Kybernetes

A nearest-neighbor algorithm for targeted interaction design in social outreach campaigns

Christopher Garcia

A nearest-neighbor algorithm for targeted interaction design in social outreach campaigns

Christopher Garcia
College of Business, University of Mary Washington, Fredericksburg, Virginia, USA

Abstract

Purpose – Organizations rely on social outreach campaigns to raise financial support, recruit volunteers, and increase public awareness. In order to maximize response rates, organizations face the challenging problem of designing appropriately tailored interactions for each user. An interaction consists of a specific combination of message, media channel, sender, tone, and possibly many other attributes. The purpose of this paper is to address the problem of how to design tailored interactions for each user to maximize the probability of a desired response.

Design/methodology/approach – A nearest-neighbor (NN) algorithm is developed for interaction design. Simulation-based experiments are then conducted to compare positive response rates obtained by two forms of this algorithm against that of several control interaction design strategies. A factorial experimental design is employed which varies three user population factors in a combinatorial manner, allowing the methods to be compared across eight distinct scenarios.

Findings – The NN algorithms significantly outperformed all three controls in seven out of the eight scenarios. Increases in response rates ranging from approximately 20 to 400 percent were observed.

Practical implications – This work proposes a data-oriented method for designing tailored interactions for individual users in social outreach campaigns which can enable significant increases in positive response rates. Additionally, the proposed algorithm is relatively easy to implement.

Originality/value – The problem of optimal interaction design in social outreach campaigns is scarcely addressed in the literature. This work proposes an effective and easy to implement solution approach for this problem.

Keywords Decision making, Social networks, Adaptation, Algorithms, Artificial intelligence

Paper type Research paper

1. Introduction

Over the past decade social media has come to occupy a place of prominence in the way communication occurs between individuals. Its pervasiveness has also transformed the way individuals interact with organizations. Organizations are now able to have rich, highly personalized interactions with their volunteers, supporters, customers, and other user groups because of the widespread availability of data as well as the ability to engage users through the same social channels in which they engage their friends and colleagues. Accordingly, social media has become a critical strategic asset for organizations to employ in outreach campaigns. In order to deliver meaningful, tailored engagements, organizations must be able to intelligently use their data to make accurate inferences about users at the individual level and then be able to determine the most appropriate content to be provided to each user.

Social media outreach campaigns are a powerful tool for organizations trying to recruit volunteers or raise support. In a social media outreach campaign an organization interacts with users through one or more channels, with the goal of

This research was funded by a Jepson Fellowship awarded by the University of Mary Washington.

Kybernetes
Vol. 45 No. 8, 2016
pp. 1243-1256
© Emerald Group Publishing Limited
0368-492X
DOI 10.1108/K-09-2015-0236

obtaining a desired response for each user. A desired response may consist of a user registering to volunteer, donating money, or even simply giving a Facebook like. Each user can be thought of as a specific combination of multiple features, and an interaction may likewise be thought of as a combination of features. For example, a user has a specific age, gender, political affinity, income, and so on. An interaction likewise has, for instance, a primary subject and tone (such as fact based or humorous). There are also other interaction features not directly part of the content but which nevertheless have strong bearing on whether a user will respond positively or not, such as which channel or through whom the interaction was sent (Booth and Matic, 2011). This work addresses the problem of how to design appropriate interactions for individual users in social outreach campaigns, with the goal of maximizing the probability of a desired response from each user. Both targeted advertising and product recommendation involve determining one or more products likely to be desired by an individual user and then presenting the most likely of these to the user. By contrast, in the interaction design problem the task is to find the right combination of features to combine together into a single interaction which maximizes the probability of a user responding positively, given the user's unique combination of features. In this work a nearest-neighbor (NN) interaction design algorithm is developed. Computational experiments are then conducted to test this algorithm under multiple scenarios with differing user populations. The results are discussed with respect to the gains in response rates that may be achieved by employing this algorithm.

2. Related work

Targeted advertising has long been recognized in the marketing literature for its effectiveness (Armstrong and Kotler, 2014). Targeted advertising involves differentiating between customers and then advertising products to each customer that they are most likely to buy. Traditionally, this consisted of identifying associations between products and relatively broad customer characteristics such as age, income, or occupation. In recent years there have been many sophisticated techniques developed which use vast amounts of customer data to provide far more tailored forms of advertising. These techniques have taken form in the widely known recommendation engines of companies such as Amazon, Netflix, Pandora, and Spotify. In the domain of recommender systems, there are several major paradigms commonly employed today: content-based filtering, collaborative filtering, and hybrid (Su and Khoshgoftaar, 2009; Jannach *et al.*, 2010). Content-based filtering makes recommendations based on aligning product descriptors to individual user profiles using domain knowledge. Collaborative filtering, by contrast, makes recommendations to users based on what other similar users have rated highly or by which items most often were purchased by other uses with a similar purchase history (Si and Jin, 2003). Hybrid methods simply combine aspects of several paradigms together.

