

Aalto University
School of Science
Master's Programme in Computer, Communication and Information Sciences

Kasper Hellström

Predicting Motives for Video-On-Demand Content

Master's Thesis
Espoo, May 31, 2018

Supervisor: Professor Aristides Gionis, Aalto University
Advisor: Teemu Kinnunen D.Sc. (Tech.)

Author:	Kasper Hellström		
Title:	Predicting Motives for Video-On-Demand Content		
Date:	May 31, 2018	Pages:	vii + 66
Major:	Machine Learning and Data Mining	Code:	SCI3044
Supervisor:	Professor Aristides Gionis		
Advisor:	Teemu Kinnunen D.Sc. (Tech.)		
<p>Media content providers need to understand their users' needs and behaviors to be able to choose what kind of content they should produce and purchase. They may also be interested in providing a vast selection of content that has something to offer everyone.</p> <p>Analytics help the providers understand when the content was consumed, by which users and by which devices, but it cannot tell why it was consumed. To acquiring the motives behind the consumption would traditionally require the use of surveys. This is however not feasible to do for all the content, so there needs to be a way to estimate the motives.</p> <p>This thesis studied how viewing motives can be predicted for video content on a video-on-demand service. The features used to predict motives were genres and contextual features (device, time of day, day of week) derived from analytics events. The problems that were studied were which features and machine learning methods worked best for predicting motives.</p> <p>In previous research, motives have been predicted for users at specific times during the day. There are also several media studies that have analyzed the consumption motives for different types of media and genres. However, to the best of our knowledge, the prediction of consumption motives has not been attempted earlier for media content.</p> <p>The best result for predicting motives was acquired using all features with a neural network with 8 hidden neurons in one hidden layer. This network had a mean RMSE 0.097 in the cross-validation. This is a significant improvement over the mean RMSE of the baseline which was 0.233.</p> <p>The kernel ridge regression models performed approximately equally well as the neural networks, and the dimensionality reduction methods improved the results slightly when used. The two genres used as features were able to predict motives better than the contextual features. The best result was however obtained by combining all the features.</p>			
Keywords:	prediction, motive, media, machine learning		
Language:	English		

Aalto-universitetet
 Höskolan för teknikvetenskaper
 Magisterprogrammet i data-, kommunikations- och infor- SAMMANDRAG AV
 mationsteknik DIPLOMARBETET

Utfört av:	Kasper Hellström		
Arbetets namn:	Motivförutsägelse för videoinnehåll		
Datum:	Den 31 maj 2018	Sidantal:	vii + 66
Huvudämne:	Maskininläring och datautvinning	Kod:	SCI3044
Övervakare:	Professor Aristides Gionis		
Handledare:	TkD Teemu Kinnunen		
<p>Medieinnehållsleverantörer måste kunna förstå sina användares behov och beteenden för att kunna välja vilken typ av innehåll de ska producera och köpa in. De kan också vara intresserade av att förse användarna med ett brett innehållsurval som har något att erbjuda alla användare.</p> <p>Analysverktyg kan berätta leverantörerna när innehållet konsumeras, av vilka användare och med vilka apparater, men det kan inte berätta orsaken bakom konsumtionen. Erhållandet av motiven bakom konsumtionen skulle dock traditionellt kräva enkätundersökningar. Detta är emellertid inte möjligt att göra för allt innehåll, och därmed behövs det sätt för att uppskatta motiven.</p> <p>Det här arbetet undersökte hur väl tittarmotiv kan förutsägas för videoinnehåll på en video på begäran-tjänst. Indatan som användes för att förutsäga motiv bestod av genrer och kontextuell data (apparat, veckodag, tidpunkt på dygnet) som härletts med analysverktyg. Problemen som studerades var vilka indata och maskininlärningsmetoder som fungerade bäst för att förutsäga motiv.</p> <p>Tidigare undersökningar har studerat motivförutsägelse för användare vid specifika tidpunkter på dygnet. Det finns också flera medieundersökningar som har studerat konsumtionsmotiven bakom olika typer av media och genrer. Till vår kännedom har dock konsumtionsmotivsförutsägelse inte prövats tidigare för medieinnehåll.</p> <p>Det bästa resultatet i motivförutsägelse fick genom att använda all indata med ett neuronnät med 8 dolda neuroner i ett dolt lager. Detta nätverk hade en medeltals RMSE på 0,097 i korsvalideringen. Detta är en märkbar förbättring jämfört med referensmodellen som hade en medeltals RMSE på 0,233.</p> <p>Kernel ridge regression-modellerna presterade ungefär lika bra som neuronnäten, och dimensionsreducering förbättrade resultaten en aning då det användes. De två genren som användes som funktioner kunde förutse motiv bättre än kontextuell data. Det bästa resultatet uppnåddes dock genom att kombinera all indata.</p>			
Nyckelord:	förutsägelse, motiv, media, maskininläring		
Språk:	Engelska		

Acknowledgements

First, I would like to thank my thesis instructor Dr Teemu Kinnunen for all the support and guidance he has provided throughout the whole project, as well as Prof. Aristides Gionis for supervising my thesis and providing feedback.

Second, I would also like to thank Eija Moisala, Jaakko Lempinen, Anne Hyvärilä, Outi Roivainen, Toni Mikkola, Katherine Icaý-Rouhiainen and everyone else who I worked with at Yle during my thesis. I had a great time working with you and really appreciate you giving me the opportunity to work on this project.

In addition, I want to thank Futurice and Anniina Lehtinen for introducing me to this project.

Finally, I would like to thank Rubing Mao for supporting and encouraging me during the process. Without you this would have been much harder.

Espoo, May 31, 2018

Kasper Hellström

Notations

x	Scalar value
\mathbf{x}	Vector
\mathbf{X}	Matrix
\mathbf{x}^T	Vector transpose
\mathbf{X}^T	Matrix transpose
\mathbf{K}	Kernel matrix
$\mathcal{X} = \{\mathbf{x}_t, \mathbf{r}_t\}_{t=1}^N$	Training set
\mathcal{E}	Model error
$\langle \cdot, \cdot \rangle$	Inner product
$\ \cdot \ $	Euclidean norm
$k(\cdot, \cdot)$	Kernel function
$\phi(\cdot)$	Activation function for node in neural network
$g(\cdot)$	Hypothesis function
$\exp(\cdot)$	Exponential function
$J(\cdot)$	Cost function
s	Parameter for scaling binary columns
k	Number of dimensions the data should be reduced to
α	Regularization parameter
γ	Free parameter for polynomial and RBF kernel
d	Degree parameter for polynomial kernel
c_0	Free parameter for polynomial kernel

Contents

Notations	v
1 Introduction	1
1.1 Background	1
1.2 Context	2
1.3 Research questions	2
1.4 Summary of results and contributions	3
2 Related work	4
2.1 Motives	4
2.2 Context	5
2.3 Predicting motives	6
2.4 Dimensionality reduction	6
2.4.1 Principal component analysis	7
2.4.2 Truncated singular value decomposition	8
2.4.3 Non-negative matrix factorization	8
2.5 Neural networks	9
2.5.1 Training	12
2.6 Regression	13
2.6.1 Linear regression	13
2.6.2 Kernel ridge regression	14
2.7 Model selection	16
2.7.1 Validation	16
2.7.2 Cross-validation	17
2.7.3 Grid search	18
3 Methods	19
3.1 Analytics	19
3.2 Surveys	19
3.2.1 Survey result	21
3.3 Features	28

3.3.1	Contextual features	28
3.3.2	Genres	29
3.4	Motives	29
3.5	Evaluation	30
4	Implementation	31
4.1	Overview	31
4.2	Prediction system	32
4.2.1	Feature sets	32
4.2.2	Transforming nominal features	33
4.2.3	Dimensionality reduction	33
4.2.4	Regression models	33
5	Results	36
5.1	Methods and parameters	36
5.1.1	Kernel ridge regression	36
5.1.2	Neural networks	36
5.2	Baseline	37
5.3	Results using context	38
5.4	Results using genres	42
5.4.1	Genre A	42
5.4.2	Genre B	45
5.4.3	Both genres	49
5.5	Results using context and genres	53
5.6	Summary of results	57
6	Discussion	58
6.1	Predicting motives	58
6.2	Generalization of the result	59
6.3	Comparison to previous research	60
6.4	Limitations	60
6.5	Future work	61
7	Summary	62

Chapter 1

Introduction

1.1 Background

Understanding how the users use a service is important for the service provider. Using knowledge about the consumption, the service provider can make changes that will improve the service and lead to better user experience. For instance, knowledge about the consumption can be used to choose which shows that should appear on the home page during different times of the day. Another possibility is to provide personal recommendations to the users. Without knowledge about the consumption the service provider would have to rely on assumptions and recommend everyone the same content.

For online services, web analytics can be extremely helpful for collecting data about the usage. Using web analytics, companies can tell who has done what, when they did it and how they did it. This is a very powerful way of collecting data and it can be used for purposes such as recommendation and personalization.

For video-on-demand (VOD) services information about the used devices and watching times can be useful for understanding the consumption. However, there is additional information about the users' context that is difficult to measure. One such piece of information is the users' motives for watching a program. Motives can tell the reasons why the users chose to watch a program and how they perceive a program. For instance, someone may watch drama to entertain themselves or pass time, and someone else could watch documentaries to educate themselves. This is valuable information to the people who produce and purchase programs for the platform.

Information about user motives need to be collected with surveys. The problem is, however, that surveys are usually answered by only a few people and surveys are time consuming to construct and analyze. To solve this

problem, we attempt to use existing analytics data and machine learning to predict motives that would otherwise require performing surveys.

