University of Southampton
School of Electronics and Computer Science

A project report submitted for the award of
MEng Computer Science/MEng Software Engineering

By:
Peter West (pw9g09)
James Skinner (js40g09)
Adam Thomas (ajt1g09)
Dexter Lowe (dl10g09)
Tom Smith (taes1g09)

Project supervisors:
Dr David Tarrant (dt2)
Dr Leslie Carr (lac)

Second examiner:
Dr Mike Wald (mw)

Your4.tv - A TV streaming Service Designed for Recommendations of Programmes and Adverts

A project for the customer
inqb8r

Date: December 6, 2012

Contents

1 Acknowledgments

Thanks to our supervisors, David Tarrant and Les Carr, whose guidance and provision of resources have ensured this.

Thanks to Peter Wood and Nick Culley from Inqb8r, developers of Project4, who provided feedback for our work and ensured that required resources were in place for the duration of the project. We would particularly like to thank Peter Wood for making himself available for queries and advice for the duration of the project.

We'd also like to thank Metabroadcast, in particular Jonathan Tweed, who have provided us with access to their Atlas API (atlasapi.org), a REST service for accessing EPG and programme data. This data is in turn provided by Press Association.

Thanks to GroupLens research for hosting MovieLens rating and demographic data, which were used to train the recommendation system. Thanks to TheTVDB for providing an API to programatically find the genres of movies, which was used for classifying new programmes by the recommender.

Finally, we'd like to thank VisualDNA for their support of this project, and their offer to provide assistance and mentoring. We would particularly like to thank the CTO of VisualDNA, Tomas Cervenka, for validating the utility of our project by showing interest in the outcome of this project and extending it further.

2 Introduction

High-speed internet has given us the capability to stream television via our computers and mobile devises, giving huge possibilities into how programmes and adverts may be presented to people.

Historically, TV has been viewed on channels, with fixed schedules and the choice of an increasing number of channels. Channels could eventually be regional, allowing some targeting of programmes (e.g., regional news) and adverts. Faster internet brought the possibility of catch-up TV, allowing people to watch programmes from the past any time they wanted.

Current Internet TV streaming services - such as BBC iPlayer[1] and 4oD[2] - allow users to watch either Live TV or programmes that have been shown previously. This can be done via a web-interface on a computer. iPlayer and 4oD both have apps available for other platforms, such as the Sony PS3 and Android smart phones[3].

The customer, inqb8r, runs a service called Project4, which similar to above, is a TV streaming service which also offers previously shown programmes. The service is aimed at students (specifically designed to be used on JA.NET) and shows Channel 4, four of its sister channels - E4, More4, 4Music and Film4 - and its own channel, studentTV.

However, the burden is on the user to choose a programme [cite] in the

Additionally, adverts presented on 4od are often unrelated to the programme[cite] and can be repeated often in the same break. There is no evidence to suggest that these adverts are targeted towards users based on their viewing behaviour[check], suggesting that adverts may not be complimenting the interests of the viewer.

In this report, we study how our product, your4.tv, allows adverts to be targeted towards users, and in doing so, use the time more effectively to show adverts that users may be interested in. Users are given a TV channel which is tailored to them - they do not need to choose a programme, it is targeted towards them

We first perform a survey, in which we asked over 60 participants how they consume adverts in different forms of media.

2.1 The Team

The team consists of five people, all completing Part 4 of Computer Science MEng / Software Engineering MEng. Since the team had previously worked together on a scripting project, we already knew each others key strengths and weaknesses, allowing us to allocate tasks effectively from the very start of the project. The team members are:

[1] http://www.bbc.co.uk/iplayer/

[2] http://www.channel4.com/programmes/4od

[3] http://apps.channel4.com/view-all-apps/, https://play.google.com/store/apps/details?id=bbc.iplayer.android

1. *Peter West* (team leader): knowledgeable in user-interface design; strong experience with JavaScript and PHP.

2. *Dexter Lowe*: worked with enterprise systems; strong experience with Java; very familiar with Eclipse.

3. *Jim Skinner*: knowledgeable in machine-learning; strong experience with Python.

4. *Tom Smith*: experienced with data visualisation; strong experience with JavaScript.

5. *Adam Thomas*: experienced web developer; strong experience with JavaScript and PHP.

3 Motivation

3.1 Live Streaming

The iPad and other tablet devices have for some time had apps like 4od available which allows on demand streaming of recorded Television content but there has been very few attempts by the major TV channels to bring live TV streaming to these devices. Channel 4 provided a Paralympics app[4] which promised live commentary. Unfortunately this only provided short videos (lacking sound) and text commentary with images much like a Twitter stream or blog. So we hoped with this project to bring live streaming media to the iPad.

