Automated analysis and planning of social network marketing

Master's Thesis in Software Engineering

Erik Brännström
Chalmers University of Technology
erikbr@student.chalmers.se

ABSTRACT

Managing online marketing campaigns is a repetitive and analytical process which is typically done manually by domain experts. This paper deals with the problem of how to use software to manage historical marketing data and use that as a foundation for decision-making with the purpose of optimizing future ad performance.

The solution presented in this thesis involves applying both data mining and automation practices to provide decision makers with knowledge that can be enacted upon. Operators of the system input the historical data, upon which the system creates an estimation model based on said data which is used to estimate the performance of, for the system, previously unseen ads. These suggested ads can either be input by the operator or automatically created by combining existing ad properties.

The solution has been validated on real-world data from an industrial partner in the social networking industry.

Keywords

Online marketing, business intelligence, performance estimation, social networks

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning; J.1 [**Administrative Data Processing**]: Marketing

1. INTRODUCTION

As more and more people are using the Internet on a daily basis, the area of online marketing is expanding as a way for organizations to reach large audiences at a relatively low cost. The process of managing the marketing material however requires a lot of manual labor, since results must be properly analyzed and applied to future decisions.

The research in this paper was carried out in association with an industrial partner that relies heavily on online marketing to promote their product to new users. The partner is a social networking company based in Sweden which will be referred to by the name of Company A. The purpose of partnering was to evaluate the prospect of automating parts of the marketing process in a real-world environment and show the validity of the presented solution.

To be able to properly assimilate the contents, the first subsection presents the problem domain and defines useful terminology which is used throughout this paper. These descriptions are relied on in the subsequent two sections, which first details the addressed problem followed by an overview of the solution.

The final section deals with the scope by stating restrictions on what will be covered by this paper.

1.1 Domain

This paper exist at the intersection of a number of different areas. As marketing is probably the field furthest from typical software engineering tasks, it will be described in some detail in the first subsection. That will then lead into the concept of business intelligence, followed by a subsection on automation

1.1.1 Online marketing

A site that displays advertisement, the publisher, is often paid by the advertiser based on the number of times visitors see or click on the ads, but it may also be coupled with other requirements, such as that the visitor goes on to buy a product from the advertiser in a given time span. The advertisement is generally tailored to the expected interests of the visiting users. This is because advertisers want to keep the costs down while still getting good results and distributors usually prefer to show only relevant information to their visitors to keep them coming back.

A domain description is shown in Figure 1. A *marketing campaign*, or simply a *campaign*, is comprised of one or more advertising messages, *ads*, that are directed to one defined audience, the *target* or *target group* (e.g. gender or people searching for certain keywords). Any ad, and by extension campaign, have numeric measures of success called *metrics*. The most commonly used are *impressions*, which is the number of times an ad has been shown, and clicks.

When referring to a user interaction, the word *action* will sometimes be used. Even though actions in the case of Company A is the same as clicks, one might be interested in other values than simple clicks such as for example the number of users who go on to register at the site after clicking. Ads have a number of properties, which depend on the media that is used. For example, textual ads in search engines typically have a title, a short text and a URL to which the user is redirected upon clicking the ad, whereas an ad on a site such as Facebook can also include an image.

The targeting properties available also depend on the publisher. For the purposes of this thesis, four different classes of online advertising are identified based on the way the ads are targeted. *Search* advertising uses the user's search terms, *social* advertising uses demographic and personal data (e.g. age, gender, location or interests), *contextual* advertising finds keywords on the page on which the ads are displayed

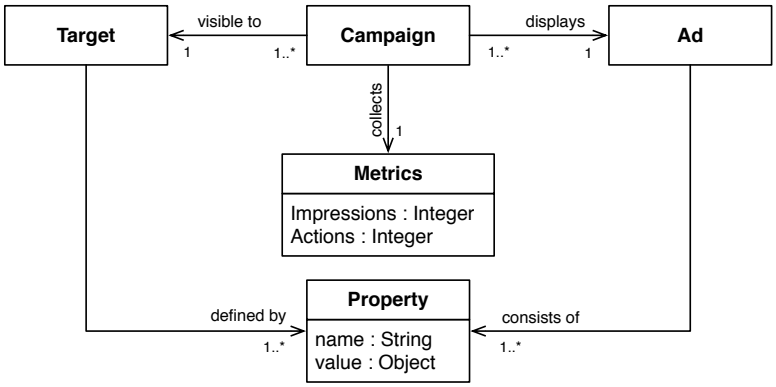

Figure 1: UML description of marketing terminology.

or uses manual categorization and finally *non-contextual* advertising, which does no relevancy matching. The separation between these classes is not necessarily distinct and a single publisher can use targeting criteria from different classes. This paper focuses on social advertising and its use by organizations such as Company A.

1.1.2 Business intelligence

Marketing, perhaps especially online, will typically generate large amounts of data. These data sets are then used as a foundation for future decision making. Business intelligence is a rather loosely defined term, but it is highly relevant to this concept of using past data to future decisions.

[12] defines business intelligence systems as those that "combine data gathering, data storage, and knowledge management with analytical tools to present complex internal and competitive information to planners and decision makers", whereas [6] uses the broader definition of "the process of turning data into information and then into knowledge". Even though neither definition explicitly mention the use of computers and software for processing the data, it is an integral part of modern business intelligence systems.

The data used in these types of system is generally classified as either structured or semi-structured, as described by [12]. Though the difference may not always be clear cut, structured data is the type of data which typically resides in databases and custom relationship management (CRM) applications so that it can easily be searched, updated, aggregated, etcetera. Semi-structured data on the other hand is that which cannot be parsed as easily by software, such as e-mails, movies, reports and phone conversations. The author also describes that data can be further categorized based on its source; either internal or external. This framework helps define the data type for which this thesis is relevant, namely internal structured data.

Business intelligence includes a large number of software engineering related areas. For example, the task of identifying useful knowledge from data sets is a field known as knowledge discovery, which in turn often utilizes data mining.

To increase the usability and efficiency of a business intelligence system, it should be able to reduce manual operation to a minimum, which requires some form of automation, either fully or partially.

1.1.3 Automation

[8] defines a framework for automation in the field of autonomic computing. Figure 2 shows an autonomic manager, which is a component that collects data from a system and, based on this data, performs actions with the purpose of improving the system. This control loop is divided into four subtasks called monitor (collect system information), analyze (model data), plan (design behavior required to reach goal) and execute (run the planned actions), sometimes referred to as MAPE. Each subtask can optionally interact with a knowledge base for storing and retrieving data.

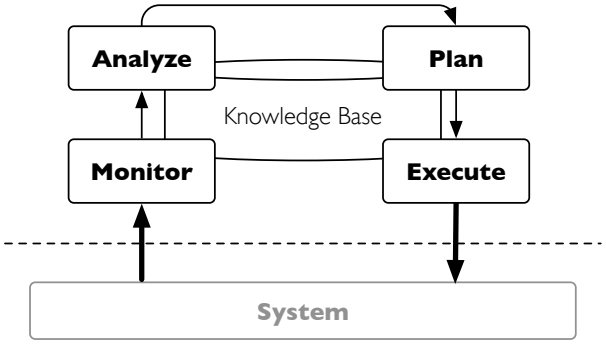

Figure 2: General MAPE control loop

In order for the system to analyze and interact with its environment there is a need for some type of input and output.The input of environmental information are called *sensors* and the output to enact the planned actions are carried out using *effectors*. For the framework to be useful it is not necessary however that these parts, nor the MAPE tasks, are computerized. It is possible to have human interaction integrated into the system description.