1.2 Context

The Finnish broadcasting company, Yle, operates several TV and radio channels and provides video and audio on demand services as well as online news articles for the Finnish audience. As the national broadcasting company in Finland it is funded by tax money and regulated by law. Therefore, Yle has the duty to provide diverse and comprehensive content for all Finnish citizens.

While Yle increasingly focuses on its online services they are able to collect precise data of the online media consumption. This gives them new opportunities to measure how they fulfill their duties. For instance, Yle is already predicting genders and age groups for the users of its online services based on what they have consumed [16]. There is, however, a need to gain a deeper understanding of the audience and the consumption than what demographics has to offer. What Yle wants is to understand the motives and context of the users and how they consume Yle's content. This type of information is important for Yle to decide what kind of shows to order and purchase, and to determine their selection of programs is wide enough.

1.3 Research questions

This thesis will explore how motives can be predicted for VOD content using genres and contextual features derived from analytics events.

Kärkkäinen [16] earlier studied at Yle how demographics and motives could be predicted for users using analytics data. As far as we know, this is the only attempt to predict motives using machine learning and web analytics data. In his research the motives were tied to users and time blocks, which resulted in the motives being difficult to predict. Therefore, in this thesis a different approach is taken where the motives are considered as relations between the user and the content.

The research questions for this thesis are the following:

1. Can viewing motives be predicted for content with contextual consumption data and genres?
 - (a) How good is the prediction when using only contextual data?
 - (b) How good is the prediction when using only genre data?

(c) How good is the prediction when using both contextual and genre data?

2. Which machine learning method suits best for predicting motives?

In the research questions, contextual consumption data refers to data that describes the consumption of content for different times and devices.

1.4 Summary of results and contributions

The results of this project show that motives can be predicted with moderate precision for programs using contextual data and genres. Genres are able to predict motives clearly better than the contextual data, and by combining both features the predictions can be improved slightly even further.

The study contributes to media research by showing that viewer motives can be predicted for programs. However, there is still room for improvement in the predictions. The predictions could be improved by trying new features and collecting a larger training set.

Chapter 2

Related work

This chapter looks into how motives and context have been defined in academic publications and explores the type of research that has been done in media studies about motives and context. In addition, this chapter explains how the research in this thesis differs from earlier research.

2.1 Motives

Motives are the reasons for actions. Burke [6] has reasoned that motives should always be examined through five key elements: *act*, *scene*, *agent*, *agency* and *purpose*. In the perspective of this thesis, these elements can be seen as part of context. Therefore, it is reasonable to believe that motives have connections with other contextual features.

Motives have been studied extensively in different media studies. Rubin [22] has studied motives for TV watching. In the study, he performed a survey with 30 statements with reasons for watching television. Using factor analysis, he then determined five motive patterns for TV watching which were *pass time/habit*, *information*, *entertainment*, *companionship* and *escape*.

Similar studies have also been done for TV show genres. For reality TV, Papacharissi and Mendelson [19] found out that the main motives for watching are *reality entertainment*, *relaxation*, *habitual pass time*, *companionship*, *social interaction* and *voyeurism*, out of which *reality entertainment* and *habitual pass time* were the most common. In another study, Rubin and Perse [23] examined the motives for soap operas. The primary motives turned out to be *exciting entertainment*, *escapist relaxation* and *pass time*, while *social utility*, *voyeurism* and *information* were less likely motives.

Motives have also been studied for new media on the internet where the users have a more active role when selecting the content. Bondad-Brown

et al. [5] looked at the differences between motives for watching TV and online user-shared videos (YouTube). The results in the study indicate that the motives for watching TV and online videos are very similar, although all motives were more likely to occur for TV watching. Similarly, Hanson and Haridakis [10] studied how motives differed for his students when they were watching traditional news and comedy-news on YouTube. Not too surprisingly, students who watched traditional news did so primarily for informational reasons, whereas those who watched comedy news did so primarily for entertainment. Pittman and Sheehan [21], on the other hand, has studied Netflix users' motives for binge-watching. The study showed that binge-watchers are more engaged in the characters and story lines than other watchers.

2.2 Context

The New Oxford American Dictionary[26] defines context as “circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood and assessed.” This is a very broad definition since so many things can be considered to be part of context. Adomavicius and Tuzhilin [2] also states that what is considered to be context for one discipline very likely differs for another discipline.

In the field of human-computer interaction, Dey and Abowd [7] has defined context as information that characterizes the situation of a participant in interaction. Thus, from the perspective of this thesis, context can be considered as information about the interaction between the user and the VOD platform. Dey and Abowd also considers there to be four primary context types: *location*, *identity*, *time* and *activity*. The user's location can be represented in many ways, e.g. using coordinates, the current city or country, or a place label such as *home* or *work*. According to Dey and Abowd, location can also include information about nearby people and objects as well as nearby activities. Identity, on the other hand, can stand for any kind of information about the user. This includes the user's name, phone number, address, list of contacts and friends. The two remaining primary context types, activity and time, are fairly self-explanatory, describing what took place and when it happened, respectively.

According to Dourish [8], contexts can be classified into two different views; a representational view or an interactional view. In the representational view the structure of observable attributes is known before and it does not change significantly over time. On the other hand, the interactional view assumes that user behavior is influenced by context which may not be ob-

servable. Thereby, the users' motives can be considered to be part of context as well.

2.3 Predicting motives

This thesis will focus on how motives can be predicted for VOD content with genres and contextual features.

Kärkkäinen [16] has explored how well motives can be predicted for users at specific time blocks using the users' read articles, watched videos and used devices. The results, however, were not too good, since nearly all motives were predicted with the same accuracy as the baselines, which always predicted the most common motive for everyone. Only predictions for information seeking and enjoying or relaxing performed slightly better than the baselines. Kärkkäinen states that the attempted approaches cannot be used to predict motives accurately, although the time block alone could potentially be used to predict motives since many of the survey responses were similar to each other.

The approach used by Kärkkäinen was based on the assumption that users have habitual motives which repeat for time blocks. This is more likely true for traditional media where TV and radio shows are broadcasted at specific times and news papers arrive in the morning. For online services, however, the user is able to chose more freely what to consume and when. Therefore, in this thesis, motives will be viewed as connections between users and TV shows.

2.4 Dimensionality reduction

As the name dimensionality reduction suggests, the method can be used for reducing the number of features in a data set. This is beneficial since, the time complexity of machine learning algorithms usually depend on the number of features, so reducing the number of features will reduce the time needed. Another advantage of dimensionality reduction is that it can reduce noise in the data. [3]

An example of dimensionality reduction could be a data set where the data points consist of houses for sale. In this case some (numerical) features could be the price, size and building year and number of rooms of the house. Some of these features may not be important for the machine learning algorithm and some features, such as the size and the price, can be dependent and therefore contain overlapping information.

From a theoretical point of view, the idea is to reduce the size of a data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ which contains n items described with d features. The resulting reduced matrix $\mathbf{Z} \in \mathbb{R}^{n \times k}$ where $k < d$. There are several techniques for achieving this and usually some data is discarded in the process.

According to Alpaydin [3] there are two types of methods for dimensionality reduction: feature selection and feature extraction. In feature selection, the goal is to select k out of d features that represent the data as well as possible. In feature extraction, on the other hand, the goal is to create k new features based on the d old features, such that $k < d$.

The following subsections will discuss the feature extraction methods used in this thesis. The methods are principal component analysis (PCA), truncated singular value decomposition (SVD) and non-negative matrix factorization (NMF).

2.4.1 Principal component analysis

Principal component analysis (PCA) was invented by Pearson [20] in 1901 and it can be interpreted as an orthogonal projection to a lower dimensional space so that the variance is maximized [1]. Using matrix notation this looks like

$$\mathbf{Z} = \mathbf{X}\mathbf{W},$$

where \mathbf{X} is a data matrix that has been column-wise centered around zero and $\mathbf{W} \in \mathbb{R}^{d \times k}$ is a projection matrix where each column w_j is a principal component. Since PCA is an orthogonal projection, all the principal components in \mathbf{W} are orthogonal to each other.

Wold et al. [29] showed that the principal components correspond to the eigenvectors of the covariance matrix for the data matrix, which is

$$\mathbf{\Sigma} = \frac{1}{n} \mathbf{X}^T \mathbf{X}.$$

Additionally, Wold et al. [29] mentioned that the eigenvectors associated with the largest eigenvalues are assumed to contain the most useful information, and therefore the eigenvectors are usually sorted in the order of descending eigenvalues. Eigenvectors corresponding to small eigenvalues can be considered to be noise in the data. Jolliffe [14] has however stated that this is not always the case and that eigenvectors with small eigenvalues can also contain important information.

2.4.2 Truncated singular value decomposition

In singular value decomposition (SVD), the data matrix \mathbf{X} is factorized into three matrices in the following way

$$\mathbf{X} = \mathbf{U} \times \mathbf{\Sigma} \times \mathbf{V}^T,$$

$n \times d$ $n \times n$ $n \times d$ $d \times d$

such that \mathbf{U} and \mathbf{V} are orthogonal matrices and $\mathbf{\Sigma}$ is diagonal matrix with non-negative numbers on the diagonal. The diagonal values of $\mathbf{\Sigma}$ are known as singular values and it is common that they are in descending order.

