University of Manchester

School of Computer Science
Project Report 2014

Using Machine Learning to predict personal expenditure

Author:
Pez Cuckow

Supervisor:
Gavin Brown

Q1: I'm still not sure on the title, the report includes: prediction, historical finances and security

Abstract

Abstract pending

P1: Abstract, assuming I need one

Keywords: MY KEYWORDS, GO HERE

Contents

List of Figures

List of Tables

Glossary

category transactors have a category and a subcategory, e.g. Tesco = Shopping, Groceries. 10, 11, 15

global transactor the system holds two collections of transactors and mapping's, the global ones are shared between all users, and only accessed with the admin panel.. 19

mapping this connects the reference found on a statement to a Transactor. e.g. Snbs, Sains =¿ Sainsbury's. 19

reference the memo or message that is included on the bank statement with a transaction. 7, 19

transaction a single movement of money from/to a Transactor. 7

transactor somewhere money is spent, e.g. Tesco, Sainsbury's, Byte Cafe.. 18

user transactor unique to each user. 19

Chapter 1

Introduction

Traditionally the management of personal finances is performed by viewing bank statements provided by the users bank. In the modern age of 'Internet banking', banks offer a limited set of tools that mimic the paper statements seen historically.

This project sets out to build an online application that can be used to manage personal finances. There are two main parts of the project; firstly users can upload bank statements, which are displayed and navigated in an intuitive manner; secondly, once the application has enough historical data, predicting the users future outflow.

1.1 Motivation

There are four main steps when producing and using a budget: recording previous expenses, sorting these into categories, using this historical information to estimate future expenditure, and evaluating the accuracy of predictions new information and adjusting accordingly.

Since the liquidity crisis of 2009 [1], budgets have been squeezed and the average persons personal disposable income has fallen signifiantly , hitting a nine-year low in 2012 [2]. With experts suggesting that "Budgets are essential for financial planning" [3], research suggesting that personal budgets lead to a "positive impact" on "mental wellbeing" [4] and guides from UCAS, the UoM SU Advice Centre and The Manchester University Crucial Guide encouraging use of budgeting, it is clear that producing a budget is of benefit.

In an informal survey[1] by the researchers, however, the majority of stu-

Q2: Refrences inline, or in a block at the end of the paragraph?

[1] Appendix A

dents questioned, did not heed this advice, and were not following a budget. Producing an easier way to manage personal finances and predict future outflow can hopefully reduce the barriers to entry for creating budgets and increase the people using one.

Increasing use of debit cards [5] means that bank statements contain more and more information about where people spend their money. With access to those bank statements now provided online, with most UK banks offering the option to export transaction history, individual users can collate a database of their personal spending habits.

The increasing availability of this data, combined with more detailed transaction history makes it potentially possible to automate the four main steps of producing a budget, and this is the main objective of the project.

1.2 Aims and Objectives

The key objectives of this project can be split into three parts, the management of statements, making predictions of future outflow using those statements and ensuring a high level of security.

1.2.1 Statement Management

Implement an intuitive way to view and manage personal finances. There are several key parts to this, upload and parsing of transactions from statements downloaded from a bank, resolving the references found on the statement to the real world business they represent and categorising the individual transactions to make them easier to understand.

Q3: Is it clear what categorising is?

1.2.2 Prediction

Accurately predicting the future transactions that a user will based on their transactions history. The prediction should be made using a model that is fitted each users individual spending patterns, and evaluated in order to improve the model. The application will need to predict whether or not spending will occur and how much money will be spent.

1.2.3 Security

The project should be secure and uphold the high levels of expectation from the users uploading their personal information. The application will deal with information of a sensitive nature, strong security techniques are of

high importance to ensure no loss of personally identifiable information. The project should take this into account, considering possible attack vectors and taking steps to mitigate those attacks.

Q4: What tense should this be in? The project took or should take

1.3 Overview of Report

P2: This report covers some of the key design decisions, implementation decisions and then what the application does

Chapter 2: Background An review of existing products in the market and an introducing to techniques used in the project.

This needs to be completed

Chapter 2

Background

As outlined in Chapter 1 the project consists of two major parts: a financial management service, which can be used to view historical spending and gain understanding of personal finances; and a forecasting element which predicts how much spending will occur in the future.

Q5: Include this?

2.1 Statement Management

There are existing applications that implement similar features to the money management aims of this project, most notably Lloyds TSB Money Manager, the first and only personal money management application provided by a UK bank and Mint.com a United States (US) only personal finance service [6], [7].

Q6: Technically it's spending or receiving "transaction" but this becomes very wordy?

Q7: Mention the current "bad" internet banking here?

2.1.1 Lloyds Money Manager

The service is available to Lloyds TSB current account holders as part of their online banking and it's features revolve around documenting historical spending [8].

The key features include:

- Categorising spending
- Creating spending plans per category
- Viewing money spent per category
- Track progress of budget targets

Customer reviews of the service highlight the usefulness of spending analysis screen, which includes a breakdown of spending in each category (Fig. 2.1, as well as the spending calendar, which displays money spent in a day by day format. The reviews, however, also highlight some shortcomings, noting that changes to categories are not reflected immediately, categories are often incorrect and that it's not possible to override the category for a single transaction, for example food bought at a petrol station is placed in the Car category and cannot be moved [9], [10].

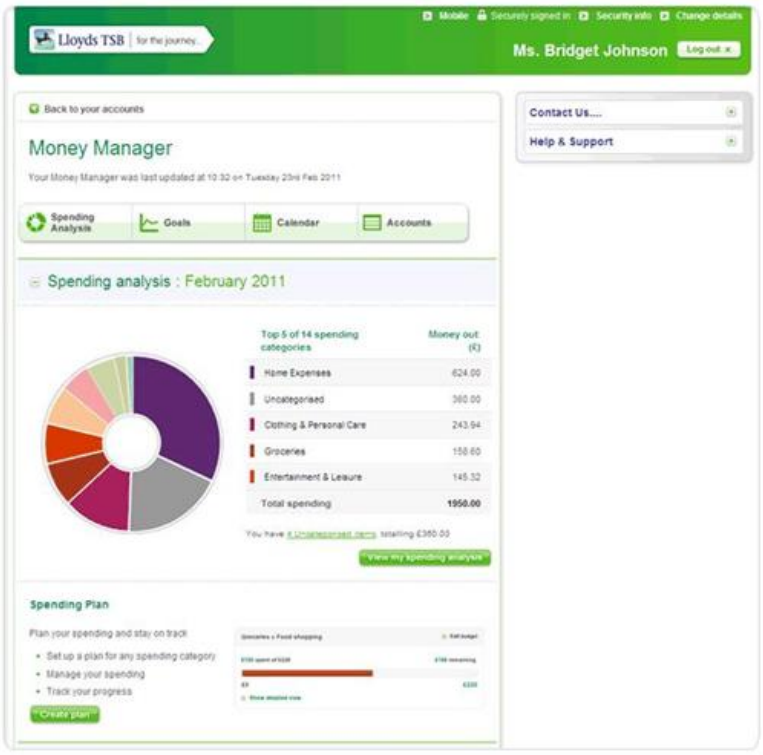

Figure 2.1: Spending Analysis by category on Money Manager [8]

A key advantage of the money manager is that Lloyds already have access to their customer data, so there is no data entry or upload required, which could be confusing and off-putting to potential users.