1.2 Problem

The current workflow for managing online marketing campaigns starts with the creation of advertising material by the advertiser. Once the ads have been run, their impact is analyzed and based on these results, material can be created, adapted or removed from circulation to better suit the goals of the organization. For example, ads with a low click rate

need to either be removed completely or modified in some way in order to increase their efficiency, whereas an ad that performs very well most likely is left as is and used as inspiration for new ads.

There is a high cost for this repetitive manual labor in terms of both time and money, since the data must be analyzed over and over again. This may not be a problem for small sets of data, but it becomes increasingly hard to maintain as the number of campaigns and ad properties grow over time. Company A, for example, has hundreds of campaigns running at once and the amount of historical data grows quickly which means that managing this data becomes cumbersome.

The problem is thus one of managing historical advertisement data and applying that data to future decisions in order to optimize the future overall click rate of the organization's ads. In business intelligence terms, the goal is therefore to apply an analytical tool to the stored data in order to provide decision makers with knowledge that can be enacted upon.

1.3 Solution

The basic requirement of a system which could solve the previously stated problem is that it should be able to analyze existing campaigns and their metrics and use this as a basis for suggesting new ads to the operator[1] based on the estimated performance.

The proposed solution is to apply the principles of automation described in the MAPE framework to the context of marketing. Monitoring is thus the collection of metrics for online advertisement. To optimize these metrics, the gathered information is analyzed and this analysis forms the basis for the planning of future campaigns to run. Once the plans are completed, they are presented to the operator who can execute them, after which new metrics are gathered and so on.

It is already common for monitoring to be automated, either using custom software or services such as Google Analytics, whereas the other parts of the process are performed manually. It is infeasible to fully automate the whole process, due to for example the creative side of advertisement including creating new photos and writing new texts. There are however certain areas that can be automated and this paper will focus on automated analysis and planning of online marketing campaigns, shown in Figure 3.

A system to automate this process requires monitored data as input, which in this context equals historical data of campaigns and their metrics as mentioned previously. This data is then analyzed and modeled so that an estimated action rate can be estimated based on the properties of the ad. Because Company A only targets their ads based on gender, between which there are significant differences in how the ads are designed, targeting will be handled by dividing the full data set into subsets based on the target and then applying the solution to each such subset independently.

Based on this model, the system will generate suggestions for new campaigns as well as recommend actions to be taken by the operator to optimize the overall average action rate of existing ads.

[1]As the word *user* is typically used to refer to a person interacting with either the web site that is being marketed or the web site on which the advertisement is shown, we will use the word *operator* when discussing a person interacting with the system that is described in this paper to avoid confusion.

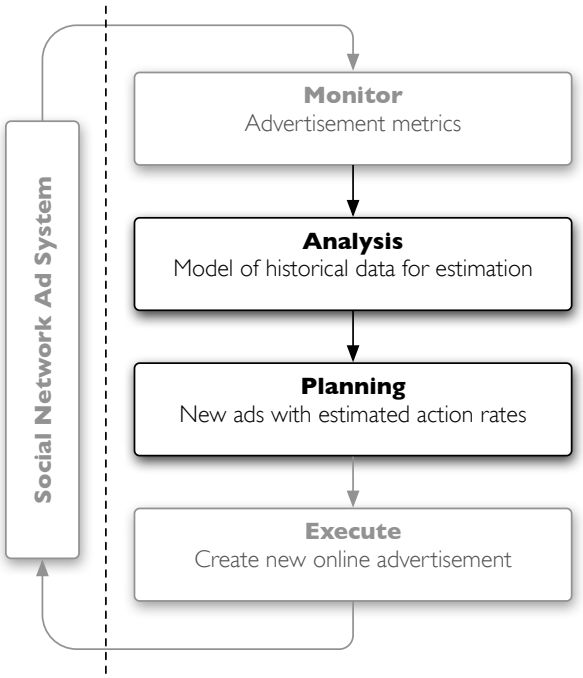

Figure 3: Online marketing automation system in control loop

1.4 Scope

Using the MAPE framework in Figure 2, only the analysis and planning tasks are considered part of this paper, whereas monitoring and execution are out of scope. The latter two are of course relevant in the implementation of the system, but neither will be covered as a research topic. In this context, this means for example that the feature of integrating this system with marketing services to automatically add new ads and campaigns will not be a part of the final system.

Furthermore, the data set will include attributes whose values are free text and images, however text mining and image recognition are beyond the scope of this project.

The model of the marketing domain purposefully excluded references to costs and budgets. Though the economy of marketing is of great interest to the advertiser, we have assumed that the most important part of the process is to optimize the number of actions. This assumption was approved by our industrial partner.

On the topic of costs and budgets, it is also necessary to mention that due to the costs of running advertisement, there was a limitation on the types of experiments we could perform. Thus large scale live testing of the system is not included in the scope of this thesis.

Finally, targeting will only be covered briefly and for the specific case of Company A. A general solution for managing different types of targeting and how they influence estimations is thus not a part of this thesis.

1.5 Thesis outline

This section has provided an overview of the domain, a description of the problem as well as a brief outline of the suggested solution. It has also defined the scope of this thesis.

Section 2 will cover research foundations for this paper, mostly in the area of knowledge discovery and data mining. This is followed by section 3 which summarizes research efforts related to the field of online marketing.

Section 4 describes the method used to develop the solution, the research questions that defined the focus of our research as well as an explanation of the data used to experiment on.

Section 5 then answers the research questions and the solution is developed and explained based on those results.

Finally, section 6 gives a summary of the results presented in this paper, discussion on risks and constraints as well as suggestions for future work.

2. FOUNDATIONS

The problem as well as solution described in the introduction show there is a need to have a software system which can parse and analyze large sets of data. To facilitate this, a number of high-level descriptions of frameworks for knowledge discovery in databases exist [2, 3] and they exhibit a number of commonalities. These include the importance of having a knowledgeable human operator guiding the process in terms of supplying domain knowledge to the system formulating the goal of the knowledge discovery; feeding discovered knowledge back into the system; and the identification and application of a discovery method, or more specifically the data mining algorithms. Because the solution is a knowledge discovery system, it is important that it too has these same characteristics.

In data mining, the input to a system can be described using the terms *concepts*, *instances* and *attributes*, where concept is the actual result of the mining, i.e. what we want to be learned; an instance is one single example of data to be mined and can be compared to a row in a database; and attribute is a property of an instance, which in the database analogy is a column [18]. This analogy is useful since it is directly applicable to the structured internal data the system uses as input, as defined in the introduction.

[10] gives a top-down explanation of data mining. The learning methods can be divided into two classes; supervised learning and unsupervised learning. In the first case, the learning is based on existing instances and the known values of a dependent variable for each such instance, whereas unsupervised learning has no knowledge of such a dependent variable and thus is only concerned with identifying structure in the input data. The structured input data will have such a dependent variable, or at least it can be calculated from existing variables (e.g. the number of clicks divided by the number of impressions gives the average click rate of a campaign). This means that we are dealing with supervised learning.

Within supervised learning there are a number of different types of learning tasks, and [10] goes on to describe classification as the most common such task. The purpose of a classification algorithm is to categorize an instance into a predefined class, based on existing instances that are already classified. This is not directly applicable to the marketing data, but [18] describe a special case of classifiers where the outcome is a real number instead of a class. Such classification is referred to as numeric prediction or numeric classification, and this technique is evaluated for use in estimation of action rates for new campaigns. In the case of marketing, another possibility is to use classifiers which provide probabilities of class association as well. By looking at each impression as a trial, each ad has a probability of belonging to the class of being clicked.

3. RELATED WORK

There have been some research in methods for estimating the performance of online advertisement, though no such research seems to exist for social network advertisement specifically. We begin by looking at related work for ad performance estimation in search advertisement, followed by techniques for improving, though not estimating, performance. Finally we describe research concerning how users experience ads and potential problems caused by fake traffic.