Strang [28] has showed that \mathbf{U} , \mathbf{V} and $\mathbf{\Sigma}$ can be calculated by noting that

$$\mathbf{X}^T \mathbf{X} = \mathbf{V} \mathbf{\Sigma} \mathbf{U}^T \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T = \mathbf{V} \mathbf{\Sigma}^2 \mathbf{V}^T,$$

which means that \mathbf{V} and $\mathbf{\Sigma}$ can be obtained from the eigendecomposition ($\mathbf{A} = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^T$) of the symmetric matrix $\mathbf{X}^T \mathbf{X}$. This means that the values on the diagonal of $\mathbf{\Sigma}$ are equal to the square roots of the eigenvalues of $\mathbf{X}^T \mathbf{X}$, and that the columns of \mathbf{V} are the eigenvectors of $\mathbf{X}^T \mathbf{X}$. \mathbf{U} (and $\mathbf{\Sigma}$) can similarly be obtained from

$$\mathbf{X} \mathbf{X}^T = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \mathbf{V} \mathbf{\Sigma} \mathbf{U}^T = \mathbf{U} \mathbf{\Sigma}^2 \mathbf{U}^T.$$

Stewart [27] has showed that truncated SVD works similarly to SVD, except that only the k largest singular values and eigenvectors are kept. Thus, the dimensions are the following:

$$\mathbf{X} \approx \mathbf{U}_t \times \mathbf{\Sigma}_t \times \mathbf{V}_t^T.$$

$n \times d$ $n \times k$ $k \times k$ $k \times d$

For dimensionality reduction purposes, we however want to transform \mathbf{X} from being an $n \times d$ matrix to an $n \times k$ matrix. With truncated SVD this is accomplished by using the matrix product of \mathbf{U}_t and $\mathbf{\Sigma}_t$:

$$\mathbf{Z} = \mathbf{U}_t \times \mathbf{\Sigma}_t \approx \mathbf{X} \times \mathbf{V}_t$$

$n \times k$ $n \times k$ $k \times k$ $n \times k$ $k \times k$

The same transformation can also be applied to new data \mathbf{X}_2 by simply multiplying it with \mathbf{V}_t :

$$\mathbf{Z}_2 = \mathbf{X}_2 \times \mathbf{V}_t$$

$m \times k$ $m \times k$ $k \times k$

2.4.3 Non-negative matrix factorization

In non-negative matrix factorization (NMF), the matrix \mathbf{X} is decomposed into two smaller matrices \mathbf{W} and \mathbf{H} . In contrast with PCA and truncated

SVD, there is a requirement that the resulting matrices \mathbf{W} and \mathbf{H} must be non-negative. Consequently, NMF can only be performed on non-negative matrices. The dimensions of the matrices in the factorizations are shown below.

$$\mathbf{X}_{n \times d} \approx \mathbf{W}_{n \times k} \times \mathbf{H}_{k \times d}$$

The factorization can usually not be solved analytically, so it is often approximated numerically. Lee and Seung [17] have shown that one way to solve NMF is to use an iterative approach for minimizing the Euclidean distance

$$\|\mathbf{X} - \mathbf{WH}\|.$$

This can be done by updating the matrices \mathbf{W} and \mathbf{H} iteratively in the following way:

$$\mathbf{H}_{[i,j]}^{n+1} = \mathbf{H}_{[i,j]}^n \frac{((\mathbf{W}^n)^T \mathbf{X})_{[i,j]}}{((\mathbf{W}^n)^T \mathbf{W}^n \mathbf{H}^n)_{[i,j]}}$$

$$\mathbf{W}_{[i,j]}^{n+1} = \mathbf{W}_{[i,j]}^n \frac{(\mathbf{X} (\mathbf{H}^n)^T)_{[i,j]}}{(\mathbf{W}^n \mathbf{H}^n (\mathbf{H}^n)^T)_{[i,j]}}.$$

For dimensionality reduction the $n \times k$ matrix \mathbf{W} can be used as the reduced form of \mathbf{X} .

2.5 Neural networks

A neural network is a machine learning algorithm that consists of neurons or nodes that are interconnected. The neurons are usually organized in layers. The layers can be divided into three different types: the income layer, hidden layers and the output layer. In most cases the layers are sequential and there are connection only between neighboring layers. These types of networks are called feed-forward networks, since the data always moves in one direction, forward. The connections between layers can either be either partially or fully connected, meaning that each neuron one of the layers is connected to all other neurons on the other layer. Each connection has a weight that is a scalar value which is used as a multiplier when calculating the values for the next layer. Nodes placed in hidden and output layers may also have activation functions $\phi(\cdot)$ that are used to bring nonlinearity neural network calculations.[11]

Figure 2.1 shows an illustration of a small neural network. This network has three neutrons in the input layer, two neurons in the hidden layer and one neuron in the output layer. The weights $w_{i,j} = [\mathbf{W}]_{i,j}$ and $u_{i,j} = [\mathbf{U}]_{i,j}$ as well as activation functions $\phi_h(\cdot)$ and $\phi_o(\cdot)$ have been added to the illustration

for demonstration. The input layer can be represented with a vector $\mathbf{i} = [i_1, i_2, i_3, 1]^T$ with three values and the constant bias, 1. The hidden layer can similarly be represented with a vector $\mathbf{h} = [h_1, h_2, 1]^T$ that is calculated from the input layer and weights in the following way:

$$\mathbf{h} = \phi_h(\mathbf{W}\mathbf{i})$$

In other words, the value for a neuron in the hidden layer is calculated with the following formula:

$$h_j = \phi_h\left(\sum_{k=1}^4 w_{k,j}i_k\right)$$

The value for the output layer is calculated with the same formula, but with its own weights and activation function:

$$\mathbf{o} = \phi_o(\mathbf{U}\mathbf{h})$$

Thus, the value for the output layer can also be written as a function of the input layer:

$$\mathbf{o} = \phi_o(\mathbf{U}\phi_h(\mathbf{W}\mathbf{i}))$$

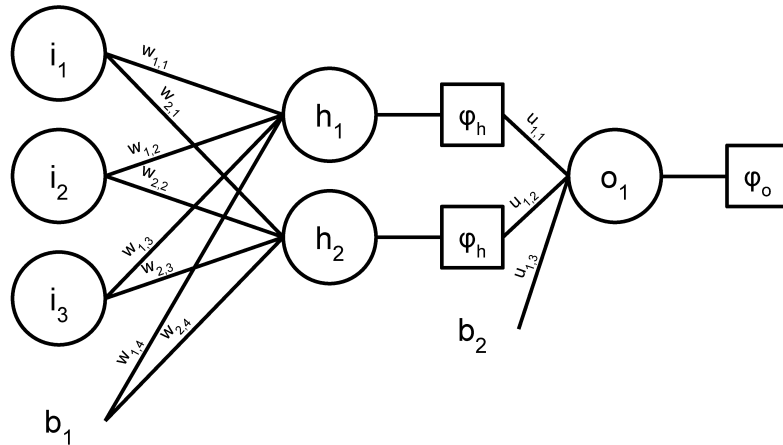


Figure 2.1: A small neural network with three input neurons, two hidden neurons and one neuron in the output layer. The weights, biases and activation functions of the neural network have been added to the figure for demonstration.

Some common activation functions that are used with neural networks are *tanh*, *sigmoid*, *rectifier*, *hard limit* and *softmax*. The formulas for these functions are given below:

- Tanh:

$$\phi(z_k) = \frac{e^{z_k} - e^{-z_k}}{e^{z_k} + e^{-z_k}}$$

- Sigmoid:

$$\phi(z_k) = \frac{1}{1 + e^{-z_k}} = \frac{e^{z_k}}{e^{z_k} + 1}$$

- Rectifier:

$$\phi(z_k) = \max(0, z_k)$$

- Hard limit:

$$\phi(z_k) = \begin{cases} 1 & \text{for } z_k \geq 0 \\ 0 & \text{for } z_k < 0 \end{cases}$$

- Softmax:

$$\phi(z_k) = \frac{e^{z_k}}{\sum_{i=1}^N e^{z_i}}$$

Note that in contrast to the others, the return value of the softmax function also depends on values of other neurons in the same layer. The graphs of these functions can be seen in Figure 2.2.[11]

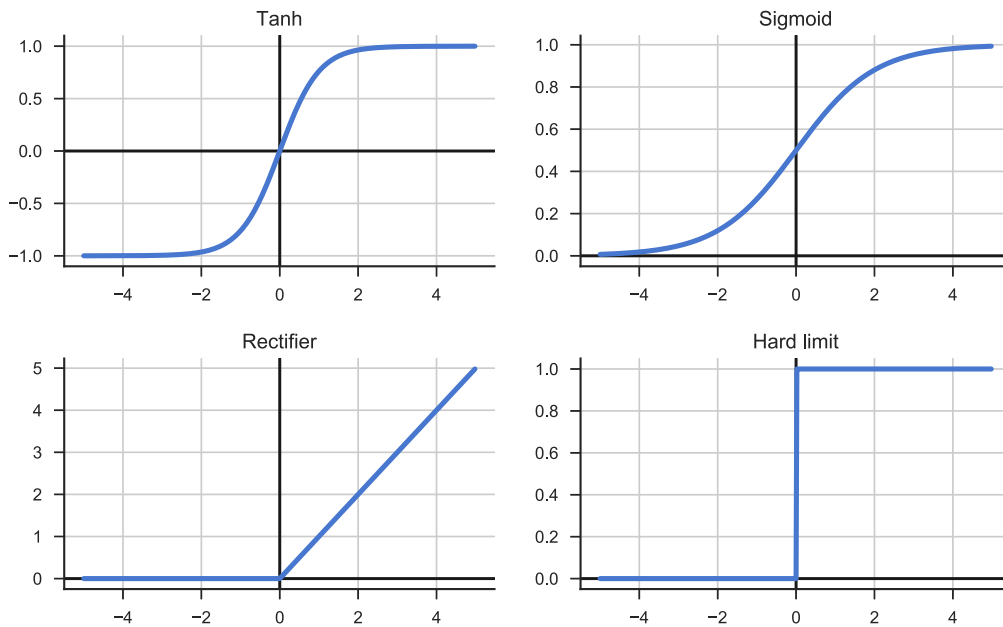


Figure 2.2: Four different activation functions commonly used with neural networks plotted for the range $[-5, 5]$.