3.2 Channel Hopping

These days whether a user has Freeview, Sky or even just standard terrestrial television they have a multitude of channels to choose from. Even under the banner of what were once single channels like Channel 4 or Sky rest a considerable number of actual channels. Channel 4 currently has 5 major channels under its name[5]. This means that users who want to watch television without a particular show in mind are first required to hunt through a considerable amount of data. While modern systems allow EPG data to be displayed and searched on the TV the fact that TV is live means that by the time they have located a desirable show it may already have started.

Some more advanced systems allow time-shifting such as TiVo allowing users to pause and rewind live TV. It achieves this by recording a rolling 30 minutes of the last 2 channels the user was watching[6]. So even these systems which are considered advanced have their limitations.

Having considered these limitations and our resources, which included all the Channel 4 stream sources, we approached the issue of missing the start, or locating an engaging programme in a different way. We decided to provide the user with only one channel which would be generated when the user accesses our system by using what is known about them to build a playlist from all the live channels our system knows of, if there is nothing the user would find appealing on, or they have missed the start it will fall back to adding pre-recorded shows to the playlist until something appropriate started.

3.3 Adverts

Another common issue with streaming media is the adverts. These raise money for the broadcaster and importantly are embedded in the stream at the source. This means that as much as users may request it there is no way to skip an ad-break. This is because the rest of the programme has not yet been broadcast so there is no media to return to. Having considered how users consume adverts we concluded that

[4] `https://itunes.apple.com/gb/app/channel-4-paralympics/id554157549`

[5] Channel 4, E4, More4, 4Music and Film4

[6] `http://www.mytivo.com.au/whatistivo/tivois/pauserewind/`

adverts are necessary but typically users do not currently enjoy consuming adverts.

3.3.1 Advert Relevance and Control

We decided that our system should maximise both user enjoyment and advertiser utility. By hypothesising that users would be more likely to enjoy an advert if it were relevant to them, we determined that our product should provide personally recommended adverts targeted to each user. This would make users seeing the adverts more interested improving the user experience whilst simultaneously providing advertisers with greater utility per impression by maximising the likelihood that a viewer will be interested in their product. Additionally we decided that a skip button allowing the user to blacklist adverts would improve the user experience by allowing users to control which adverts they see and if they provide a reason why they skipped, this is valuable feedback for advertisers to allow them to improve future adverts.

3.3.2 Interaction and Engagement

We hypothesised that if a user is less engaged by an advert then they may ignore it or even leave the room giving advertisers little or no utility and only annoying the users. By realising that current adverts do not take advantage of the interactive platform provided by tablet devices we predicted that by providing the adverts on an interactive plane we could maximise user attention by providing useful or fun interactive content to reinforce the advert. Furthermore by keeping the user focussed on the advert even if they do not pay direct attention to the content the interaction provides another chance for users to remember the advert and indeed provides greater benefit in advert repetition as users may interact with the advert even if they have seen the content before, if it is entertaining or useful. for example a showings finder embedded in an advert for a film. When the user first sees the advert they may not be looking for a film, but later at the prompting of the advert they then have an instant interaction which may allow them to discover a showing at a local cinema starting soon thereby eliminating the search on a cinema's website and maximising the chance that the viewer would see the advertised film over another which may have been shown on the cinema site.

3.4 Personal Focus

iPads and indeed most portable devices such as smart phones, tablets and laptops are typically used by a single person. Because of this many personal systems are already tightly integrated such as social media, like Twitter or Facebook. Additionally while a standard television may be watched by a whole family or more anything watched on a tablet will likely be viewed by a single person, the owner. This means that adverts can be targeted individually to members of a family rather than trying to cover the full spectrum, additionally social interaction such as the action of liking something on Facebook can be easily integrated into adverts allowing users to save information on their personal profiles for later referral.

3.5 Summary

In summary our project improves the user experience by making live media personal, targeting both shows and adverts to each user as well as integrating seamlessly with the social experience. Additionally it reduces frustration at missed shows as these are transparently skipped in favour of something the user can watch from the start. In addition it provides a zero-interaction discovery system to remove the time consuming hunt for something to watch and perhaps aid users in discovering new and exciting content. Moreover it improves the experience of consuming adverts by providing users with interactivity, personally targeted adverts and the ability to control which adverts they are shown.

4 Background

4.1 Television Broadcasting & Online Video

Typically, live television channels and streaming services display programme content with video advertisements apportioned amongst this content at both regular intervals, and between distinguishable items of content. Advertisements typically vary in subject and style in a number of ways in order to increase the efficiency of the advert by maximising the captured audience who fit the target audience for the given advert.

Television broadcasts typically have very low granularity - as they are, by their nature, viewed simultaneously by everyone who is watching a particular channel. This leaves a limited amount of factors for advertisers to maximise their efficiency. These factors include the time of day; the adjacently showing content; the typical target audience of the channel; and location at a very broad scope (regions, e.g. East Midlands).

