Affinity Filtering: A Novel Approach to Social Recommendation

Riley Kidd

A subthesis submitted in partial fulfillment of the degree of
Bachelor of Software Engineering at
The Department of Computer Science
Australian National University

October 2012

Typeset in Palatino by TEX and LATEX 2ε.

Except where otherwise indicated, this thesis is my own original work.

Riley Kidd
24 October 2012

Abstract

Social networks such as Facebook allow users to create a rich and verbose profile composed of both user specific interactions (such as comment and message passing, tags, likes, etc) and user preferences (such as favourite movies and music, group memberships, page likes, etc). These interactions and preferences define implicit groups having varying affinities to the preferences of a given user; these affinities can be learned from data and then leveraged to predict a user's own preferences, e.g., whether a user will "like" a certain news article or blog post.

The objective of this thesis is to decipher which of these aforementioned affinity measures are truly predictive of a user's like preferences.

The success of our predictions are evaluated using the machine learning algorithms of *Naive Bayes*, *Logistic Regression* and *Support Vector Machines*, and the results are compared to previous work using the state of the art social collaborative filtering technique of *Social Matchbox* as a baseline. The data is sourced from a set of over 100 Facebook users and their interactions with over 39,000 friends during a four month period.

Our analysis has shown that user interactions in themselves are not highly predictive of user likes, however user preferences are. This increasingly predictive trend continues as we limit user exposure (ensuring some k number of users u have liked a link v) over links. We conclude by analysing a combination of the most predictive user preference affinity measures, offer a summary of our work to date and propose recommendations for further research in this area.

Contents

Chapter 1

Introduction

The Internet is quickly becoming a network of people, providing a myriad of expanding social information and user driven content. This social presence on the web is continually expanding. With the emergence of services such as Facebook, Myspace, LinkedIn, Twitter and Google+, what defines a user and their online *user interactions* (such as comment and message passing, tags, likes, etc) and *user preferences* (such as favourite movies and music, group memberships, page likes, etc) is an expanding graph of rich social content.

From this premise, the ultimate question we wish to address in this thesis becomes: How can we leverage this user information to decipher which *user interaction* or *user preference* affinity features are most predictive of user "likes" (e.g., whether a user will "like" a certain news article or blog post)?

We address this question by comparing and contrasting varied potential affinity relationships in our data against appropriate baselines and ultimately offer a combination of features which proposes the most predictive solution to the question posed above.

In this chapter we will outline the objectives of this research, summarise our contributions and provide an outline for the remaining chapters.

1.1 Objectives

The primary objective of this thesis is to compare and contrast differing potential affinity features across both *user interactions* and *user preferences*. Using state of the art machine learning concepts of *Naive Bayes* (NB), *Logistic Regression* (LR) and *Support Vector Machines* (SVM) compared with our appropriate baselines of *Social Matchbox* (SMB) and *Constant Classifiers* (Constant). With the ultimate aim of discovering which affinity features are most predictive or a user's like preferences.

Based on the insight that social influence can play a crucial role in a range of behavioural phenomena [Granovetter 1978; Watts and Strogatz 1998] and that positive social annotations on search items add perceived utility to the worth of a result [Pantel and Haas 2012] we will also test while limiting user exposure to a link. This exposure hold out technique involves ensuring some k number of users u have liked some link v.

Based on the results found during our *user interaction* and *user preference* affinity analysis we will also extract individual feature weights to analyse which explicit features are most predictive of a user's like preferences (ie, for groups: which group sizes, group types, group localities, etc are most predictive).

Finally, we will assess and compare the effect of combining the individually predictive affinity features found during our analysis.

1.2 Contributions

Most previous work has simply compressed all social information into a single "user similarity" metric, which does not leverage the fact that very specific interactions or preferences can be very predictive. Our approach involves recommendation via explicit affinity groups. This is a novel approach to social recommendation as we can leverage very specific fine-grained social features.

Specific contributions made during this thesis show:

- Both *interactions* (tags, likes, etc) and *messages* (incoming and outgoing messages posted between users) are not more predictive than previously applied SMB techniques.
- The *user preference* affinities of *favourites* (favourite movies, music, etc), *group* memberships (Australian National University, Students in Canberra, etc) and *page* likes (Google Chrome, The Simpsons, etc) are the most predictive affinity features.
- Comparing both affinity types of *user interactions* and *user preferences* against an exposure limit results in a substantial improvement over previous techniques as this exposure increases.
- Individual *groups* which were most predictive were highly localised with a medium user frequency, while the most predictive *pages* were much more general and of a higher relative user frequency.
- Combination of the most predictive individual affinity features outlined above present the most predictive results found during our analysis.

Overall, we discover which user affinities are most predictive of a user's like preferences, analyse the effect of user exposure across links, evaluate which explicit affinity features contribute the most weight during prediction and assess the predictive qualities of combining the individually most predictive affinities features.

1.3 Outline

The remaining chapters in this thesis are organised as follows:

- **Chapter 2 (Background)**: We first outline appropriate background information for the reader. Including information pertaining to the source of our data set, mathematical notation used throughout this thesis, previous work in this area and our research approach and methodology.
- **Chapter 3 (User Interactions)**: In this chapter we discuss different affinity features for *user interactions* and the results of applying these features to NB, SVM and LR in comparison with our baselines.
- **Chapter 4 (User Preferences)**: A similar affinity feature analysis as above is applied, however the features we utilise are for *user preferences*.
- **Chapter 5 (Feature Combination)**: In this chapter we discuss the effect of combining individually predictive affinity features based on results gained in the previous sections and propose an ideal feature hybrid.
- **Chapter 6 (Conclusion)**: Finally, we draw the work done throughout this thesis to a conclusion and offer avenues for future work in this area.

The sum of these chapters represents a novel approach to exploiting and analysing *user interactions* and *user preferences* affinity relationships to ascertain which features are most predictive of user likes and present an approach of combining these useful individual feature components into an effective classification paradigm.

Chapter 2

Background

In this chapter, we define the social network Facebook central to this study, the source of our data set, notation used throughout this thesis, our choice of classification algorithms and finally our testing approach and methodology.

2.1 Facebook

Facebook is the largest and most active social media service in the world (as of September 2012 it had more than 1 billion active users [Facebook 2012]). Facebook users can create a profile containing personal *preferences* and information including their favourite music, favourite movies, inspirational people, interests, age, birthday, etc and have friendships and *interactions* between other users.

The four main interactions between users are posts (posting something on a friends' wall), tags (being mentioned in a friends post or comment), comments (written data on a post) and likes (clicking a like button if a user likes a post or comment). The mediums for these interactions are across links (some URL), posts (some Facebook post), photos (some uploaded Facebook photo) and videos (some uploaded Facebook video).

Given the enormous scope of *interaction* and *preference* information available for each user, NICTA have developed an application (app) capable of tracking and recording all pertinent user information. This app will be discussed in the following section.

2.2 Data Set

One issue present in the Facebook paradigm is deciphering whether a user does not like an item, a users Facebook feed (current visible personalised Facebook information) is comprised of recent activity between their friends, groups, pages, etc giving an enormous scope for potential feed items. Given the high rate of posting, these top feed items are only displayed for a short period of time. Coupled with the fact that Facebook allows users to explicitly like an item, but not dislike it - distinguishing between what a user does and does not like becomes difficult.

Given this fact, a Facebook app named *LinkR*[1] was developed [www 2012]. This app collected information about users, their interactions and preferences as well as a subset of available information about their friends. Additionally, the app proposed links to users and asked them to explicitly rate each link as either a like or dislike. The app tracked and stored this information for over 100 app users and their 39,000 friends over a 4-month time period. Which is a sufficiently large data-set for performing our analysis.

The table below summarises the interactions data collected from both app users and their friends used during subsequent analysis.

App Users	**Posts**	**Tags**	**Comments**	**Likes**
Wall	36,539	7,711	18,266	15,999
Link	5,304	-	5,757	6,566
Photo	4,933	28,341	8,677	8,612
Video	245	2,525	1,687	843
App Users and Friends	**Posts**	**Tags**	**Comments**	**Likes**
Wall	4,301,306	1,215,382	3,122,019	1,887,497
Link	678,612	-	693,930	995,214
Photo	1,268,816	9,620,708	3,431,321	2,469,859
Video	59,244	904,604	486,677	332,619

Table 2.1: Data records for interactions between users. Rows are the type of interaction, columns are the medium of interaction.

2.3 Notation

The mathematical notation utilised during this thesis is outlined below.

- A set of users U of size N each with an associated I-element user feature vector X where $X \in \mathbb{R}^I$ (alternatively if a second user is needed $Z \in \mathbb{R}^I$) where the length and components of I are uniquely defined for each affinity feature, under their appropriate sections.
- A set of items V.
- A friend function $Friend_{u,z}$ which is *True* when users u and z are friends.
- A liked function $Likes_{u,v}$ which is *True* when user u likes item v.
- A relationship between user u over some feature index i where this $Relationship_{u,i}$ is uniquely defined under each affinity features section.

[1]The main developer of the LinkR Facebook App is Khoi-Nguyen Tran, a PhD student at the Australian National University.

- An alters set for each user u item v pair over some feature index i, based on some relationship between other users z and where each of u and z have liked item v. Where $alters_{u,v,i} = \{z | Relationship_{u,i} \wedge Likes_{u,v} \wedge Likes_{z,v}\}$.

  This alters set can be visualised in the figure below:

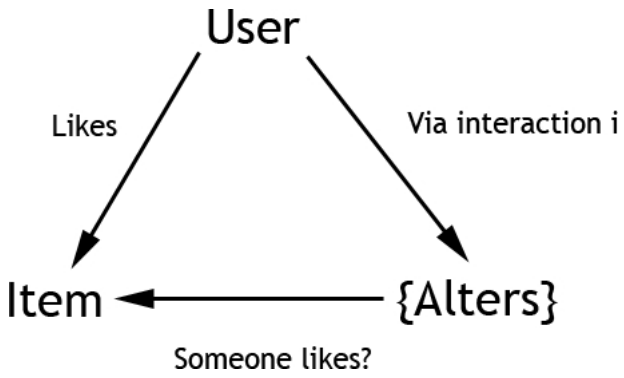

Figure 2.1: Alters paradigm. A user u likes some item v, a $Relationship_{u,z}$ is defined via some affinity i (in this example we use *interactions*) uniquely defined for each affinity feature, to create our set of $alters_{u,v,i}$.

- Exposure of a user u to an item v is the number of friends of u who have liked v. An exposure limit where $Exposure_{u,v} = \sum_{z}^{N}(Friend_{u,z} \wedge Liked_{z,v})$ where this exposure can be limited by some k with the condition $Exposure_{u,v} >= k$. Thus ensuring some k number of friends have liked a link v.

  This exposure can be visualised in the figure below:

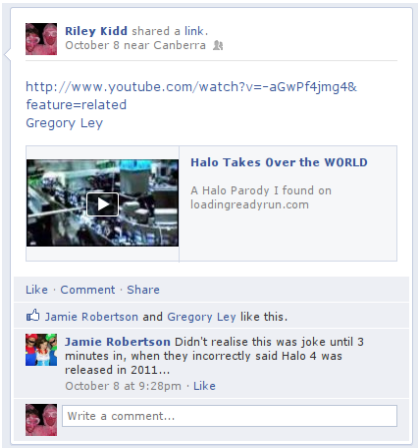

Figure 2.2: Here we see an example of a link v posted to a friends wall, which has subsequently been liked by two friends z. This demonstrates an exposure of 2 for item v.

- A data-set D comprised of $D = \{(u, v, x) \rightarrow y\}$ where $u \in U$, $v \in V$ and the binary response $y \in \{0, 1\}$ where 0 represents a dislike and 1 represents a like.

2.4 Affinity Features

Given the vast amount of potential information available about users on Facebook, we need to break this information down into individual features.The two distinct categories we will group these features under are *user interactions* and *user preferences*. *User interactions* involve explicit affinity relationships between users, while *user preferences* involve latent similarities expressed between users.

The individual components of these affinity based categories are displayed below:
User interactions:

- **Interactions** : Posts, tags, likes, comments between users.
- **Outgoing Messages** : Messages sent to other users.
- **Incoming Messages** : Messages received from other users.

User preferences:

- **Demographics** : Age, gender and location of a user.
- **Favourites** : A users favourite preferences for activities, books, athletes, teams, movies, music, sports, television, people and interests.
- **Groups** : All groups a user has joined.
- **Pages** : All pages a user has liked.

Each of these affinity features will be used individually to train and test our classifiers defined below and are discussed in explicit detail under their separate sections of this thesis. During our analysis we will compare the predictiveness of each of these affinity features individually and in combination. Each feature will also be tested against some exposure to analyse the differences in results.

2.5 Previous Work

Two general approaches to prediction in a social context are *content-based filtering* (CBF) [Lang 1995] which exploits item features based on items a user has previously liked and *collaborative filtering* (CF) [Resnick and Varian 1997] which exploits the current users preferences as well as those of other users.

Previous work defined the term *social* CF (SCF) [Noel 2011] which augments traditional CF methods with additional social network information, the results of this previous work and analysis using live user trials came to the conclusion that the approach of SMB provided the best results for this data set and as such will be used as a baseline.

These methods of CBF, CF and SCF result in a user gaining some similarity measure between other users, while the affinity features we explore during this thesis are based on explicit individual *user interaction* and *user preference* features and result in different models and predictions based on our feature selection.

2.6 Training and Testing

All evaluation is applied using 10 fold cross validation wherein the data is partitioned into 10 complementary subsets, eight folds (80% of data) for training and two folds (20% of data) for testing.

The training and testing process is repeated 10 times for each set of fold data. These results are then averaged to produce our estimates and standard error. The benefit of this method over repeated sub-sampling is all data points are used for both training and validation.

2.7 Recommendation Algorithms and Baselines

Each classification algorithm is passed the training data for each fold as outlined above. The classifier builds a model representation of the data and applies this model to the test set to classify each test item into either a dislike 0 or a like 1.