[14] describe a model for predicting the click-through rates of ads given a set of properties in the context of search engine advertisement. For the estimation, the authors use logistic regression on a large number of ad features (e.g. number of characters in text and segments in URL). This approach is chosen in the paper because logistic regression provides a probability for class association, and a click probability is the goal. There is however no further discussion or experimentation to elicit why this choice is more appropriate than for example numeric classifiers. The results are nevertheless promising in regard to the predictive powers of logistic regression and as such it will be evaluated as a potential estimation model in this paper.

Another paper on predicting click-through rate (CTR) is [13], where a cluster analysis approach is used. In order to estimate the CTR of a search term, a number of related search terms with known performance are identified and used as a basis for the prediction. Though the method performs well, it is tightly bound to the field of search engine advertisement. The same can be said for [14], which for example utilizes correlation between search terms to improve estimates. Because the concept of search terms does not exist in social networking advertisement, neither approach is directly applicable.

Automation in marketing is not a completely novel idea. [9] present a technique for automatically finding related keywords for broader targeting of ads by creating a graph of search terms based on the domain knowledge contained in search engines. Continuation of that work is that of [17], where keywords are also extracted from the landing page to further improve the results. Both approaches are however only relevant for contextual and search engine advertisement as they use data exclusive to the respective advertisement class.

Marketing in the context of social media has also gained interest in the academic community. [19] propose a technique for identifying subgroups of users in social networks based on their interactions and then use that information to create targeted advertising. The results could very well be applied in parallel to the results of this thesis in order to better target ads, however it does not deal with ad performance in any way.

Research has also shown that using targeted advertisement is effective given that the ad is not experienced as being obtrusive, though obtrusive ads that are not targeted (i.e. non-contextual) also increase performance [5]. Another

study presented results that suggest that users on social networks generally tend to not consciously take notice of advertisement and, perhaps of this meaning that ads are not considered to be obtrusive, they do not have any negative feelings toward the existence of advertising [7]. By automating the process of estimating ad performance, these properties for ad management are expected to be implicitly handled by the model, since properties that make users experience the ad as obtrusive or unnoticeable will perform worse and thus those results will impact future estimates.

In online advertisement there have been problems of so called click-spam or click-fraud, meaning that the number of clicks on ads is increased in a fraudulent manner, for example using bots. [1] provides a methodology to measure click-spam in their networks as well as a study of the severity of the problem on different classes of networks. Their results show that established search advertising on for example Google and Bing is fairly accurate in their filtering of click-spam, whereas the problem is greater for contextual and social advertisement, and severe for mobile advertisement. A related paper by [20] describes a methodology for advertisers to evaluate the quality of click traffic and use this to assess the difference in quality between bulk traffic vendors and established pay-per-click networks. It may be of importance to be aware of these issues as they could carry some impact on the research, however managing traffic quality and click-spam is not in the scope of this paper.

In conclusion, though predicting performance of ads has seen some limited research, the area of social networking advertisement is largely unexplored, especially in terms of performance prediction.

4. METHOD

This section describes the process that was used to identify a solution to the problem described in the introduction. The first step of the process was a literature review to identify existing research in this area and examine the foundations. Simultaneously, meetings were held with marketing experts from Company A to define the domain in which to apply the results of the project.

The next step was obtaining a sample of real world marketing data for analysis and experimentation. This process began with manual analysis and then formalized experiments to confirm the idea that the data could be modeled in order to provide better-than-random estimates. Once that had been established, the appropriate choice of classifier was evaluated.

Once the modeling and estimation part of the research was concluded, the question of planning was approached. This dealt with defining how an operator should select ads from the system's suggestions and analyzing if ad performance declines over time in order to suggest when an existing ad has run its course.

The problems described above were formulated as research questions that are covered in more detail in section 4.1.

The last subsection in this section describes the sample data provided by Company A to help the reader understand the structure of a typical data set.

4.1 Research questions

In formulating the research questions, the definitions provided by [16] were used in order to provide clear goals for what the thesis should provide.

RQ 1. *How to automate the creation and estimation of good ads in online marketing?* The main research question for this thesis is the identification of a method of development that can be used in organizations to support decisions in online marketing. The solution presented is a procedure that is applicable to online advertisers.

Because this question is very broad, the validation step consisted of dividing the question into four sub-questions and validating them each in turn.

RQ 1.1. *Do ad properties have differing impact on performance?* It would be reasonable to assume that different properties of an ad may have different impact on the user it is shown to in terms of likelihood of him or her taking an action, however the size of those differences are harder to hypothesize about. This is an important first question in order to get an indication of how, if at all, properties relate to the click rate.

To answer the question, a controlled experiment was defined were ad properties were carefully selected by domain experts and then tested in a real world environment. From the resulting metrics, the correlation between properties and performance was calculated and analyzed.

RQ 1.2. *How well do classifiers perform in estimating click rates?* The first step is to identify whether or not a numerical classifier can, based on historical data, adequately estimate the action rate of previously unseen ads. It is thus a matter of evaluating the performance of existing algorithms.

To provide an answer to the question an experiment was defined, where the success rate of a regression model was compared to those of random values from statistical distributions, one normally distributed and one uniform distribution with the same minimum and maximum values as the data set, as well as comparing success rates of different classifiers to each other. The results answer both how well classifiers perform in relation to simpler models, and if they performs better, which classifier is most suited for the task.

RQ 1.3. *Which ad selection strategy provides the highest average click rate?* It is unlikely that a classifier would correctly estimate future ad performance every time. To potentially minimize the impact of bad estimations, multiple suggested ads could be selected by the operator and run together in order to improve the average action rate.

To evaluate this six different strategies were tested, where one, two and three suggested ads were selected based on their estimated performance after which the average real performance both with and without weighting based on the estimates were calculated. The results were then analyzed to identify the appropriate strategy to use.

RQ 1.4. *How does time affect the performance of ads?* A marketing executive suggested that the performance of ads declines over time. To evaluate the correctness of this statement, data was gathered over a period of two weeks where the same ads were run continuously and then analyzed. Depending on the answer, the final solution would possibly have to account for such a decline in estimating the future performance of existing ads.

4.2 Data set

The main data set used for analysis and experiments was provided by Company A and consisted of ads run in the Polish market between July 17^{th} and August 17^{th} 2012. The set contained 959 instances of ads run on Facebook in the mentioned time interval. All ads had however not been run

for the whole time.

There were a total of eight (8) unique titles, but six of those where almost equal with only difference being the city they referred to. The body texts had three (3) distinct values, and by counting the titles referring to a city as a single title, every title was used exclusively together with a single body text so that only three possible title/body combinations existed. A total of eleven (11) unique images were used, out of which five (5) were used when targeting both genders and each gender having three (3) exclusive images. Figure 4 shows the layout used by Facebook, with the title above the image and the body text to the right of the image. This layout is fixed and cannot be modified by the operator, who can only set the individual properties for the ad (and of course also targeting properties).

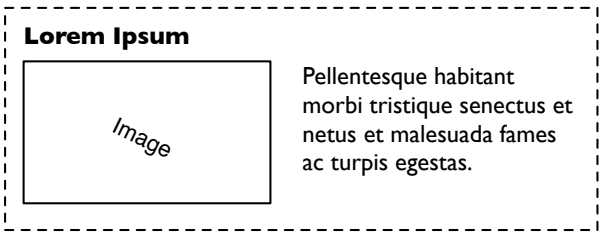

Figure 4: Layout for ad presentation

The average number of impressions for each ad in the data set was 143875.17 and the average number of clicks was 32.96, both values with two decimals of precision. By dividing the clicks with the impressions we get an average click-through rate of about 0.0229%.