A neural network can also lack a hidden layer or contain several hidden layers. Figure 2.3 and Figure 2.4 show the architectures of two neural networks with one and two hidden layers respectively.

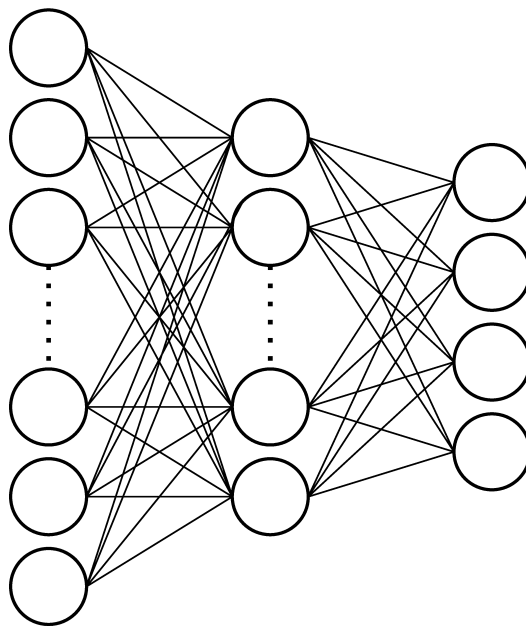


Figure 2.3: A fully connected feed-forward neural network with one hidden layer.

2.5.1 Training

In supervised learning a neural network is trained with a training set $\mathcal{X} = \{\mathbf{x}_t, \mathbf{r}_t\}_{t=1}^N$ where \mathbf{x}_t contains the feature values that are fed to the input layer and \mathbf{r}_t contains the target values that the network is supposed to produce at the output layer. A common algorithm used in training feed-forward neural networks is the backpropagation algorithm. The algorithm works by taking a batch of the training set, giving it to the network and then calculating the error by comparing the output with the expected result. The gradient of the error function is then used to determine how to update the weights of the neural network to reduce the error. This is repeated until a local minima is reached.[4]

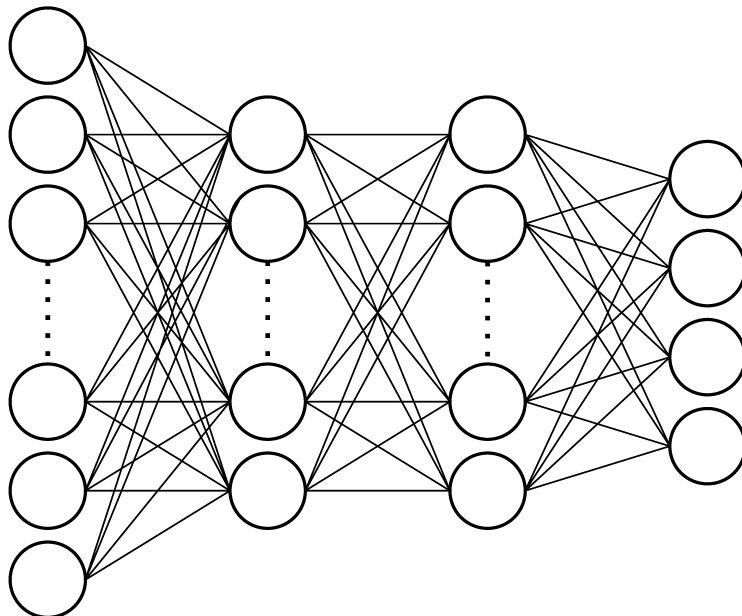


Figure 2.4: A fully connected feed-forward neural network with two hidden layers.

2.6 Regression

Regression is about modeling the relationship between dependent variable and explanatory variables. A common use case for a regression model is to predict the dependent variables using the values of the explanatory variables.

In the following subsections, $\mathcal{X} = \{\mathbf{x}_t, r_t\}_{t=1}^N$ denotes the training set of size N , where \mathbf{x}_t contains the explanatory variables or features and $r_t \in \mathbb{R}$ is the dependent variable. The matrix \mathbf{X} contains all the explanatory variables \mathbf{x}_t on its rows. The hypothesis functions are denoted by $g(\cdot)$ and they try to approximate $g(\mathbf{x}) \approx r_t$.

2.6.1 Linear regression

Linear regression is a simple regression model where $g(\mathbf{x}_t) = \langle \mathbf{w}, \mathbf{x}_t \rangle$ is a linear function. The model in linear regression is chosen so that mean squared error (MSE)

$$\mathcal{E} = \frac{1}{N} \sum_{t=1}^N (r_t - g(\mathbf{x}_t))^2$$

is minimized. Murphy [18] has shown that this can be minimized using the ordinary least squares solution for \mathbf{w} :

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{r}.$$

2.6.2 Kernel ridge regression

In many cases the relationship between the explanatory variables and the dependent variable is nonlinear, which means that it cannot be accurately be modeled with linear regression. In other words, the linear regression model is underfitting. A solution for this is to map the original feature space into a new nonlinear feature space where the relationship is linear. This can be done with a mapping function $\phi(\cdot)$ which takes a feature vector and returns a vector in a nonlinear space that usually has a higher dimension. In kernel ridge regression this mapping is simulated with kernels.[24]

A kernel function $k(x, y)$ simulates the inner product of two mapped vectors using the original feature vectors. Essentially, the kernel function does the following without explicitly performing the mapping $\phi(\cdot)$ to a nonlinear space:

$$k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle.$$

This is known as the kernel trick. Some common kernels include the linear kernel, polynomial kernel and the RBF kernel. Out of these the linear kernel is the simplest since it equals to a kernel where the mapping function is equal to the identity function.

- Linear kernel:

$$k(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$$

- Polynomial kernel:

$$k(\mathbf{x}, \mathbf{y}) = (\gamma \langle \mathbf{x}, \mathbf{y} \rangle + c_0)^d$$

- RBF kernel:

$$k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2)$$

According to Murphy [18], a common problem that rises with high-dimension feature spaces and kernels is overfitting, where the data is modeled too precisely so that new predictions will become unreliable. In kernel ridge

regression this is counteracted with regularization. The idea of regularization is to lower the complexity of the model so that it would not overfit. Murphy [18] shows that in kernel ridge regression this is done by adding the squared euclidean norm of the weight vector to the cost function. Thus the cost function looks like this:

$$J(w) = \min_w \sum_{t=1}^N (\langle \mathbf{w}, \phi(\mathbf{x}_t) \rangle - r_t)^2 + \lambda \|\mathbf{w}\|^2$$

where λ is the regularization parameter. Let \mathbf{z}_t and \mathbf{Z} refer to feature vectors and matrices that have been mapped to another feature space. In matrix form the cost function looks the following:

$$J(w) = (\mathbf{r} - \mathbf{Z}\mathbf{w})^T (\mathbf{r} - \mathbf{Z}\mathbf{w}) + \lambda \mathbf{w}^T \mathbf{w} \quad (2.1)$$

$$= \mathbf{r}^T \mathbf{r} - 2\mathbf{w}^T \mathbf{Z}^T \mathbf{r} + \mathbf{w}^T \mathbf{Z}^T \mathbf{Z} \mathbf{w} + \lambda \mathbf{w}^T \mathbf{w} \quad (2.2)$$

The cost function is derived with respect to \mathbf{w} to obtain the optimal \mathbf{w} that minimizes the cost:

$$\frac{\partial}{\partial \mathbf{w}} J(\mathbf{w}) = -2\mathbf{Z}^T \mathbf{r} + 2\mathbf{Z}^T \mathbf{Z} \mathbf{w} + 2\lambda \mathbf{w} = 0 \quad (2.3)$$

$$-\mathbf{Z}^T \mathbf{r} + \mathbf{Z}^T \mathbf{Z} \mathbf{w} + \lambda \mathbf{w} = 0 \quad (2.4)$$

$$\mathbf{Z}^T \mathbf{Z} \mathbf{w} + \lambda \mathbf{w} = \mathbf{Z}^T \mathbf{r} \quad (2.5)$$

$$(\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I}) \mathbf{w} = \mathbf{Z}^T \mathbf{r} \quad (2.6)$$

$$\mathbf{w} = (\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I})^{-1} \mathbf{Z}^T \mathbf{r} \quad (2.7)$$

$$\mathbf{w} = \mathbf{Z}^T \boldsymbol{\alpha} \quad (2.8)$$

$$\mathbf{w} = \sum_{i=1}^N \alpha_i \mathbf{z}_i \quad (2.9)$$

where $\boldsymbol{\alpha} = (\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I})^{-1} \mathbf{r}$.