With television advertising costs increasing and an ever emerging online advertising market, broadcasters are noting dropping TV ad effectiveness levels. In 2012, 53% of advertisers cited targeting as the most valuable aspect of online video. Online video typically boasts significantly more factors to target upon, including browsing history (behavioural data), user demographics and interests amongst others. On the contrary, in 2012 advertisers expressed a 41% drop compared to the previous year in the important of reach (size of audience) which is attributed to an increase comfort that online video has a large enough audience, which in turn emphasises the importance on targeting specific audiences. This is emphasised by 35% of advertisers agreeing that demographics data is the most valuable form of targeting - the largest targeting type in comparison to others such as contextual and geographic. (**?**)

Google, an online advertising leader, has reinforced this shift in ideology towards targeted video by supporting a "Cost Per View" (CPV) pricing strategy. Cost per view differs from cost per impression (CPI) in that pricing is based upon users who choose to watch a video instead of video that is simply passively playing. This strategy can also incorporate advert skipping. YouTube, a Google product, offers "TrueView" which only charges an advertiser if the viewer reaches the end of the advert without using a skip button (**?**). This is an attractive option for advertisers, as TV advertising does not return any metrics to identify who out of the audience reach is engaged by the advertisement.

Television advertisers are further concerned by the apparent loss of engagement in this form of passive advertising. Nielson reports a growing trend in which viewers choose the best medium available (television, online and mobile) available to them to consume content according to the viewers perception of the quality of the content and its availability. 31% of internet activity occurs when consumers are also watching television. A growing number of households are adopting Digital Video Recorder (DVR) technology, with 29% of US homes able to time shift television which allows viewing only content the viewer is interested in, including the skipping of adverts. However, live TV viewing still accounts for the majority of video consumption, but

is growing at a slower rate than the richer online platforms. (**?**)

The introduction of online video has introduced an interactive plane which can not exist on TV. Online advertisements have long used animated images and flash technology to introduce this level of interactivity to the user. Recent developments in web standards such as HTML5 have given rise to rich video. Rich video allows the inclusion of out of band metadata in videos and the opportunity for viewers to manipulate aspects of the video advert. For example, YouTube[7] allows content owners to annotate their videos with extra clickable information at specific points in time. A study by the advertising network DoubleClick highlighted how this form of user engagement can increase advert efficiency, revealing "Rich Media with Video" to be the best choice for brand awareness, brand favour and purchase intent (**?**).

4.1.1 Mobile Video

Smartphone usage in the UK has risen year on year. As of 2012 51% of the UK population owns a smartphone device. As mobile advertising is a strong growth industry, it is of interest to content owners to expose their products and services on this platform. By the very nature of the devices on the market, more users are relying on their smartphone outside of the home in order to research and purchase products; watch video; communicate and to stay informed. 66% of smartphone users watch video on their device and 84% of users notice mobile ads while using their phone. (**?**) These two factors together present an opportunity for content owners to engage users with their services by employing the use of rich video with advertising. With over 31% of advertisers considering mobile video as the area in which advertising spend will increase most (**?**), TV broadcasters need to shift towards a more engaging platform.

4.2 Inqb8r

Inqb8r[8] is a UK based business which provides a platform for content owners and advertisers alike to expose their own products and services to a university student focused audience. Inqb8r also provides content creation services to their customers in order to aid them in maximising their contact with students.

4.2.1 Project4

Project4 is an addressable advertising service

Younger viewers in the student age range (18-24) also appear to watch online video at a higher rate than others...

4.3 Interactive Advertising

Advertising has traditionally been static imagery or pre-recorded video with, particularly in the case of TV advertising, a broad target audience, coarse targeting and

[7] http://www.youtube.com
[8] http://inqb8r.tv

minimal feedback, with all information flowing from the advertiser to the audience. The rise of internet media streaming services has suddenly added two new possibilities: the possibility of an information flow from the audience back to the advertiser, and the possibility of a user on a device such as a laptop to easily interact with adverts. This plane of interactivity allows for advertisers to design fun, memorable and informative interactions into their adverts, and collect user information through the audience→ advertiser information flow.

4.4 Programme Recommendation

Recommendation systems are a mature area of research, with a huge variety of implementations existing to back up a large theoretical base. Recommendation systems are widely used in internet streaming media services such as YouTube (`www.youtube.com`), 4od (`www.channel4.com/programmes/4od`) and iplayer (`www.bbc.co.uk/iplayer`), and are of enormous commercial value, as shown by the US$1,000,000 2009 Netflix prize (**?**) for improving upon the Cinematch algorithm.

By using user feedback, a recommender system may improve by learning what a particular user likes. Asking the user explicitly for ratings on programmes is sufficient to provide this information, but can appear as an additional chore to the user with no immediate apparent reward. This can lead to users being unwilling to provide explicit programme ratings, meaning a recommender system has no information from which to improve. In addition, explicit ratings are prone to biases from user subjectivity, item popularity and rating habits (**?**, p. 304). Collecting ratings implicitly combats this, which may be done using any detectable user interaction assuming it correlates with user preferences, such as percentage user play time of a programme (**?**, p. 305).