There are two primary forms of the CF problem: user-based CF and item-based CF. The user-based CF problem consists of a set of m users and n items, resulting in an $m \times n$ matrix in which each row contains a user's ratings of the different items. Using this information the objective is to present a set of the top-N recommendations to each user based on what similar users have rated highly (Su and Khoshgoftaar, 2009). In item-based CF there is an $n \times n$ matrix which contains item-item similarities derived from customer rating or purchasing history. This information is then used to determine the top recommendations for each user based on what they have viewed or purchased (Sarwar *et al.*, 2001).

In their comprehensive review of CF techniques, Su and Khoshgoftaar (2009) broadly categorized the techniques widely employed as either memory based, model based, or hybrid. Memory-based involve computing the user-item or item-item matrices periodically and then storing them. Memory-based CF algorithms employ NN approaches to find the top-N recommendations by applying a distance or similarity metric. Common distance metrics include the Pearson correlation, the constrained Pearson correlation, the Spearman rank correlation, and vector-based cosine similarity (McLaughlin and Herlocker, 2004; Herlocker *et al.*, 2004; Goldberg *et al.*, 2001). Because results are pre-computed, there is little computational work needed to quickly retrieve a set of recommendations for users. Memory-based techniques are thus easy to implement and effective in many instances (Hofmann, 2004; Linden *et al.*, 2003). One of the drawbacks encountered with memory-based techniques, however, lies in the often highly sparse matrices encountered. Accordingly, model-based techniques apply quantitative techniques suitable to sparse data to the problem of CF. Model-based CF techniques include TAN-ELR, which combines naïve Bayesian methods with logistic regression (Su and Khoshgoftaar, 2006; Greinemr *et al.*, 2005), association retrieval, which employs activation algorithms for exploring associations (Huang *et al.*, 2004), maximum margin matrix factorization techniques (Srebro *et al.*, 2005), and imputation techniques (Su *et al.*, 2008; DeCoste, 2006; Noh *et al.*, 2004). In addition, clustering techniques (Rongfei *et al.*, 2010; Chee *et al.*, 2001), Markov decision processes (Shani *et al.*, 2005), and latent semantic models (Hofmann, 2004). Hofmann, 2004 have also been employed in model-based CF. Hybrid techniques combine content analysis with CF (Melville *et al.*, 2002) as well as combining multiple CF techniques together (Melville *et al.*, 2002; Pennock *et al.*, 2000).

In US politics, particularly in presidential campaigns, the use of sophisticated analytical approaches to support social media outreach has proven highly successful. As a consequence, the application of analytics to political campaigns has generated significant public interest (Issenberg, 2012; Harfoush, 2009). Nickerson and Rogers (2014) provide a recent survey on the types of analytical problems and techniques which are frequently encountered in modern political campaigns. Analytical activities tend to focus in constructing three types of scores: behavior scores, support scores, and responsiveness scores. Behavior scores yield probabilities that individuals will engage in certain types of political activities, support scores predict political preferences of individuals, and responsiveness scores predict how individuals will respond to campaign outreach. The latter of these three, responsiveness scores, have particular relevance for this work. Randomized field experiments are often used to measure average responsiveness of different campaign tactics, and the resulting responsiveness scores are then used to guide targeting decisions (Arceneaux and Nickerson, 2010; Nickerson and Rogers, 2010). Like the interaction design problem addressed in this paper, responsiveness scores are employed to increase the likelihood of a desired response. One of the differences, however, is that interactions are inherently multidimensional, and finding the optimal interaction for a particular user requires finding a particular combination of features.

In summary, targeted advertising methods, collaborative filtering, and political campaign models are all targeted methodologies aimed at increasing the likelihood of obtaining a desirable response from a user. However, the problem of modeling and optimally designing multidimensional interactions for individual users in outreach campaigns appears to be scarcely addressed in the literature. Accordingly, in this work an interaction design algorithm is presented and evaluated.

3. Problem definition

In this problem a user X is represented as a vector of features such as age, geographic location, political affinity, and so on. An interaction design Y (interaction hereafter) is represented by a vector of categorical features such as theme, tone, sender, media channel, and so on. Each interaction is composed of the categorical features $F = \{F_1, F_2, \ldots, F_n\}$, and each feature F_i takes on a value from the set of levels $L(F_i)$ in each interaction. As an example, a user could be represented by the vector (AGE = 25, GEOGRAPHIC_LOCATION = "Eastern USA," POLITICAL_AFFINITY = Democrat") and an interaction design could be represented by the vector (THEME = "Environmental," SENDER = "Al Gore," MEDIA_CHANNEL = "Twitter"). For each user X and interaction Y there is a response $R \in \{0, 1\}$ where 1 denotes that the user responds to the interaction as desired by the organization (called a positive response), and 0 denotes that they do not. The objective is to find for each user X a corresponding interaction Y that maximizes the probability $P(R = 1|X, Y)$.

From the organization's perspective, each user is simply a specific combination of features (as is each interaction). It may be the case that there are many users an organization interacts with which have exactly the same set of feature values. It is thus helpful to think of users as equivalence classes rather than individuals, and we will treat a user and user class as one and the same. In an organization's data it is expected that for a user class X there are multiple records of responses from X to different interactions. We let $M(X)$ represent the bag of all interaction instances to which a user of class X responded. Note that there may be many duplicate interaction instances in $M(X)$ since there can be many users of the same class who responded to the same interaction. We further define $M^+(X) \subseteq M(X)$ as the bag of all interactions sent to user class X which received a positive response.