All affinity feature analysis carried out in this thesis will be performed on the following classification algorithms:

2.7.1 Constant

The constant predictor returns a constant result irrespective of the affinity feature selected. Namely, this predictor returns dislike regardless of the affinity feature represented by X. The most common result in our data set is dislike and hence this dislike predictor is displayed in all comparison analysis, tables and graphs.

2.7.2 Social Match Box

SMB is an extension of existing SCF techniques [Yang et al. 2011; Cui et al. 2011] which constrain the latent space to enforce users who have similar preferences to maintain similar latent representations when they interact heavily.

SMB uses the social regularization method which incorporates user features to learn similarities between users in the latent space which allows us to incorporate the social information of the Facebook data [Noel 2011].

This objective component constrains users with a high similarity rating to have the same values in the latent feature space, which models the assumption that users who are similar socially should also have similar preferences for items.

2.7.3 Naive Bayes

NB is a basic probabilistic classifier which involves applying Bayes' theorem using strong conditional independence assumptions between each feature in X. During training each element i in X is devised to contribute some evidence that this x_i belongs to either a like or dislike classification, during testing the class with the highest probability when applied to the model is the classification predicted.

With feature vector $f \in \mathbb{R}^F$ derived from $(x, y) \in D$, denoted as $f_{x,y}$. NB learns a conditional model of the form $p(C|F_1, \ldots, F_n)$ over a dependent class variable C conditioned on the feature variables $F_1, \ldots, F_n$. Applying both Bayes' rule and conditional independence assumptions the model can be rewritten as $p(C|F_1, \ldots, F_n) = p(C)\prod_{i=1}^{n} p(F_i|C)$.

Classification of our test vector is achieved by choosing the most probable class of either like (1) or dislike (0).

$$\text{classify}(f_1, \ldots, f_n) = \underset{c \in \{1,0\}}{\text{argmax}}\, p(C = c)\prod_{i=1}^{n} p(F_i = f_i|C = c).$$

2.7.4 Logistic Regression

LR directly estimates parameters based on the training data assuming a parametric form of the distribution. LR predicts the odds of a feature vector X being either a like or a dislike by converting a dependent variable and one or more continuous independent variable(s) into probability odds.

The probability p_i is modelled using a linear predictor function $l(i)$, the linear predictor function of a particular point d is written as $l(i) = \beta_0 + \beta_1 x_{1,i} + \cdots + \beta_M x_{m,i}$, where each data point d is associated with an explanatory feature vector X and $\beta_0, \ldots, \beta_M$ are regression co-efficients indicating the relative effect of a particular explanatory variable $x_{m,i}$ on the prediction.

The probability of a particular outcome is linked to the linear prediction function, $\text{logit}(\mathbb{E}[Y_i|x_{1,i}, \ldots, x_{m,i}]) = \text{logit}(p_i) = \ln\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_{1,i} + \cdots + \beta_M x_{m,i}$ Where the class of either dislike (0) or like (1) with the higher probability is the prediction made.

The LR implementation used during this thesis is *LingPipe* [Alias-i. 2008. LingPipe 4.1.0. http://alias-i.com/lingpipe (accessed October 1 2011].

2.7.5 Support Vector Machine

SVM is a supervised learning machine based on a set of basis functions which help construct a separating hyperplane between data points. Training involves building the relevant hyperplanes which can then be used for testing. Each data point is classified as a like or dislike depending on which side of the hyperplane it falls.

With feature vector $f \in \mathbb{R}^F$ derived from $(x, y) \in D$, denoted as $f_{x,y}$. A linear SVM learns a weight vector $w \in \mathbb{R}^F$ such that $w^T f_{x,y} > 0$ indicates a like classification of $f_{x,y}$ and $w^T f_{x,y} \leq 0$ indicates a dislike classification.

The SVM implementation used during this thesis is *SVMLibLinear* [Chang and Lin 2011].

2.8 Evaluation Metrics

When evaluating the success of each affinity feature at correctly classifying an item, the following metrics are used:

- A *true positive* (TP) prediction refers to when the prediction correctly identifies the class as true.
- A *false positive* (FP) occurs when the prediction is true, but the true class was false.
- A *false negative* (FN) occurs when the prediction is false but the actual class is true.

These definitions can be visualised using the table below:

		y	
		T	**F**
$\hat{y}$	**T**	TP	FP
	F	FN	TN

Table 2.2: Actual and prediction comparison table.

Where y represents the true class value $y \in \{0, 1\}$: *actual* and $\hat{y}$ represents the class prediction $\hat{y} \in \{0, 1\}$: *prediction*.

Accuracy relates to the closeness to the true value. In the context of our results, the accuracy refers to the number of correct classifications divided by the size of the data set.

$$\text{accuracy} = \frac{\text{number of correct classifications}}{\text{size of the test data set}}$$

Precision relates to the number of retrieved predictions which are relevant. In the context of our results, the precision refers to the number of TP predictions divided by the sum of the TP and FP predictions.

$$\text{precision} = \frac{\text{number of TP}}{\text{number of TP + number of FP}}$$

Recall refers to the number of relevant predictions that are retrieved. In the context of our results, recall refers to the number of TP predictions divided by the sum of the TP and FN predictions.

$$\text{recall} = \frac{\text{number of TP}}{\text{number of TP + number of FN}}$$

The f-score combines and balances both precision and recall and is interpreted as the weighted average of both precision and recall.

$$\text{f-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

The main metric we use for analysis, tabulation and graphing in our results is accuracy.

Chapter 3

User Interactions

This chapter is dedicated to analysing the different *user interaction* features present in Facebook.The *user interactions* we examine can be broken down into two distinct groups:

- **Interactions** : Posts, tags, likes and comments between users.
- **Messages** : Both outgoing and incoming messages sent between users.

These interactions give implicit networks of friendships, previous methods [www 2012] have claimed if people interact frequently they will like the same things, in this section we break this idea down into the smaller implicit overlaps of these inherent interactions.

3.1 Interactions

Interactions between users in Facebook can be summarised under the following categories:

- **Direction**: Interaction directionality has been shown to be highly reflective of user preferences [Saez-Trumper et al. 2011] and can be grouped as either *incoming* (for example where a message is posted to some user) or *outgoing* (where some user posts a message to another user).
- **Modality**: The medium some user employs to interact with another user via either *links*, *posts*, *photos* or *videos*.
- **Type**: The category some user employs to interact with another user via either *comments*, *tags* or *likes*.

Some of these online *interactions* suggest a real world relationship (such as being tagged in photos or videos) so we expect these to be predictive of user like preferences.

For *user interactions* each feature vector X where $X \in \mathbb{R}^I$ is composed of the cross product between the above components where:

$$I = \{Incoming, Outgoing\} \times \{Posts, Photos, Videos, Links\} \times \{Comments, Tags, Likes\}$$

The alters set for each i is conditioned by the relationship:

$$Relationship_{u,i} = \{z | Interacted_{u,z,i}\}$$

In this case the $Interacted_{u,z,i}$ function returns *True* if a user u has interacted with a user v via the current interaction i, for example whether user v has been tagged in a photo uploaded by user u.

Applying this *interactions* affinity feature to our classification algorithms we obtain:

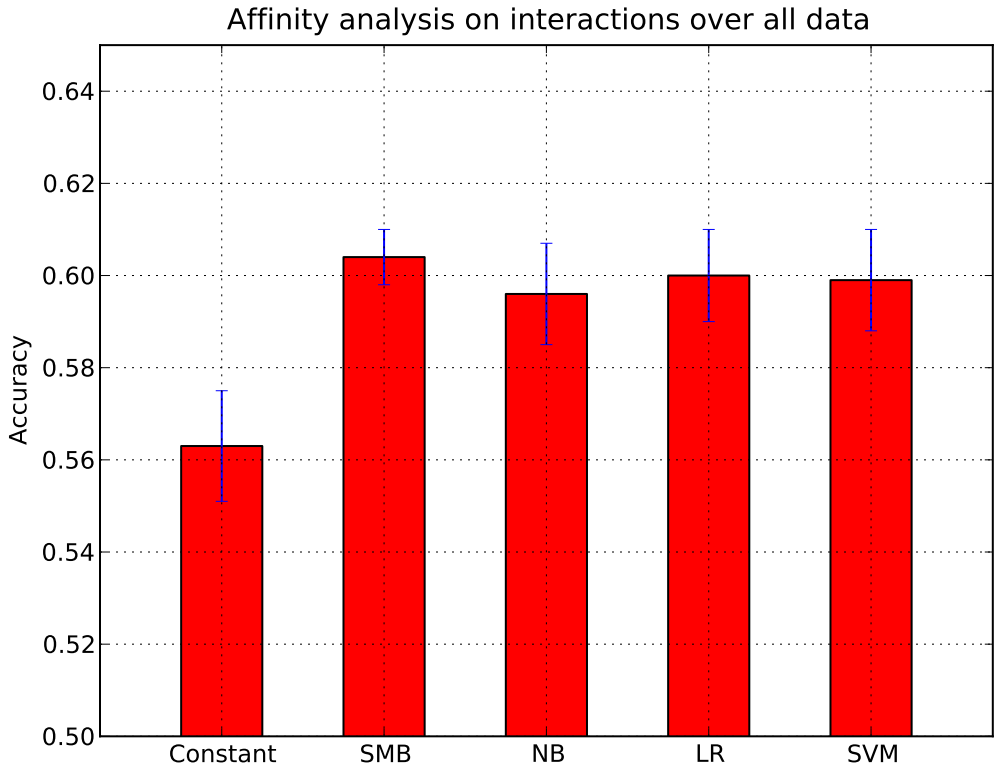

Figure 3.1: Accuracy results using *user interactions* against all data. *User interactions* are less predictive then our baselines.

User interactions in themselves are not more predictive then our SMB baseline. One reason for this result could be we can not track information passing outside of Facebook, users who frequently interact could be real world friends and hence share information via email or word of mouth and not over the Facebook medium.

Comparing *user interactions* against an exposure across k we obtain:

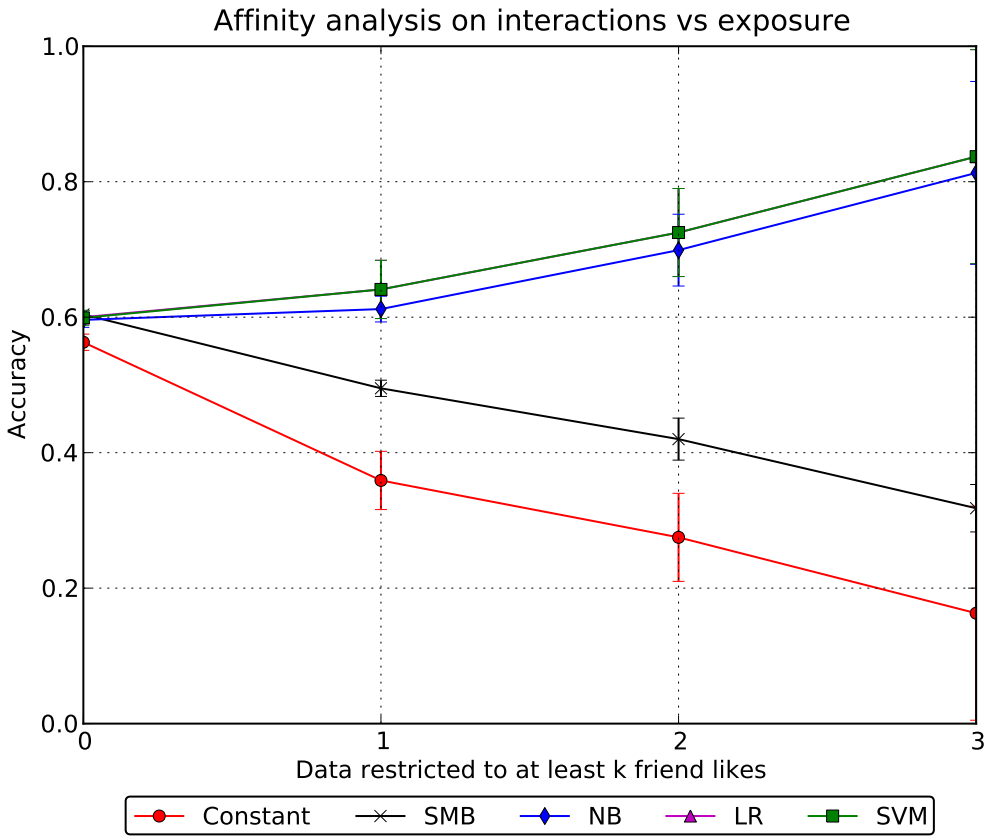

Figure 3.2: Accuracy results against exposure using *user interaction* features. *User interactions* provide a drastic improvement over our baselines as k increases, suggesting SMB is not always the best classifier. This demonstrates the intuitive assumption that *user interactions* can not improve prediction when interactions do not exist between users, while our k ensures they do. Note in this case LR and SVM both learnt the same result.

Our comparison has shown that as our data is restricted across an exposure, the performance of our classifiers improves. This implies that for *user interactions* simply having one user liking an item is enough to improve upon our baselines. This is intuitively correct as our classifiers can not learn when interacts do not exist between users.

3.2 Conversation

The next *user interactions* we compare are messages passed between users. These messages can be broken down based on their directionality of either *outgoing* (messages sent to other users) or *incoming* (messages received from other users).

We extracted the most common words exchanged between our users, the top 40 are displayed in the table below:

Rank	Word	Frequency
1	:)	292,733
2	like	198,289
3	good	164,387
4	thanks	159,238
5	one	156,696
6	love	139,939
7	:p	121,904
8	time	106,995
9	think	106,459
10	see	103,690
11	nice	99,672
12	now	94,947
13	well	92,735
14	happy	84,381
15	:d	83,698
16	much	78,719
17	oh	77,321
18	yeah	76,564
19	back	76,032
20	great	70,514
21	going	70,447
22	still	68,245
23	new	67,430
24	day	65,579
25	come	63,837
26	;)	62,936
27	year	61,771
28	look	60,608
29	yes	59,774
30	want	59,514
31	tag	58,633
32	hahaha	57,448
33	also	56,414
34	need	55,921
35	make	54,949
36	sure	54,395
37	thank	54,112
38	people	53,211
39	miss	53,182
40	guys	52,855

Table 3.1: Top conversation content data for all users. We see very common words and online expressions have a high frequency in our data set. Highly emotive and sentimental words are very common, implying these interactions occur between close friends.