Collaborative-filtering based recommendation systems are faced with the cold start problem, which is the problem of the recommender being unable to give high-quality recommendations on items for which sufficient information has not yet been gathered. In a live TV recommendation system, at the point when a TV show is recommended to users, it has not yet been aired and therefore cannot use collaborative data to recommend the programme. After airing, the show may pick up many viewer ratings, but in a system that deals only with live TV, these ratings are now useless as the programme will not be shown and hence not have the chance to be recommended again. Designing recommender systems to deal with the cold start problem is an active area of research, and studies have shown that item-based algorithms outperform SVD-based algorithms in early stages of recommendation(**?**). In addition, collaborative approaches have been combined with content-based approaches to provide an initial source of information(**?**).

Research is being performed into the extension of programme recommenders into advertisement recommenders (**?**). While similar algorithms are used, advert recommenders have a different set of considerations in providing optimal adverts: who is watching, what is being watched, programme popularity and business rules (e.g., the amount paid for the advertisement) (**?**).

5 Planning

The project would be completed over a 10 week period, so it was essential that the project be started promptly. During the first week on the project the team met with the customer (represented by Peter Wood and Nick Culley) and supervisor, Les Carr, the specification that the customer had provided was discussed and the primary tasks of the project identified. This was key to agreeing on a project that would be completed successfully given the time and resource constraints.

Having established a project, the team compiled a brief (see Appendix A) that described the deliverable product and research focus. We also drew up a time plan in the form of a Gantt Chart (see Appendix **??**), which we could refer to throughout the project to ensure we were on track.

5.1 Task Allocation

It was clear that the project could be divided into separate tasks:

1. Video streaming - requires understanding of streaming and use of Java Dexter

2. Programme and advert recommendation - machine learning, use of datasets Jim

3. iPad client - HTML5 video, JavaScript, CSS Peter and Adam

4. Statistics visualisation - JavaScript Tom

5.2 Agile team roles

5.2.1 Leader

Responsible to ensuring that the team work well together by allocating tasks effectively, scheduling meetings and milestones and ensuring that resources are available where they are needed. Must ensure the project remains within the agreed constraints of scope and time limit. Also responsible for keeping the supervisors and customers informed of progress.

5.2.2 Architect

Understands and oversees how the system integrates together to a low level and ensures it is well documented. If a team member has any doubt in the architecture, it is up to the architect to deliver an answer or brief the team if any changes are required. Must oversee areas where subsystems interact. This primarily involved ensuring that internal APIs are agreed, well designed and thoroughly documented.

5.2.3 Lead tester

Is responsible for ensuring that all parts of the system are tested adequately by use of software testing technique and that all testing has been documented. While they may not necessarily perform all testing themselves, they should have confidence that all parts of the system are well tested in order to avoid failure of the system as a whole.

5.2.4 Analyst

Keeps the product on track to the specification agreed with the customer and supervisors. Performs an ongoing risk assessment and informs the architect's designs. Should ensure that all parts of the system are documented with regard to the customer's spec - i.e., how certain parts cover points in the spec. Must ensure that project remains within scope and does not exceed time and resource constraints.

5.2.5 Documentor

Responsible for ensuring that documentation is kept throughout the project. They take minutes in meetings and from other communications and ensure that all documentation is well organised in preparation for final write-up at the end of the project. They should ensure that documentation complements the expected standard of delivery (i.e. the markscheme).

5.3 Meetings

5.3.1 Weekly progress meeting

Tuesday 14:00-16:00 Attendees: Team, Supervisors (every other week), Customers (occasional weeks) This meeting is for discussing progress during the week. It ensures that the team is on-track, and that the project is on schedule. The meeting should start with a briefing from each member on what they've accomplished and how they've documented it.

5.3.2 Weekly developing session

Wednesday 10:00 - 14:00 Attendees: Team This will start with 30 minutes - 1 hour of meeting time to discuss any development or design issues. a 3-4 hour programming session will follow in the Undergraduate labs.

5.3.3 Weekly pair-programming sessions

Flexible times Attendees: Developer pair Team members should ensure they work together for a few hours a week.

5.4 Risk Analysis

Resources unavailable

Network outage

Injury

6 Prestudy

6.1 How People Consume TV Streaming Services

Details about the survey

6.2 Analysis

Survey result analysis

... suggests that the product we propose will improve the ...

7 Design

7.1 System Layers

The user must be able to access the app via a web browser.

It was decided that the TV and programme streams should be delivered to the client via Wowza media server, which would be run on a Windows server. Peter Wood (inqb8r) has had extensive experience with Wowza, and was able promptly to have the server configured to stream, releaving us of any lengthy configuring that might have had to take place with any other system. Additionally, inqb8r were able to provide us with a licensed Wowza server.