Individual interaction feature values may each increase or decrease the propensity of a specific user to respond positively. A user X, for example, may be more interested in hearing from a business leader than an actress. It may then be expected that, all other aspects being equal, an interaction sent by a business leader is at least as likely to result in a positive response as one sent by an actress. Similarly, a user may be more concerned with environmental issues than economic ones. Accordingly, we expect the user to be at least as likely to respond positively to an interaction with an environmental theme as one with an economic theme, all other aspects being equal. This property is referred to as the feature carrying assumption; its violation would clearly constitute unusual circumstances. This property is assumed, and may be formally stated as follows: given user X and interaction Y containing feature subset $S \subseteq F$ and feature $W \in F$ such that $W \notin S$, $P(R = 1|W = a) \geqslant P(R = 1|W = b)$ implies $P\left(R = 1\middle|\cap_{F_i \in S} F_i = v_i \cap W = a\right) \geqslant P\left(R = 1\middle|\cap_{F_i \in S} F_i = v_i \cap W = b\right)$ for all $a, b \in L(W)$ and $v_i \in L(F_i)$.

4. NN algorithm

In this section we present a NN algorithm for constructing interactions for each user to maximize the probability of positive response. We begin by a restricted case and then generalize this to the full problem.

4.1 A restricted case: a single user class

We first consider the case of a single user X, where there is a bag of interactions $M(X)$ to which X responded and a sub-bag $M^+(X)$ of positive responses. By examining this case, we may derive properties which will be used in the general case of many users. We begin by noting that the organization is on control of which interactions a user X receives. Furthermore, it is clear that if user X receives 50 interactions with an environmental theme but only two with an economic theme then it is impossible to draw conclusions about the probability of a positive response to environmental themes relative to economic ones. To properly do so, an approximately balanced and sufficiently sized sample of interactions must be sent to the user. For this analysis we thus assume a balanced sampling of feature values reflected in $M(X)$. In essence this means that all combinations of interaction features are equally likely to occur in $M(X)$. This is formally stated as follows:

$$P\left(\bigcap_{Fi \in s} F_i = v_i\right) = P\left(\bigcap_{Fi \in s} F_i = v'_i\right) \text{ for all } v_i, v'_i \in L(F_i) \text{ and } S \subseteq F \tag{1}$$

We now define the positive mode $\mu_i^+ \in L(F_i)$ for feature F_i and user X. The positive mode is the value μ_i^+ which occurs most frequently for feature F_i in $M^+(X)$. Specifically, this means that:

$$P\left(R = 1 \cap F_i = \mu_i^+\right) \geqslant P(R = 1 \cap F_i = a) \text{ for all } a \in L(F_i) \tag{2}$$

Theorem 1. For user X, let interaction $Y = (v_1, v_2, \ldots, v_n)$ be such that v_i is the positive mode for each feature F_i (i.e. $v_i = \mu_i^+$). Then Y maximizes $P(R = 1)$ for user X.

Proof. By the basic axiom of conditional probability:

$$P\left(R = 1 \middle| \bigcap_{F_i \in F} F_i = v_i\right) = \frac{P\left(R = 1 \cap_{F_i \in F} F_i = v_i\right)}{P\left(\cap_{F_i \in F} F_i = v_i\right)}$$

Furthermore, the following holds by (1) for all $S \subseteq F$:

$$\frac{P\left(R = 1 \cap_{F_i \in S} F_i = v_i\right)}{P\left(\cap_{F_i \in S} F_i = v_i\right)} \geqslant \frac{P\left(R = 1 \cap_{F_i \in S} F_i = v'_i\right)}{P\left(\cap_{F_i \in S} F_i = v'_i\right)}$$

$$\equiv P\left(R = 1 \bigcap_{F_i \in S} F_i = v_i\right) \geqslant P\left(R = 1 \bigcap_{F_i \in S} F_i = v'_i\right)$$

Let P_k^* designate the maximum probability $P(R = 1)$ possible given optimal values for features $F_1, F_2, \ldots, F_k$, and assume that:

$$P_k^* = P\left(R = 1 \bigcap_{i=1}^{k} F_i = \mu_i^+\right)$$

By (2) $P\left(R = 1 \cap F = \mu_i^+\right) \geqslant P(R = 1 \cap F_i = a)$ for all $a \in L(F_i)$. Now by the feature carrying assumption we have:

$$P\left(R = 1 \bigcap_{i=1}^{k} F_i = \mu_i^+ \bigcap F_{k+1} = \mu_{k+1}^+\right) \geqslant P\left(R = 1 \bigcap_{i=1}^{k} F_i = \mu_i^+ \bigcap F_{k+1} = v\right)$$

for all $v \in L(F_{k+1})$. It is trivially true by (2) that $P\left(R = 1 \cap F_1 = \mu_1^+\right) \geqslant P(R = 1 \cap F_1 = a)$ for all $a \in L(F_1)$, and it then immediately follows inductively that $P(R = 1)$ is maximized overall features $F_1, F_2, \ldots, F_n$ when $F_i = \mu_i^+$ for each attribute F_i. ■

Theorem 1 shows that when there is a single user class the interaction constructed by taking the positive mode for each feature maximizes the probability of a positive response, assuming that all combinations of interaction features are equally likely to occur in $M(X)$. In the next section we use this result to develop an algorithm for the case of many users.