These words show a high degree of emotion and sentiment is common between our users in words such as ”thanks” and ”love”, this implies that these interactions are occurring between people who are real friends and not online associates.

For *messages* each feature vector X where $X \in \mathbb{R}^I$ is composed of:

For the *outgoing* case:

$$I = \{Outgoing\ Words\} \times \{Optimal\ Outgoing\ Size\}$$

For the *incoming* case:

$$I = \{Incoming\ Words\} \times \{Optimal\ Incoming\ Size\}$$

Where the optimal sizes for *outgoing* and *incoming* words are defined for each classifier in their corresponding sections below.

The alters set for each i is conditioned by the relationship:

For the *outgoing* case:

$$Relationship_{u,i} = \{z | Messaged\ Outgoing_{u,z,i}\}$$

For the *incoming* case:

$$Relationship_{u,i} = \{z | Messaged\ Incoming_{u,z,i}\}$$

In these cases the *Messaged Outgoing* and *Messaged Incoming* functions return whether the user u has messaged the user z (outgoing) or whether the user z has messaged the user u (incoming) the word currently at index i in the most popular words list (a limited version of this list is shown in *Table 3.1*).

3.2.1 Outgoing

The first issue is to determine the most predictive number of *outgoing* words for use by our classifiers. Given the expansive size of potential messages and memory constraints in the testing environment we decided to test within a range of $(100 - 1000)$ with an incremental step size of 100 for each test.

The most predictive *outgoing* words sizes for each of our classifiers are:

- **Naive Bayes**: 500.
- **Logistic Regression**: 200.
- **Support Vector Machine**: 900.

The results of testing based on differing sizes of *outgoing* words can be seen below.

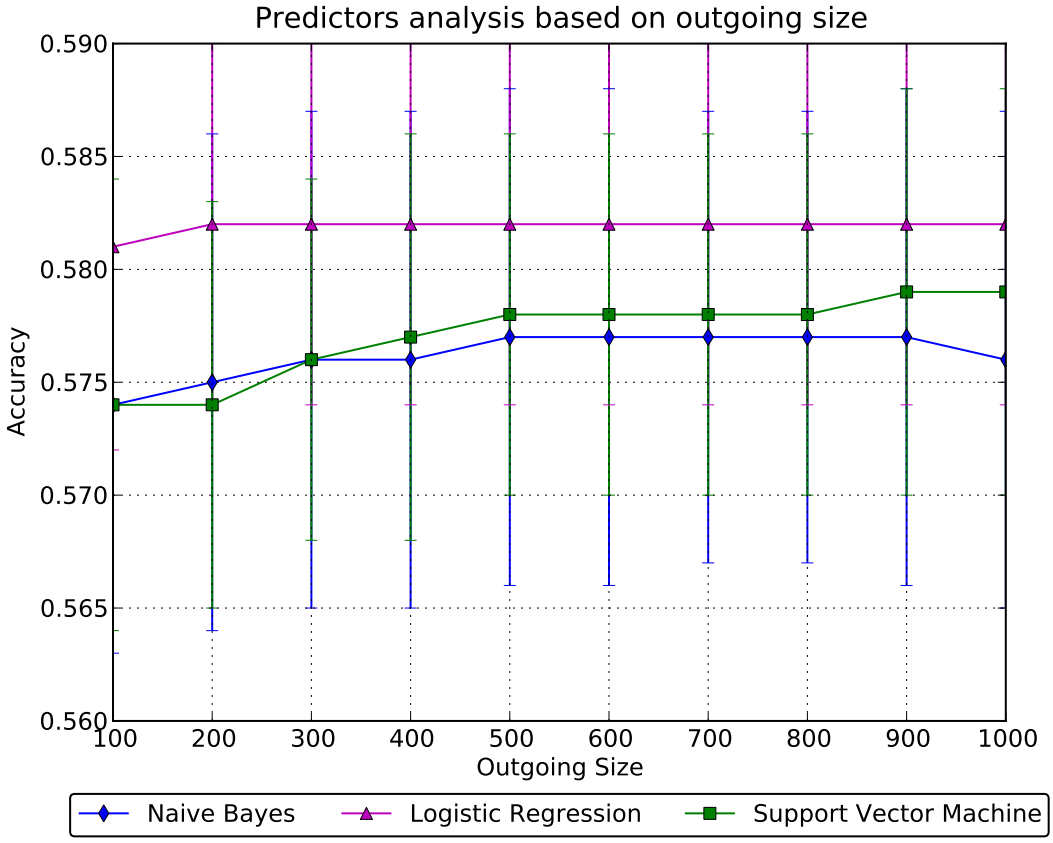

Figure 3.3: Accuracy results for different *outgoing* words sizes. Best performance can be found using LR with a relatively small word size of only 200 more words do not appear to increase the predictiveness for *outgoing* words.

Using these most predictive word sizes for each of our classifiers and building our feature vector as defined above we obtain the following results:

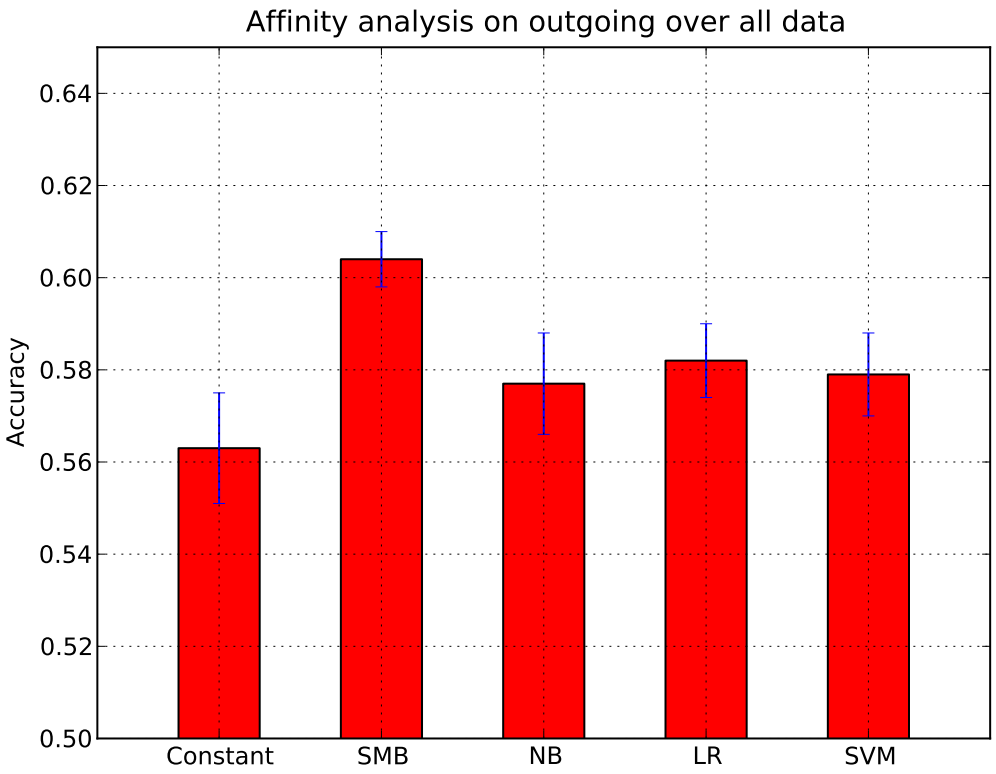

Figure 3.4: Accuracy results using *outgoing* words. *Outgoing* words are clearly less predictive then both our baseline and *user interactions*. As shown in *Figure 3.1*.

These results do not show an improvement over our *interactions* or our baselines and are only a marginal improvement over the *constant* baseline. Demonstrating that *outgoing words* are not predictive of a user's like preferences.

A possible reason for this poor predictive performance could be we don't see infrequent words often enough to get low-variance parameter estimates.

Comparing *outgoing* words against exposure we obtain:

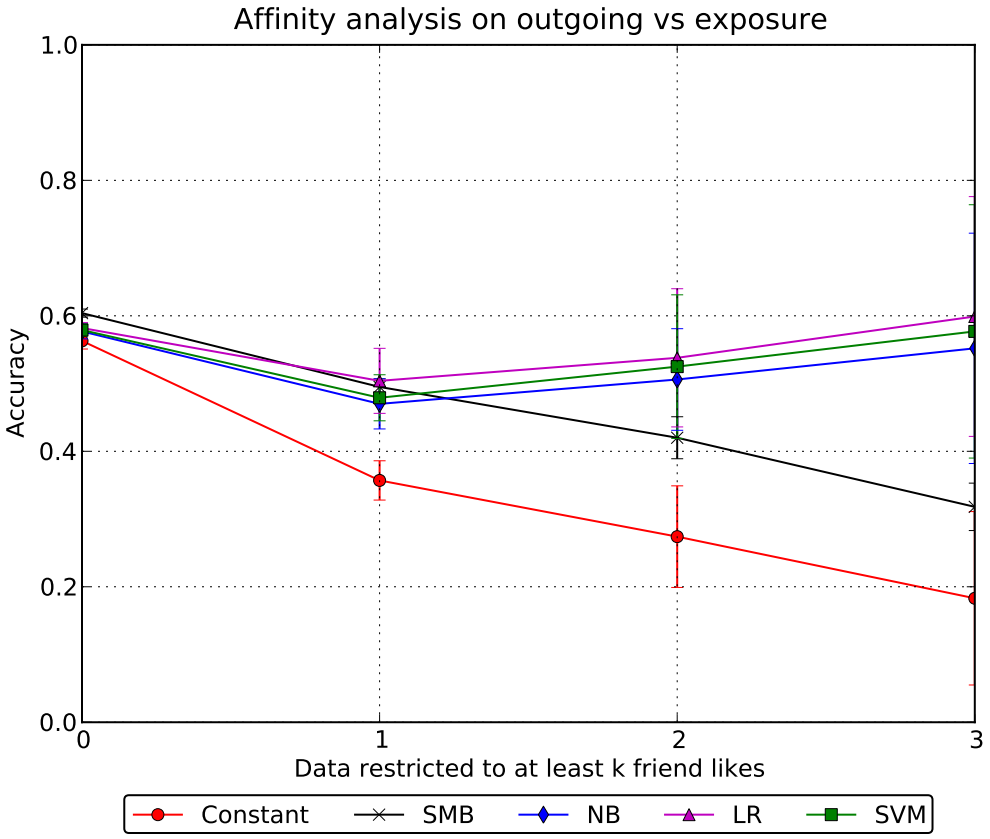

Figure 3.5: Accuracy results against exposure using the *outgoing* words feature. *Outgoing* words predictiveness improve as k increases, but this is negligible when compared with the exposure results from *user interactions* as seen in *Figure 3.2*.

The exposure offers an increase in the predictiveness of the *outgoing* words feature. However even when $k = 3$, this feature still does not outperform SMB when $k = 0$. Indicating that *outgoing* words are not a useful feature for prediction.

3.2.2 Incoming

Similarly for *incoming* words we need to discover the most predictive word size for use by our classifiers, using the same methodology as described above for *outgoing* words we obtain the following graph:

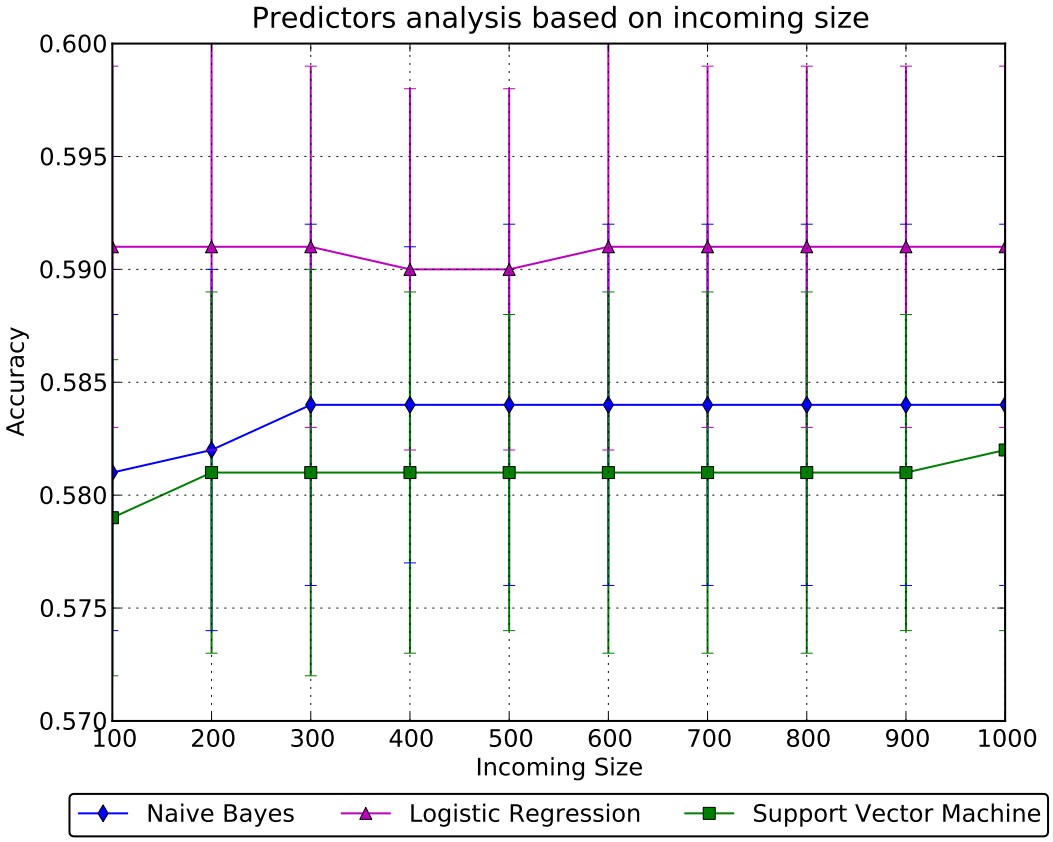

Figure 3.6: Accuracy results for different *incoming* words sizes. *Incoming* words are more predictive then *outgoing* words, but follow the similar trend that relatively small sizes offer close to optimal predictions for this feature as seen in *Figure 3.3*.