The model itself can thus be written as:

$$g(\mathbf{x}_t) = \langle \mathbf{w}, \phi(\mathbf{x}_t) \rangle \quad (2.10)$$

$$= \sum_{i=1}^N \alpha_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_t) \rangle \quad (2.11)$$

$$= \sum_{i=1}^N \alpha_i k(\mathbf{x}_i, \mathbf{x}_t), \quad (2.12)$$

which utilizes the kernel trick. The kernel trick can also be used in the calculation of α where $\mathbf{Z}^T\mathbf{Z}$ can be expressed using the kernel matrix \mathbf{K} . [18]

$$\mathbf{K} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & k(\mathbf{x}_1, \mathbf{x}_2) & \dots & k(\mathbf{x}_1, \mathbf{x}_N) \\ k(\mathbf{x}_2, \mathbf{x}_1) & k(\mathbf{x}_2, \mathbf{x}_2) & \dots & k(\mathbf{x}_2, \mathbf{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \mathbf{x}_1) & k(\mathbf{x}_N, \mathbf{x}_2) & \dots & k(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}$$

2.7 Model selection

Machine learning models have several parameters that can take many different values. This means that there are many ways to select the different parameters. Changes the parameters can affect the results substantially, and it is therefore very important to pick suitable parameters. This section discusses methods that can be used for selecting a model with a good set of parameters.

2.7.1 Validation

When trying to find a good set of parameters for a machine learning algorithm, it is common to try several different parameter combinations. Their performance is usually evaluated by comparing the predicted values to the expected values using a loss function such as mean squared error (MSE). The training error for a model is obtained by predicting values for the training set that was used to train the model. A low training error does, however, not necessarily mean that the model is performing well on data that the model has not seen before. This phenomenon is known as overfitting. Therefore, it is recommended to have a separate validation set that is not used in training and is only used for calculating a validation error. The model with the parameter combination that produces the lowest validation error should then be chosen as the best model. The relation between the model complexity and the training and validation error can be seen in Figure 2.5 where the training error decreases as the model becomes more complex, but the validation error begins to increase when the model becomes too complex. The validation error for the selected model is however biased and does not reflect the true error that the model will have when making predictions for unseen samples. To acquire an unbiased error estimate for the model a third untouched set is needed. This set is called a test set and it should only be used after the model has been selected to determine how well the model will actually perform.[13]

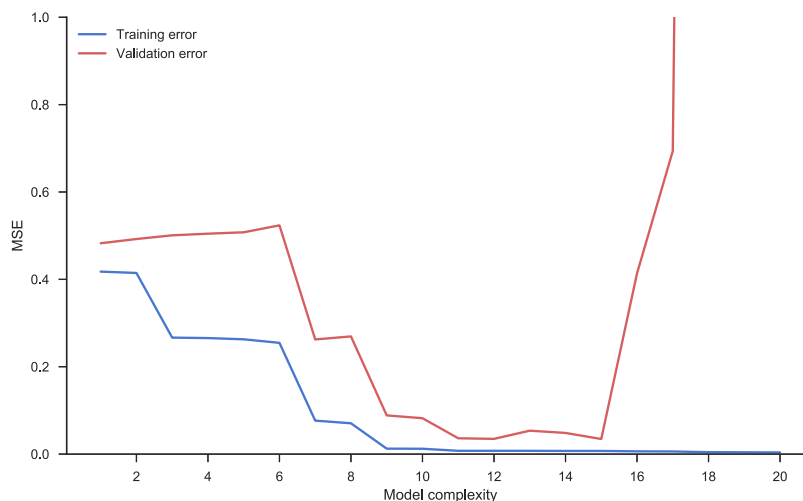


Figure 2.5: Training and validation error for polynomial regression model with different complexities. The training error decreases as the model becomes more complex, but the validation error begins to increase when the model becomes too complex.

2.7.2 Cross-validation

Splitting the labelled data into a training, validation and test set is a good idea, but this requires a lot of data to provide beneficial results with low variance. With smaller data the results can vary greatly depending on how the data is split between the training and validation set. One way to reduce the variance with smaller sets is to use k -fold cross-validation where the data set is divided into several training and validation sets.[25]

In k -fold cross-validation, the labelled data set is divided into k parts. These parts are then used to form k different training and validation set pairs in a way such that each part is once used solely as the validation set and the remaining data is used as the training set. This means that the model is both trained k times and validated k times. Figure 2.6 demonstrates how the different parts are used for training and validation during the different runs. The average of the k validation errors can then be used as a low variance estimate of the model performance.[9]

Using a larger k in k -fold cross-validation means that the training sets will be larger. On the other hand, it also means that the validation sets will be smaller and that the model will be trained more often which can be computationally expensive. Alpaydin [3] suggests that a larger k can be used when the size of the data set, N , is small. Similarly, k can be reduced when



Figure 2.6: An example of 5-fold cross-validation and how the five different parts are used for training and validation during the different runs.

N grows larger. A special case of Cross-validation that can be used with very small data sets is called leave-one-out. In this case $k = N - 1$.

2.7.3 Grid search

One method for finding the optimal parameters is to just try every possible combination of parameter options for a machine learning model. This brute-force approach is called grid search. Since some machine learning models may have real numbered or unbounded parameters, it is often necessary to manually discretize and set a range to the parameter options.[12]

Take a simple model M that takes two parameters a and b as its input, where

$$a \in \mathcal{A} = \{a_1, a_2\}$$

and

$$b \in \mathcal{B} = \{b_1, b_2, b_3\}.$$

For this model grid search would try all the six parameter combinations of a and b that can be obtained with the cartesian product:

$$\mathcal{A} \times \mathcal{B} = \{(a_1, b_1), (a_1, b_2), (a_1, b_3), (a_2, b_1), (a_2, b_2), (a_2, b_3)\}$$

Thus, the time complexity of grid search is equal to the product of cardinalities of the parameter sets. For the model in this example the time complexity is $O(|\mathcal{A}||\mathcal{B}|)$. [12]

Chapter 3

Methods

3.1 Analytics

Like many other online service providers, Yle also collects information about the usage of their service. This provides Yle with useful information which they can utilize when making data-based decisions for improving their service. Since this thesis focuses on the VOD context, only the VOD analytics will be explained below.

Yle's VOD platform, Yle Areena, is an online streaming service which shows video and audio content produced and purchased by Yle. Users are able to watch live broadcasts or content that has already been broadcasted. The time the broadcasted content is available online varies from show to show according to the license. Some content is also only available online through the VOD platform.

The VOD platform is accessible by an internet browser and different apps. Apps are available for all major mobile operating systems as well as some smart TVs and game consoles.

The analytics system collects information about which videos the users are watching, when they hit play and pause, how much they have watched and what devices they are using. This information is stored to a database from which it can easily be queried.

3.2 Surveys

To be able to predict motives, we needed to collect the viewing motives of the users for the training sets. Therefore, two surveys with questionnaires were conducted.

Each survey consisted of a set of identical questions for 10–13 TV shows. To acquire a good training set, TV shows of the surveys were selected so that they would represent a wide range of different genres and formats. In addition, the shows needed to have relatively many views so that there would be enough responses for each show.

The surveys were sent by email invitations to registered Yle users who according to the analytics had began watching at least one of the TV shows in the survey within the past 3 days. In the survey, the users were first asked which of the TV shows they had watched on the VOD platform. Then, for the shows they had watched, they were asked about their reasons and motives to watch the show, if they had watched the show alone or together and where they were watching the show. The motive statements asked by the users in both surveys were the following:

- I want to entertain myself
- I feel the program is entertaining
- I want to stay up-to-date with the latest news
- I want to deepen my knowledge of the world
- I want to improve myself and learn more
- I want content related to where I live
- I watched the show because I was bored
- I picked the show because it was easily available
- I want to find something that has not yet been seen by everyone
- I want to feel being part of a group
- I watched the show together with my friends or family

These statements were not mutually exclusive, so each user could pick 0–11 statements for each show they had watched. Another context feature that was asked for was where the user had watched the show. The options for where the show had been watched were the following:

- At home
- At work/school

- On the move
- Elsewhere

Besides the questions mentioned above, the users were also asked about their age and gender in order to check if the group of answerers were misrepresenting. The first survey was sent to 21,586 users and the second survey was sent to 8,397 users. The survey samples were mutually exclusive. Both surveys were open for one week after the email invitations were sent.

After the surveys had closed, each user's answers for each TV show were connected to the user's viewing history during the time of the survey for that particular TV show. For instance, if a user had answered that they watched news together with family at home to stay up-to-date, it could be connected to the watching time and used device type.

3.2.1 Survey result

The first and second survey received 3090 and 1149 responses respectively. The age and gender distributions for the respondents in the survey can be seen from Figure 3.1. In the first survey, 68% of the respondents were women, and the overall distribution seems to be skewed strongly towards teenage girls and slightly towards middle-aged people. In the second survey, the gender distribution was more equal with 47% being women. The age distributions appeared to be slightly skewed towards middle-aged people.

Table 3.1 shows the distribution of the motive statements for different programs. The 10 first rows in the table respond to the first survey and the 13 latter rows to the second survey. The motive group for the statement is mentioned in the parenthesis behind the statements. From the table it can be seen that many of the surveyed programs were watched because people wanted to be entertained.

Figure 3.2 shows the motive values for the programs that have been calculated as described in Section 3.4. It can be seen that the programs from the first survey were watched more often because of entertainment motives compared to the programs in the second survey. One can also see that information motives are often in contrast with entertainment motives. The figure also shows that pass time and social motives are far less common. In fact, pass time reaches 49% of the viewers only once. Similarly social rises to 37% of the viewers only for one show.