4.2 The NN algorithm

NN algorithms are one of the oldest machine learning methods, first employed for classification (Cover and Hart, 1967). NN algorithms have a number of desirable properties for machine learning: they make virtually no assumptions about the data on which they are applied, they are intuitively understandable, and they are easy to implement. NN classification works by employing a distance or similarity measure that can be calculated between any two data points. Using a set of labeled observations, each new data point is assigned a predicted label according to the most frequent class among its k-NNs for some fixed k. This approach is readily adapted into a non-parametric regression method for solving problems with numeric response variables by taking the average response of the k-NNs (Altman, 1992). Because of their power and simplicity, NN algorithms have been widely used in diverse applications including cancer classification and gene expression (Parry *et al.*, 2010), text classification (Tan, 2005; Han *et al.*, 2001), predicting chemical properties (Nigsch *et al.*, 2006), remote sensing (Li and Chen, 2009), and others. In collaborative filtering, NN methods have been used successfully in many instances (Su and Khoshgoftaar, 2009).

One of the key assumptions of CF is that users who are similar will respond similarly to the same recommendations, and the widespread success of CF-based recommendation systems gives strong empirical confirmation to this assumption (Jannach *et al.*, 2010). This same principle may be applied to the interaction design problem in that similar users may be expected to respond similarly to a given interaction. The first component of the proposed algorithm is thus a similarity function $S(X_1, X_2)$, where $S(X_1, X_2) > S(X_1, X_3)$ denotes that user X_1 is more similar to X_2 than to X_3. There are many widely used similarity functions for numeric, categorical, and mixed features (Zhang *et al.*, 2014; Boriah *et al.*, 2008) which may be adapted for use in this context.

Under ideal circumstances we would have a large and fully balanced sample of interactions in $M(X)$ with corresponding responses recorded for each user X, and we could then simply build an interaction of positive modes for each user to maximize the likelihood of a positive response. Given that users and interactions may consist of tens or even hundreds of features, however, this assumption is unrealistic. If similar users respond similarly to interactions, however, we may look at users similar to X to infer how X would likely respond to an interaction. This leads to an intuitive strategy for designing an effective interaction for user X: take the positive modes of the users most similar to X.

A common practice in using NN methods for classification is to weight each neighbor's vote based on similarity (Samworth, 2012). The intuition for this is that neighbors nearer to a data point X should count more in voting for the predicted label than those further away. This idea may be incorporated into interaction design by computing the positive modes using a weighted-point system rather than simple frequency. In this case a weighting function may be used so that feature values from more similar users contribute more points than those from less similar users. Thus for feature F_i, the value $v \in L(F_i)$ with the greatest number of weighted points is deemed the weighted-positive mode. A weighting function $W(X_1, X_2)$ may be used to weight any feature value occurrence in $M^+(X_2)$ relative to X_1. Presumably $W(X_1, X_2) > W(X_1, X_3)$ if $S(X_1, X_2) > S(X_1, X_3)$, although W can be tailored to weight more similar users more or less heavily.

The complete interaction design algorithm, shown in Algorithm 1, employs W for finding positive modes based on the k-most similar users.

Algorithm 1. Optimal interaction design for user X given k and data set of users U

```
$NN = k$-NNs of $X$ by similarity function $S$ in $U$
$M^+ =$ all interactions in $NN$ with positive response
FOR $F_i \in F$:
    $\mu_i^+ =$ positive mode for feature $F_i$ taken from interactions in $M^+$ by weighting function $W$
RETURN optimal interaction $Y_X = (\mu_1^+, \mu_2^+, \ldots, \mu_n^+)$.
```

One of the limitations in using NN methods for CF is the sparsity in the user-item matrix, which has resulted in using model-based approaches as an alternative (Su and Khoshgoftaar, 2009). In the interaction design problem, however, user similarities are determined by user features and there is no sparse analog to a user-item matrix. As a consequence, this problem is not generally vulnerable to the sparsity problem which occurs in CF.

5. Computational experiments

In order to test the methods developed we performed simulation experiments which apply the interaction design algorithm to populations composed of individuals with different propensities to respond to different types of interaction designs. In these experiments there are subpopulations of individuals who each have specific propensities to respond positively to different types of interactions. By altering the number of subpopulations present, the specificity of interaction required for a positive response, and the average probability of positive response given an appropriate interaction design, we can gain insights into the increase in positive response rates possible by using this approach. Below we report on the experiment concept, design, and results obtained.