LAMP server

7.2 Programme Recommendation

A your4.tv user is presented with a personal channel populated with a list of programmes as recommended by a programme recommendation system. This system only recommends programmes, and is separate from the targetted advert retreiver (described in Section 7.3). At an abstract level, each programme is given a vector $\mathbf{p}$, which describes what the programme is like. The vector $\mathbf{p}$ lies within the programme space $\mathcal{P}$, and is calculated using the function `get_programme_vector` from programme information. The concrete implementation of this used in your4.tv is that $\mathcal{P}$ is 19-dimentional, where each dimention represents one of the genres:

Action and Adventure	*Animation*	*Children*	*Comedy*	*Documentary*
Drama	*Game Show*	*Home and Garden*	*Mini-Series*	*News*
Reality	*Science-Fiction*	*Fantasy*	*Soap*	*Special Interest*
Sport	*Talk Show*	*Western*	*Unclassified*	

which is the full set of allowable genres on The Tvdb[9] plus 'Unclassified', which is reserved for programmes which `get_programme_vector` cannot assign an appropriate vector. To map a programme into this space, `get_programme_vector` uses the supplied programme name to query the Tvdb api[10] for a programme's genres, and returns a binary vector of genre memberships (for each element: 1 if the programme belongs in genre; 0 otherwise). As an example, the programme 'Grand Designs' has genres *Documentary*, *Home and Garden* and *Reality*, so would be assigned the vector $[0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]$.

So that a user may be recommended programmes, each user is given a vector, $\mathbf{u}$, also within the space $\mathcal{P}$, which represents the users ideal programme. The vector is initialised with the function `get_user_vector`, which takes a vector of a users demographic information, $\mathbf{d}$, and returns the user vector, $\mathbf{u}$. The function `get_user_vector` is learned using a set of training data containing mappings between users, their demographics and shows they enjoy. While a user uses your4.tv, their actions influence their user vector. Upon giving a programme a rating, $\mathbf{u}$ is

[9] `http://thetvdb.com/`

[10] `http://thetvdb.com/wiki/index.php/Programmers_API`

moved either towards or away from $\mathbf{p}$ by an amount which depends upon the magnitude of the rating, the initial distance between $\mathbf{u}$ and $\mathbf{p}$, and the learning rate of the recommender which may be tweaked or made inversely proportional to the number of ratings made by the user. The exact change to $\mathbf{u}$, where $\mathbf{u}_i$ is the i^{th} component of vector $\mathbf{u}$, $\mathbf{u}'$ is the new user vector, $0 \leq L \leq 1$ is the learning rate and $-1 \leq r \leq 1$ is the rating given is given by:

$$\mathbf{u}'_i = \mathbf{u}_i + \begin{cases} |\mathbf{p}_i - \mathbf{u}_i| \, Lr \times \frac{\mathbf{p}_i - \mathbf{u}_i}{|\mathbf{p}_i - \mathbf{u}_i|}, & \text{if } r \geq 0 \\ (1 - |\mathbf{p}_i - \mathbf{u}_i|) Lr \times \frac{\mathbf{p}_i - \mathbf{u}_i}{|\mathbf{p}_i - \mathbf{u}_i|}, & \text{otherwise} \end{cases}$$

Because of division by 0 for the case when the user and programme vectors are, on a particular dimension, identical ($\mathbf{p}_i - \mathbf{u}_i = 0$), a special case was required:

$$\mathbf{u}'_i = \mathbf{u}_i + \begin{cases} 0, & \text{if } r \geq 0 \\ \text{sign}(\text{random}() - 0.5) \times Lr, & \text{otherwise} \end{cases}$$

Genres from Tvdb were chosen as a basis for $\mathcal{P}$ instead of the Project4 or Atlas genres, as this solved the problem of finding training data for the learned function `get_user_vector`. No programme viewing data was available using Project4 or Atlas genres, so the MovieLens 1M dataset[11] was downloaded, and the movies mapped to their corresponding Tvdb genres, giving a large dataset of user demographics, movie ratings, and movie vectors within the programme space $\mathcal{P}$ which serve as suitable training data. To map new programmes into the $\mathcal{P}$, `get_programme_vector` looks up the Tvdb genres for a given programme, using these to construct $\mathbf{p}$.

The space $\mathcal{P}$ is, by design, coupled only with the functions `get_user_vector` and `get_programme_vector`. As a result of this, `get_programme_vector` may be modified, changing the structure of $\mathcal{P}$, as long as `get_user_vector` is re-learnt using the new programme representation. This is desirable, as programme representations better suited to recommendation are possible; i.e., more similar programmes are closer together, more different programmes are further apart, and similar programmes are not given identical representations. Possible ways of accomplishing this are discussed in section 10.3.1.

7.3 Targeted Adverts

Targetted adverts are shown in your4.tv during programme advert breaks, replacing the adverts which would otherwise appear. Users are able to skip an advert, prompting a box asking the reason they chose to skip the advert, possibly resulting in it being blacklisted for the user. Skipping adverts does not reduce the amount of time the user spends watching adverts, instead simply allows the collection of user preference data and replaces the advert for another which the user hopefully finds more relevant.