The most predictive *incoming* words sizes for each of our classifiers are:

- **Naive Bayes**: 300.
- **Logistic Regression**: 100.
- **Support Vector Machine**: 1000.

Using these most predictive word sizes for each of our classifiers we obtain the following results:

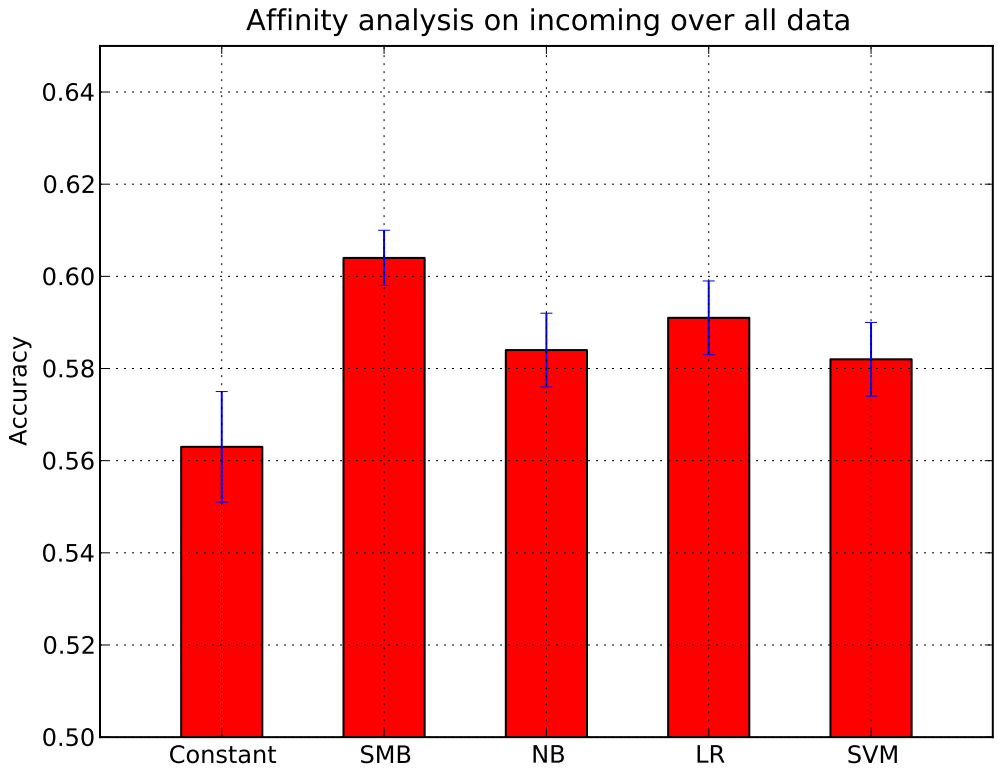

Figure 3.7: Accuracy results using the *incoming* words features. *Incoming* words are a stronger predictor then *outgoing* words implying *incoming* words are more predictive, however they are still a weaker predictor then *user interactions* and our baselines.

Incoming words are more predictive then *outgoing* words, however *incoming* words themselves are not more predictive then our baselines. Comparing *incoming* words against exposure we obtain:

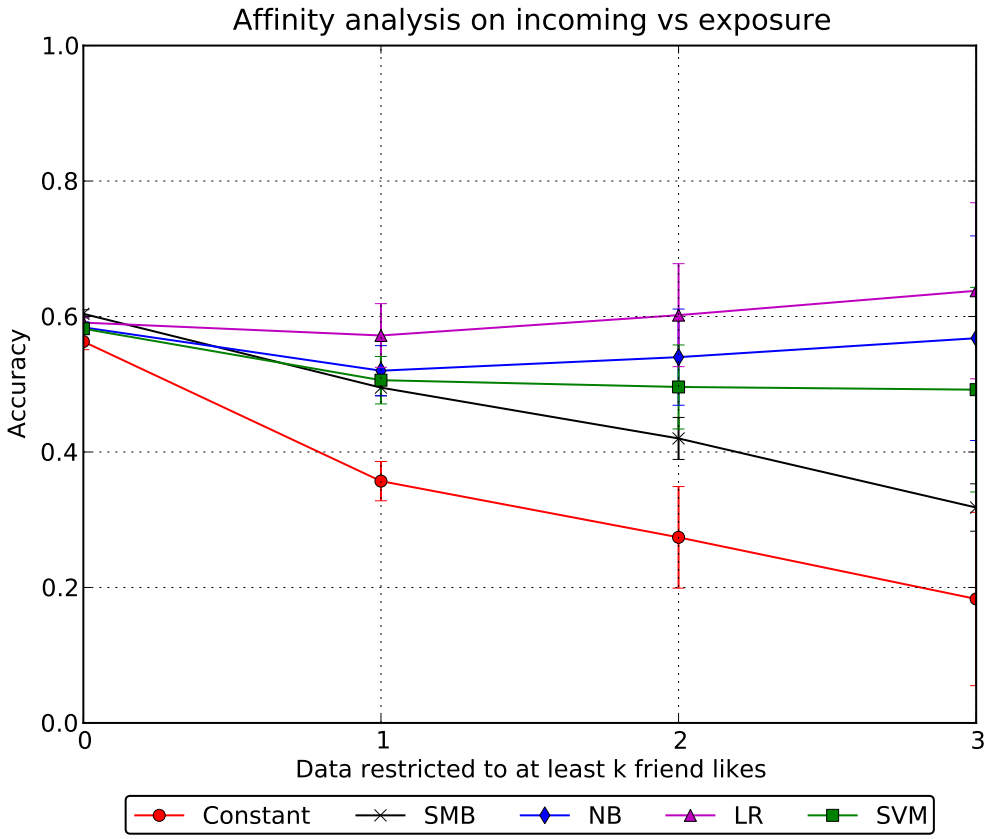

Figure 3.8: Accuracy results against exposure using *incoming* words features. *Incoming* words predictions improve as k increases and outperform SMB, however they are still less predictive then *user interactions*.

Incoming words improve upon our baselines as k increases, however this predictive increase is negligible in comparison with SMB when $k = 0$ and *user interactions* across $k = 2$ and $k = 3$ and hence *incoming* words are not predictive of user likes.

Incoming words show an improvement over *outgoing* words, however neither outperforms *user interactions*.

3.3 Conclusion

Throughout this chapter we have explored the different interaction types available between users on Facebook.

We have found that words, irrespective of their directionality do not assist with improving predictions. [Anderson et al. 2012] concluded that it is less important what users say, then who they interact with, which we also found in our results. Additionally, it has been shown for *user interactions outgoing* interactions are more important then *incoming* [www 2012], while our results have found that for words, *incoming* are

(slightly) more important then *outgoing*. Overall, *messages* are a much weaker at prediction compared with both *interactions* and our baselines.

Our results have shown, that for *interactions* to improve upon our baseline predictions it is enough for some user to have previously liked the item, this allows an improvement in predictiveness as k increases because there are at least k users for each item which our predictors can learn from. An important result we have found is SMB clearly fails when there are *interactions* to learn from and this is where these *user interactions* can help with prediction.

In the next chapter we proceed to other ways to define affinity groups based on *user preferences* to see if they are more or less predictive than the affinity groups based on *user interactions* explored above.

Chapter 4

User Preferences

In this chapter we will compare the predictiveness across the following different types of *user preferences* available in Facebook:

- **Demographics** : Age, gender and location of a user.
- **Favourites** : A users favourite preferences for activities, books, athletes, teams, movies, music, sports, television, people and interests.
- **Groups** : All groups a user has joined.
- **Pages** : All pages a user has liked.

4.1 Demographics

The first *user preference* we will examine is *demographics*, the specific *demographics* data we are interested in includes:

- **Users age**.
- **Users birthday**.
- **Users location**.

Below we will give a basic analysis of this *demographics* data in our data set.
Gender breakdown:

Male	Female	Undisclosed
85	33	1

Table 4.1: Gender breakdown for app users. We see a strong male bias due to the majority of app users being computer science students (or graduates).

Despite this clear male bias [Ugander and Marlow 2011] found that in a social setting, there are no strong gender homophily tendencies. Hence the male skew should not negatively affect our results. Additionally [Backstrom et al. 2011] have shown that different genders have differing tendencies to disperse interactions across genders, implying this gender bias will not negatively effect prediction and hence gender information will be used as a *demographics* feature.

Birthday breakdown:

Year	Frequency
Undisclosed	1
1901-1905	1
1906-1910	0
1911-1915	1
1916-1920	0
1921-1925	0
1926-1930	0
1931-1935	0
1936-1940	1
1941-1945	0
1946-1950	0
1951-1955	0
1956-1960	2
1961-1965	1
1966-1970	4
1971-1975	10
1976-1980	12
1981-1985	25
1986-1990	34
1991-1995	25
1996-2000	2

Table 4.2: Birthday breakdown for app users. There is clearly a densely populated age range of around $18 - 30$, similarly as in *Table 4.1* due to the majority of app users being computer science students (or graduates).

Birthdays are grouped in a distinct range, most users in this data set are grouped in the range of around $18 - 30$. [Ugander and Marlow 2011] have found that there is a strong effect of age on friendship preferences, which implies that many of our app users should share similar preferences given their age similarities and hence birthday information will be a useful component of this feature.

Location breakdown:

Location	**Frequency**
Undisclosed	33
Ahmedabad, India	1
Bangi, Malaysia	1
Bathurst, New South Wales	1
Bellevue, Washington	1
Braddon, Australian Capital Territory, Australia	1
Brisbane, Queensland, Australia	2
Canberra, Australian Capital Territory	56
Culver City, California	1
Frederick, Maryland	3
Geelong, Victoria	1

Table 4.3: Location breakdown for app users. Most users are either undisclosed or based in Canberra where this app was developed.

Given the fact that most app users are either situated in Canberra (location of the app development and deployment) or are undisclosed, location information will not be used by this feature vector.

For *demographics* each feature vector X where $X \in \mathbb{R}^I$ is composed of a combination of the above components where:

- $x_1 =$ Whether the user u is male, $gender_u = male$.
- $x_2 =$ Whether the user u is female, $gender_u = female$.
- $x_3 =$ Whether the user u and any user in the alters share the same gender defined by the relationship:

$$Relationship_{u,3} = \{z | gender_u = gender_z\}$$

- $x_4 =$ Whether the user u and any user in the alters are of a different gender defined by the relationship:

$$Relationship_{u,4} = \{z | gender_u \neq gender_z\}$$

- $x_5 =$ Whether the user u and any user in the alters set share the same birth range defined by the relationship:

$$Relationship_{u,5} = \{z | birthrange_u = birthrange_z\}$$

In this case the functions $gender_u$ returns the gender of u where $gender \in \{male, female\}$ and $birthrange_u$ returns the birthrange of u where $birthrange \in \mathbb{R}$.

Applying this *demographics* feature to our classifiers we obtain:

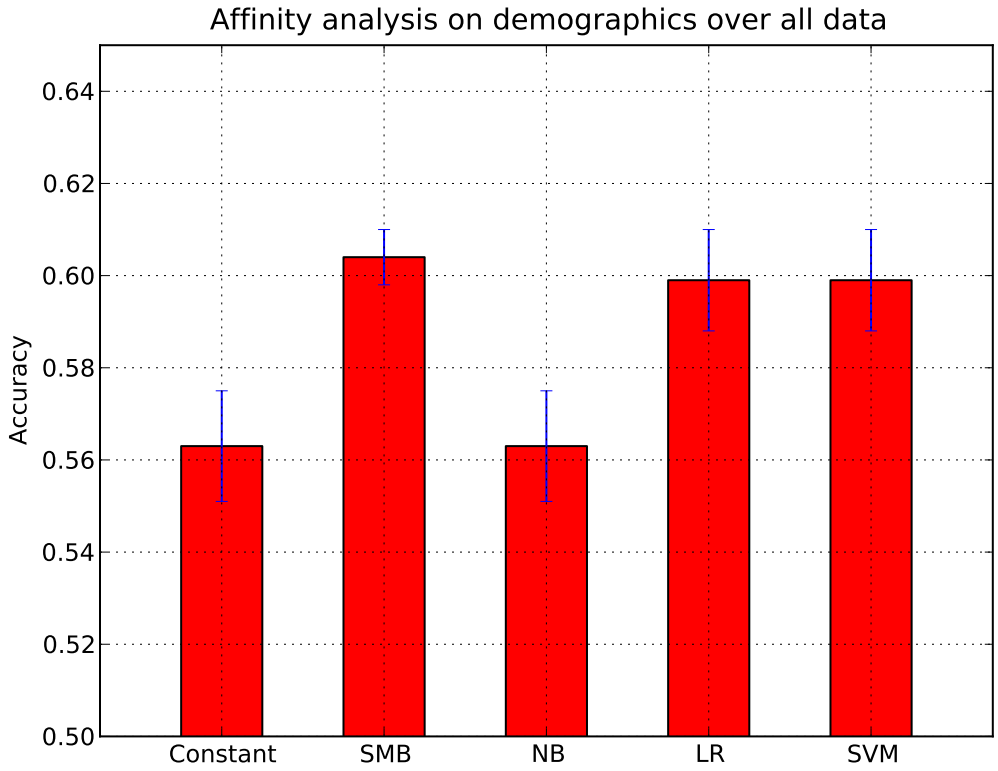

Figure 4.1: Accuracy results using *demographics* features. *Demographics* are our best performing feature so far for $k = 0$ implying *demographics* are predictive of a users preferences, however we still do not outperform our SMB baseline.