5.1 Simulation concept

In these experiments users are exposed to interactions generated by different strategies including the NN algorithm, and the response rates are compared. Both users and interactions are represented as vectors of categorical features. These experiments model a population of users containing a number of different subpopulations within. Each subpopulation is characterized by a subset of common user features. For users not in any specific subpopulation there is a (relatively low) baseline probability B with which they will respond positively to any interaction. In each subpopulation there is a specific, increased probability for users to respond positively to interaction designs which have a specific subset of common features; otherwise they respond positively with probability B. Each user subpopulation may be thought of as a tuple (U, M, p) where U denotes a specific subset of user features, M denotes a specific subset of interaction features, and p denotes the specific probability that a user with features in U responds positively to an interaction with features in M. Here, U and M may be thought of as templates. For example, if a user is represented as a vector of five features, one subpopulation might be composed of everyone who has features $F_1 = A$ and $F_2 = B$ regardless of what values they have for features F_2, F_3, and F_4. Thus if we have subpopulation (U, M, p) where $U = \{F_1 = A, F_2 = B\}$ and $M = \{F_1 = C, F_4 = D\}$, any user having features $F_1 = A$ and $F_2 = B$ will respond positively with probability p to any interaction with features $F_1 = C$ and $F_4 = D$. It is further possible to have subpopulation (U', M', p') where $U' = \{F_1 = A, F_2 = B, F_3 = P\}$ and $M' = \{F_1 = C, F_4 = D, F_5 = Q\}$. In this case every user matching U' also matches U, and every interaction matching M' also matches M. Under such conditions, the probability of a positive response for user matching U' and interaction matching M' will be p', since this is the closest match. The probability P that any user X responds positively to any interaction Y is thus determined as follows: find the subpopulation (U, M, p) which most closely matches X to U and Y to M. If such a subpopulation exists then $P = p$, otherwise $P = B$.

5.2 Experiment design

In this section we describe the details of the simulation experiments. We begin with a discussion of basic representations and how individual trials were conducted, and then proceed to discuss the experiment design in terms of the experimental variables, levels used, and the number of replications. Both users and interactions consist of vectors of ten features each. During each trial, the number of levels for each feature (in both user and interaction vectors) was randomly set to between two and four inclusive. Users and interactions were generated by simply randomly choosing a level from the set of levels for each feature. Each trial proceeded as follows:

(1) 10,000 users were generated and randomly split into three groups: 5,000 training users, 2,500 calibration users, and 2,500 test users.

(2) 300 interactions were randomly generated.

(3) Each training user X was given an interaction Y randomly chosen from the 300, resulting in response R. The resulting (X, Y, R) tuples were used as NN data points in Algorithm 1.

(4) The calibration users were used to find the best k for each variety of Algorithm 1 and to calibrate the three control strategies (discussed below).

(5) Each test user was given interactions generated by each variety of Algorithm 1 and by each of the control strategies, and the responses recorded.

Two versions of Algorithm 1 were used which employed two different weighting functions for computing the positive modes: $W_1(X, X') = 1$ and $W_2(X, X') = 10^{S(X,X')}$. W_1 weights all values equally and thus produces an unweighted mode, whereas W_2 weights values which are found in more similar users significantly higher. In addition to Algorithm 1, three control strategies were used to produce interactions for the test users: test user is given a randomly selected interaction from the 300; test user is given the interaction out of the 300 which produced the highest overall response rate on the calibration users; and test user is given a randomly selected interaction out of the top 15 with the highest response rates from the calibration users. Control 1 provides a worst-case control, while controls 2 and 3 approximate traditional marketing methods. Thus, there were five methods in total tested in each trial: the two versions of Algorithm 1 and the three control strategies.

The simulation experiments follow a 2^3 factorial design which examines the impact of manipulating three distinct population factors at low (−) and high (+) levels: required interaction specificity (MS); number of subpopulations (NS); and subgroup response propensity (RP). The MS factor determines how many specific features an interaction must have on average for a subpopulation to be more likely to respond positively. Thus, for subpopulation (U, M, p), $|M|$ will be large when this factor is high and small when it is low. Higher levels of MS simulate cases where users are more likely to respond positively only when they receive a highly specific interaction. The NS factor determines how many specific subpopulations are present in the population. Finally, the RP factor determines the probability of positive response of a subpopulation when a matching interaction is encountered. When RP is low, the probability of a positive response is only slightly higher than the baseline when a matching interaction is encountered; it is significantly higher when RP is high. The 2^3 design gives a total of eight distinct scenarios: every possible combination of the three factors at both low and high levels. Each scenario used 20 trial replicates, giving a final total of 160 trials run.

Each trial was randomly generated by a set of parameters, shown in Table I, and the values of certain parameters were determined by the experimental design factors above. In these parameters, B represents the baseline probability of a positive response as discussed above. The Number of Subpopulations Present is determined by the NP design factor. The Number of Target User Attributes per Subpopulation determines the number of specific user feature values required in each subpopulation. Thus, for each subpopulation (U, M, p) this parameter determines the size of $|U|$. The Number of Target Interaction Attributes per Subpopulation determines the number of specific interaction features each subpopulation requires for a positive response probability increase, and is itself determined by the MS design factor. For each subpopulation (U, M, p) this parameter determines the size of $|M|$. Finally, the Subpopulation Positive Response Probability parameter determines the probability of positive response p for each subpopulation (U, M, p). This parameter is determined by the RP design factor.