Over the course of this project, Company A made some changes to their marketing strategy. The most noticeable difference was the removal of distinct age groups for targeting, which left gender as the only targeting property. To still be able to use the data set, the metrics of campaigns that only differed in the targeted age group were aggregated. This aggregation reduced the number of unique instances to 64. Other changes to the strategy included using the same title on all ads and changing the regions in which marketing took place, however the data set was not adapted to reflect these changes since those changes could not be simulated using for example aggregation.

An early idea on how to modify the data to provide more insights for the operator was to apply tags to the images and texts. Tags are a set of words that describe an object, so that similarities can be discerned more easily by a computer system. There were however two main problems with this method. First of all, there is the question of subjectivity. This is especially a problem for pictures, whose information content is highly unstructured making the set of possible attributes that are important hard to define. As an example, imagine an ad with a picture of a girl sitting in a coffee shop. A user's likelihood of clicking the ad may either increase or decrease depending on the looks of the girl, the lighting in the room, what beverage the girl is drinking, whether there are other people around or not, etcetera. The subjectivity involved (who decides whether a person is good-looking or not?) and the nearly unlimited possible attributes were the main reasons for ruling out image tagging.

For texts it is possible to remove a lot of the problems with image tagging by only focusing on the actual words that appear. This does however lead to a new problem, which would also have affected image tagging if it was applied despite the concerns raised. Consider the case of two high performing ads, one with the tags "beach" and "volleyball", the other one with tags "snow" and "skiing". Because a classifier is not aware of the context, it would likely estimate that an ad focusing on for example "beach" and "snow" would have a very high click rate, though this is unlikely. Also considering that tagging could easily introduce thousands or millions of possible combinations to be estimated, this would mean the operator would need to do a lot of extra work in ruling out these misguided suggestions and the whole purpose of the system would be defeated.

A visual analysis of a graph of the click rates hinted of a normal distribution. To judge whether this was the case, the extended Shapiro-Wilks normality test [15] as implemented in R[2] was used. Ads that had not received any clicks were considered to be outliers and removed from the sampling to not skew the results. This turned out to only affect a single ad.

A significance level, α, of 0.05 was chosen for the null hypothesis H_0, which would mean the data is normal. The test returned the test statistic $W = 0.9654$ and a p-value of 0.07339. Since $0.07339 > \alpha$, we cannot reject H_0 and therefore we assume a normal distribution of the data set.

When no specification is made in regard to which data set was used for a specific experiment, it is implied that the results are based on the part of the aggregated data set which targeted women. Though all experiments have been run on both subsets, we have chosen to only give the results of one set as long as the conclusions of the results are the same.

5. SOLUTION

The proposed solution to the problem is to automate the creation of good ads for online marketing using a software system. Figure 5 gives an overview of the architecture for such a system. The main components include a data manager for interacting with the knowledge base, an estimator that based on historical data can estimate the performance of new ads and an ad factory that can provide new ads with unknown performance to, for example, an estimator.

In order to answer the research questions, the estimator, which is the central part of the system, must be evaluated. By having an evaluator which controls the interfaces to and from the estimator as in Figure 6, we can create a number of different evaluators to test different aspects of the estimation.

Because the evaluator controls the input to the evaluator component, ads whose performance is known can be input to the application and its output compared to the actual values. This comparison is useful for gaining insight into how well the estimates reflect reality. More specifically, by repeating such an experiment many times but change the data set used, we get a reliable indicator of how well an estimator performs, namely the percentage of trials for which the estimate was reasonably close to the actual value. This of course requires a definition of what reasonably close means, something which will be covered in section 5.4. These indicator values can also be compared for different estimators to

[2]http://www.r-project.org/

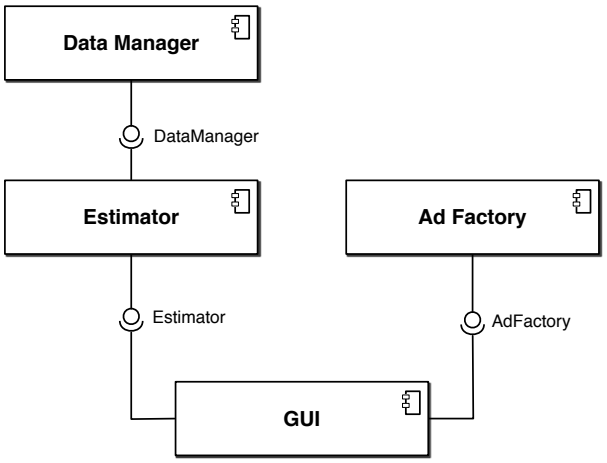

Figure 5: Component diagram of software solution

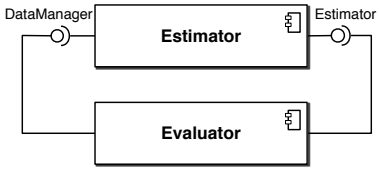

Figure 6: Component diagram of system evaluation

identify which is the most accurate one, which is also done in the mentioned section.

It is important to know that the developed system takes advantage of the Weka library [4] by using its classes for modeling the data sets and its implementations of classifiers. It also uses the Colt library[3] for dealing with statistical distributions.

The Data package defines the main classes that model the data. The main purpose of *Ads* is to manage a list of single *Ad* objects that all store the same type of information. It is data from these classes that is used by the logic in the Core package.

The main interface in Core is the *Estimator*, whose purpose it is to predict the performance of an ad whose performance is not known. The different realizations of this interface represents different methods of prediction, with the *NumericEstimator* utilizing numeric classifiers, the *NominalEstimator* returns the probability that the ad belongs to the class of clicked ads, and the *DistributionEstimator* uses statistical distributions to randomly assign a prediction.

An AdFactory is basically a provider of ads which do not have an estimated action rate. The source of these ads can be for example combinations of ad properties for which no metrics exist, or suggestions input by an operator. The realization of this interface is closely tied to the environment in which the system will exist.

The final package is Evaluation, whose only interface is *Evaluator*. The requirements of its implementations are to provide a textual description of what it does and of its results, which were used in the validation of this system. The

[3]http://acs.lbl.gov/software/colt/

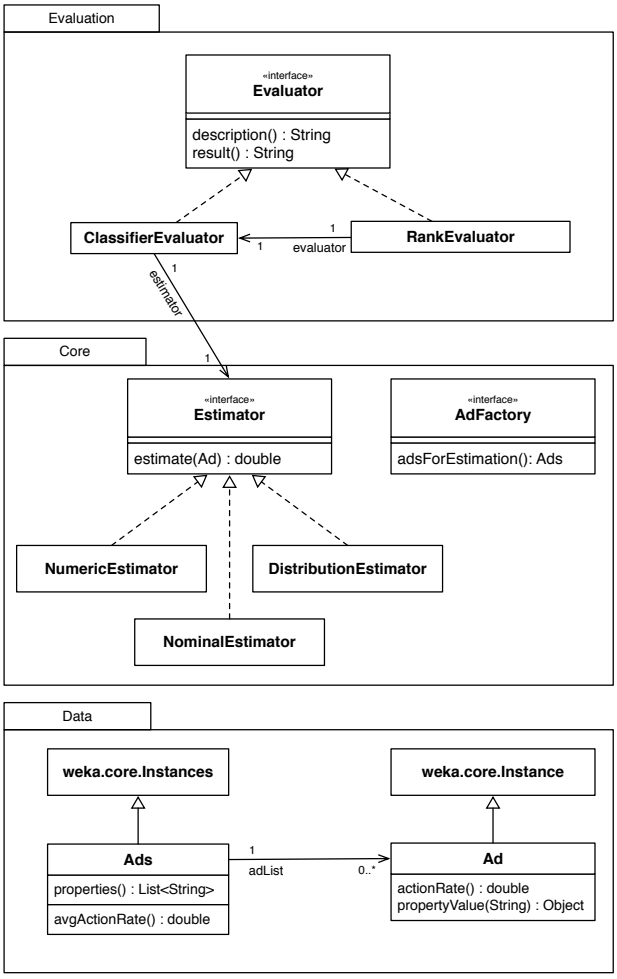

Figure 7: Class diagram of system

ClassifierEvaluator creates its results by instantiating an Estimator with known data, and can then compare the estimates to some known reference values. One such example was described in the beginning of the section, where the action rates of the ads in the AdFactory are saved and compared to the estimated results.