The *demographics* features provide our best results so far for the case when $k = 0$ implying *demographics* are predictive, however they are still less predictive then our SMB baseline.

Comparing *demographics* against exposure we find:

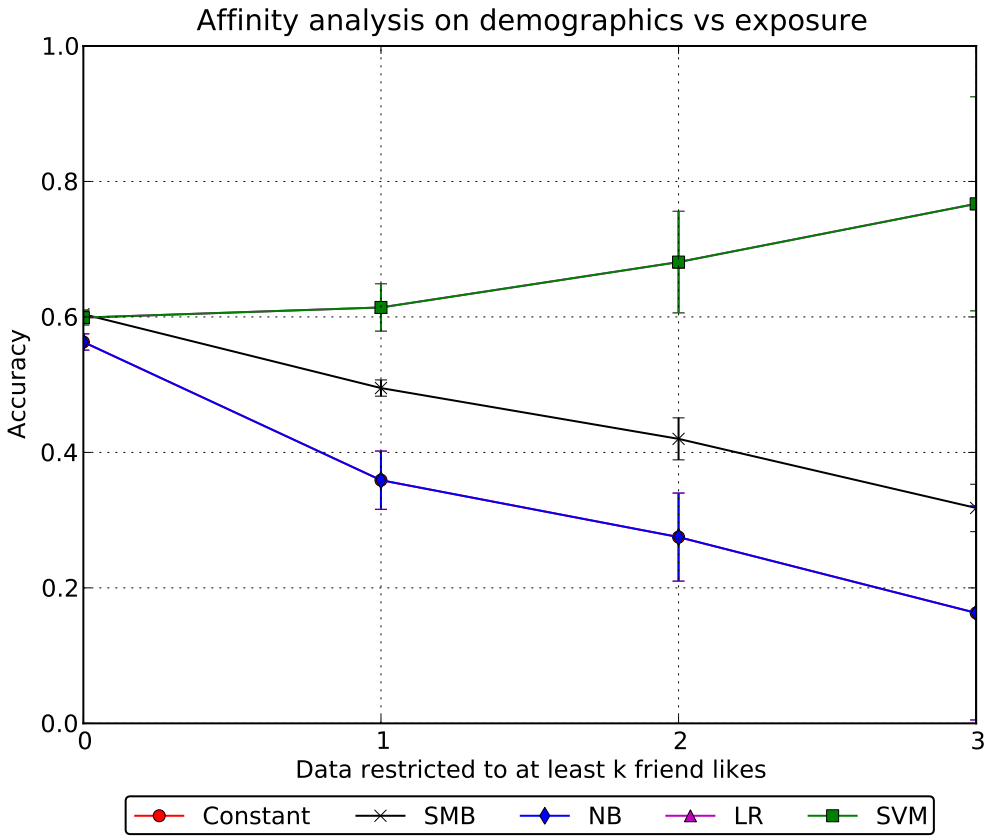

Figure 4.2: Accuracy results for exposure using the *demographics* features. As k increases the predictiveness of *demographics* substantially increases with it, clearly gender and age are both predictive as exposure increases. Note in this case Constant = NB and LR = SVM.

The exposure for *demographics* shows a sizable improvement over our baselines as k increases. This demonstrates that as the number of friends who like an item increase, the predictiveness of our *demographics* feature increases, due to affinity groups requiring more consensus (higher exposure) to see improved prediction performance.

Hence, both gender and age are predictive *user features*.

4.2 Favourites

Facebook facilitates a wide variety of user selected *favourites*. These *favourites* allow a user to associate themselves with other people who share their same favourite tendencies under the below headings:

- **Activities** i.e, Sleeping, eating, reading.

- **Athletes** i.e, Roger Federer, Rafael Nadal, Leo Messi.
- **Books** i.e, Harry Potter, The Bible, Terry Pratchett.
- **Interests** i.e, Movies, Music, Cooking.
- **Movies** i.e, Inception, Avatar, Fight Club.
- **Music** i.e, Daft Punk, Muse, Pink Floyd.
- **People** i.e, Alan Turing, Maurice Moss, Steve Jobs.
- **Sports** i.e, Badminton, Basketball, Cycling.
- **Teams** i.e, Manchester United, Liverpool FC, Canberra Raiders.
- **Television** i.e, The Big Bang Theory, How I Met Your Mother, The Simpsons.

Based on the membership frequencies of our app users for these different *favourite* categories they can be partitioned into three distinct sets of high, medium and low frequency. Below we display example tables of these categories:

F	Television
20	The Big Bang [..]
19	How I Met [..]
14	The Simpsons
13	Top Gear
12	Futurama
12	Scrubs
11	Black Books
10	Black Books
10	South Park
10	Family Guy
9	The Daily Show
8	The IT Crowd
8	FRIENDS
7	True Blood
7	MythBusters

Table 4.4: *Television*.

F	Interest
5	Movies
5	Music
3	Cooking
3	Sports
2	Psychology
2	Internet
2	Video Games
2	Martial arts
2	Literature
2	Economics
2	Tennis
2	Badminton
2	Artificial intelligence
2	Computers
2	Travel

Table 4.5: *Interests*.

F	People
2	Alan Turing
1	Bender
1	Maurice Moss
1	Steve Jobs
1	Sean Parker
1	Pope Benedict XVI
1	Martin Luther
1	Alistair McGrath
1	St Augustine
1	Dennis Ritchie
1	Linus Torvalds
1	Richard Stallman
1	C. S. Lewis
1	Mike Oldfield
1	Ryan Giggs

Table 4.6: *People*.

Where the first column *F* displays the frequency for this *favourite* and the second column denotes the type of *favourite* displayed. Here we can see *television* represents an example of a high frequency *favourite* of around $20 - 10$, *interests* represent an example of a medium frequency *favourite* of around $9 - 5$ and *People* represent an example of a low frequency *favourite* of around $4 - 1$.

The different frequency levels are summarised between the *favourites* categories below:

- **High Locality**: *Music*, *movies*, *television* - Showing our app users appear to share similar *favourites* in a media setting.
- **Medium Locality**: *Activities*, *books*, *interests*, *sports* - Showing our app users share some degree of similar preferences across these *favourites*.
- **Low Locality**: *People*, *athletes*, *teams* - Showing our app users do not share many similar preferences across *favourites* involving specific people and teams.

For *favourites* each feature vector X where $X \in \mathbb{R}^I$ is composed of the above components where:

$$I = \{activities, athletes, books, interests, movies, music, people, sports, teams, television\}$$

The alters set for each i is conditioned by the relationship:

$$Relationship_{u,i} = \{z | SameMembership_{u,z,i}\}$$

In this case the $SameMembership_{u,z,i}$ function returns *True* if a user u shares a membership with user v via the current *favourite* i.

Applying this feature vector to our classifiers we obtain:

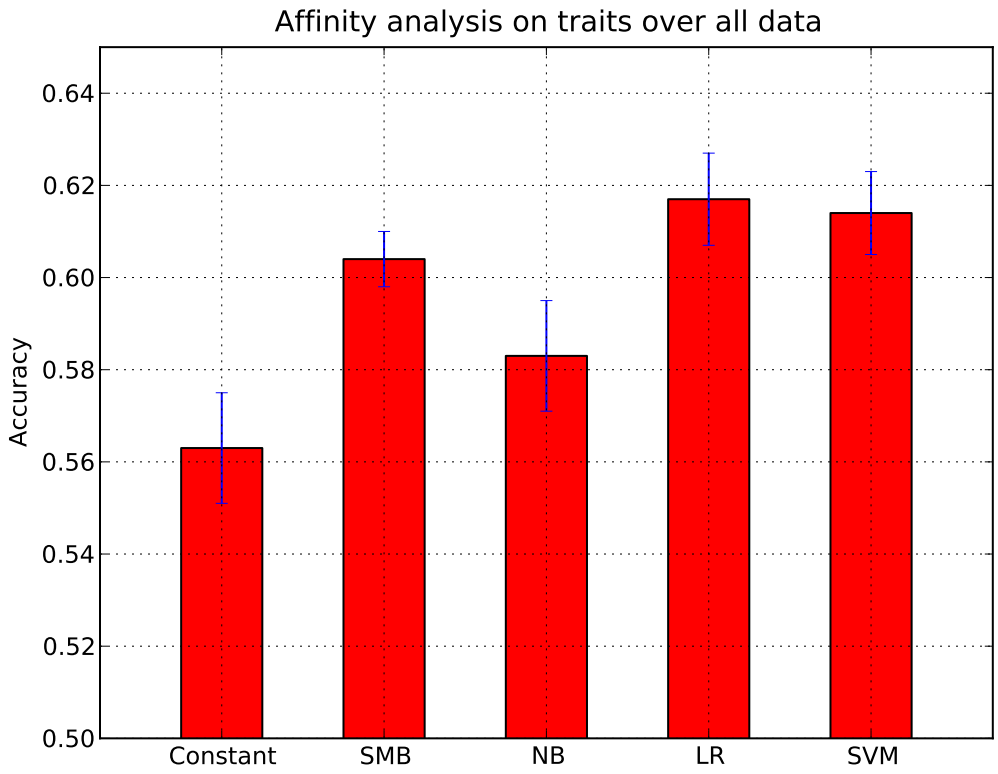

Figure 4.3: Accuracy results using the *favourites* feature. *Favourites* are our first feature more predictive then SMB for the case when $k = 0$, indicating that *favourites* are highly predictive of a user's preferences. Demonstrating the novel insight that affinity filtering can offer better results than collaborative filtering.

The *favourites* feature shows our first improvement over our SMB baseline in both the LR and SVM case for $k = 0$ demonstrating that *favourites* are more predictive then any previously applied feature or method.

Comparing *favourites* across exposure we obtain:

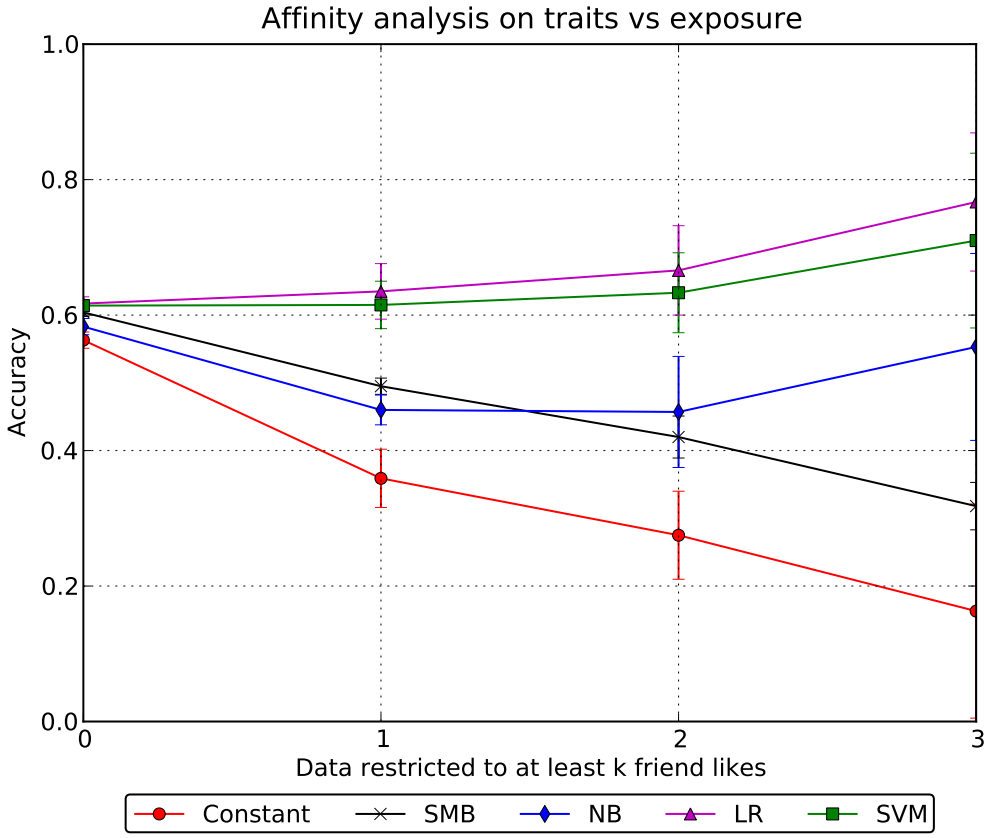

Figure 4.4: Accuracy results against exposure using the *favourites* features. Clearly, *favourites* are more predictive of user preferences compared to our baselines and all *user interactions* across all exposures k, due to the fact (unlike *interaction filtering*) *user preferences* do not require explicit interactions with friends.

This predictive trend continues with exposure where each successive increase of k causes the predictiveness of our classifiers to increase. Clearly, *favourites* are highly predictive of a user's like preferences.

Given the highly predictive nature of *favourites*, we extract the model weights from the most predictive classifier of LR for the case where $k = 0$. In the following table we can see explicitly which *favourites* are most predictive:

Favourite	Weight	Frequency
Activities	5.927 ± 0.001	281
Television	5.210 ± 0.0	1,029
Music	3.409 ± 0.001	629
Movies	2.668 ± 0.001	454
Interests	1.921 ± 0.001	64
Sports	1.820 ± 0.001	27
Books	1.769 ± 0.0	163

Table 4.7: LR feature weights. The *favourite* column displays the current *favourite*. The *weight* column shows the weighting given for this *favourite* and the *frequency* column displays the number of times this *favourite* was set to 1. We find that high frequency *favourites* are most predictive of dislikes.

This table shows that the *favourites* which exhibit a high frequency have a larger influence of indicating a like during prediction. These most predictive like *favourites* are mostly focused on media and leisure, notably no explicit *favourite* is given a negative feature weight which would be predictive of a dislike.