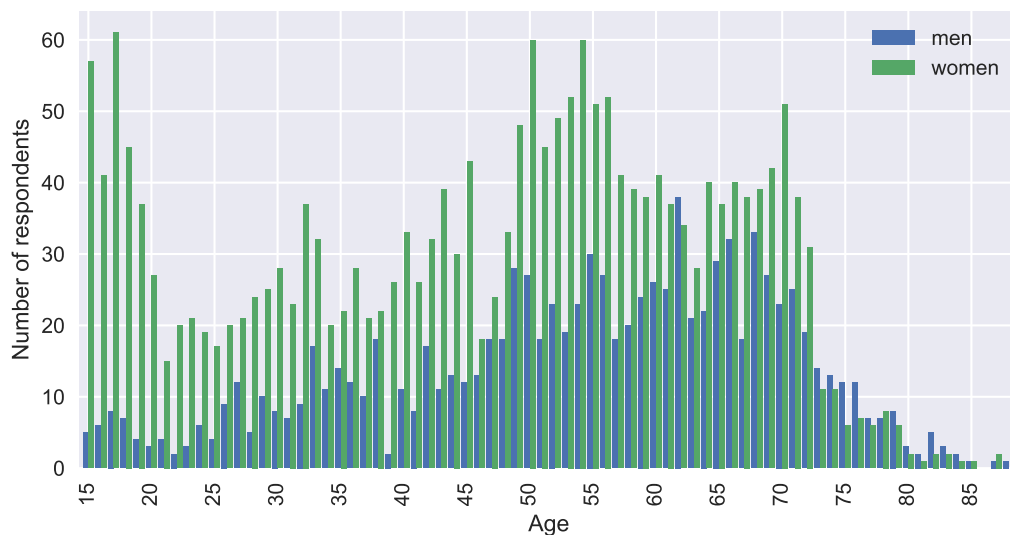
The survey also asked about where the users had watched the program. The distribution of places for watching each surveyed program can be seen in Figure 3.3. The vast majority of the viewers had watched the programs

Table 3.1: The distribution of motive statements in the surveys per program

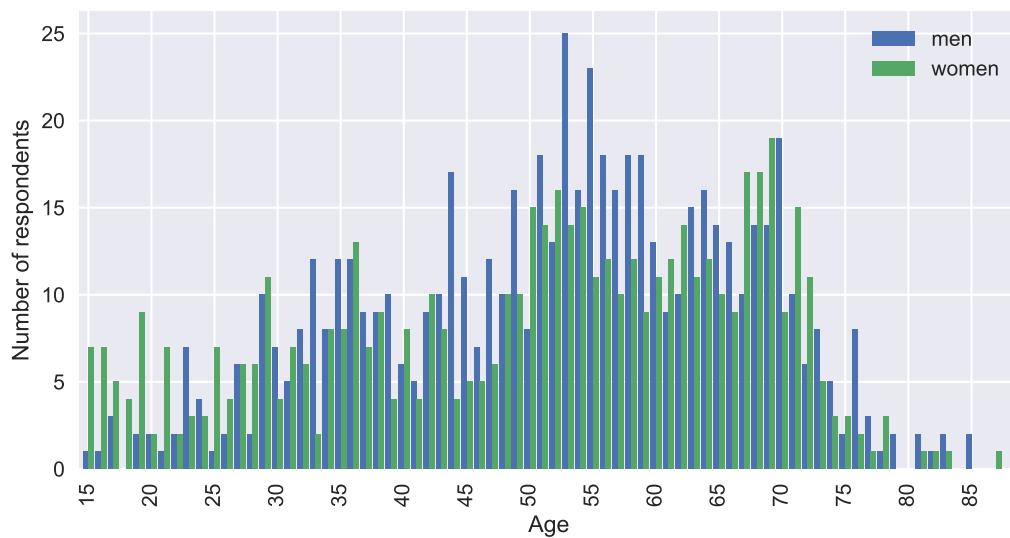
	I want to entertain myself (entertainment)	I feel the program is entertaining (entertainment)	I want to stay up-to-date with the latest news (information)	I want to deepen my knowledge of the world (information)	I want content related to where I live (information)	I want to improve myself and learn more (information)	I watch the show because I was bored (pass time)	I picked the show because it was easily available (pass time)	I want to find something that has not yet been seen by everyone (social)	I want to feel being part of a group (social)	I watch the show together with my friends or family (social)
Au pairit Kanadassa	211	226	14	46	1	24	147	65	7	22	35
Catastrophe	488	628	3	10	3	8	119	74	31	2	42
Holby City	277	371	13	28	2	40	58	44	12	8	19
Midsomer Murders	448	641	6	16	1	16	68	84	7	4	80
NHL videos	35	32	32	9	0	6	17	14	5	2	3
Noin viikon uutiset	598	627	318	179	13	102	78	79	34	40	97
Skam	263	263	34	61	4	53	71	57	34	76	44
Sohvaperunat	389	391	31	15	3	8	92	46	10	78	114
Uusi päivä	408	540	30	12	10	17	135	80	15	21	68
Yle News (20.30 or 18.00 broadcast)	39	42	768	472	90	201	16	97	21	35	114
Avaruuskansiot	10	10	4	19	0	17	3	5	4	1	3
Banshee	35	34	0	1	0	0	4	6	2	0	1
Bonusperhe	107	120	1	6	0	8	17	21	6	1	18
MOT	17	27	109	94	8	62	8	20	22	7	12
Our girl	178	226	2	34	1	16	31	34	20	2	20
Planet Earth II	148	170	31	268	5	176	12	28	25	6	68
Pressiklubi	76	90	97	65	4	36	12	15	9	12	18
Suomi on suomalainen	46	56	20	109	23	80	11	13	13	9	13
Under the Hammer of the Nazis	29	26	13	136	0	99	8	16	20	5	8
Urheilujuttuja	63	71	75	10	1	7	12	27	3	14	7
Uutisvuoto	118	116	44	18	3	16	22	24	7	7	26
Wild at Heart	71	92	0	11	0	6	15	14	6	1	20
World Figure Skating Championships	89	89	26	12	1	10	7	16	2	13	29

at home. This was true for all the surveyed programs. Out of the surveyed programs the ones that were watched the least at home still had 86% of the views from homes. All other locations were used by less than 10% of the viewers of each program. Since most of the users watched the programs from home, the location was considered as an uninteresting feature in this project.

The last surveyed question about the program was about the who if any the program was watched together with. The results for the viewing company can be seen in Figure 3.4. For this question the most common answer was to watch the program alone. The program that was watched least alone was Planet Earth II, for which 35% of the respondents said that they had watched together with family and friends. Although the location contained more variation than the viewing location, this feature was also thought to be too monotonous for this project.



(a) First survey



(b) Second survey

Figure 3.1: Age distributions for male and female respondents of the surveys.

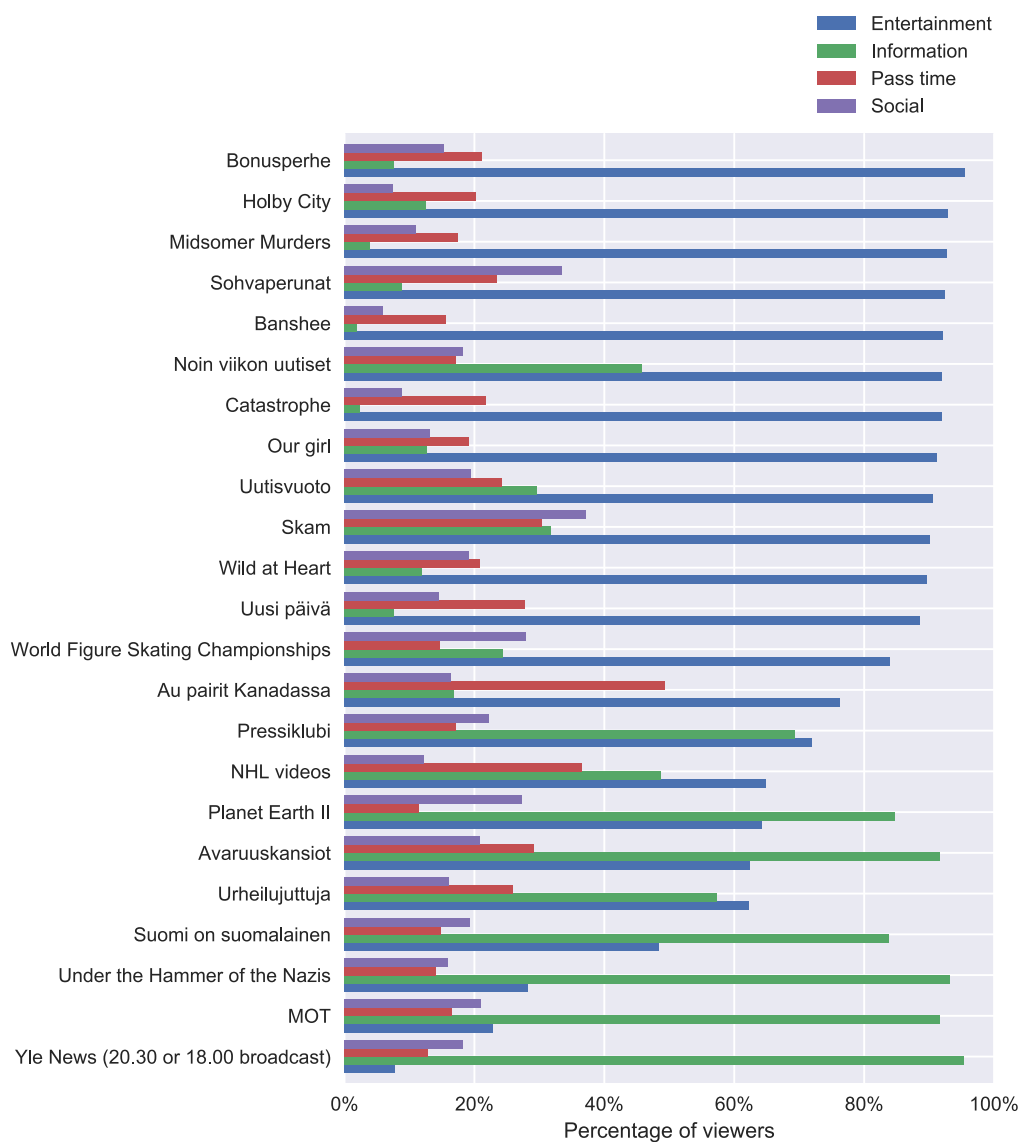


Figure 3.2: The distribution of motives for the surveyed programs.