5.3 *Similarity measure and choice of k*

In the implementations of Algorithm 1 used in these experiments, the Overlap metric was used to determine similarity. Overlap is the count of common features between two users, and this metric was chosen due to its simplicity. In real-world applications, it is likely that domain-specific knowledge about the nature of user data may be incorporated into the similarity function to provide more sophisticated similarity measures. For finding the best value of k in k-NN classification, bootstrap resampling provides an effective approach (Hall *et al.*, 2008). The interaction design problem requires responses to be generated by users, however, making a similar bootstrapping

Parameter	Value or distribution
Baseline probability, B	0.02
Num. subpopulations present	5 (NS−), 20 (NS+)
Num. target user attributes active per subpopulation	Uniform random integer in [2, 5]
Num. target interaction attributes active per subpopulation	Uniform random integer in [1, 3] (MS−), Uniform random integer in [4, 5] (MS+)
Subpopulation positive response probability	Uniform random in [0.1, 0.25] (RP−), Uniform random in [0.5, 0.85] (RP+)

Table I. Trial generation parameters and corresponding values

approach impractical in any real-world application. Instead, a simpler method was employed which could realistically be used within the context of a marketing experiment. Calibration users were each given an interaction generated by Algorithm 1 using k-values ranging from 1 to 15, and the value of k which produced the highest positive response rate was taken.

5.4 Results and discussion

The average positive response rates for each method (controls C1, C2, and C3; NNU = NN with unweighted-positive mode, NNW = NN with weighted-positive mode) are shown in Table II. Table III displays the results of significance testing for the unweighted- and weighted-positive mode NN algorithms. For the significance testing, a two-proportion test was applied comparing positive response rates of each NN algorithm to each control, and also comparing response rates of the weighted-positive mode form against the unweighted form. The null hypothesis in each case is that there is no difference in positive response rates of the two methods compared, while the alternate is that positive response rates from the left-hand-side method are less than those from the right-hand-side. The p-values for each hypothesis test are shown in Table III by alternate hypothesis.

A cursory look at Table II shows that both the weighted and unweighted-positive mode forms of Algorithm 1 produce significantly better positive response rates than any of the controls in all scenarios, with the sole exception of the (+ − −) scenario. This is confirmed in Table III which shows p-values < 0.0001 in these cases. For these cases, the NN algorithms produced average positive response rates ranging from 1.25 to over eight times the average response rates of the controls. The sole exceptional

Table II. Average response rates for each interaction design method by scenario

			Interaction design method				
MS	NS	RP	C1 (%)	C2 (%)	C3 (%)	NNU (%)	NNW (%)
−	−	−	4.08	3.84	4.27	5.19	5.18
−	−	+	9.06	8.45	7.34	20.33	20.20
−	+	−	7.69	7.91	7.74	9.58	9.6
−	+	+	26.64	26.13	27.63	44.37	45.03
+	−	−	2.28	1.96	2.17	2.27	2.17
+	−	+	2.65	3.70	2.58	5.93	6.20
+	+	−	3.64	2.98	3.15	10.43	10.73
+	+	+	3.91	5.64	4.12	12.26	12.63

Table III. p-values for significance testing by scenario

			Alternate hypotheses (H0: resp. rate 1 = resp. rate 2)						
MS	NS	RP	NNU > C1	NNU > C2	NNU > C3	NNW > C1	NNW > C2	NNW > C3	NNW > NNU
−	−	−	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.855
−	−	+	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.938
−	+	−	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.817
−	+	+	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.135
+	−	−	0.851	0.006	0.454	0.984	0.804	0.836	0.981
+	−	+	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.219
+	+	−	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.297
+	+	+	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	0.219

scenario (+ − −) consisted of a high interaction specificity requirement coupled with a low number of subpopulations responsive to specific interaction designs and a trivial increase in positive response probability for each subpopulation given appropriate interactions. When such conditions are present there is little responsiveness to any form of interaction from anywhere within the population and as a consequence, no interaction design strategy will be expected to significantly increase positive response rates. It is also seen that the weighted algorithm form did not significantly outperform the unweighted form in these experiments.

These scenarios reflect a wide range of underlying amounts of responsiveness to specific types of interactions among subgroups of the population. In some scenarios, there were many subgroups which would respond positively to the proper interactions with high probability while in others, there were fewer responsive subgroups and those that existed were only slightly more likely to respond positively to the right interactions. Across scenarios, both forms of the proposed algorithm consistently outperformed the control strategies and were able to generate interactions which resulted in increased response rates. In scenarios with less responsive populations the increases in positive response rates were on the order of 20 to 50 percent, while in scenarios with more responsive populations the increases were on the order of 200 to 400 percent.

6. Conclusions and future work

This work has examined the problem of how to design targeted interactions for individual users to maximize the probability of a positive response within the context of a social media outreach campaign. A NN algorithm for interaction design was proposed, and a series of simulation experiments were conducted to test this algorithm under eight different scenarios with differing population characteristics. Specifically, two forms of the NN algorithm were tested (weighted- and unweighted-positive mode) and compared against three control strategies. In seven out of the eight scenarios each of the NN algorithms significantly outperformed all three controls. In the single scenario where this was not the case there was very little sensitivity within the population to any forms of interactions, and under such circumstances response rates cannot be significantly impacted by interaction design. The weighted-positive mode NN algorithm did not significantly outperform the unweighted version. In summary, the NN algorithms were highly effective in increasing response rates across populations with varying characteristics.