[Brandtzg and Nov 2011] found that virtual interactions help reveal common interests, while real world interactions help support friendships. Our data supports this, these common interests or *favourites* investigated above are clearly predictive of what a user will not like and providing our most predictive results so far.

4.3 Groups

Facebook facilitates users to join *groups* for a large number of different types ranging from local sports teams and political preferences to computer games. These *groups* facilitate users to associate themselves with other people who share a similar *group* preference. We outline the most popular groups for our app users in the table below:

Frequency	Group Name
27	ANU StalkerSpace
20	Facebook Developers
15	ANU CSSA
14	CSSA
13	Australian National University
11	ANU - ML and AI Stanford Course
10	iDiscount ANU
10	Our Hero: Clem Baker-Finch
9	Students In Canberra
7	I grew up in Australia in the 90s
7	Grow up Australia - R18+ Rating for Computer Games
7	ANU Engineering Students' Association (ANUESA) 2010
7	ANU Postgraduate and Research Student Association (PARSA)
6	No, I Don't Care If I Die At 12AM, I Refuse To Pass On Your Chain Letter.
6	No Australian Internet Censorship
6	The Chaser Appreciation Society
6	Feed a Child with a Click
6	ANU Mathematics Society
6	ANU International Student Services, CRICOS Provider Number 00120C
6	2011 New & Returning Burton & Garran Hall
5	If You Can't Differentiate Between "Your" and "You're" You Deserve To Die
5	Keep the ANU Supermarket!!!
5	If 1m people join, girlfriend will let me turn our house into a pirate ship
5	The Great Australian Internet Blackout
5	When I was your age, Pluto was a planet.

Table 4.8: Popular *groups* breakdown for our app users. The *groups* joined by our app users exhibit a high degree of locality, many of these top rated *groups* have an ANU and/or Canberra focus.

The most popular *groups* joined by our app users exhibit a high degree of ANU and/or Canberra based locality. The frequencies of these top *groups* are consistently higher then the most predictive *favourites* outlined in the previous section.

For *groups* each feature vector X where $X \in \mathbb{R}^I$ is composed of the product between:

$$I = \{Groups\} \times \{Optimal\ Group\ Size\}$$

Where the optimal *group* size is defined below for each classifier. The alters set for each i is conditioned by the relationship:

$$Relationship_{u,i} = \{z | SameGroup_{u,z,i}\}$$

In this case the $SameGroup_{u,z,i}$ function returns *True* if a user u and user z are both members of the same top *group* at index i in the top *groups* list, where a limited version of this list is shown in *Table 4.8* above.

Given the quantity of *groups* on Facebook, we first need to find the optimal *group* test size for our data set. Given memory and time constraints we tested within a range of $(100 - 1000)$ with an incremental step size of 100 for each test.

The results for these tests are shown below:

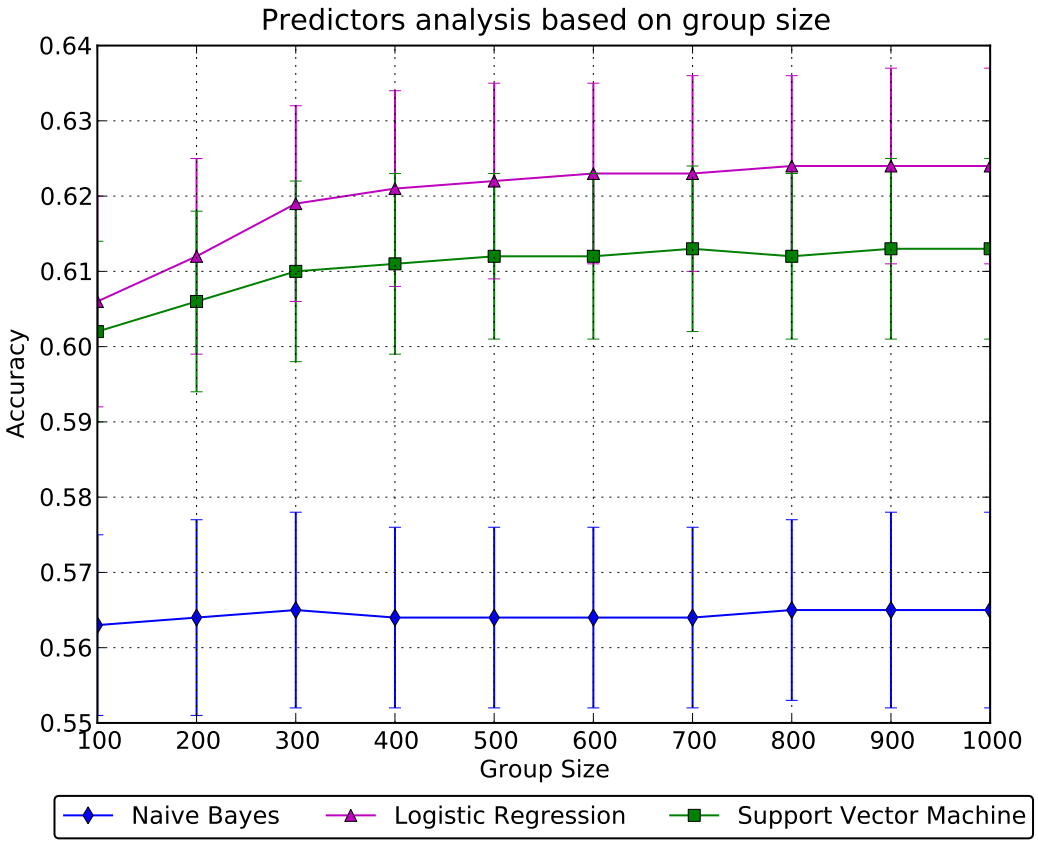

Figure 4.5: Accuracy results for testing using different *group* sizes. Unlike *messages*, the predictiveness of *groups* increases as the *groups* size increases, implying that the more *groups* considered the more predictive *groups* will be.

Here we see as the *group* size increases, the predictiveness of this feature also increases. LR and SVM show a gradual increase as this *group* size increases, alluding to the possibility of an even higher *group* size being optimal.

The most predictive *group* sizes for each of our classifiers are:

- **Naive Bayes**: 300.
- **Logistic Regression**: 900.
- **Support Vector Machine**: 800.

Using the most predictive *group* sizes defined above and comparing to our baselines we obtain:

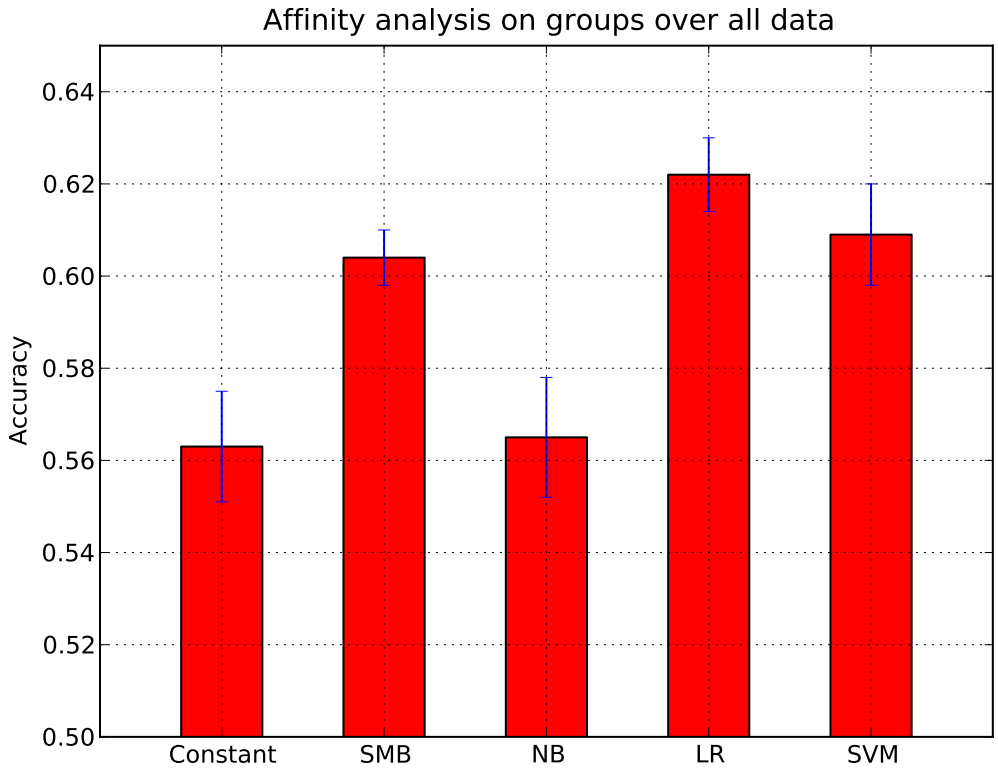

Figure 4.6: Accuracy results using the *groups* feature vector. We see *groups* are highly predictive of a user's like preferences, showing an improvement over *favourites* and our baselines.

Both LR and SVM show an improvement over *favourites* and our baseline for the case of $k = 0$ demonstrating that *groups* are highly predictive of a users like preferences.

Applying the *groups* feature across our exposure, we obtain:

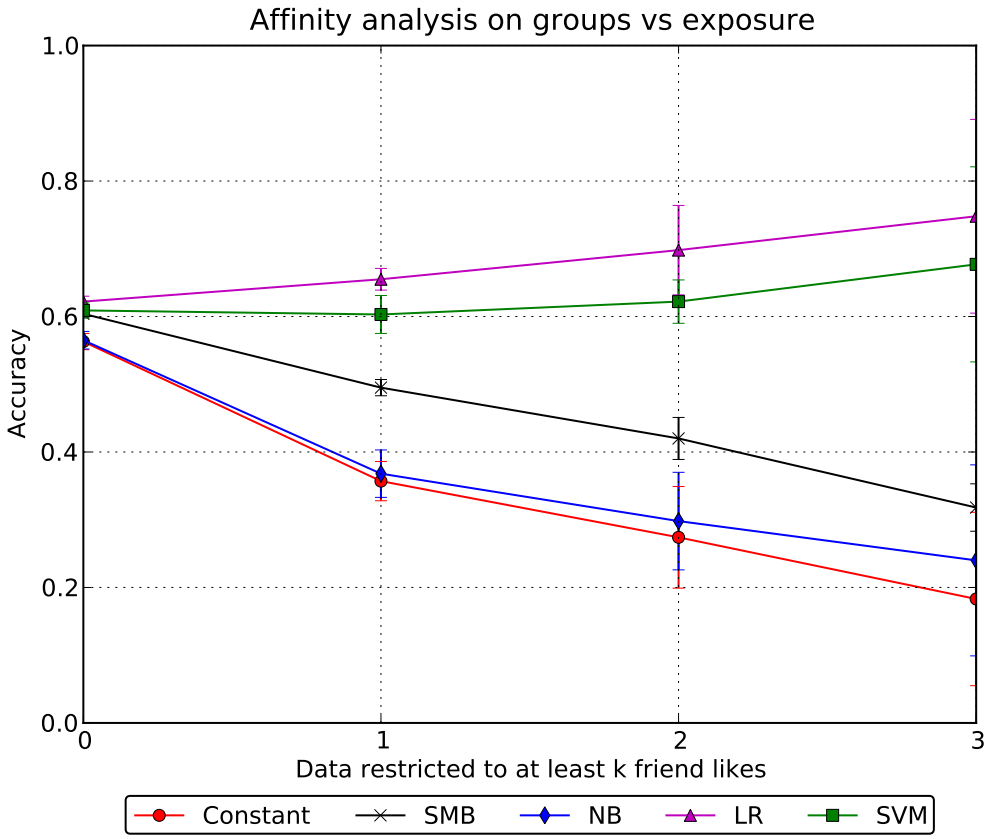

Figure 4.7: Accuracy results across exposure using the *groups* features. Here we see that the predictive trend for the $k = 0$ case continues as the predictive tendencies of *groups* increases across exposure.

The predictive qualities of *groups* continues with exposure where each successive increase of k causes the predictiveness of our LR and SVM classifiers to increase, demonstrating the intuition that the more users available to learn from, the better our classifiers can perform.

Given the predictive tendencies of *groups*, we extract the model weights for the case where $k = 0$ to see which specific *groups* contribute most to the predictions:

We have found *groups* are highly predictive of a user's like preferences, we see from the weights table above that *groups* which are highly local to the ANU and/or Canberra and have a high relative frequency of app users are most predictive of a user's likes. Notably, no *groups* contribute explicit dislike (negative) weights. The overall sizes of these *groups* is relatively small which adheres to the common sense notion that large groups appeal more broadly and are less predictive of a user's preferences.

Name	Size	Weight	Frequency
ANU StalkerSpace	1292	7.236 ± 0	453
Facebook Developers	487	3.442 ± 0	177
ANU CSSA	38	2.742 ± 0	191
Australian National University	619	2.565 ± 0	70
Overheard at the Ateneo de Manila University	253	2.462 ± 0	26
iDiscount ANU	338	2.203 ± 0	88
PETITION FOR FACEBOOK TO INSTALL A DISLIKE BUTTON	683	2.018 ± 0	92
I grew up in Australia in the 90s	731	1.991 ± 0	75
Grow up Australia - R18+ Rating for Computer Games	222	1.951 ± 0	102
Heavy Metal - CANBERRA METAL	30	1.694 ± 0	42

Table 4.9: LR feature weights. The *name* column displays the current *group* name. The *size* column shows the total size of this group across all users. The *weight* column shows the weighting given for this *group* and the *frequency* column displays the number of times this *group* was set to 1. We find that highly local *groups* with a high relative frequency of app users are most predictive of a users like preferences.

4.4 Pages

Facebook facilitates users to like *pages* for 'things' they want to associate themselves with across a wide spectrum of varied areas ranging from web browsers and TV shows to schools.