Each advert is assigned to a campaign, where a campaign has multiple adverts and an advert may apply to multiple campaigns. A campaign may be given a set of

[11] `http://www.grouplens.org/node/73`

restrictions, where the adverts belonging to a campaign are shown whenever the restrictions are met. Restrictions may apply to programmes, users and times, where the restrictable attributes are:

Programmes	**Times**	**Users**	
genre	time of day	gender	latitude
individual programme	weekday	age	longitude
liveness			occupation

Attributes may have multiple restrictions, where the restrictions are single points or ranges where appropriate. When a campaign has multiple restrictions on a single attribute, that attribute is considered satisfied if any of its restrictions are satisfied; e.g.: if a campaign has weekday restrictions of Monday and Wednesday, the weekday restriction is considered satisfied on either a Monday or a Wednesday. A campaign only applies when all of its restrictions are satisfied, though a campaign need not have restrictions for all of the above attributes, or indeed any restrictions at all.

Whenever a campaign is satisfied, its adverts are shown with a frequency directly proportional to the value of a campaign metric we are calling nicheness, defined as:

$$\text{nicheness} = \frac{(1 - \text{nicheness}_{\text{programme}}) + (1 - \text{nicheness}_{\text{time}}) + (1 - \text{nicheness}_{\text{user}})}{3}$$

where

$$\text{nicheness}_{\text{programme}} = 1 - \frac{\text{satisfying programmes}}{\text{total programmes}}$$

$$\text{nicheness}_{\text{time}} = 1 - \frac{\text{satisfied time during campaign}}{\text{total time during campaign}}$$

$$\text{nicheness}_{\text{user}} = 1 - \frac{\text{satisfying users}}{\text{total users}}$$

meaning that for a particular person, programme and time in which multiple campaigns apply, more niche campaigns will have their adverts display more often. This design decision was made to reward advertisers for creating niche campaigns over broad campaigns. By encouraging a greater number of more highly targetted advertisements in this way, users will be shown a far more relevant advert set. According to our prestudy (section 6), such relevance has a large influence on user attentivity, and is also known to positively influences the user experience(**?**).

7.4 Streaming

Each programme will be recorded using the EPG data. Since adverts will be replaced by ones recommended to viewers, it must be known when adverts are due to start and finish. This information is available as a GPI[?] pulse, sent by Channel 4 [more technical details]

- detail of stream origin and how many systems it goes through before reaching us - detail about any delay this may cause - what limitations are imposed by this? is useful data stripped? does the delay cause inaccuracies?

8 Implementation

8.1 Languages and Techniques

As this project involved interacting with many different systems, a diverse range of programming languages were used.

8.1.1 Client-side web interfaces

1. *HTML*
2. *CSS*
3. *JavaScript*

8.1.2 Server-side and data-layer

1. *PHP*
2. *Python*

Communication between server and client is done through a REST interface.

Datebase is MySQL.

8.1.3 Streaming server

1. *Java* to interface with the Wowza streaming server
2. *Python* to interpret to GPI[?] pulse in order to establish when breaks in live programmes are

8.2 Libraries and Frameworks

8.3 Detailed Architecture

8.3.1 Database structure

8.3.2 Programme recommendation

8.3.3 Advert selection

8.3.4 REST services

8.3.5 Playback

Plays HLS streams in HTML5 Video tag, or RTMP stream in Flash Video.

The player must be capable to displaying two types of media: *video* (for programmes and video adverts) and *stills* (for adverts which consist of just one picture). Additionally, some adverts may have an overlay, which consists of some HTML specified by the advertiser. A challenge here is hiding the buffering of videos or loading of images. A solution is to hide any media until it has loaded. Layers:

1. *Black layer*: used to hide any loading media
2. *Overlay layer*: an iFrame which will display the HTML that the advertiser has specified for the overlay
3. *Still layer*: an image that will display any still adverts
4. *Video layer*: the video component to show the video

8.3.6 Data visualisation

8.4 Code Statistics

Lines of code in JS, PHP, Java, Python, HTML, CSS etc

9 User Study

9.1 Overview

A study was performed in which the change in user engagement was measured upon being presented with interactive and non-interactive advertisements. To estimate user engagement, two known correlating measurements were taken: information recall() and time perception(**?**).

9.2 Methodology

Using a set of 119 student targetted adverts taken from Project4, 8 were given interactive html/css overlays containing interactive content likely to be typical for an interactive TV advert. Each participant in the study was shown two rounds of adverts—one interactive and one non-interactive—with half the participants being shown the interactive adverts first, and the other half shown non-interactive adverts first. The length of the rounds varied between 4, 5, and 6 minutes, and were evenly split between longer interactive rounds, longer non-interactive rounds and equal length rounds; the purpose of which was to measure differences in the participants perceived time and actual round time.