The *RankEvaluator* in turn does not require an Estimator directly, but rather utilizes the results of the ClassifierEvaluator to establish how well classifiers perform for different selection strategies, as described in RQ 1.3.

This overview of components and classes has explained how the software was designed. The following subsections will first suggest how to create the knowledge base required by the system, followed by descriptions of the validation experiments and their results.

5.1 Knowledge base

The data manager was mentioned as the component which interacts with the knowledge base. The knowledge base contains all historical data used by the system and must thus not only be able to provide data, but also be updatable. The actual contents of the knowledge base should be limited to only the data which is used by the other components to con-

serve space, unless that specification of what is useful and not is likely to change.

An efficient way of storing the structured data that we are dealing with is a database, and was also the solution used in our implementation. By storing only the necessary data in a database and updating it whenever new monitored metrics are available, not only is the data manager's task simplified but the data can also be queried from outside the system which avoids lock-in of data, something that could be a problem for some organizations.

5.2 Impact of properties

To show whether different properties carry different weight in regard to how well an ad performs was an important first step in order to get an indication of the potential merits of modeling the data. For example, if an ad consists of an image and a text, are the choices of image and text equally important in the search of a high-performing ad or does one carry a stronger influence on the final result than the other? Or even more importantly, if the properties carry no correlation to the click-rate, that would imply performance has nothing to do with the ad properties defined by the advertiser.

Domain experts from two geographical markets, Brazil and Argentina, were asked to choose three that they had previously run in their market which they identified as representative of high, average and low performance respectively. The three performance levels were chosen to get a baseline performance as well as more extreme values in both directions. The selection was done for men and women separately, so a total of six ads were selected for each region. It is important to note that these were existing ads, so while they performed differently the choice of for example the lowest performing ad was not a manufactured ad whose purpose was to perform badly (e.g. by selecting an offensive image). Finally combinations of the three images and three texts within each region and gender were formed and put online.

The budget for the experiment was 720 USD, or 20 USD per ad. For Brazil this generated an average of 179124 impressions and 77 clicks, and for Argentina those numbers were 382791 and 126 respectively. Once the budget was spent, the results were collected and the correlation coefficient was calculated for each property and the resulting click rate. Each regions data set was also split by gender to see differences between target groups. Finally, the data was gathered and the correlation was calculated. We used the Pearson coefficient of correlation [11], which is defined as

$$\rho = \frac{Cov(X,Y)}{\sqrt{(VarX)(VarY)}} \tag{1}$$

In our case, the random variables X and Y represent the choice of value for a given property and the click rate respectively. Because the covariance and variance are unknown they have to be estimated based on the sample.

$$\hat{\rho} = \frac{\widehat{Cov(X,Y)}}{\sqrt{(\widehat{VarX})(\widehat{VarY})}} = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{(\sum_{i=1}^{n}(X_i-\bar{X})^2)(\sum_{i=1}^{n}(Y_i-\bar{Y})^2)}} \tag{2}$$

Because we are only interested in how big the impact is of different properties and not if said impact is positive or negative, it is the magnitude of the correlation which is important and not the direction. Since $-1 \leq \hat{\rho} \leq 1$, we therefore used the absolute value $|\hat{\rho}|$ for comparison. As an example, even if the correlation for a property is near -1, the conclusion is still that it has a large impact on the performance. In other words, the interpretation of $|\hat{\rho}|$ is that if it is 1 there is an exact linear correlation between the two random variables whereas 0 means there is no linear correlation at all. The results of the experiment are presented in Figure 8.

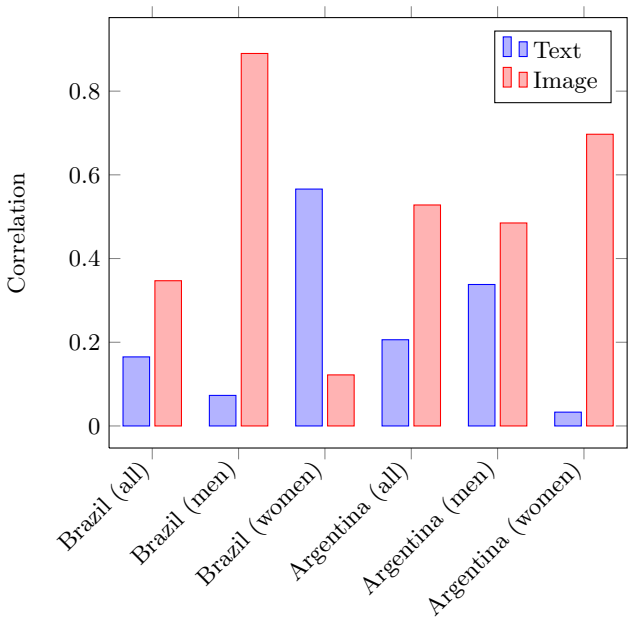

Figure 8: Impact of text and image in ad performance

The results clearly show that the impact of the choice of image is larger than that of the choice of text. Most notable are perhaps the correlation values for men in Brazil and women in Argentina where the image is of utmost importance, as well as the result that women in Brazil are the only group where the text has a higher impact than the image.

Because of the large differences, the results were discussed with one of the domain experts involved in the ad selection. Overall he considered the results to be in line with his expectations, even though the high correlation between text and CTR for women in Brazil and men in Argentina were slightly higher than he would have guessed. A suggested reason for this was that the gender of the domain expert in each region was the opposite of the gender for which the texts were more important than expected (i.e. a heterosexual man is better at selecting high performing images of women and vice versa). Because of this, further experiments will be made at Company A to compare image selection between men and women, however those experiments will not be part of this paper.

The results nonetheless provide an answer to RQ 1.1, as there is a noticeable difference in the impact of the two properties used in this example.

5.3 Rate estimation

The working hypothesis was that a numerical classifier

could estimate the action rate of new ads sufficiently well. By manual experimentation with the data set in the Weka workbench, a support vector regression algorithm called Sequential Minimal Optimization (SMO) was identified as a good candidate for further experimentation.

The experiment was designed as follows. Initially, the data set was randomly split into two separate, non-overlapping sets called training and validation. The training set was then used to create the regression model, after which the action rate of the validation instances was estimated. For each such estimate, an estimate was considered successful if the following inequality held true.

$$c_i - \epsilon \leq \hat{c}_i \leq c_i + \epsilon$$

c_i is the actual action rate for ad i in the validation set and $\hat{c}_i$ is the estimated value. The tolerance limit is defined as $\epsilon = k * \bar{c}$, where k is a constant between 0 and 1. This process was repeated 5000 times and both the total number of estimations as well as the number of successful estimations were counted. From this a success percentage was calculated for different values of k.

For comparison, the same process was also applied with two simple statistical models of the training set. The simplest model was defining a uniform distribution between the minimum and maximum values in the data set, and the slightly more advanced model assumed a normal distribution. The latter was used due to our previous assumption that the data is in fact normally distributed. The estimations from these models were then made by drawing a random value from the distribution. Table 1 and Figure 9 show the results of the experiment for all three models.

Method	k=0.01	k=0.05	k=0.10	k=0.20
Uniform	1.51%	7.41%	14.32%	29.74%
Normal	2.55%	12.99%	25.72%	48.50%
SMO	1.84%	14.66%	34.19%	72.96%

Table 1: Estimation success percentage over 5000 rounds

The value of k defines how large the interval of success is, and thus if k is increased enough, success is always guaranteed no matter the estimate. Figure 10 shows the size of the success spans for the values used in the previously mentioned experiment, with estimate $\hat{c} = 0.00025$ and mean $\bar{c} = 0.00020$.