2.1.2 Mint.com

Mint.com offers very similar features to LLoyds but it limited to the US. However, Mint automatically logs into the users online bank account and downloads their statements authenticating with their banking username and password. It's reported that this feature relies on the use of application programming interface's (API) at each bank which Intuit (the company behind Mint) have negotiated access individually, though Intuit have published no information to support or dispute this [11], [12]. Although this feature is clearly useful and saves time for the users, it does make Mint responsible for storing their customers Internet banking passwords and presumably involves fee payments to the banks providing these API's. For these reasons it was decided that automatic statement uploading was outside of the budget and scope of this project, however, the project should support manual upload of statements to avoid date entry of users.

Q8: Should I have more detail on what mint/lloyds do?

Q9: Does this need backing up?

2.1.3 Mobile Apps

Mobile applications or 'apps' as they are commonly known have seen a surge in popularity since the release of startphones and are a common target for small pieces of software, such as financial organisation [13].

The three most popular iPhone personal financial applications [14], at the time of planning the project, all offered features very similar to those found in the Lloyds Money Manager and Mint. The most popular features being grouping money by category and graphs of spending history. However, they all had the same drawback, the user had to manually enter all of their transactions and set categories for them, which appears time consuming and error prone, particularly on a mobile app [15]–[17].

The increase of mobile usage should be considered when planning the features of the project, with the project ensuring mobile compatibility and if possible, avoiding manual data entry.

P3: This needs improving

2.2 Prediction

There are various approaches to making predictions of financial spending, each with their own advantages and disadvantages . Predicting future transactions before they occur is technically similar to the work done by investors on the stock market, where the objective is to predict whether the value of a stock will fall or increase in order to make buy/sell decisions.

Q10: Not sure if this is relevant

Preifer and Carraway demonstrated that Markov Chain Models can be used to model customer relationships with a business and predict the expected value of a marketing engagement with an individual customer. By creating a transition matrix of a particular customer transitioning from not spending to spending and visa versa over five periods[1], they were able to estimate the likely-hood of a spend occurring in a given period, Fig. 2.2 shows a graphical representation of the model that was produced, the states represent the five periods, where p_i is the probability of the transition occurring during period i [18]. The paper is able to calculate the expected loan to value ratio (LTV) for the customer over the periods, by taking a matrix costs and gains associated with a purchase in each period and multiplying that by the probability of a purchase occurring taken from the transition matrix. This gives the expected present value for each period, which can be used to decide when to end a relationship with a customer (preventing the costs). They demonstrate applying Markov Chain Models to a larger dataset, calculating the optimal policy for ending relationships with customers depending on varying costs concluding that the use of MCM's is an effective way of making customer relationship decisions. However, this paper assumes the company performing these predictions already knows how much money a customer will spent during each interactions and is focussed around calculating the probability of a spend occurring. An implementation applied to the personal spending space will require a way to predict the value of the future transaction.

Q11: Mention what HMM's are here?

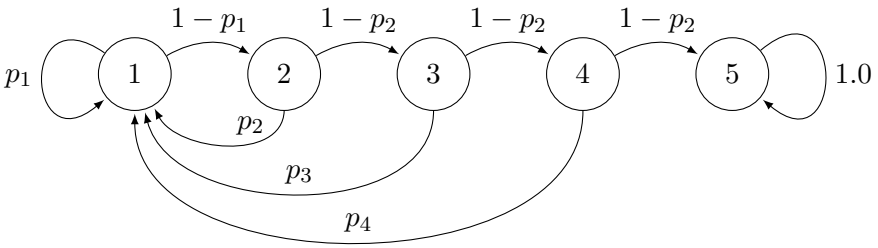

Figure 2.2: Markov Chain Model for a particular customer over five periods [18, Adapted from Fig. 1]

Research by Singh et al., from the Massachusetts Institute of Technology, studied the spending behaviour of 52 adults and investigated the impact

[1] An 'illustration' assuming a customer will never return after 5 months of not spending

of social interactions, including text messages, phone calls and face-to-face meetings, on the participants spending in order to predict their spending behaviour. Using a Naïve Bayes classifier and selecting a subset of their available features using an Information Gain approach, choosing those with most relevance to each classification task, they were able to correctly classify whether the participant would overspend, explore a diverse range of businesses and remain loyal to a business with 72% overall accuracy. They concluded that social factors, were better "predictors of spending behaviour" than personality traits, which had been previously studied [19]. Although this paper did not study the affects of the participants previous transactions on spending, they were able to predict the users spending behaviour, highlighting that factors other than the transaction history may be of importance when trying to predict a users future outflow. However, the paper does not attempt to make a prediction of the amount spent or how many transactions occur.

Smoothing is typically applied to financial market data, for example the value of a particular stock on the FTSE 100. The most common techniques are simple, weighted and exponential moving averages, which all reduce the noise found in the data potentially revealing an orderly process, by removing outliers found in the data. The result of this effect can be seen in Fig 2.3 [20].

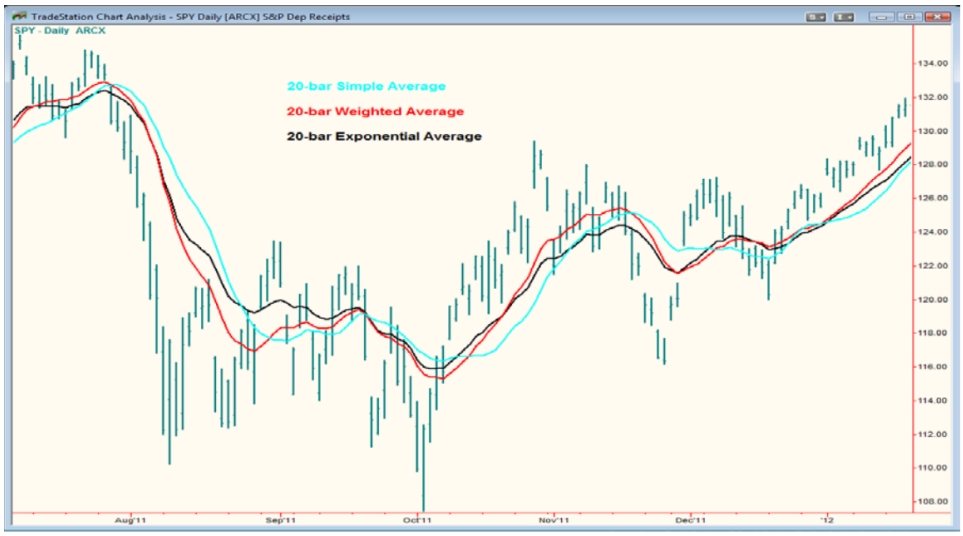

Figure 2.3: Simple Moving Average (MA) [blue], Weighted MA [red] and Exponential MA [black] of the S&P500 [3][20, Fig. 5]

