get_more_often_than_one
64. uses_this_more_often_than_about
65. uses_thanks_more_often_than_know
66. uses_get_more_often_than_and
67. uses_on_more_often_than_like
68. uses_get_more_often_than_have
69. uses_get_more_often_than_was
70. uses_are_more_often_than_me
71. frequency_of_the_ngram_come_here
72. frequency_of_the_ngram_there_</S>
73. frequency_of_the_ngram_i_went_to
74. frequency_of_the_word_jump
75. uses_thanks_more_often_than_what
76. uses_about_more_often_than_out
77. uses_cool_more_often_than_me
78. uses_did_more_often_than_get
79. uses_dont_more_often_than_me
80. uses_not_more_often_than_me
81. uses_its_more_often_than_get
82. uses_for_more_often_than_lol
83. uses_oh_more_often_than_me
84. uses_dont_more_often_than_get
85. uses_right_more_often_than_oh
86. uses_lol_more_often_than_and
87. uses_me_more_often_than_is
88. frequency_of_the_ngram_<S>_hold_on
89. frequency_of_the_ngram_<S>_u_</S>
90. frequency_of_the_ngram_sexy_</S>
91. uses_get_more_often_than_that
92. uses_about_more_often_than_at
93. frequency_of_the_ngram_i'll_have
94. frequency_of_the_ngram_i'll_have_to
95. frequency_of_the_ngram_<S>_i_kno
96. frequency_of_the_ngram_:p_</S>
97. frequency_of_the_ngram_an_event
98. frequency_of_the_ngram_<S>_show_</S>
99. frequency_of_the_ngram_anyone_help_me
100. frequency_of_the_ngram_anyone_help
101. frequency_of_the_ngram_to_dance_and
102. frequency_of_the_ngram_say_button_</S>
103. frequency_of_the_ngram_say_button
104. frequency_of_the_ngram_very_cool
105. frequency_of_the_ngram_there_a_game
106. frequency_of_the_ngram_the_say
107. frequency_of_the_ngram_the_say_button
108. uses_now_more_often_than_we

109.uses_get_more_often_than_so
110.frequency_of_the_word_pig
111.frequency_of_the_word_project
112.frequency_of_the_word_unfortunately
113.frequency_of_the_word_visited
114.frequency_of_the_word_tiny
115.frequency_of_the_word_tia
116.frequency_of_the_word_italiano
117.frequency_of_the_word_paul
118.frequency_of_the_word_japanese
119.frequency_of_the_word_japan
120.frequency_of_the_word_landmark
121.frequency_of_the_word_hello
122.frequency_of_the_word_hm
123.frequency_of_the_word_heyyyy
124.frequency_of_the_word_lying
125.frequency_of_the_word_libraries
126.frequency_of_the_word_makeup
127.frequency_of_the_word_meditation
128.frequency_of_the_ngram_hi_(:)
129.frequency_of_the_ngram_hi_(:)_</S>
130.frequency_of_the_ngram_i_kno
131.frequency_of_the_ngram_hi_how_are
132.frequency_of_the_ngram_i_add
133.frequency_of_the_ngram_my_time
134.frequency_of_the_ngram_lag_</S>
135.frequency_of_the_ngram_ok_thanks_</S>
136.frequency_of_the_ngram_kno_</S>
137.frequency_of_the_ngram_<S>_hi_(:)
138.frequency_of_the_ngram_<S>_please_</S>
139.frequency_of_the_ngram_amazing_</S>
140.frequency_of_the_ngram_<S>_very_cool
141.frequency_of_the_ngram_a_friend
142.frequency_of_the_word_completely
143.frequency_of_the_word_consider
144.frequency_of_the_word_fellow
145.frequency_of_the_word_figuring
146.frequency_of_the_ngram_tomorrow_</S>
147.frequency_of_the_word_description
148.frequency_of_the_word_80s
149.frequency_of_the_word_anchor
150.frequency_of_the_word_p
151.frequency_of_the_word_*
152.frequency_of_the_word_ahhhhhh
153.frequency_of_the_word_alex
154.uses_thanks_more_often_than_on
155.uses_can_more_often_than_why
156.uses_for_more_often_than_me
157.frequency_of_the_word_lucky
158.uses_want_more_often_than_be
159.uses_up_more_often_than_so
160.frequency_of_the_word_l
161.uses_got_more_often_than_know
162.uses_up_more_often_than_have
163.uses_are_more_often_than_get
164.uses_up_more_often_than_and
165.uses_hey_more_often_than_a
166.uses_up_more_often_than_sl
167.uses_up_more_often_than_much
168.uses_where_more_often_than_about
169.uses_get_more_often_than_hello
170.frequency_of_the_ngram_ya_i