As k increases it becomes clear that the regression model is noticeably better than the statistical distributions. At $k \approx 0.036$ the regression model and the normal distribution perform equally well, and as k increases the regression model becomes increasingly better in relation to the distribution, which answers RQ 1.2. Additional experiments also revealed that the regression model reached a 50% success rate at $k \approx 0.135$.

5.4 Choice of classifier

Even though the SMO algorithm performed well for estimation, the choice of algorithm was mainly an educated guess. As different classifiers tend to be best suited for different situations, the estimation experiment was revisited to identify the best choice of classifier for this purpose. The only difference was that for this experiment the statistical

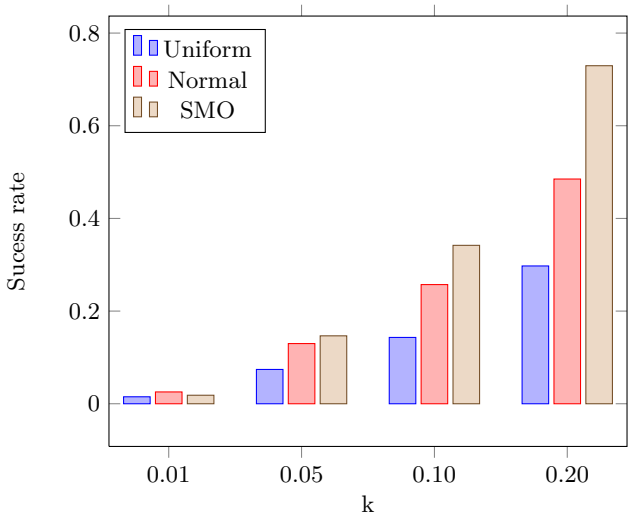

Figure 9: Bar chart of Table 1

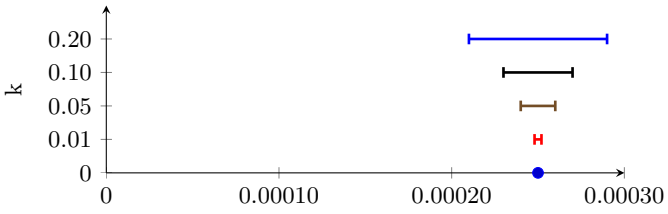

Figure 10: Impact of k-values on success intervals

distributions were replaced with additional classifiers from the Weka library. Each classifier was still evaluated 5000 times, but this time for $k \in \{0.01, 0.02, ..., 0.20\}$.

All numerical classifiers from the library were explored. Besides the previously mentioned SMO algorithm, the classifiers used were standard linear regression (*LinearRegression*), a model tree that can have linear models at the leaves (*M5P*), an extension of the nearest neighbor approach called instance-based learning (*IBk*), a back-propagation classifier (*MultilayerPerceptron*) and a regression tree (*REPTree*) [18]. In addition to this, the library's logistic regression implementation (*Logistic*) was evaluated. Figure 11 shows the results.

Because the increase of the success rate is not exactly linear, there is no clear best choice at first glance. However, because there is little use for an operator to be told that there is less than a 50% chance that an estimated interval is correct, since such a statement would imply that it is in fact more likely that the real value lies outside the interval than within, we can focus on the parts of the graphs above that threshold. Above the 50%-line there are only two classifiers that at some point have the highest success rate, the logistic regression and the REPTree. The rates of these classifiers follow each other closely up to $k = 0.15$, at which point the latter no longer increases in accuracy.

Based on these results, we then selected the four classifiers with the highest success rates above the 50% threshold to run additional experiments on. The first one was to analyze in more detail the size of the estimation errors. To do this,

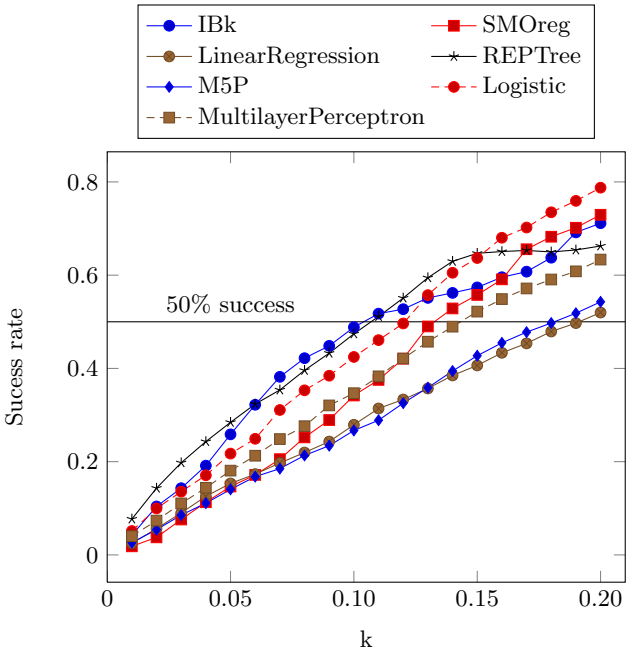

Figure 11: Success rates for different classifiers

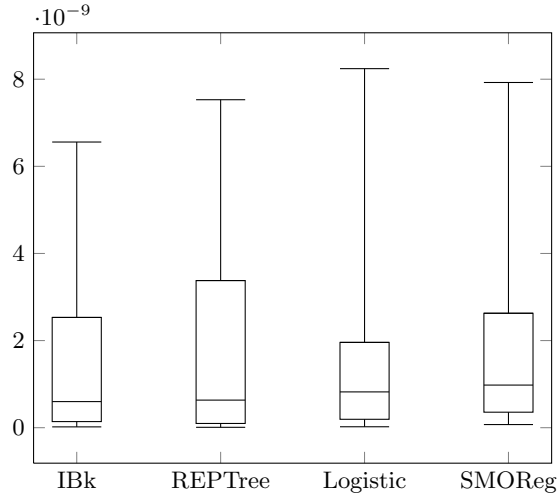

Figure 12: Mean square error for classifiers with highest success rate

the same experiment as before was run, but instead of calculating success rates we instead calculated the mean squared error over 5000 runs. The results are presented in Figure 12. The box contains the values between the 25^{th} and 75^{th} percentiles with the line inside marking the median value. The bottom and top whiskers show the 10^{th} and 90^{th} percentile respectively.

It is interesting that logistic regression has the lowest upper limit of the box, which means most of its estimation errors are very small, however it also has the highest top whisker which implies that there are also times when estimations are completely off. IBk on the other hand has the lowest top whisker, and as such has a lower spread, even though it may not be quite as accurate as the regression in many cases. That being said, IBk still has the second lowest upper limit of its box and those two properties put together make it an interesting choice.

There is however another aspect of classifiers that is interesting to analyze further before drawing any final conclusions, namely performance. The four classifiers with the highest success rates above the 50% threshold were analyzed from a timing point of view. The results are in Figure 13, where the time measured in milliseconds is shown on a logarithmic scale.

For each data point, a new set of data was generated with N attribute values for two attributes (representing the body text and image used at Company A). The data set thus contained N^2 instances from which N instances were removed and put into a set of test data. The removed instances were those combinations of text i and image j where $i = j$. This systematic removal was used to guarantee that no property value would be completely removed, which could potentially happen with a random removal process. The remaining $N(N-1)$ instances were left as the training set. Each instance in the training data was given metrics, which were drawn from a random distribution.

After this, the time taken for each classifier to initialize itself (i.e. create the underlying data model based on the training data) plus the time required to estimate the N instances from the test data was measured. This was repeated three times for every data point and the average time was calculated.