The most popular *pages* liked by our app users are shown in the table below:

Frequency	Page Name
33	ANU Computer Science Students' Association (ANU CSSA) 2011
32	The Australian National University
31	ANU Stalkerspace
21	Humans vs Zombies @ ANU
20	The Big Bang Theory
19	Australian National University
19	How I Met Your Mother
18	ANU LinkR
18	ANU ducks
17	Australian National University Students' Association
16	Google
15	Google Chrome
15	ANU XSA
15	Facebook
14	YouTube
14	The Simpsons
13	Portal
13	Top Gear
13	Music
13	ANU Memes
12	Futurama
12	Scrubs
12	ANU O-Week 2012: Escape to the East
12	The Stig
11	Black Books

Table 4.10: Popular *pages* breakdown for app users. *Pages* exhibit less locality preferences when compared with *groups*, while some *pages* are still ANU/Canberra focused, many are also more general. Additionally app user membership frequencies for *pages* are relatively much higher then *groups*.

In comparison with *groups*, top *pages* show a relatively higher frequency among app users and have less of an ANU/Canberra centric focus.

For *pages* each feature X where $X \in \mathbb{R}^I$ is composed of the product between:

$$I = \{Pages\} \times \{Optimal\ Page\ Size\}$$

Where the optimal *page* size is defined below for each classifier. The alters set for each i is conditioned by the relationship:

$$Relationship_{u,i} = \{z | SamePage_{u,z,i}\}$$

In this case the $SamePage_{u,z,i}$ function returns *True* if a user u and user z are both

members of the same top *page* list at index i, where a limited version of this top list is shown in *Table 4.10* above.

Given the quantity of *pages* on Facebook, we need to find some optimal test size for each classifier. Given memory and time constraints we tested within a range of $(100 - 1000)$ with an incremental step size of 100 for each test.

The results of these tests are shown below:

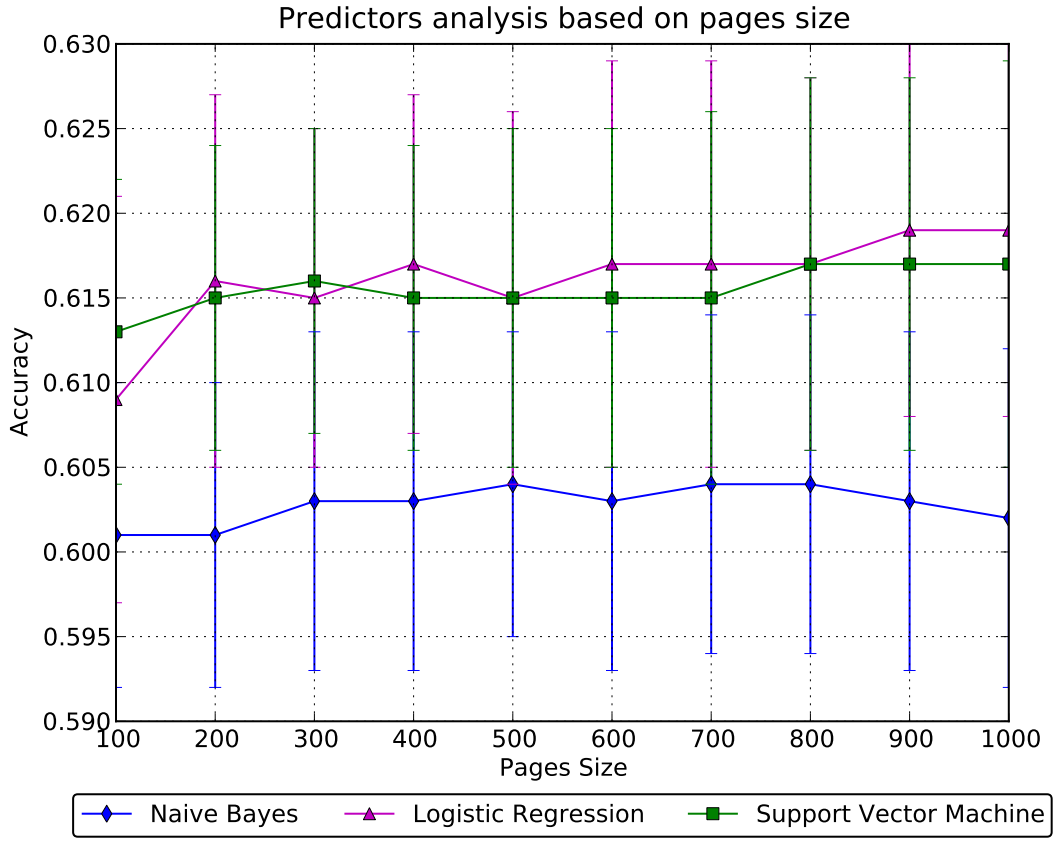

Figure 4.8: Accuracy results for different *page* sizes. Results for *page* sizes are quite jumpy across relatively medium page sizes, however a similar trend from *groups* follows that the more *pages* we use for testing, the more predictive our results.

LR and SVM exhibit a gradual (though jumpy) increase as our *page* size increases, alluding to the possibility of an even higher *page* size being optimal for prediction.

The most predictive *page* sizes for each of our classifiers are:

- **Naive Bayes**: 500.
- **Logistic Regression**: 900.
- **Support Vector Machine**: 800.

Using the most predictive *page* sizes as defined above we obtain:

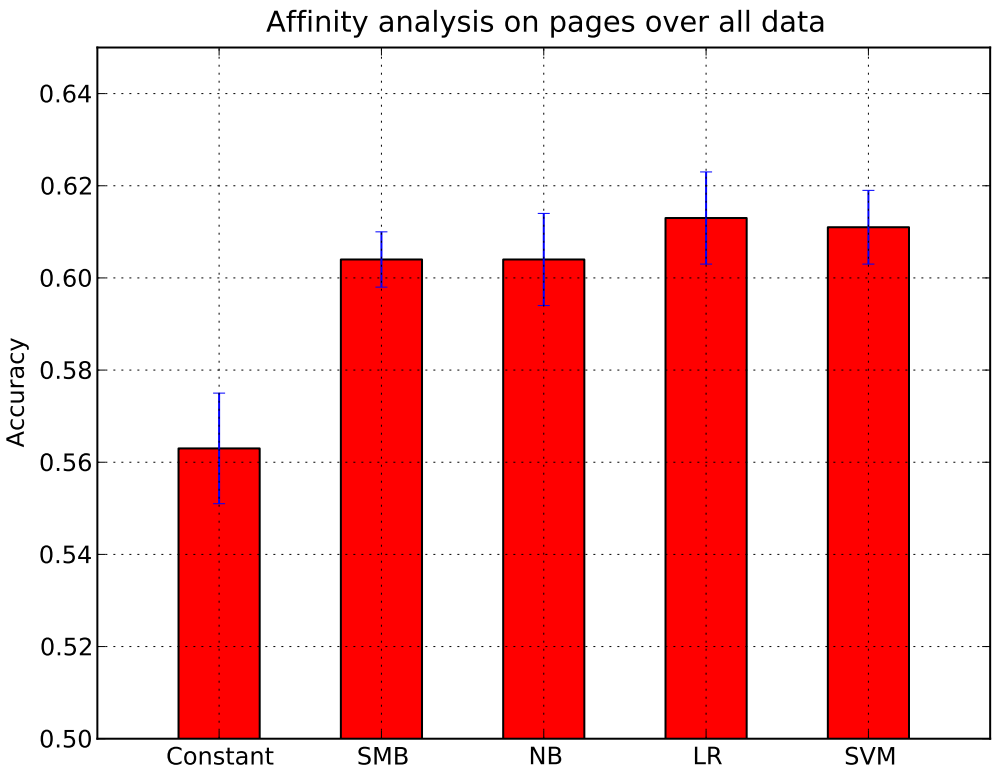

Figure 4.9: Accuracy results using the *pages* feature, we can see an improvement over our baselines for both LR and SVM. Demonstrating that while *pages* are predictive of a user's like preferences, they are not as predictive as *groups* or *favourites*.

We can see LR and SVM show a prediction improvement over our SMB baseline for the case of $k = 0$ demonstrating that *pages* are also highly predictive. Though not as predictive as *groups* and *favourites*. Possibly due to the fact that *groups* are generally more local and they have a more relatively mid-ranged frequency among app users.

Applying *pages* against exposure, we obtain:

Figure 4.10: Accuracy results against exposure using the *pages* feature. We see a similar trend as demonstrated in *groups* and *favourites* where the predictiveness of this feature improves with exposure.

The similar trend of improved predictiveness over exposure similarly continues with *pages*. As more data is available for the classifiers to learn from, the accuracy of the prediction increases.

By extracting the model weights from the case where $k = 0$ we can see which explicit *pages* are most predictive of a user's like preferences:

Name	Size	Weight	Frequency
Sorry mate i can't, i've got Quidditch	254	1.799 ± 0	18
Avatar: The Last Airbender	324	1.514 ± 0.001	13
National Geographic	662	1.437 ± 0.001	18
The Simpsons	1552	1.414 ± 0	170
Sushi	387	1.33 ± 0.001	9
House	1746	1.291 ± 0	66
Seinfeld	609	1.249 ± 0	15
Starbucks	1548	1.249 ± 0	7
American Dad	540	1.215 ± 0.001	18
friends don't let friends vote for Tony Abbott	551	1.206 ± 0.001	19

Table 4.11: Positive LR feature weights. The *name* column displays the current *page* name. The *size* column shows the total size of this group across all users. The *weight* column shows the weighting given for this *page* and the *frequency* column displays the number of times this *page* was set to 1. We find non-local and a high range of *page* sizes to be most predictive.

Name	Size	Weight	Frequency
CatDog	259	-1.815 ± 0.001	12
Worst. Idea. Ever. [pause] Let's do it.	227	-1.737 ± 0	21
Grug	279	-1.698 ± 0	9
Kings Of Leon	840	-1.607 ± 0.001	14
Planking Australia	166	-1.598 ± 0.001	4
Dr. House	964	-1.588 ± 0	28
Suit Up	466	-1.389 ± 0.001	17
Don't you hate it when Gandalf marks your [..]	110	-1.372 ± 0.001	19
Paramore	1004	-1.343 ± 0.001	31
Tintin	250	-1.339 ± 0.001	11

Table 4.12: Negative LR feature weights. The *name* column displays the current *page* name. The *size* column shows the total size of this group across all users. The *weight* column shows the weighting given for this *page* and the *frequency* column displays the number of times this *page* was set to 1. We find non-local and a high range of *page* sizes to be most predictive.

For this case we find individual *pages* predictive for both likes (positive weights) and dislikes (negative weights). The individually most predictive *pages* in our data are non-local (which was the opposite case to *groups*) and relatively highly varied. This trend continues for *pages* across both like and dislike weight influence.

4.5 Conclusion

Throughout this chapter we have explored different *user preferences* available for users to demonstrate their personal preferences across a range of different topics and mediums.

Others have pointed out that non-social information is more predictive of user likes [www 2012] and we observe that too. We have found that *user preferences* are more predictive of user likes compared with *user interactions*, this is particularly true for *favourites*, *groups* and *pages*. This observation holds true for the exposure case of $k = 0$ and these accurate predictions continue to improve with each successive increase in k (notably at the detriment of our baseline). Notably, in terms of their individual weight components *favourites* and *groups* actively weight to predict what a user will dislike, while *page* weights are used for both like and dislike predictions.

The novel insight we find throughout this chapter is *user preference* affinity filtering can offer more predictive results then previously applied collaborative filtering techniques.

In the next chapter we will investigate the effect of combining the individually predictive affinity features found in this chapter.

Chapter 5

Feature Combination

Given the vast number of affinity features outlined in the previous sections and the scope of users and associated data on Facebook, it is computationally costly, time consuming and ineffective to naively combine all features. Hence, it is crucial we provide a practical approach which facilitates the combination of the individually most predictive affinity features, from which we can efficiently learn.

In this chapter we combine these individually most predictive affinity features together and examine the results.

5.1 Affinity Feature Selection

Based on results found during our *user interactions* and *user preference* analysis, the most predictive features we have found were:

- **Favourites**.
- **Groups**.
- **Pages**.

The following results and analysis are based on these above features combined.

Applying this combined affinity feature to the data set we obtain:

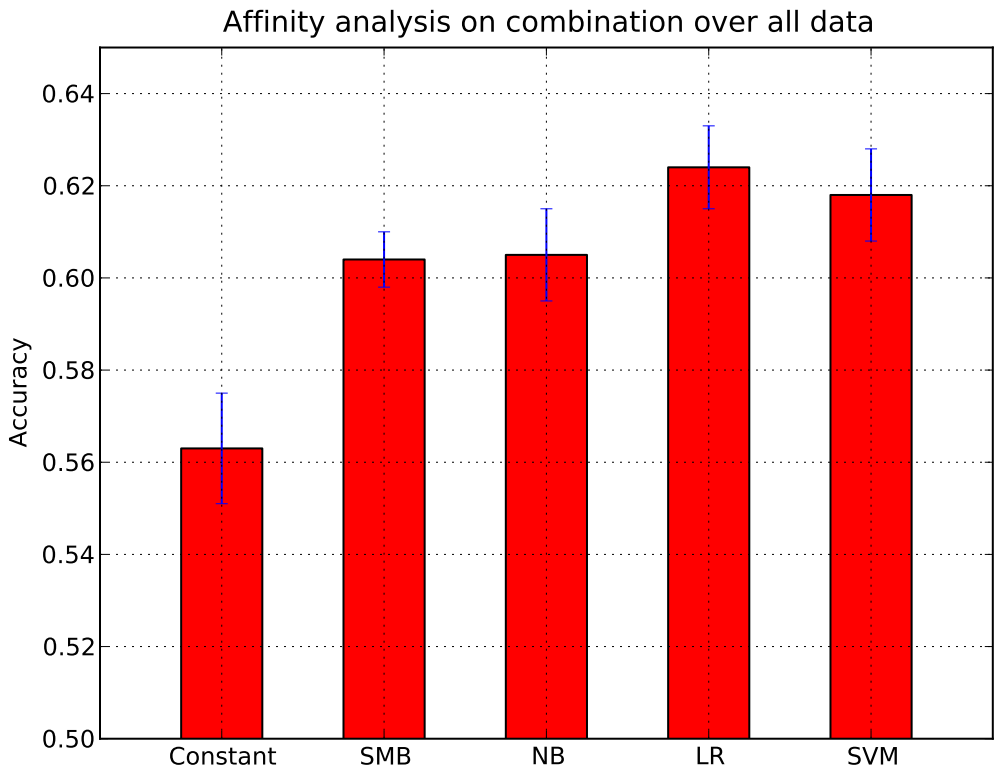

Figure 5.1: Accuracy results using the *combined* feature set. This *combined* features give our best prediction in comparison to all other individual features and we see a large improvement over our SMB baseline, particularly for the LR case.