171.frequency_of_the_ngram_<S>_ya_i
172.frequency_of_the_ngram_what_up_</S>
173.frequency_of_the_ngram_yo_</S>
174.frequency_of_the_word_sitting
175.uses_right_more_often_than_was
176.frequency_of_the_ngram_<S>_hey_whats
177.frequency_of_the_ngram_<S>_sup
178.frequency_of_the_ngram_the_game_</S>
179.frequency_of_the_ngram_hey_whats
180.uses_want_more_often_than_so
181.frequency_of_the_ngram_<S>_o_</S>
182.frequency_of_the_ngram_on_a
183.frequency_of_the_word_fun
184.frequency_of_the_ngram_u_got
185.frequency_of_the_ngram_<S>_i_want
186.frequency_of_the_ngram_<S>_how_much
187.frequency_of_the_word_yo
188.frequency_of_the_ngram_fag_</S>
189.frequency_of_the_ngram_let's_go
190.frequency_of_the_ngram_stuck_</S>
191.uses_now_more_often_than_in
192.frequency_of_the_ngram_get_one_</S>
193.frequency_of_the_ngram_is_awesome
194.frequency_of_the_ngram_tired_</S>
195.uses_right_more_often_than_so
196.frequency_of_ascii_character_109
197.frequency_of_the_word_about
198.frequency_of_the_ngram_hips_are
199.frequency_of_the_ngram_hopefully_i'll
200.frequency_of_the_ngram_don't_really_have
201.frequency_of_the_ngram_game_starting
202.frequency_of_the_ngram_my_time_</S>
203.frequency_of_the_ngram_ok_this_is
204.frequency_of_the_ngram_it_but_i
205.frequency_of_the_ngram_is_my_card
206.frequency_of_the_ngram_<S>_i've_never
207.frequency_of_the_ngram_<S>_/kiss
208.frequency_of_the_ngram_<S>_clap_clap
209.frequency_of_the_ngram_again_soon
210.frequency_of_the_ngram_and_there
211.frequency_of_the_ngram_an_attack
212.frequency_of_the_ngram_click_one_of
213.frequency_of_the_ngram_clap_</S>
214.frequency_of_the_ngram_speak_japanese_</S>
215.frequency_of_the_ngram_the_inside
216.frequency_of_the_ngram_<S>_later
217.uses_right_more_often_than_there
218.uses_got_more_often_than_there
219.uses_right_more_often_than_just
220.uses_of_more_often_than_you

221.uses_some_more_often_than_so
222.frequency_of_the_word_thanks
223.frequency_of_ascii_character_41
224.frequency_of_the_ngram_let_me_get
225.frequency_of_the_ngram_for_me_to
226.frequency_of_the_ngram_right_back
227.frequency_of_the_ngram_<S>_let's_go
228.frequency_of_the_ngram_me_some
229.uses_won't_more_often_than_wont
230.frequency_of_the_word_catch
231.frequency_of_the_word_beat
232.frequency_of_the_word_bottom
233.frequency_of_the_word_fight
234.frequency_of_the_ngram_<S>_yo
235.frequency_of_the_ngram_i_want
236.uses_now_more_often_than_really
237.uses_want_more_often_than_was
238.uses_doing_more_often_than_hi
239.uses_now_more_often_than_about
240.uses_want_more_often_than_hi
241.uses_got_more_often_than_hi
242.uses_so_more_often_than_got
243.frequency_of_the_ngram_gay_</S>
244.uses_right_more_often_than_from
245.uses_thanks_more_often_than_look
246.uses_hey_more_often_than_are
247.frequency_of_the_ngram_hey_girls
248.frequency_of_the_ngram_of_here
249.frequency_of_the_ngram_<S>_i_never
250.frequency_of_the_ngram_want_you
251.uses_oh_more_often_than_lol
252.uses_of_more_often_than_get
253.uses_me_more_often_than_here
254.uses_thanks_more_often_than_out
255.uses_thanks_more_often_than_im
256.uses_oh_more_often_than_much
257.uses_up_more_often_than_be
258.uses_thanks_more_often_than_right
259.uses_oh_more_often_than_can
260.uses_not_more_often_than_get
261.uses_of_more_often_than_hey
262.uses_of_more_often_than_this
263.uses_get_more_often_than_at
264.uses_oh_more_often_than_where
265.uses_thanks_more_often_than_can
266.uses_its_more_often_than_up
267.uses_get_more_often_than_why
268.frequency_of_the_word_realized
269.frequency_of_the_word_clearly
270.frequency_of_the_word_explain
271.uses_up_more_often_than_make
272.uses_thanks_more_often_than_hey
273.uses_time_more_often_than_too
274.uses_but_more_often_than_this
275.frequency_of_the_ngram_job_</S>