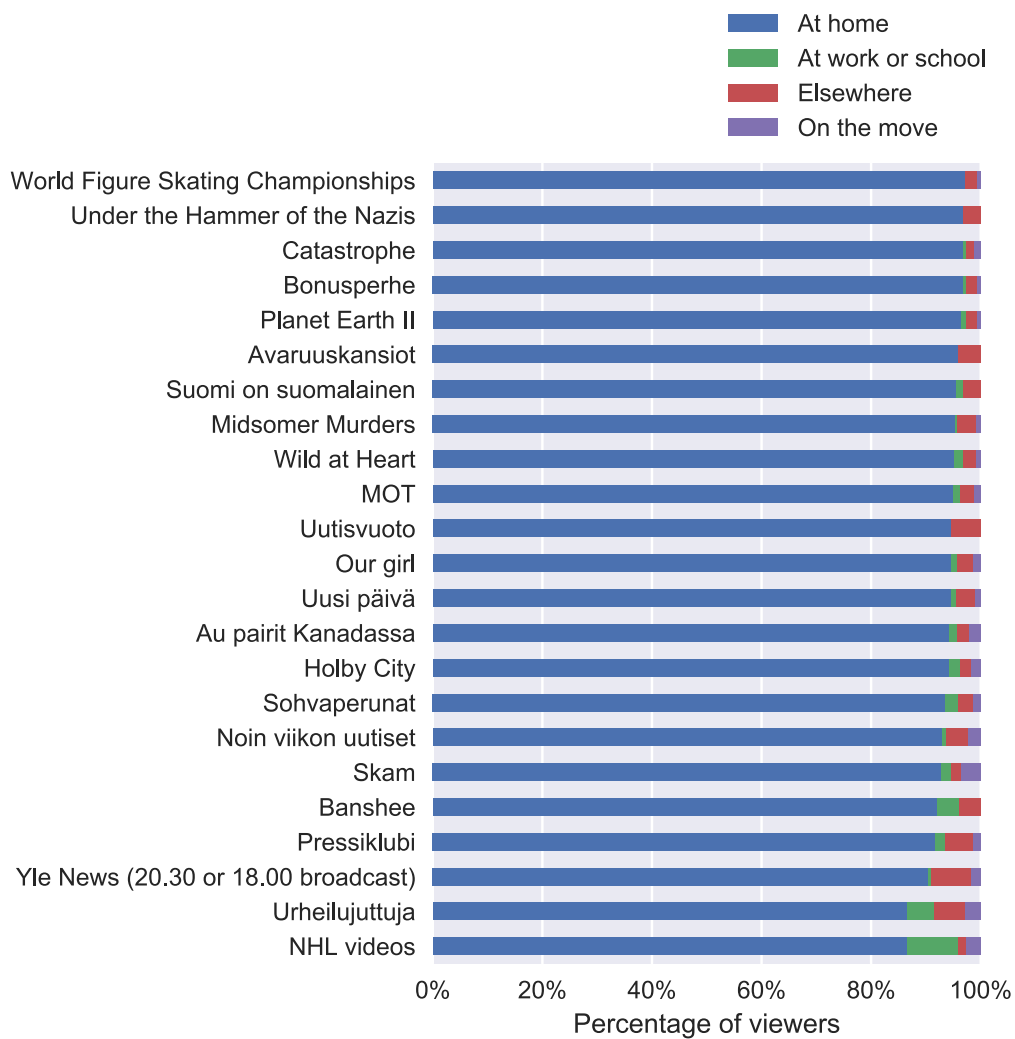


Figure 3.3: The distribution of places for watching the surveyed programs.

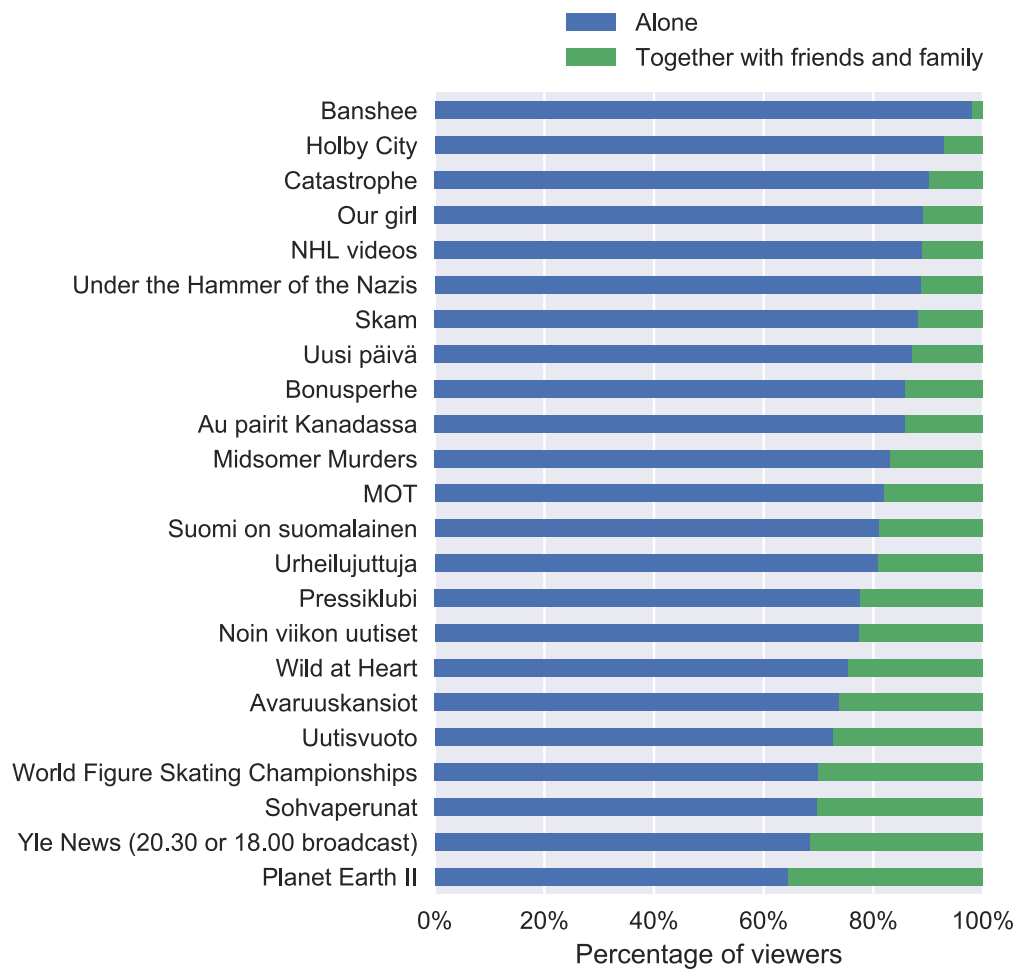


Figure 3.4: The distribution of watching the surveyed programs alone or together with friends and family.

3.3 Features

The features used to describe programs in this project were chosen to be contextual features and genres. In this project, the contextual features describe in what type of context the show was watched. The context information that was available in the analytics database was the timestamp and device details of each watching event. Another option for program features could have been the users that had watched the show. In this case the feature matrix X had been an item-user matrix, the transpose of a user-item matrix often used by recommendation systems [15]. Contextual features and genres were, however, chosen over the item-user matrix since it was considered more time-invariant. Time-invariance was considered important for features due to the difficulty of collecting data. Some programs on the VOD platform have relatively short lifespans, which means that motives data collected for them could become unusable when their features would have changed. For instance in the case of the item-user matrix, the predictions would become unreliable when too few people watch both the programs that the motives were collected for and the programs that motives should be predicted for.

3.3.1 Contextual features

The contextual features that were constructed described how the views for a program were divided between different contexts. The different contexts, in this case, consisted of three components: *device type*, *timeslot* and *day type*. The four different device types were: *phone*, *tablet*, *PC* and *TV*. These device types were already used at Yle and they were used in this project because they provide a good level of granularity. The time slots that were used were 3-hour time slots, the first one being between 0:00 and 2:59, and the last one being between 21:00 and 23:59, using the 24-hour clock notation. The two different day types used were weekday (Monday – Friday) and weekend (Saturday and Sunday). These day types were chosen because there was previous knowledge that the consumption differs on weekends. Additionally, programs that were broadcasted on a certain day affected the online release and watching. Therefore, weekday was chosen as the second day type to ensure that two programs watched on different weekdays would not always have differing features.