We find that the *combined* affinity feature gives the most predictive results when compared with both our baselines and all other individual features tested during this research.

The most predictive individual results of *favourites*, *groups* and *pages* against the *combined* feature are tabulated below:

Classifier	**Accuracy**
NB	0.583 ± 0.012
LR	0.617 ± 0.01
SVM	0.614 ± 0.009

Table 5.1: *Favourite* feature results for $k = 0$.

Classifier	**Accuracy**
NB	0.604 ± 0.01
LR	0.613 ± 0.01
SVM	0.611 ± 0.008

Table 5.2: *Pages* feature results for $k = 0$.

Classifier	**Accuracy**
NB	0.565 ± 0.013
LR	0.622 ± 0.008
SVM	0.609 ± 0.011

Table 5.3: *Groups* feature results for $k = 0$.

Classifier	**Accuracy**
NB	0.605 ± 0.01
LR	0.624 ± 0.009
SVM	0.618 ± 0.01

Table 5.4: *Combined* feature results for $k = 0$.

These tables clearly show that the most predictive feature vector is a combination of the individually most predictive features found during our analysis. This trend continues across our exposure k and this *combined* feature offers the most predictive results found during this research over all exposures.

This can be seen in the graph below:

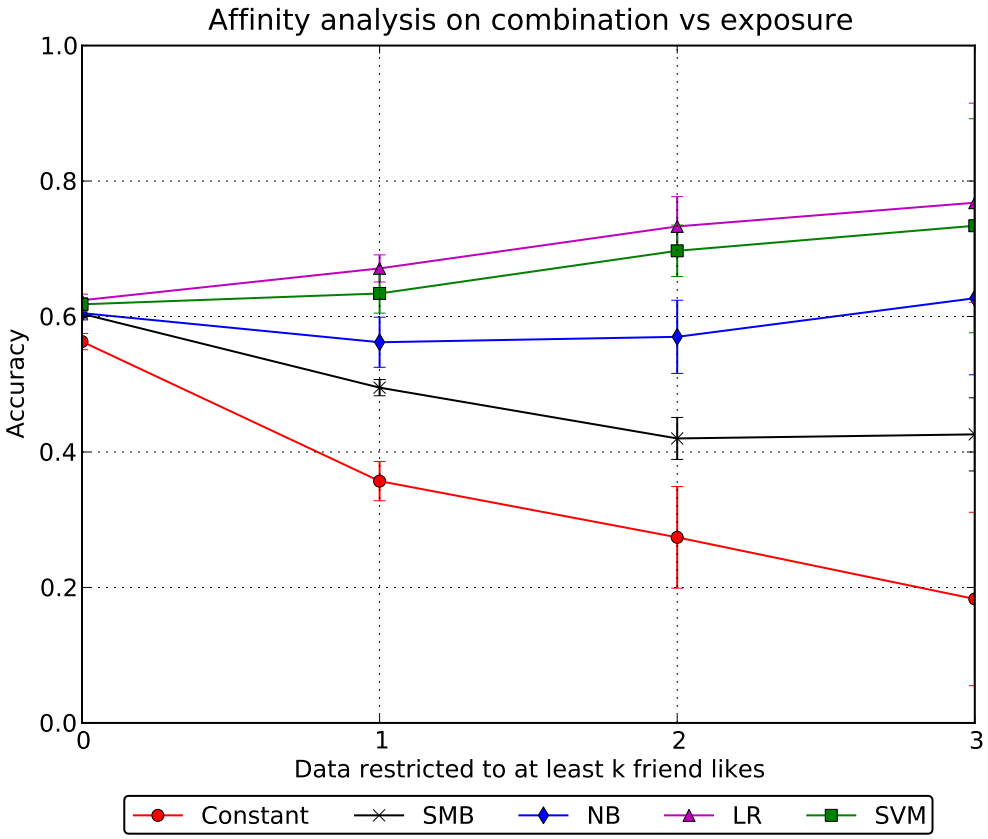

Figure 5.2: Accuracy results across exposure using the *combined* feature set. This feature offers the most predictive results discovered during this research for all k. As our exposure increases, the improved predictive performance of this *combined* feature increases too.

By extracting the model feature weights we can see which *combined* features were most predictive:

Name	Size	Weight	Frequency
(Page) Avatar: The Last Airbender	324	1.68 ± 0.001	13
(Page) I'm late. Got attacked by a wild Pokemon	161	1.609 ± 0	20
(Group) Overheard at the Ateneo de Manila	253	1.527 ± 0.001	26
(Page) Sorry mate i can't, i've got Quidditch	254	1.501 ± 0	18
(Group) I would.....for Escapium.	50	1.467 ± 0.001	11
(Group) Burgtoons	34	1.37 ± 0.001	7
(Page) The Simpsons	1552	1.355 ± 0.001	170
(Group) City Gate Hall	27	1.346 ± 0	5
(Page) Victoria's Secret	764	1.337 ± 0	11
(Page) Starbucks	1548	1.313 ± 0	7

Table 5.5: LR feature weights extracted for the positive case. The *Name* column displays the name of the feature. *Size* represents the size of the *Page* or *Group*. *Weight* represents the weight this feature vector received. *Frequency* displays the number of times this feature vector was set to 1 for a user.

Name	Size	Weight	Frequency
(Page) Don't you hate it when Gandalf marks [...]	110	-1.627 ± 0.001	19
(Page) Goodberry's	318	-1.591 ± 0	73
(Page) Worst. Idea. Ever. [pause] Let's do it.	227	-1.561 ± 0	21
(Page) CatDog	259	-1.531 ± 0.001	12
(Page) Planking Australia	166	-1.501 ± 0.001	4
(Page) Avenged Sevenfold	351	-1.471 ± 0	6
(Page) Grug	279	-1.465 ± 0	9
(Page) Dr. House	964	-1.451 ± 0	28
(Group) If 1m people join, girlfriend will let [...]	416	-1.362 ± 0	68
(Page) Do you ride kangaroos? no mate the [...]	321	-1.333 ± 0.001	23

Table 5.6: LR feature weights extracted for the negative case. The *Name* column displays the name of the feature. *Size* represents the size of the *Page* or *Group*. *Weight* represents the weight this feature vector received. *Frequency* displays the number of times this feature vector was set to 1 for a user.

The positive LR weights are equally broken up into *pages* and *groups* as contributing highly to a prediction of a like. The size ranges for both *pages* and *groups* varies greatly from as little as 27 up to as high as 1552 and there is little corelation between frequency and sizes, in fact lower frequencies of around 20 contribute most to a like prediction.

The negative LR weights are more focused on the predictiveness of *pages* for a dislike and the sizes of the *pages* are in a much more consistent range, while the frequencies still vary.

These LR weights show that both *groups* and especially *pages* are highly predictive of a user's like preferences and particular sizes or frequencies do not appear to be more predictive then others. Both *groups* and *pages* are indicative of a like prediction while *pages* are most predictive for a dislike. Additionally under this *combined* paradigm, *favourites* do not appear to contribute strongly to our results.

We have shown in this chapter that the *combined* feature of our individually most predictive features results in the most accurate and concise predictions from our data set, when combined *favourites* are less predictive then *groups* and *groups* are less predictive then *pages*. *Pages* and *groups* both contribute to a dislike prediction, while a like prediction is more strongly associated with *pages* - which is not surprising as *pages* are more predictive then *groups*. Additionally, the sizes and frequencies of these *groups* and *pages* do not appear to be consistent in their predictive qualities.

Chapter 6

Conclusion

In this chapter we will outline a summary of the work completed during this thesis and discuss our proposal for future work in this area.

6.1 Summary

In this thesis we have tested and compared an exhaustive list of different affinity features across varied exposure sizes.

We have found that *user interaction* affinity features in themselves are not predictive of user likes, however coupled against exposure, they show a comprehensive predictive improvement over our baselines.

Conversely, we have shown the interesting result that *user preference* affinity features are more predictive of user likes and this trend continues across user exposure.

We have investigated individual groups and found that the most predictive were highly localised with a medium user frequency, while the most predictive pages were much more general and of a higher relative user frequency.

To answer the question initially proposed for this thesis, we have shown that the affinity features which provide the highest predictiveness of user likes come from *user preferences* and not *user interactions*. The most predictive features found in this analysis are *favourites*, *group memberships* and *page likes*. Additionally a combination of these individually predictive features results in the most predictive classification.

Which summarises the exciting and novel insights examined during this thesis.

6.2 Future Work

Proposed future work can be summarised under the following points:

- **Increase size ranges**: Given our maximum test sizes used for *groups* and *pages* and the trend that an increased size range resulted in better prediction, this size could be increased to find the optimal testing range for each of our classifiers.

- **Learning from only Positive Label Data**: Given the Facebook model of allowing users to like but not dislike data, explicit dislike data can not be gleaned

from Facebook, which is why the active likes sourced from the NICTA app were used during this evaluation. An approach could be developed which can predict whether a user will have seen an item (online timestamps, recent interactions with a user, etc) and we can infer that if the user did not like the item then they disliked it. This data set could then be applied to the testing methodology outlined in the above chapters.

- **General user set**: As outlined in our *demographics* section, the user set used during this analysis is mostly composed of computer science undergraduates (and graduates). Applying the methods outlined above to a more general user set such as in the study done by [Ugander and Marlow 2011] which comprised of the entire active social network of 721 million users (as of May 2011), which is more indicative of the general Facebook user population could offer more generalisable results.

- **Bayesian Model Averaging**: Combining the most predictive qualities of the individually trained classifiers by sampling and weighting the results by applying Bayes' law, this approach does not require explicit feature selection (as outlined in *Chapter 5*).

6.3 Concluding Remarks

Our approach to affinity filtering has not been previously explored, the rich Facebook data-set available has allowed us to compare the importance of individual affinity features and discover which are most predictive of a user's like preferences. In contrast to previous social CF work, we leverage fine-grained *user interactions* and common *user preferences* and show that these are highly predictive, even without an individual latent user/item model.

This methodology (and associated empirical results) are the exciting and novel insights examined in this research; we have shown the impressive result that the explicit *user preferences* of *favourites*, *groups* and *pages* (individually as well as in combination) outperform SMB across the board. By applying these non user-specific features, we have developed a highly scalable and more predictive approach than previous methods, making this the current state-of-the-art technique for recommendation on rich social networks such as Facebook.

Bibliography

2012. New objective functions for social collaborative filtering. In *WWW-12* (2012). (pp. 6, 13, 23, 46)

ALIAS-I. 2008. LINGPIPE 4.1.0. HTTP://ALIAS-I.COM/LINGPIPE (ACCESSED OCTOBER 1, . 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology 2*, 27:1–27:27. (p. 10)

ANDERSON, A., HUTTENLOCHER, D., KLEINBERG, J., AND LESKOVEC, J. 2012. Effects of User Similarity in Social Media. In *ACM International Conference on Web Search and Data Mining (WSDM)* (2012). (p. 23)

BACKSTROM, L., BAKSHY, E., KLEINBERG, J., LENTO, T., AND ROSENN, I. 2011. Center of attention: How facebook users allocate attention across friends. ICWSM'11 (2011). (p. 25)

BRANDTZG, P. B. AND NOV, O. 2011. Facebook use and social capital — a longitudinal study. ICWSM'11 (2011). (p. 34)

CHANG, C.-C. AND LIN, C.-J. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology 2*, 27:1–27:27. (p. 11)

CUI, P., WANG, F., LIU, S., OU, M., AND YANG, S. 2011. Who should share what? item-level social influence prediction for users and posts ranking. In *International ACM SIGIR Conference (SIGIR)* (2011). (p. 9)

FACEBOOK. 2012. Facebook - 1 billion active people fact sheet (2012). (p. 5)

GRANOVETTER, M. S. 1978. Threshold models of collective behavior. *Am. J. Sociol 83(6):14201443*. (p. 1)

LANG, K. 1995. NewsWeeder: Learning to filter netnews. In *12th International Conference on Machine Learning ICML-95* (1995), pp. 331–339. (p. 8)

NOEL, J. G. 2011. New social collaborative filtering algorithms for recommendation on facebook (2011). (pp. 8, 9)

PANTEL, A., GAMON AND HAAS. 2012. *Proceeding SIGIR '12 Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*. ACM. (p. 1)

RESNICK, P. AND VARIAN, H. R. 1997. Recommender systems. *Communications of the ACM 40*, 56–58. (p. 8)

SAEZ-TRUMPER, D., NETTLETON, D., AND BAEZA-YATES, R. 2011. High correlation between incoming and outgoing activity: A distinctive property of online

social networks? In *Fifth International AAAI Conference on Weblogs and Social Media*, ICWSM '11 (2011). (p. 13)

UGANDER, B., KARRER AND MARLOW. 2011. The anatomy of the facebook social graph. *CoRR abs/1111.4503*. (pp. 25, 26, 54)

WATTS, D. J. AND STROGATZ, S. H. 1998. Collective dynamics of small-world networks. *Nature 393(6684):440442*. (p. 1)

YANG, LONG, SMOLA, SADAGOPAN, ZHENG, AND ZHA. 2011. Like like alike: Joint friendship and interest propagation in social networks. In *WWW-11* (2011). (p. 9)