These techniques can be applied to a discrete set of numerical time series data, such as personal expenditure over time in order to make an estimate of what the next value in the series will be [21]. A prediction can be made using the formula in Fig. 2.4, where w_i is the weight and x_i is the value at time period i. Simple smoothing is the equivalent of $w_i = 1$, while exponential smoothing is based around a negative exponential law such as $w_i = e^{-n+i}$, both are examples of weighted smoothing and the weights can be decided in different ways depending on what is being predicted. Time periods with a higher weight have a greater affect on the mean, so in order to make a future prediction, the most recent time period would have a higher weight.

Q12: The chosen implementation details in the design section

$$\frac{w_1x_1 + w_2x_2 + \cdots + w_nx_n}{w_1 + w_2 + \cdots + w_n}.$$

Figure 2.4: Using weighted smoothing to predict a future value

Smoothing (and therefore prediction) can be extended to take into account trends and possible seasonal fluctuations using double and triple smoothing, respectively. A technique known as 'Holt-Winters double exponential smoothing' takes into account trends in data, which single smoothing does perform accurately with, by factoring the weighted average growth between previous the time series when calculating the average for each period [22]. Extending the calculation into double and triple smoothing when estimating a users future outflow was decided as a possible extension for the project.

Q13: A demonstration of this is in chapter X?

Q14: Where to mention this?

2.3 Security

Q15: Should I mention the password entropy paper here? Currently in design

[3] A stock market index of 500 American companies, the US equivalent of the FTSE 500

Chapter 3

Design

This chapter covers design of the system, including an overview of the architecture and descriptions of the key components.

3.1 Statement Management

The statement management features of the application were selected based on the functionality observed during the background research and conversations with potential users, asking what features they enjoyed from their current Internet banking and what additional features they would find useful to manage their statements

The key features include; parsing of files downloaded from Internet banking, mapping transactions found in the files to real world businesses (transactors), organising transaction history by category or transactor and viewing all transactions at a particular transactor.

3.1.1 Upload

To get a users transaction history they must first upload a file containing their historical transactions.

The major UK banks tested[1] provided statement downloads in Quicken, Microsoft Money or Microsoft Excel format. Further investigation revealed that the underlying formats were Quicken Interchange Format (QIF), Open Financial Exchange (OFX) and comma-separated values (CSV).

As there is no pre-defined standard for bank statements in CSV format, upon investigation, it became clear that the banks used completely different

[1] Natwest, First Direct and HSBC

structures. It was decided the application would parse the QIF and OFX formats, following their respective specifications. Examples of QIF and OFX can be seen in Fig. 3.2.

It was quickly identified that although the QIF/OFX files were following the same specification, depending on the bank they had different structures, and in some cases the structures even varied from the same bank, depending on the exact wording of the download. Interestingly a OFX file downloaded from First Direct was found to be in QIF format, despite an .ofx suffix.

Notably there were discrepancies with the formatting of dates in QIF. The specification from Intuit[2] doesn't specify a date format [23]. The sample files tested included dates in D-M-Y, M-D-Y and Y-M-D format.

To combat this three steps of resiliency were added to the design of the upload system, seen in Fig. 3.3 .

Recompile this figure

Having uploaded the file the system first identifies the filetype by looking inside the file and parsing its contents, ignoring the extension and rejecting the file if it matches neither format.

If the file is QIF the parser parses all transactions up front and evaluates the format of the dates. If the dates are found to have the format \d00−00−0000 (\d{1,2}[/−]\d{1,2}[/−]\d{2,4} in regular expression) the system needs to decide whether that's D-M-Y or M-D-Y, otherwise performing standard date parsing of the string. To decide between D-M-Y or M-D-Y the application goes through all the dates and attempts to parse in both formats, if a format fails it is marked as incorrect. This leaves the application in one of the four states, seen in Table 3.1. In the case of state 1 or 4 there is ambiguity and the application prompts the user. This ambiguity can be caused by dates that are malformed or a collection of dates falling within a range that doesn't have a day value over 12, as both formats parse correctly.

In provisional user testing, it was discovered that users had a tendency to upload the same file more than once or to upload statements with an overlapping date range. To account for this, before creating a new Transaction the application checks for an identical transaction for the current user in the database and if one is found, skips creating a new Transaction. For speed this is done using a stored unique value, resulting from a SHA512 hash of date posted, transaction value, transactor, memo and transaction id[3], which is generated when saving a Transaction to the database.

[2]The developers of Quicken and QIF

[3]If a one was provided by the users bank

```
% QIF FORMAT
!Type:Bank
D28-06-13
PASDA SUPERSTORE      TROWBRIDGE
T-15.00
^
D28-06-13
PPAYPAL PAYMENT
T-12.50
^
```

Figure 3.1: Two transactions in QIF format

```
% OFX FORMAT
<STMTTRN>
<TRNTYPE>POS</TRNTYPE>
<DTPOSTED>20130628</DTPOSTED>
<TRNAMT>-15.00</TRNAMT>
<NAME>ASDA SUPERSTORE</NAME>
<MEMO>TROWBRIDGE</MEMO>
</STMTTRN>
<STMTTRN>
<TRNTYPE>DEBIT</TRNTYPE>
<DTPOSTED>20130618</DTPOSTED>
<TRNAMT>-12.50</TRNAMT>
<NAME>PAYPAL PAYMENT</NAME>
</STMTTRN>
```

Figure 3.2: Two transactions in OFX format

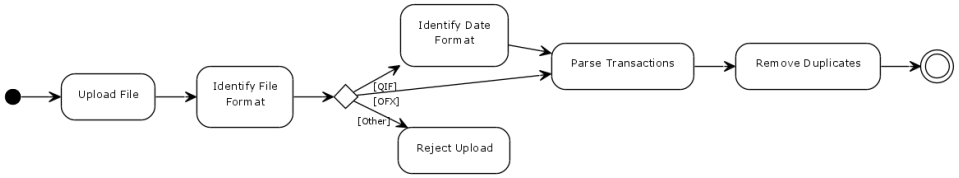

Figure 3.3: Activity diagram for statement uploads

State	D-M-Y	M-D-Y
1	true	true
2	true	false
3	false	true
4	false	false

Table 3.1: Possible states following evaluation of transaction dates

3.1.2 Named Entity Resolution

Almost all functionality of the project relies on successfully mapping the text found on a bank statement that represents a business or person to a single entity in the application, known as a transactor by the system. After a cleanup of different suffixes that banks append it was found that transactors are often referenced using several names.