One of the limitations of this study lies in its sole use of simulation experiments, due to the unavailability of real data. Moreover, even if real data were available there are significant dissimilarities between interaction design vs prediction or collaborative filtering problems which make it impractical to test outside the context of a live scenario. In particular, it is impossible to know how any specific user would have responded to any different interactions other than the ones they received. By contrast, in prediction or collaborative filtering problems there are response variables or rated items which can be compared against predictions or recommendations. Consequently, one direction of future research is to test this method in the context of a live outreach campaign. As another potential future direction, in some cases there may be a maximum number of interactions which may be designed out of a much larger number possible. The challenge in this case is to determine both the optimal subset of interactions to utilize as well as which type of interaction from this subset should go to each user to maximize the positive response rate.

References

Altman, N.S. (1992), "An introduction to kernel and nearest-neighbor nonparametric regression", *The American Statistician*, Vol. 46 No. 3, pp. 175-185.

Arceneaux, K. and Nickerson, D. (2010), "Comparing negative and positive campaign messages: evidence from two field experiments", *American Politics Research*, Vol. 38 No. 1, pp. 54-83.

Armstrong, G. and Kotler, P. (2014), *Marketing: An Introduction*, 12th ed., Prentice-Hall, Upper Saddle River, NJ.

Booth, N. and Matic, J.A. (2011), "Mapping and leveraging influencers in social media to shape corporate brand perceptions", *Corporate Communications: An International Journal*, Vol. 16 No. 3, pp. 184-191.

Boriah, S., Chandola, V. and Kumar, V. (2008), "Similarity measures for categorical data: a comparative evaluation", *Proceedings of the Eighth SIAM International Conference on Data Mining*, *Atlanta, GA*, pp. 243-254.

Cover, T. and Hart, P. (1967), "Nearest neighbor pattern classification", *IEEE Transactions on Information Theory*, Vol. 13 No. 1, pp. 21-27, doi: 10.1109/TIT.1967.1053964.

Chee, S.H.S., Han, J. and Wang, K. (2001), "RecTree: an efficient collaborative filtering method", *Proceedings of the Third Annual Conference on Data Warehousing and Knowledge Discovery, Munich, September 5-7*, pp. 141-151.

DeCoste, D. (2006), "Collaborative prediction using ensembles of maximum margin matrix factorizations", *Proceedings of the 23rd International Conference on Machine Learning (ICML'06)*, *Pittsburgh, PA*, pp. 249-256.

Goldberg, K., Roeder, T., Gupta, D. and Perkins, C. (2001), "Eigentaste: a constant time collaborative filtering algorithm", *Information Retrieval*, Vol. 4 No. 2, pp. 133-151.

Greinemr, R., Su, X., Shen, B. and Zhou, W. (2005), "Structural extension to logistic regression: discriminative parameter learning of belief net classifiers", *Machine Learning*, Vol. 59 No. 3, pp. 297-322.

Hall, P., Park, B.U. and Samworth, R.J. (2008), "Choice of neighbor order in nearest-neighbor classification", *Annals of Statistics*, Vol. 36 No. 5, pp. 2135-2152, doi: 10.1214/07-AOS537.

Han, E.H., Karypis, G. and Kumar, V. (2001), "Text categorization using weight adjusted k-nearest neighbor classification", *Proceedings of 5th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, Hong Kong, April 16-18*, pp. 53-65.

Harfoush, R. (2009), *Yes We Did! An Inside Look at how Social Media Built the Obama Brand*, New Riders, Berkeley, CA.

Herlocker, J.L., Konstan, J.A., Terveen, T.L. and Riedl, J.T. (2004), "Evaluating collaborative filtering recommender systems", *ACM Transactions on Information Systems*, Vol. 22 No. 1, pp. 5-53.

Hofmann, T. (2004), "Latent semantic models for collaborative filtering", *ACM Transactions on Information Systems*, Vol. 22 No. 1, pp. 89-115.

Huang, Z., Chen, H. and Zeng, D. (2004), "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering", *ACM Transactions on Information Systems*, Vol. 22 No. 1, pp. 116-142.

Issenberg, S. (2012), *The Victory Lab: The Secret Science of Winning Campaigns*, Crown Publishers, New York, NY.

Jannach, D., Zanker, M., Felfering, A. and Friedrich, G. (2010), *An Introduction to Recommender Systems*, Cambridge University Press, New York, NY.

Li, Y. and Chen, B. (2009), "An improved k-nearest neighbor algorithm and its application to high resolution remote sensing image classification", *Proceedings of the 17th International Conference on Geoinformatics, IEEE, Fairfax, VA*, pp. 1-4, doi: 10.1109/GEOINFORMATICS.2009.5293389.

Linden, G., Smith, B. and York, J. (2003), "Amazon.com recommendations: item-to-item collaborative filtering", *IEEE Internet Computing*, Vol. 7 No. 1, pp. 76-80, doi: 10.1109/MIC.2003.1167344.

McLaughlin, M.R. and Herlocker, J.L. (2004), "A collaborative filtering algorithm and evaluation metric that accurately model the user experience", *Proceedings of 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)*, *Sheffield, July 25-29*, pp. 329-336.