The study was scripted, which has been included in Appendix B. While participants were neither encouraged nor discouraged from interacting with the interactive set of adverts, they were informed that interactions were possible, and how to interact, removing the initial learning period. As part of the study, an video introduction was given, which is included in Appendix C.

9.3 Results

Analysing the answer to question 2b (ii), "*Please estimate how much time more or less you were watching in the second session?*", shows how users perceived time during the live and non-live rounds.

Product	Interactive content
Pot Noodle	**Poll**: The user is encouraged to vote for their favorite flavour of pot noodle.
Smirnoff Vodka	**Social**: The user may 'like' the product, interacting with the social networking website Facebook.
Movie	**Information**: The user may enter their postcode prompting a map to be expanded showing nearby cinema showings.
Scotland Holidays	**Information**: The user may enter their email address to request more holiday-related information.
Fosters Lager	**Social**: The user may 'like' the product, interacting with the social networking website Facebook.
Unified Insurance Cover	**Information**: The user may select possessions they own, and are given an estimated insurance quote.
Dominoes Pizza	**Purchase**: The user may interact witht he advert to order a pizza.
`www.thetrainline.com`	**Poll**: The user may vote whether or not they think train tickets are too expensive, and are shown the poll results upon voting.

Table 1: *Interactive content added to advertisements*

10 Discussion

10.1 Summary

10.2 Limitations

10.3 Further Work

10.3.1 Programme Recommendation

Currently, programmes are assigned binary vectors within $\mathcal{P}$; programmes either do or do not belong in each of the 18 genres. As a result, programme vectors may lie only in the corners of the hypercube geometrically representing $\mathcal{P}$. A logical improvement to the recommender system would be to allow fuzzy genre memberships, allowing programme vectors to exist anywhere within $\mathcal{P}$ and hence allowing fine-grained differences between similar programmes to be properly represented in the system.

To initialize a programme with a fuzzy programme vector, `get_programme_vector` will be required to make use of more information than the current list of programme genres. Possible avenues to explore could include modifying $\mathcal{P}$ such that points are represented by genres pulled from multiple sources and reduced to a lower dimensionality feature space through Principal Component Analysis, where the dimensionality of the feature space would be set to minimise the number of dimensions while maximising the retained information. If additional external information processing is undesirable, user rating data could be used to modify programme vectors which are initialised as binary, although this is only useful in the case of recommending non-live programmes due to the cold start problem(**?**).

If non-binary programme vectors are introduced to $\mathcal{P}$, a change is required in how a user vector is modified upon a negative programme rating. Under the current architecture, a user is pushed away from the vector of a negatively rated programme; if programme vectors exist away from the vertices of $\mathcal{P}$, a user who's vector somehow ends up at a vertex describing programmes they dislike will be unable to move away from the vertex by giving negative ratings, leaving them stuck. This is a difficult problem to solve and is outside the scope of this project; repeated bad ratings must not converge to a single point but explore the programme space, but must not pull a user vector away from a known 'good' area. While a jump with random direction may work, storing the users previous rating will allow for exploitation of rating gradients, enabling use of more complex gradient-climbing techniques.

10.3.2 Advert Recommendation

If recommendation techniques are implemented to improve advert relevance, the data already being collected on user preferences will be of great value in predicting adverts the user will enjoy/engage with. Techniques have been developed (**?**) which utilize this preference information, along with demographic (**?**) information, which is also collected by your4.tv. A third datasource utilised by the recommender sys-

tem described by (**?**) is user viewing histories, which your4.tv does not currently collect, though has potential to improve not only advert recommendations, but also programme recommendations, as mentioned in Section 10.3.1.

10.4 Conclusion

11 Bibliography

Appendix A Brief

The deliverable outcome of this project will be a prototype Channel 4 TV streaming web-application for the iPad, which will provide the viewer with a personal entertainment channel tailored to them specifically. The custom channel will consist of live streams from Channel 4, programs that are available via TV catchup and advertisements targeted towards the viewer. Programmes and streams will be available from Channel 4 and its sister channels: Film4, E4, More4 and 4Music.

While functionality exists in current TV streaming and catchup websites to recommend similar programmes, they are not integral to the service and simply offer a browsable list of shows. For example 4od[12] and iPlayer[13] offer a list of similar programs separate to the video player. User-specific data, such as demographics and viewing history, are not taken into account by these services.

In order to determine what programmes and adverts would be appropriate for the viewer, a diverse set of data sources will be utilised. By monitoring what people choose to view, we can target the viewer with programmes and adverts that complement their interests or are trending at that time. We will capture personal data via social networks such as location information, age, gender and friends to further improve the targeting of programmes and adverts.