Seen in Table 3.2, Sainsbury's was referred to nine different ways in the statement data uploaded by the research participants and similar results are found for most transactors.

Reference	Occurrences
sainsburys s/mkts	46
sainsburys s/mkt	9
sainsburys s/mkts cd	7
js online grocery	2
sainsbury s/mkt cd	2
sainsburys smkt	2
js online grocer	1
sainsburys superma	1
sainsburys-superma	1

Table 3.2: References to the entity 'Sainsbury's' found in participant data

Mapping to Entities

In consideration of this, the concept of mappings was added to the system. A mapping is a single reference to a transactor, such as 'sainsbury s/mkt'. A transactor has multiple mappings. Fig. 3.4 shows this structure.

Figure 3.4: Overview of Mappings

Global vs User

As identified in the background research, it should be possible for users to both categorise and organise transactions according to their preferences and override existing categories, however categories chosen by a particular user should not affect affect other users.

To support this the application stores two sets of mappings and transactions, User and Global. The structure of the relevant objects is shown in Fig. 3.5. A Transaction can have both a UserMapping and a GlobalMapping, in which case the UserMapping overrides the GlobalMapping when calling methods such as getMapping() on the Transaction.

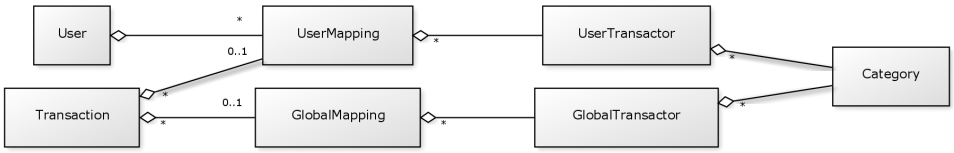

Figure 3.5: Overview of User Mappings

3.1.3 Suggestions

Having mapped references to entities, the system is able to use this knowledge to make suggestions of appropriate entities for unseen references in some cases to help streamline the naming process for potential users. This is performed by taking the list of mappings and finding those with the smallest difference to the unseen reference. Difference can be calculated in several different ways, including the Levenshtein distance which calculates the

number of single-character edits to transform between the two strings, implementation details for this project can be found in Section 4.2.2 [24].

3.2 Prediction

In order to make a prediction of how much money a user will spend and receive in a given period two steps need to be completed, predicting whether or not each individual transaction will occur in a given month and estimating how much money will be involved.

Drawing from the research detailed in Chapter 2, the system uses a First-order Markov Chain model to decide whether or not transactions will occur, and weighted arithmetic means to predict how much money will be spent.

3.2.1 Markov Chain

An easy way to visualise the Markov Chain Models the system is creating is through a directed graph. Two frequent examples are shown in Fig. 3.6 and 3.7, taken from participant data, where 0 represents a transaction not occurring, 1 is the opposite and the edges are labelled with the probability of transitioning from one to the other.

Overuse of the word occurring

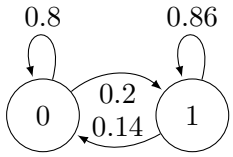

Figure 3.6: Transition diagram for a monthly pay check

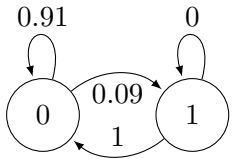

Figure 3.7: Transition diagram for a one off purchase

3.2.2 Weighted Arithmetic Mean

Having predicted whether a transaction will occur or not the system needed to predict how much money would be spent. A simple way to make this prediction us taking the mean, however initial testing in MatLab revealed that simply taking an average is affected highly by changes in spending patterns and skewed by outliers, in addition a spending pattern that suddenly changes (for example one caused by the user changing supermarket) takes too long to be reflected in the prediction.

Supported by the background research on weighted smoothing, the system uses weighted averages to account for this. A weighted average is similar to an average but each value is scaled in it's effect by a weighting factor, this allows the system to give a higher weight to more recent transactions.

The weighted average calculation used is shown in Fig. 3.8 where $w(t)$ is the weighting function for time t, the most recent month is $t = 0$ and $t = n - 1$ is the oldest month.

$$\bar{x} = \frac{\sum_{t=0}^{n-1} w(t) \times x_t}{\sum_{t=0}^{n-1} w(t)}$$

Figure 3.8: Weighted arithmetic mean

3.2.3 Five Model System

Using weighted averages is only half the story, the system needs to choose appropriate weights for each transaction and the weights chosen will have a different suitability depending on the spending patterns of the user. During initial research on weighted averages, it was observed that due to the variety of spending patterns caused by users different spending habits, there was not a 'one fits all' solution to weighting. For this reason five different weighting functions were selected and when making a prediction the application selects the weighting algorithm most appropriate for the user.

The five weighting functions were selected from a set of eight (shown in Fig. 3.9) after experimentation with personal finance data in MatLab. They were selected for significant differences in behaviour. There are four main function types: Exponential relationship $w_x = e^x$, decay $w_x = 1/x + 1$,

power $w_x = x^1$ and static $w_x = 1$. One exponential, power and static were selected, and two examples of decay with a variable to affecting the speed of the decay. It would be possible to include a scaling parameter (decay constant) for each weighting function, leading to adaptive weights and to use a learning algorithm to select the optimal value for that parameter, this is discussed in Section 7.3.2.

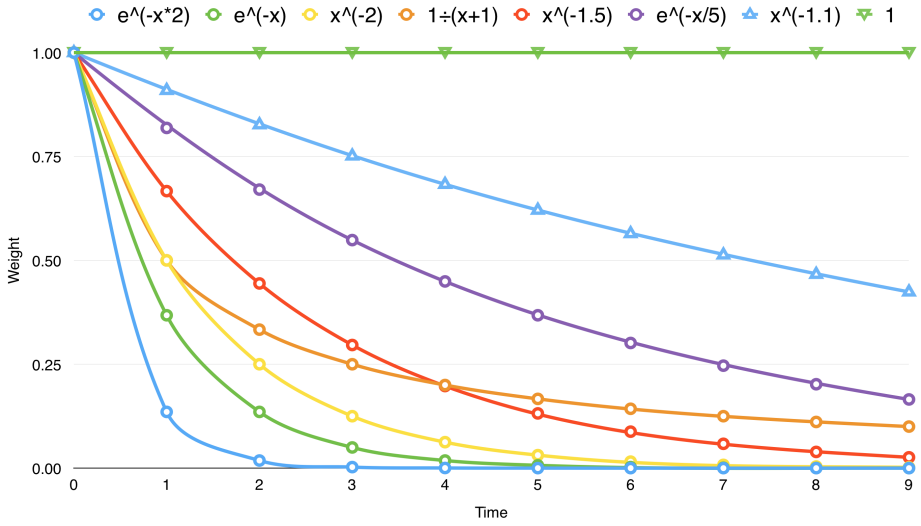

Figure 3.9: The original eight prototype weighting functions

To select the best fit weighting function for each user, the system splits the complete months[4] from the users transaction history into two parts. Training contains 75% of months, starting with the oldest and the remaining 25% (the most recent) is used for testing. If less than four months are available the most recent month is used for testing and the remainder for training. In the case where between zero and two months are available the application falls back on a simple average.