As can be seen in Figure 13, the IBk and REPTree algorithms are really fast, almost constant no matter the size of N. Then comes logistic regression which takes about 21 seconds for $N = 100$, whereas SMO regression is highly unsuitable for larger data sets since it required more than 14 minutes at the same value for N.

Another aspect of time performance is increasing the number of occurrences of the same ad, each with its own metrics. Continuing with the previous experiment, we fixed N at 10 and added a second variable M, which is the number of times each ad should occur. The reason for this experiment is to get an idea of whether it is the possible combinations or the actual number of instances which has the most impact on the time it takes to build the estimation model and estimate ads. The results are displayed in Figure 14.

The time performance in the two experiments is comparable. The main difference is that logistic regression seems to handle the increase in the number of instances much better than the the increase in combinations of property values. Since $M = 1$ in the first experiment, there are a total of 10000 instances in the generated data set when $N = 100$, which is the same amount as the second experiment has for $M = 100$ when $N = 10$. Still, the time taken is higher by a factor of 16 for the first experiment.

The most time consuming task for the two slower algorithms is building the actual classifier. Depending on how often an organization needs to update the knowledge base, this could have an effect on the choice of classifier. The accuracy of logistic regression will outweigh the cost in time in some cases, however for large sets of data it may be more useful to suffer the slight loss in accuracy in order to efficiently incorporate new data in the model, in which case

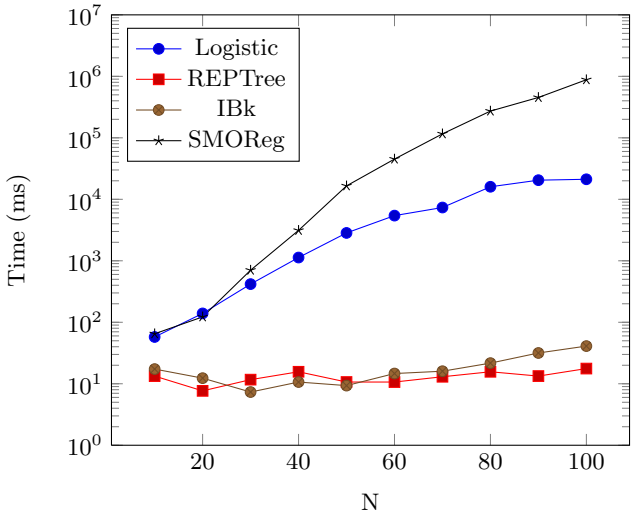

Figure 13: Classifier performance (combinations)

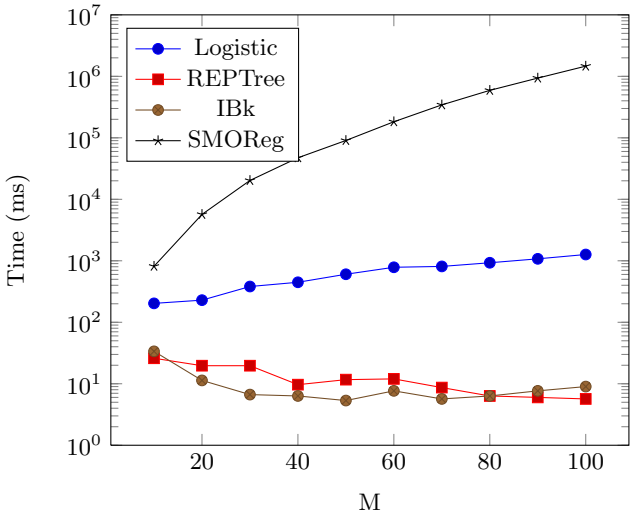

Figure 14: Classifier performance (instances)

the IBk is recommended, since its mean squared error is lower than the REPTree.

For Company A we decided to use the faster IBk algorithm, because they expected to reach the limits of what logistic regression can handle without requiring so much time that it affects the user experience. Another option discussed was having functionality in the user interface to switch between algorithms, but the idea was discarded for two reasons. First of all, the data set would most likely grow fast enough so that a switch would be required very soon anyway, and secondly there was the problem of making such a function understandable for the average operator, since they are not expected to know about classification algorithms.

5.5 Selection strategy

Once the estimations are made the question becomes how to select which ads to run. As described in RQ 1.3, if the goal is to gain the highest average click rate over all running ads, it could perhaps be more effective to choose multiple suggested ads to minimize the impact of erroneous estimations.

Strategy N is defined as the selection strategy where the N highest estimated ads are run. In addition to this, the ads were run either weighted or unweighted. The latter means that all ads are shown the same amount of times, whereas the former approach calculates a weight for each ad based on the normalized value of the estimate. Functions for the calculations are shown in Equation 3 and 4 respectively. This process was then repeated 5000 times and the values averaged to identify the optimal selection strategy. The results are available in Table 2 for both logistic regression and IBk.

$$c_{unweighted} = \sum_{i=1}^{N} \frac{c_i}{N} \tag{3}$$

$$c_{weighted} = \frac{\sum_{i=1}^{N} \hat{c}_i \cdot c_i}{\sum_{i=1}^{N} \hat{c}_i} \tag{4}$$

As the results reveal, the best approach over time from a purely statistical point of view is to only select the ad

Algorithm	N	Unweighted	Weighted
Logistic	1	0.0209%	0.0208%
	2	0.0201%	0.0201%
	3	0.0192%	0.0192%
IBk	1	0.0205%	0.0203%
	2	0.0197%	0.0197%
	3	0.0190%	0.0190%

Table 2: Averaged real click rate for selection strategies N

with the highest estimated performance, no matter which of the two algorithms is used. Other reasons for choosing multiple suggestions may of course exist, such as diversifying for potential target groups for example, but those choices need to be made by a domain expert. It is also interesting to note that the weighting of ads carries no additional boost nor any significant negative effect to the average performance.

While these results do not support any more complicated selection strategy than "best first" as answer to RQ 1.3, they do strengthen the proof for that the classifiers indeed provides a good estimate. A strategy where $N > 1$ would only be better suited if the estimator tended to overestimate the value of the $N-1$ first ads, so that in fact the N^{th} was the best choice of ad. In other words, the results indicate that the ranking of the suggested ads on $\hat{c}$ reflect the expected ranking based on c.

5.6 Influence of time

To accurately be able to compare the estimated performance of a new ad with existing ads, it is important to also estimate future performance of already existing ads. If ad performance declines over time, even the best performing ad may reach a point in time where its performance is expected to be worse than that of a new ad. The reason for such a decline could be for example because the advertiser will not show a specific ad to a user who has previously clicked it.

In order to be able to analyze this question, described in RQ 1.4, data was collected daily over a period of two weeks

for two separate ads, one targeting men and one targeting women. The reason for choosing two weeks was a trade-off between being able to identify differences between weekdays and those over time (e.g. perhaps weekends see a higher performance due to an increase in online presence) while not incurring too high costs.

The performance for the ads is presented in Figure 15 and 16 respectively. The dots represent the mean click rate for each day, whereas the bars on each coordinate represent the 90% confidence interval for the click rate. We can calculate this interval since we have made the assumption that the data is normally distributed, as was described in the data set description. The values on the x-axis is the day of the week of the measurement. The data was collected between October 3rd and October 17th 2012. It is worth mentioning that October 12, the second Friday in the data set, was a holiday in the targeted region.