Melville, P., Mooney, R.J. and Nagarajan, R. (2002), "Content boosted collaborative filtering for improved recommendations", *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI'02)*, *Edmonton*, pp. 187-192.

Nickerson, D. and Rogers, T. (2010), "Do you have a voting plan? Intentions, voter turnout, and organic plan making", *Psychological Science*, Vol. 21 No. 2, pp. 194-199.

Nickerson, D.W. and Rogers, T. (2014), "Political campaigns and big data", *Journal of Economic Perspectives*, Vol. 28 No. 2, pp. 51-74.

Nigsch, F., Bender, A., van Buuren, B., Tissen, J., Nigsch, E. and Mitchell, J.B.O. (2006), "Melting point prediction employing k-nearest neighbor algorithms and genetic parameter optimization", *Journal of Chemical Information and Modeling*, Vol. 46 No. 6, pp. 2412-2422, doi: 10.1021/ci060149f.

Noh, H., Kwak, M. and Han, I. (2004), "Improving the prediction performance of customer behavior through multiple imputation", *Intelligent Data Analysis*, Vol. 8 No. 6, pp. 563-577.

Parry, R.M., Jones, W., Stokes, T.H., Phan, J.H., Moffitt, R.A., Fang, H., Shi, L., Oberthuer, A., Fischer, M., Tong, W. and Wang, M.D. (2010), "k-Nearest neighbor models for microarray gene expression analysis and clinical outcome prediction", *The Pharmacogenomics Journal*, Vol. 10 No. 4, pp. 292-309, doi: 10.1038/tpj.2010.56.

Pennock, D.M., Horvitz, E., Lawrence, S. and Giles, C.L. (2000), "Collaborative filtering by personality diagnosis: a hybrid memory- and model-based approach", *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI '00)*, pp. 473-480.

Rongfei, J., Maozhong, J. and Chao, L. (2010), "A new clustering method for collaborative filtering", *Proceedings of the 2010 International. Conference on Networking and Information Technology*, *Manila*, pp. 488-492.

Samworth, R.J. (2012), "Optimal weighted nearest neighbour classifiers", *Annals of Statistics*, Vol. 40 No. 5, pp. 2733-2763, doi: 10.1214/12-AOS1049.

Sarwar, B.M., Karypis, G., Konstan, J.A. and Riedl, J. (2001), "Item based collaborative filtering recommendation algorithms", *Proceedings of the 10th International Conference on World Wide Web (WWW'01)*, *Hong Kong*, pp. 285-295.

Shani, G., Heckerman, D. and Brafman, R.I. (2005), "An MDP-based recommender system", *Journal of Machine Learning Research*, Vol. 6 No. 1, pp. 1265-1295.

Si, L. and Jin, R. (2003), "Flexible mixture model for collaborative filtering", *Proceedings of the 20th International Conference on Machine Learning (ICML'03)*, *Vol. 2*, *Washington, DC*, pp. 704-711.

Srebro, N., Rennie, J.D.M. and Jaakkola, T. (2005), "Maximum margin matrix factorization", *Advances in Neural Information Processing Systems*, Vol. 17, pp. 1329-1336.

Su, X. and Khoshgoftaar, T.M. (2006), "Collaborative filtering for multi-class data using belief nets algorithms", *Proceedings of the International Conference on Tools with Artificial Intelligence (ICTAI'06)*, pp. 497-504.

Su, X. and Khoshgoftaar, T.M. (2009), "A survey of collaborative filtering techniques", *Advances in Artificial Intelligence*, Vol. 2009 No. 1, pp. 1-19, doi: 10.1155/2009/421425.

Su, X., Khoshgoftaar, T.M. and Greiner, R. (2008), "A mixture imputation-boosted collaborative filter", *Proceedings of the 21th International Florida Artificial Intelligence Research Society Conference (FLAIRS, 08)*, *Coconut Grove, FL*, pp. 312-317.

Tan, S. (2005), "Neighbor-weighted K-nearest neighbor for unbalanced text corpus", *Expert Systems with Applications*, Vol. 28 No. 4, pp. 667-671, doi: 10.1016/j.eswa.2004.12.023.

Zhang, S., Zong, M., Sun, K., Liu, Y. and Cheng, D. (2014), "Efficient kNN algorithm based on sparse graph reconstruction", *Proceedings of 10th International Conference on Advanced Data Mining and Applications, Guilin, December 19-21*, pp. 356-369.

Further reading

Ansari, A., Essegaier, S. and Kohli, R. (2000), "Internet recommendation systems", *Journal of Marketing Research*, Vol. 37 No. 3, pp. 363-375.

Sarwar, B.M., Karypis, G., Konstan, J.A. and Riedl, J. (2000), "Analysis of recommendation algorithms for E-commerce", *Proceedings of the ACM E-Commerce*, *Minneapolis, MN*, pp. 158-167.

Yang, W.S., Dia, J.B., Cheng, H.C. and Lin, H.Z. (2006), "Mining social networks for targeted advertising", *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*, *Kauia, HI*, p. 137a, doi: 10.1109/HICSS.2006.272.

Corresponding author

Christopher Garcia can be contacted at: cgarcia@umw.edu