Having split the data the application loops through all the weighting functions available calculating the weighted average of the testing data and evaluating the mean absolute error (Fig. 3.10) on the training data, by comparing the prediction to the actual value. The function with the least absolute error is best fit to the users overall spending pattern, and so it is selected.

In testing it was discovered that users spending money in similar cate-

[4]Months that have passed fully

gories often best suited the same weighting model. Upon further investigation it was discovered that by finding the best weighting model per category in addition to user, on average the absolute error was less. For this reason the most recent implementation of the five model system goes through a users spending in each category, selects the best model for each and uses that to make the prediction in the category. Further research could be done into different levels of modelling and the effect of this level on overfitting, this is discussed in Section 7.3.1.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i|$$

Figure 3.10: Mean absolute error where f_i is the prediction and y_i is the true value

Q16: Show how the one with the least error is selected?

3.2.4 Confidence

As the prediction from the Markov Chain Model is based on probabilities it is unstable. To account for this, when making a prediction the application repeats the process of reading from the MCM up to $10,000$ times in one second. The repetition of the reading is used to produce a confidence level of the final prediction once all the results are combined, which is displayed to the user. The confidence level is displayed as a plus/minus value next to the prediction and gives an indication of how sure the application is.

Assuming the results follow a normal distribution the 95% confidence interval is calculated by Fig. 3.11 where $\bar{x}$ is the arithmetic mean of the predictions and x_i is the value of prediction i.

$$\bar{x} \pm z\frac{\sqrt{\frac{1}{n}\sum_{i=0}^{n}(x_i - \bar{x})^2}}{\sqrt{n}}$$

Figure 3.11: Confidence Interval formula

3.3 Security Considerations

Strong security is expected of this project, the design considers possible attack vectors and takes steps to prevent or reduce the effectiveness of those attacks.

3.3.1 Account Hijacking

Over HTTP information sent between the users web browser and the remote server in plain text and can easily be read using a man-in-the-middle attack. This risk is compounded if accessing the Internet via an an unencrypted WiFi connection[5] which would allow anyone in the local area to 'sniff' the information by simply scanning for capturing the transmitted packet.

If a website involves authentication, this becomes a serious security risk. Authentication is usually performed by sending the username and password in plain text to a remote server, which is validated and if issuing the user with a session cookie. A potential attacker could observe and store these usernames and passwords, which is why commonly websites such as Facebook use HTTPS for the login, ensuring the usernames and passwords are sent encrypted to the server.

If a website falls back to HTTP following authentication, the risk of unauthorised account access is still prevalent. In order to identify to the server which user the browser is authenticated as the browser sends their session cookie to the remote server with each request. Although the attacker can't observe the username and password, the session cookie is being sent in plain text with every request, and the attacker can perform a session hijack by downloading the content of that cookie to their local machine (Fig. 3.12) and then sending it to the remote server with their HTTP request 'proving' they are the user and gaining access to their account (Fig. 3.13) [25]. Firesheep was a proof of concept plugin for Firefox released in 2010 that demonstrated this vulnerability, showing that session hijacking could be performed on popular sites including Google, Facebook, Twitter, Dropbox and Flickr, which until recently were not using sitewide SSL to protect their cookies [26], [27]. More recently 'WhatsApp Sniffer', available on Google Play until May 2012 was able to display messages addressed to other WhatsApp users connected to the same network using this technique [28].

For this reason all of the project uses HTTPS, marks cookies as HTTPS

[5] Common at Coffee Shops and Universities

only[6], and force redirects users to the HTTPS version if they attempt to access via HTTP. This ensures user data is sent encrypted end to end and cannot be intercepted, preventing access to their authentication details or session cookie. In addition cookies are marked at HttpOnly, ensuring access via non-HTTPS methods such as client side javascript is not possible. This means that even if a users browser is infected with a malicious script for example using XSS (see 3.3.4), the contents of the cookie cannot be read.

Figure 3.12: Obtaining a users cookie using a MitM attack or Sniffing [25]

Figure 3.13: Performing a session hijack using another users cookie [25]

[6]Using the Secure attribute

3.3.2 Password Security

A common cracking technique used to gain unauthorised access to a user account is a known as a brute force attack. If an attacker knows a particular users username they can perform targeted guessing guessing of the password by enumerating through all possibilities. A websites ability to resist this kind of attack is called the 'password guessing resistance'. It is for this reason that many websites, enforce password rules in an attempt to increase the number of possible combinations for a password, or the entropy [29].

Shannon Entropy can be used estimate the strength of a passwords resistance to this kind of attack. The entropy is calculated using $H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i)$ where $p(x_i)$ is the probability of the value x occurring [30]. The paper suggests a predefined set of rules for estimating entropy based on Shannon's work studying English text, however other papers found that using this predefined set of rules was not a valid measure of password strength [31].

The project uses Shannon's original equation, calculating the probability of guessing each individual character using a formula that takes into account using a larger character set (such as numbers and symbols) decreases the likely hood of successfully guessing the next character and rejecting a password if it falls below a predefined entropy. Enforcing an entropy threshold rather than enforcing a set of restricting 'password rules', was preferred as it gives the user more flexibility hopefully avoiding the annoyance of rules and increases the search space of the passwords. A very long alphanumeric password such as 'correct horse battery staple' would be just as valid as a short password containing numbers and symbols such as '6?@7a?Y5R='.

As part of a brute force attack, the attacker may use a dictionary of popular passwords to reduce the testing space before attempting an exhaustion attack. In order to reduce the effectiveness of this kind of attack the project test's any user provided password against a dictionary of at least 50,000 common passwords sourced from password cracking resources [30].

In addition, to limit the overall effectiveness of brute force attacks, the website rate limits login attempts. If a user attempts to login more than 5 times within one minute, they must wait thirty seconds before they are able to attempt to login again. Rate limiting was chosen over CAPCHA[7] found on many websites as CAPCHA's slow down users, are often illegible and visual CAPCHA's can prevent visually impaired users from accessing the website [32], [33]. Additionally CAPCHA's can now be solved automatically with a very high success rate using computer vision techniques, and these

[7] Completely Automated Public Turing test to tell Computers and Humans Apart

techniques are already being integrated into brute force software available online [34]–[37].