Viewers will be able to express a positive or negative attitude towards the currently displayed programme or advert - opting to skip it if it does not interest them. Upon expressing a negative attitude, the user will be asked to indicate why, so that their future playlist of programmes and adverts can be tuned. An EPG (Electronic Programme Guide) will be included in the application, allowing the viewer to browse live TV listings and non-live programmes on catchup, and add them to their own personal channel. A simple keyword search will also be offered such that a user can determine if and when a particular programme is available.

Further work could include a friend system allowing a registered user to view other users personal streams, with advertisement breaks remaining focused on the currently logged in user. Data regarding the viewing history of friends can also be used to tune which programmes are included in the current personalised playlist. Friends could possibly be inferred from linked social media accounts.

As an example use-case; consider a new user, John, a 21 year old male. He signs in to the service using his Facebook account. The listings for his personal channel can be seen, populated by programmes complementing his interests, including programmes which are popular within his demographics. John can bring up an EPG, which he can use to browse and select programmes to watch. As it comes closer to Valentines day, John is shown more and more romantically themed adverts - but as he has just come out of a long-term relationship, he would rather these were not shown to him. By rating these adverts poorly, they are no-longer shown, and are instead replaced by adverts for cars, which John prefers. Over time, as John chooses, skips and rates programmes, the programmes offered will more accurately reflect Johns interests.

[12] `http://www.channel4.com/programmes/4od`
[13] `http://www.bbc.co.uk/iplayer`

Appendix B Interview Script

Hi. Thanks for agreeing to participate in our study, this should not take more than half an hour. In a moment you will be given an iPad which will show you a short video to introduce you to the rest of the study. Do you have any other questions before we begin.

⟨participant watches intro and each round of adverts⟩

Id like to emphasise that this is not a test, were just interested in seeing what your impressions were and how much you remember.

1. What were the adverts you just watched about? Describe what happened and what they were advertising in as much detail as you can.

Youve just watched two streams - one with some interactive adverts, and one without any. [Swap order as appropriate]

2. How long do you believe you were watching each stream? 2b. Were you watching for a much longer period, a little longer, the same amount of time, a little less or a lot less? 2b (ii). Please estimate how much time more or less you were watching in the second session?

3. Did you pay more attention to adverts that were relevant to you? 4. During which session did you pay more attention to adverts (if either)? 5. During which session do you remember more about the adverts shown (if either)? 6. Which session did you enjoy more - interactive or passive?

7. If you were able to choose, which system would you prefer to be used for live streaming media on iPads and why.

Im now going to ask you some questions specifically about the interactive adverts, in the [first/second] session.

Xa(i). Which adverts had interactive elements if any and what were they? Xa(ii). which did you interact with and why did you choose to interact at that time?

You saw some of the adverts twice - once with an interactive overlay, and once without.

Xa(iii). did you prefer the adverts with an interactive element to them? Xa(iv). Were you able to easily work out how to interact with an advert? Xb. Of the interactive adverts, which advert(s) did you like best and why?

Last few questions:

8. Do you believe that if adverts were more relevant to you, you would watch them more often?

There are multiple pieces of information that can be used to target adverts (make them more relevant). Im going to mention a number of categories, and Id like you to

tell me how comfortable you are with the use of that information for advertisement targeting.

9. What levels of targeting would you be comfortable with in these categories: a- demographics (anonymous information about age, gender and occupation) b- current location, c- the time you watch the advert (lunchtime, late evening) d- the context you watch the advert in, so the programme youve just watched/are watching/about to watch, e- browsing history, f- significant changes to info on Facebook (e.g. relationship status changing, new occupation) and g- learned by the system from previous interaction (adverts youve clicked on before, programmes youve chosen)

Final question:

10. Do you have any other comments, questions or comparisons about the two systems youve used today, or any other advertising systems that you know of?

Thank you for participating in our study

Appendix C Video Script

Hi, thanks for volunteering for this study. In this study, we want to find out a bit more about how people view TV adverts.

Were about to show you two sets of adverts. After these, well ask you a series of questions on your experience.

We ask that during this study you remain focused on the screen, and dont communicate with the researcher (unless there is a problem) and please dont use your phone or other devices.

When youre ready to begin, press start.

If you would like to watch this introduction again, press replay.

[Round 1: passive]

The adverts youre about to be shown are adverts as you might see on TV. All you need to do is watch these adverts and well ask you some questions about your experience at the end of the study.

Press start to begin. Press replay to watch these instructions again.

[Round 2: interactive]

In this round, you will be shown adverts as you might be shown adverts on TV, except some adverts will allow you to interact with them. During this round, you should watch the adverts, you may also wish to tap or swipe the screen in order to interact with adverts that allow you to.

You can touch the screen in order to interact with parts of these adverts. Well ask you some questions about your experience these adverts at the end of the study.

Press start to begin. Press replay to watch these instructions again.

[Finish round]

Thank for participating in the experiment. The researcher will now ask you a series of questions about your experiences. Try to answer honestly and as completely as possible. Do not be afraid to ask for clarification if you do not understand a question.

[END]