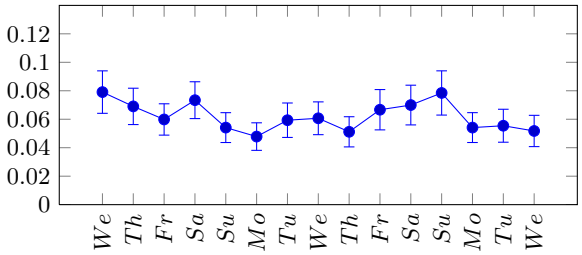

Figure 15: CTR over time for ad targeting women

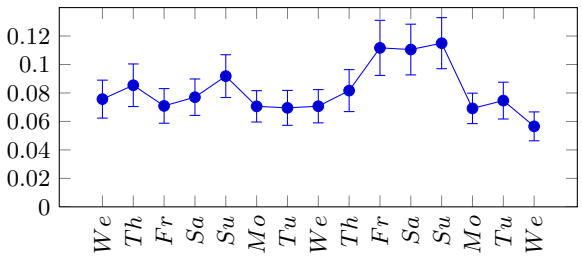

Figure 16: CTR over time for ad targeting men

By following the performance over the three Wednesdays in the data set, both ads do show a declining trend. It is not enough however to draw any strong conclusions on, since the confidence intervals are overlapping for all three data points in both sets. Comparing only two data points shows no strong decline, and in the case of for example Sundays there is a strong increase in both data sets. The biggest differences appear in Figure 16, where it becomes clear that with 90% confidence there was an increase in click rate between Fridays.

There is some indication that given a longer time period there might be a decline, though this experiment cannot prove such a theory.

5.7 User interface

eb ▶*Screenshot and short discussion*◀

6. CONCLUSION

In this thesis we have shown a technique for developing an automated system that provides decision support to organizations working with social network marketing. We have also described and performed experiments which validate our solution.

The experiments indicated that the most accurate estimator for ad click rates was the logistic regression, however a performance analysis revealed that it might be too slow for large data sets. For such a case, a faster algorithm such as the REPTree is recommended, which does not suffer from the same exponential increase in time while still retaining a useful level of accuracy.

The software solution consists of three basic components. The Data Manager is responsible for interacting with the knowledge base of all previously collected information. This data is then used by the Estimator, which models the data in order to provide estimates of new ads. The ads to be estimated are generated by the Ad Factory, and typically does so by generating combinations of ad properties which do not exist in the knowledge base. To interact with the system, these components need to be combined in a user interface, which most likely will be graphical though that is not a requirement.

The next subsection will discuss possible concerns regarding the validity of the research shown in this paper, followed by possible future work to extend and improve upon this work.

6.1 Validity

Though great care has been taken to be thorough in the research for this thesis, this section will discuss some factors that may be of interest for the reader to both understand potential limitations of the research in terms of constraints and risk factors, as well as potential future additions to the results presented in the paper.

6.1.1 Constraints

Running ads through a publisher obviously incurs a cost in terms of both money and time, and as such it became a constraint for this study, which of course is not in any way an unusual limitation. It was sometimes possible to use existing ad data, while at other times it was necessary to specify requirements which were passed along to the marketing department and put online. The time ads could be kept online was a trade-off between the needs of this study and the budget of Company A. The existing data sets were very useful, but in an optimal study there would be a larger set of data with long running ads, including low-performing ads that for business reasons are typically removed quickly.

6.1.2 Risk factors

Because the output of the system is dependent on historical data, an assumption has been made that older data is still representative of the current state of the marketing. This is definitely not true for campaigns that are adapted for Christmas, Valentine's Day or other special occasions. The assumption is that the amount of such time-dependent data is so small that it will not impact the final results. If however there is reason to believe that this set of data would influence the output, the recommended approach is to remove it from the knowledge base.

6.2 Future work

Due to circumstances described in the introduction, the targeting of ads was removed as a goal of this thesis. For advertisers with many campaigns with differing targeting it would most likely be interesting to extend the system with capabilities to handle target groups. For targeting which is considered binary (either a person has the attribute, or they do not), such as for example gender or nationality, the described system can be used by simply separating the data sets from each other. This is a simple solution which has the downside of not modeling potential similarities in behavior between groups, which could increase the efficiency of the modeling.

A more advanced variant of the above problem is that of attributes which can be described using scalar values. An example of this could be age, where it would be reasonable to expect that people whose age is only a year or two apart would be attracted to the same properties in an ad. Simply dividing the data sets by age would probably lead to an even greater loss of information than the cases described previously.

Though this thesis has limited the scope of the system for use with social networks, it would be useful to develop the technique further to deal with multiple classes of advertisement, such as for example search and contextual advertisement, since many organization combine classes to expand their audience.

Finally, in the discussion regarding the impact of different properties, the idea that the gender of a marketing expert could strongly influence how well ads perform was presented. This could very well be the hypothesis of future work in the field of marketing.

7. ACKNOWLEDGEMENTS

We would like to thank our thesis supervisor Matthias Tichy for his support and the ideas he provided throughout this project. We would also like to thank our industrial supervisor as well as the marketing people at Company A for sharing their knowledge and giving us feedback on our work.

References

[1] V. Dave, S. Guha, and Y. Zhang. Measuring and fingerprinting click-spam in ad networks. *Proceedings of the Special Interest Group on Data Communication (SIGCOMM)*, 2012.

[2] U. Fayyad, G. Piatetsky-shapiro, and P. Smyth. Knowledge discovery and data mining: Towards a unifying framework. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, August 2-4, 1996*, pages 82–88. AAAI Press, 1996.

[3] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus. Knowledge discovery in databases: An overview. *AI Magazine*, 13(3):57–70, 1992.

[4] S. R. Garner. Weka: The waikato environment for knowledge analysis. In *Proceedings of the New Zealand computer science research students conference*, pages 57–64. Citeseer, 1995.

[5] A. Goldfarb and C. Tucker. Online display advertising: Targeting and obtrusiveness. 2011.

[6] M. Golfarelli, S. Rizzi, and I. Cella. Beyond data warehousing: what's next in business intelligence? In *Proceedings of the 7th ACM international workshop on Data warehousing and OLAP*, pages 1–6. ACM, 2004.

[7] Z. Hadija, S. Barnes, and N. Hair. Why we ignore social networking advertising. *Qualitative Market Research: An International Journal*, 15(1):19–32, 2012.

[8] IBM. An architectural blueprint for autonomic computing. *Quality*, 36(June):34, 2006.

[9] A. Joshi and R. Motwani. Keyword generation for search engine advertising. In *Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference on*, pages 490–496. IEEE, 2006.

[10] M. Kantardzic. *Data mining: concepts, models, methods, and algorithms*. Wiley-IEEE Press, 2011.

[11] J. Milton and J. Arnold. *Introduction to probability and statistics: principles and applications for engineering and the computing sciences*. McGraw-Hill, Inc., 4th edition, 2002.

[12] S. Negash. Business intelligence. *Communications of the Association for Information Systems*, 13(1):177–195, 2004.

[13] M. Regelson and D. Fain. Predicting click-through rate using keyword clusters. In *Proceedings of the Second Workshop on Sponsored Search Auctions*, volume 9623, 2006.

[14] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*, pages 521–530. ACM, 2007.

[15] J. Royston. An extension of shapiro and wilk's w test for normality to large samples. *Applied Statistics*, pages 115–124, 1982.

[16] M. Shaw. What makes good research in software engineering? *International Journal on Software Tools for Technology Transfer (STTT)*, 4(1):1–7, 2002.

[17] S. Thomaidou and M. Vazirgiannis. Multiword keyword recommendation system for online advertising. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 423–427. IEEE, 2011.

[18] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, 3rd edition, 2011.

[19] W. Yang, J. Dia, H. Cheng, and H. Lin. Mining social networks for targeted advertising. In *System Sciences, 2006. HICSS'06. Proceedings of the 39th Annual Hawaii International Conference on*, volume 6, pages 137a–137a. IEEE, 2006.

[20] Q. Zhang, T. Ristenpart, S. Savage, and G. Voelker. Got traffic?: an evaluation of click traffic providers. In *Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality*, pages 19–26. ACM, 2011.