3.3.3 Database Storage

Unfortunately, it's common for the contents of a websites database to be leaked, whether by an administrator of the website or using other techniques such as SQL injection [38], [39]. It's important to think about the security of the data held within the database, as well as the security of the front end.

This project uses three main techniques to help ensure the security of the users information stored in the database.

Passwords

If a users password is stored in a reversible state, whether that is plain text or encrypted and the database is leaked that users account can be accessed and in addition, the data can be to attack other websites and compromise users using the same username and password. Encryption is equivalent to plain text in that, it is simply a case of finding the encryption key, which is very possible using brute force and all the data is in plain text. This is why standard security practice is to hash passwords. The only way to 'decrypt' a hash is to guess the original input by brute force and see if that matches the output. However crackers often make use of Rainbow tables, collections of precalculated hashes and the input used to create them, allowing an attacker to simply lookup a hash in their database to get the result rather than enumerating all the possibilities [40]. To reduce the effectiveness of this attack method, 'salting' is commonly used. Salting involves adding a random collection of numbers and letters to each password. This means that the generated hash is dependent on both the users password and the salt, and therefore a Rainbow Table would need to be generated for each user, cancelling out the advantage of precalculation and making the tables useless.

Having decided to hash and salt passwords, different hashing functions were investigated. Traditionally functions such as MD5, SHA1 and SHA256 are used to perform the hashing, however due to advances in modern computer equipment it is possible to generate these at an incredibly fast rate, reducing the time taken to brute force a hash. Using a deliberately slow hashing function is designed avoid this problem. Blowfish written by Bruce Schneier is commonly suggested, as it is designed as a computationally expensive operation [41] . This was evaluated with a simple test, calculating

as many hashes per possible in one second on the server hosting the project. Table 3.3.3, shows the results, which found that on average Blowfish took significantly longer to generate each hash[8].

For this reason the project salts all passwords and hashes them using Blowfish.

	Hashes Per Second		
	Average	Standard Deviation	95% Confidence Interval
MD5	2,296,667	12,923	±8010
SHA1	1,869,725	14,783	±9162
BLOWFISH	17	0	±0

Table 3.3: Average number of hashes completed per second on a 2.7Ghz i7

Personally Identifiable Data

Another concern is personally identifiable data being leaked. In an attempt to avoid this the application encrypts all information stored in the user table, that is needed at a later date using the AES128 encryption standard. This standard was selected for the project as was endorsed by the U.S. National Institute of Standards and Technology, when outlined by NIST in 2001 and has become the "encryption standard for commercial transactions in the private sector" [42], [43].

Hashing of Usernames

In addition to the encrypting data needed at a later date the username of each user is hashed so it is only known to the person using that account. In the rare case that any of the passwords were brute forced, the relevant username would also need to be brute forced in order to attempt a login or use the details on another website.

3.3.4 Other

Other attack vectors including SQL injection and cross-site scripting (XSS) were also considered.

It was decided that the project would use a prepared statements to reduce the risk of SQL injection. By sending the query followed by the

[8]The code used to perform the test can be found in Appendix B

parameters as literal values the database server would not interpret them as an executable portion of SQL and attacks, relying on escaping SQL such as ' OR 1 are prevented.

In order to mitigate the possibility of XSS the project will need escape all content before displaying it to the user or saving to the database. It was decided that the project would use a templating language that escaped output by default, requiring the output be explicitly marked to avoid escaping. By escaping all content before displaying it to the user a maliciously crafted piece of text such as <**script**>alert(1);</**script**> would be sent to the users browser as <script>alert(1);</script> and not interpreted as a script.

3.4 Technical Design

Q17: Do I actually need a technical design section? The plan was to include some class diagrams of the project. Leaving this for now.

3.4.1 Object Orientation

P4: Patterns used

3.4.2 Domain Class

P5: Diagram and list of ALL the objects

3.4.3 UI Design

P6: Add photos showing the evolution of the UI

3.4.4 Database Class

P7: Diagram of database

3.4.5 External Software and Frameworks

P8: What frameworks were used

3.4.6 Platform

P9: I decided to use a browser based web application instead of a desktop application

Chapter 4

Implementation

P10: Limited to programming languages only

4.1 Security

4.2 Prediction

P11: All the steps that were needed for the prediction

4.2.1 Named Entity Resolution

P12: Use of MySQL to find similar, how that works, alternatives

4.2.2 Suggestions

P13: How it uses the above, how the whole user/global thing works

4.2.3 Markov Chain

P14: Why I chose first order markov chains, alternatives to that etc...

4.2.4 Weighted Averages

P15: How does that work, why is it good?

4.2.5 5 Model System

P16: How was is a model chosen for the user?

Chapter 5

Results

P17: Print Screen Step Through of the Main Stuff the Application Does

5.1 System Walk-through

5.2 Responsive Web Design

A key feature of the application is being able to access it at any time from any device, particularly when taking into account the rapid increase in the use of mobile devices. Interacting with a website on a smartphone or tablet is not the same as interacting using a computer, due to the smaller screen size and use of touch over a mouse.

Forbes reported that 24% of their 2013 website visits came from mobiles and 13% from tablets, down from a total of 15% in 2012. particularly with the high percentage of website visits coming from mobile devices [44].

In order to ensure the project is accessible from a variety of different devices the core UI uses Responsive Web Design (RWD) to layout the website differently depending on the size of the device to ensure an optimal viewing experience.

The differences depending on the device are highlighted in Figs. 5.1-5.4.

Figure 5.1: Layout on a standard laptop

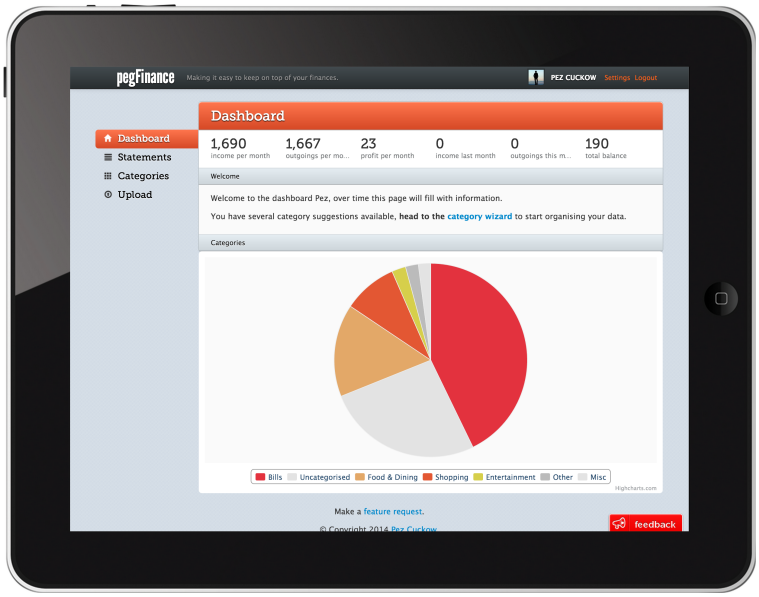

Figure 5.2: Layout on a tablet in landscape

Figure 5.3: Layout on a tablet in portrait

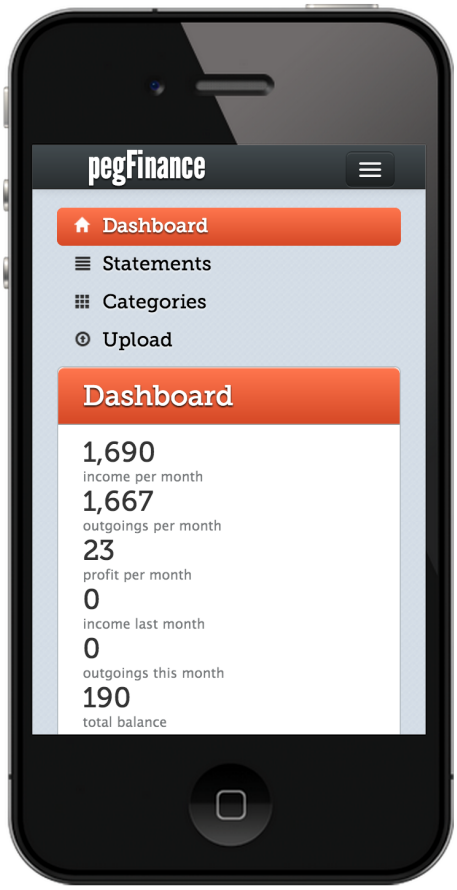

Figure 5.4: Layout on a smaller smartphone

Chapter 6

Testing and Evaluation

6.1 Unit Tests

P18: Not sure if this fits here!

6.2 Evaluation of the system

P19: How did I check that it was working? Testing with personal data, users trying it and running tests of everything

Need figures representing accuracy, error, etc

6.3 User feedback

P20: User surveys to check what they wanted

Chapter 7

Conclusions

7.1 Key Things Learnt

P21: How to do prediction, password entropy, other stuff

7.2 Does the system do what I set out to do?

P22: Does it make accurate predictions (summarise the evaluation section)

7.3 Further research

P23: Would like to classify users, would like to take into account global spending patterns

7.3.1 Overfitting models

P24: Include why having too many models is bad and explore those a bit

7.3.2 Using Learning to Select a Scaling Parameter

P25: How learning algorithms could be used to fit the weights even more (custom one for each user!)

7.4 Limitations of research?

P26: Limited set of testers, just students and friends, add more stuff and then go global.

Bibliography

[1] C. Gore, "The global recession of 2009 in a long-term development perspective," *Journal of International Development*, vol. 22, no. 6, pp. 714–738, 2010.

[2] A. Barnard, "The economic position of households, q1 2012," *Office Of National Statistics*, 2012. [Online]. Available: http://www.ons.gov.uk/ons/rel/hsa/the-economic-position-of-households/q1-2012/art---the-economic-position-of-households--q1-2012.html (visited on 04/08/2014).

[3] The Wall Street Journal. (2013). The experts: is creating a personal budget a good idea?

[4] Think Local Act Personal. (2013). 2nd national personal budget survey. L. Boyd, Ed., [Online]. Available: http://www.thinklocalactpersonal.org.uk/Latest/Resource/?cid=9503.

[5] BBC News. (2010). Debit card spending in uk overtakes cash for first time, [Online]. Available: http://www.bbc.co.uk/news/business-11901953 (visited on 04/09/2014).

[6] Lloyds Bank. (2014). Lloyds bank - private banking - internet banking, [Online]. Available: http://www.lloydsbank.com/private-banking/contact-us/internet-banking.asp (visited on 04/20/2014).

[7] Mint. (2014). What is mint, Intuit, Inc, [Online]. Available: https://www.mint.com/what-is-mint/ (visited on 04/20/2014).

[8] Lloyds Bank, Ed. (2014). Money manager, The free, easy way to keep track of your money, [Online]. Available: http://www.lloydsbank.com/online-banking/benefits-online-banking/money-manager.asp (visited on 04/09/2014).

[9] Money Watch. (2011). Lloyds tsb money manager, [Online]. Available: http://money-watch.co.uk/7992/lloyds-tsb-money-manager (visited on 04/18/2014).

[10] 'Deals' *et al.* (2011). Anyone tried money manager at lloyds tsb? MoneySavingExport.com Forum, [Online]. Available: `http://forums.moneysavingexpert.com/showthread.php?t=3114854` (visited on 04/21/2014).

[11] Stack Overflow Users. (2010). Banking api/protocol, Stack Overflow, [Online]. Available: `http://stackoverflow.com/questions/3469628/banking-api-protocol` (visited on 04/19/2014).

[12] ——, (2010). Is there an api to get bank transaction and bank balance? Stack Overflow, [Online]. Available: `http://stackoverflow.com/questions/7269668/is-there-an-api-to-get-bank-transaction-and-bank-balance` (visited on 04/20/2014).

[13] K. Purcell, "Half of adult cell phone owners have apps on their phones," *Pew Internet & American Life Project*, 2011.

[14] iTunes. (2013). Top 10 apps - finance, Apple, [Online]. Available: `http://www.apple.com/euro/itunes/charts/apps/top10appstorefinance.html` (visited on 04/20/2014).

[15] SpendeeApp, Ed. (2014). Spendee - see where your money goes, [Online]. Available: `http://www.spendeeapp.com` (visited on 04/09/2014).

[16] S. Flückiger. (2013). Budgt, [Online]. Available: `http://www.spendeeapp.com` (visited on 04/09/2014).

[17] BlueTags, Ed. (2014). Pocket expense, [Online]. Available: `http://www.bluetgs.com/Home.aspx` (visited on 04/09/2014).

[18] P. E. Pfeifer and R. L. Carraway, "Modeling customer relationships as markov chains," *Journal of Interactive Marketing*,

[19] V. Singh, L. Freeman, B. Lepri, and A. Pentland, "Predicting spending behavior using socio-mobile features," in *Social Computing (SocialCom), 2013 International Conference on*, 2013, pp. 174–179. DOI: `10.1109/SocialCom.2013.33`.

[20] S. Dash, "A comparative study of moving averages: simple, weighted and exponential," Tech. Rep., 2012. [Online]. Available: `https://www.tradestation.com/education/labs/analysis-concepts/a-comparative-study-of-moving-averages` (visited on 04/17/2014).

[21] J. Filliben *et al.*, "Introduction to time series analysis," in *NIST/SEMTECH Handbook of Statistical Methods*, National Institute of Standards and Technology, 2003.

[22] P. S. Kalekar, "Time series forecasting using holt-winters exponential smoothing," *Kanwal Rekhi School of Information Technology*, vol. 4329008, pp. 1–13, 2004.

[23] Quicken Support. (). Quicken interchange format (qif) files, [Online]. Available: `http://web.archive.org/web/20100203121050/http://web.intuit.com/support/quicken/docs/d_qif.html`.

[24] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," in *Soviet physics doklady*, vol. 10, 1966, p. 707.

[25] J. Williams, A. Jex, *et al.* (2011). Session hijacking attack, Open Web Application Security Project (OWASP), [Online]. Available: `https://www.owasp.org/index.php/Session_hijacking_attack` (visited on 04/22/2014).

[26] E. Butler, "Hey web 2.0: start protecting user privacy instead of pretending to," GitHub, Presented at Toorcon 12 San Diego, 2010. [Online]. Available: `http://codebutler.github.io/firesheep/tc12/` (visited on 04/19/2014).

[27] ——, (2014). Codebutler/firesheep, GitHub, [Online]. Available: `https://github.com/codebutler/firesheep` (visited on 04/22/2014).

[28] D. Walker-Morgan. (2012). Sniffer tool displays other people's whatsapp messages: news and features, The H Security, [Online]. Available: `http://www.h-online.com/news/item/Sniffer-tool-displays-other-people-s-WhatsApp-messages-1574382.html` (visited on 04/21/2014).

[29] K. Helkala and E. Snekkenes, "A method for ranking authentication products," in *Proceedings of the Second International Symposium on Human Aspects of Information Security & Assurance*, 2008, ISBN: 9781841021898.

[30] W. Burr, D. Dodson, E. Newton, R. Perlner, W. Polk, S. Gupta, and E. Nabbus, "Electronic authentication guideline," *National Institute of Standards and Technology*, 2013, Special Publication 800-63-2.

[31] M. Weir, S. Aggarwal, M. Collins, and H. Stern, "Testing metrics for password creation policies by attacking large sets of revealed passwords," in *Proceedings of the 17th ACM Conference on Computer and Communications Security*, ser. CCS '10, Chicago, Illinois, USA: ACM, 2010, pp. 162–175, ISBN: 978-1-4503-0245-6. DOI: `10.1145/1866307.1866327`. [Online]. Available: `http://doi.acm.org/10.1145/1866307.1866327`.

[32] M. May, "Inaccessibility of captcha," *Alternatives to Visual Turing Tests on the Web, W3C, Editor*, W3C, 2005.

[33] S. Hegarty. (2012). The evolution of those annoying online security tests, BBC News, [Online]. Available: `http://www.bbc.co.uk/news/magazine-18367017` (visited on 04/20/2014).

[34] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet, "Multi-digit number recognition from street view imagery using deep convolutional neural networks," *CoRR*, vol. abs/1312.6082, 2013.

[35] 9kw.eu. (2014). Captcha service for the user - captcha solver, [Online]. Available: `http://www.9kw.eu/index.html` (visited on 04/22/2014).

[36] D. Danchev. (2014). Google's recaptcha under automatic fire from a newly launched recaptcha-solving/breaking service, Webroot Threat Blog, [Online]. Available: `http://www.webroot.com/blog/2014/01/21/googles-recaptcha-automatic-fire-newly-launched-recaptcha-solving-breaking-service/` (visited on 04/21/2014).

[37] I. Savinkin. (2013). 8 best captcha solvers, [Online]. Available: `http://scraping.pro/8-best-captcha-solving-services-and-tools/` (visited on 04/20/2014).

[38] (2012). Linkedin passwords leaked by hackers, BBC News, [Online]. Available: `http://www.bbc.co.uk/news/technology-18338956` (visited on 04/19/2014).

[39] D. Chechik. (2013). Look what i found: moar pony! TrustWave Security Firm, [Online]. Available: `http://web.archive.org/web/20131208203540/http://blog.spiderlabs.com/2013/12/look-what-i-found-moar-pony.html` (visited on 04/22/2014).

[40] M. W. Jorgensen. (2012). Free rainbow tables - distributed rainbow table generation, Distributed Rainbow Table Project, [Online]. Available: `https://www.freerainbowtables.com/tables/` (visited on 04/22/2014).

[41] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (blowfish)," in *Fast Software Encryption*, Springer, 1994, pp. 191–204.

[42] National Institute Of Standards And Technology, "Announcing the advanced encryption standard (aes)," 2001.

[43] R. M. Stair and G. W. Reynolds, *Principles of Information Systems: a Managerial Approach*, 9th ed. South-Western, 2009, p. 245, ISBN: 0324665288.

[44] J. Steimle. (2013). Why your business needs a responsive website before 2014, [Online]. Available: `http://www.forbes.com/sites/joshsteimle/2013/11/08/why-your-business-needs-a-responsive-website-before-2014/` (visited on 04/06/2014).

Appendices

Appendix A

Survey

Informal survey of 12 CS students in the third year lab.
Questions:

1. Do you currently make a budget?
2. Do you stick to that budget?
3. Do you find your budget has a 'positive' impact?

Table A.1: Survey Results

	Question		
Answer	1.	2.	3.
yes	5	1	3
no	7	4	4
n/a	0	7	5

Appendix B

Hashing Test

Implemented in PHP, test was run on a 2.7 Ghz Intel Core i7 with 16 Gb of 1600 Mhz DDR3 RAM.

```
<?php

$timeTarget = 1;
ini_set('memory_limit', '-1');

$strings = array();
for($i = 0; $i < 3000000; $i++) {
        $strings[$i] = randomString(16);
}

$hashingFunctions = array('MD5', 'SHA1', 'BCRYPT');

foreach($hashingFunctions as $hash) {
  for($i = 0; $i < 10; $i++) {
        $start = microtime(true);
        $count = 0;
        do {
      $count++;

      if($hash == $hashingFunctions[0])
        md5($strings[$count]);
      elseif($hash == $hashingFunctions[1])
        sha1($strings[$count]);
      elseif($hash == $hashingFunctions[2])
        password_hash($strings[$count], PASSWORD_BCRYPT);
      else
        echo "ERROR: Unknown function";
```

```
        $end = microtime(true);
          } while (($end - $start) < $timeTarget);
          echo "$hash\t$count\n";
    }
}

function randomString($length = 10) {
    $characters = '0123456789
       abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
       ';
    $randomString = '';
    for ($i = 0; $i < $length; $i++) {
        $randomString .= $characters[rand(0, strlen(
           $characters) - 1)];
    }
    return $randomString;
}
```