These three context components (*device type*, *timeslot* and *day type*) yield 64 mutually exclusive context combinations. For each context combination a feature was created which described the proportion of weekly views that fell into that particular context. For instance, if one program had 100,000

views in the previous seven days and 6,000 of these views took place on weekdays between 12:00 and 14:59 on tablet devices, then the program feature corresponding to the mentioned context would have the value 0.06. The proportion of weekly views was used instead of the number of weekly views since the popularity of programs on the platform varied greatly and the use of proportion can thus normalize the data.

3.3.2 Genres

Besides contextual features, two different program genres were used. The two genres are referred to *Genre A* and *Genre B* in this project. The programs in the training set belong to 8 different genres in *Genre A* and 11 different genres in *Genre B*.

The genre data type is, however, categorical, which meant that the genres needed to be transformed to a numerical form to be used by the machine learning algorithm. This was done with one-hot encoding. In some cases the one-hot encoding was combined with scaling so that the features would be of approximately the same order of magnitude. In these cases, each genre first created its own mutually exclusive binary column, and then the columns were multiplied with a scalar value. When scaling was used, this scalar value was one of the optimized values of the pipeline.

3.4 Motives

The motives data for the training set was collected using two surveys. In these surveys, registered users were asked which programs they had watched and to select 0–11 statements which described the reason why they watched the program. The statements were then grouped to determine the motives the user had for the program in question. The four motives in this project were: *entertainment*, *information*, *pass time* and *social*. The 11 statements were grouped under these motives in the following way:

- Entertainment:
 - I want to entertain myself
 - I feel the program is entertaining
- Information:
 - I want to stay up-to-date with the latest news
 - I want to deepen my knowledge of the world

- I want to improve myself and learn more
- I want content related to where I live
- Pass time:
 - I watched the show because I was bored
 - I picked the show because it was easily available
- Social:
 - I want to find something that has not yet been seen by everyone
 - I want to feel being part of a group
 - I watched the show together with my friends or family

Each user who had chosen at least one statement in a motive group was considered to have that motive when watching the program. Therefore, a user could have 0–4 motives per program. After this the proportion of users with certain motives were calculated for each program. This resulted in four motive values between zero and one for each program. For instance, if 95% of the users watched news with informational motives and 5% watched it with entertainment motives, then the news program would have 0.95 as its information value and 0.05 as its entertainment value. These motive values were combined with the features described in the previous section to create the training set for the model.

3.5 Evaluation

The performance of different models was evaluated with leave-one-out cross-validation. Since, the training set consisted of 23 samples, each model was trained 23 times with the same parameters so that each time one of the samples was left out of the training set. The left out sample was then used for testing while the remaining training set was used to train the model. The test error was then calculated as the root mean of square errors (RMSE) for each sample:

$$\text{RMSE} = \sqrt{\frac{\sum_{j=1}^M (\hat{y}_j^{(i)} - y_j^{(i)})^2}{M}},$$

where M is the number of motives. The error for a set of parameters was obtained by taking the mean of all sample errors. The median and standard deviation of the sample errors was also used to compare the performance of different models.

Chapter 4

Implementation

The goal of this project was to predict watching motives for VOD content using available data. To accomplish this, a training set was created and used to train a model and then the model was used to predict motives for the rest of the programs. This chapter discusses the different parts involved in the implementation and training of the motives predictor.

4.1 Overview

Figure 4.1 shows the different parts of this prediction system.

The available data in this project consisted of the data collected by the analytics system as well as two different content genres. All this data was stored at a VOD database from where it could be queried to create features for programs. Since the genres were nominal, they needed to be converted to quantitative features to be used by the machine learning algorithms.

Information about viewer motives for selected programs was gathered with two surveys. The motives were then combined with different features for the surveyed programs to create a training set. The training set was used to train supervised models. These models could then be given features that were created using data from the database to make predictions for any program.

The following sections will explain the model and parameter selection more in-depth.

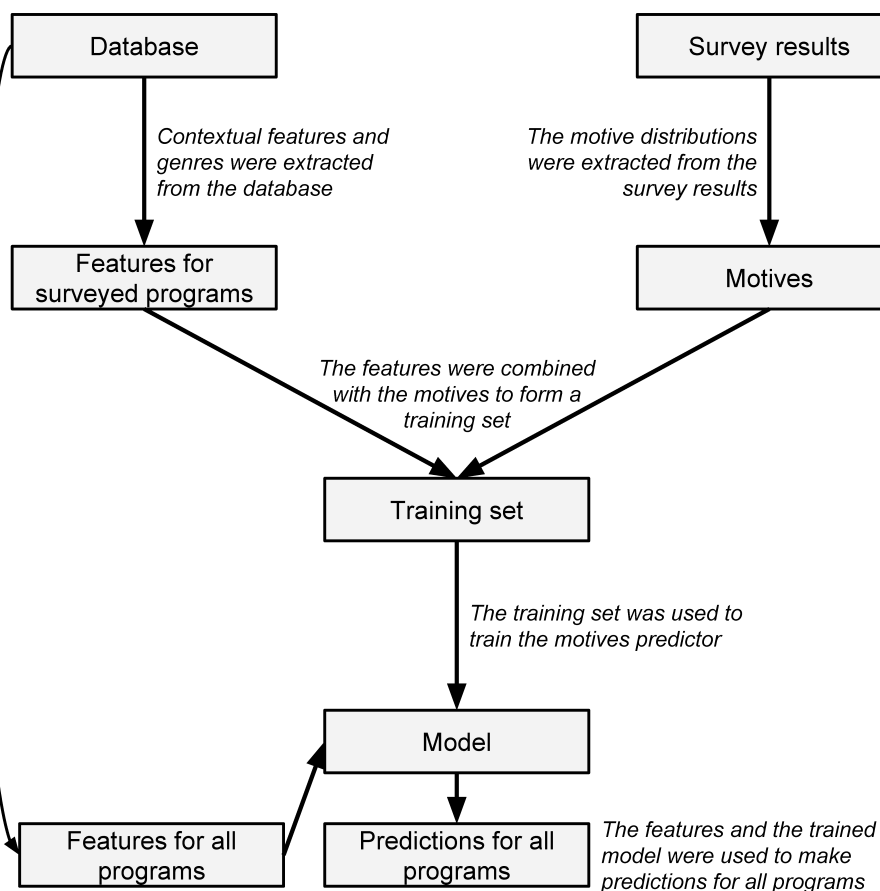


Figure 4.1: The system structure of the motives predictor.

4.2 Prediction system

In order to find out which methods and features improve the prediction results the most, several experiments were conducted with different setups. This section discusses the different variants and parameters that were tried of the prediction system.

4.2.1 Feature sets

One of the goals of this project was to find out how well different features perform to predict motives. To accomplish this the prediction model was trained with different features and feature combinations. The following feature sets were used:

- Contextual features
- Genres
 - Genre A
 - Genre B
 - Both Genre A and Genre B
- Contextual features combined with both Genre A and Genre B

4.2.2 Transforming nominal features

The genres which were a part of the features were nominal, so they needed to be transformed to numerical form before they could be used by the dimensionality reduction. This was accomplished with one-hot encoding to transform the labels to mutually exclusive binary columns. When dimensionality reduction was used, the binary columns were multiplied with a scalar value s in order to scale the binary values to approximately the same size as the contextual features. This was done because features with large variance can affect dimensionality reduction algorithms such as PCA [29]. The scalar value s was optimized among other parameters when it was used.

4.2.3 Dimensionality reduction

When contextual features were used with kernel ridge regression, dimensionality reduction was used to transform the features before training the model. The purpose of dimensionality reduction was to reduce the risk of overfitting, which was substantial since the dataset was very small and there were 64 contextual features. The different methods that were tried were PCA, truncated SVG and NMF. All these methods have a parameter k which specifies the number of dimensions after the reduction. This parameter was also optimized using grid search. Kernel ridge regression models were also trained without dimensionality reduction for comparison.

4.2.4 Regression models

Two different types of regression models were used in this project: kernel ridge regression and neural networks. This subsection discusses the different regression models and the different parameters that they were used with.

Kernel ridge regression

Kernel ridge regression was chosen as one of the regression models due to its regularization and the use of the kernel trick. As mentioned earlier, the training set was very small and therefore regularization is useful to avoid overfitting. The benefit of the kernel trick is that it can turn linear models, such as ridge regression, to nonlinear models. This is very useful if the predicted function is not linear function of its input.[24]

Kernel ridge regression has an α parameter which can be used to adjust the regularization. In addition to α , different kernels can be tried with the model. The tried kernels were:

- Linear kernel:

$$k(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$$

- Polynomial kernel:

$$k(\mathbf{x}, \mathbf{y}) = (\gamma \langle \mathbf{x}, \mathbf{y} \rangle + c_0)^d$$

- RBF kernel:

$$k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2)$$

The parameters α , γ , c_0 and d were all optimized using grid search for the kernels they appeared in.

Parameters The parameters used with kernel ridge regression were:

s Used to scale binary columns

k Number of dimensions the data was reduced to

α Regularization parameter

γ Free parameter for polynomial and RBF kernel

d Degree parameter for polynomial kernel

c_0 Free parameter for polynomial kernel

Neural networks

The second regression method that was used was neural networks. Neural networks was chosen since the activation function can easily limit the output between 0–1 and because the predicted motives might be related. The networks used to predict had either one or two hidden layers. The activation function used for the neurons of the hidden layers was *tanh*. For the output layer *sigmoid* was used as the activation function to automatically scale all the motives between 0 and 1. The optimizing algorithm used for training was RMSProp. In the experiments, several neural networks were trained with different numbers of hidden layers and hidden neurons. prediction result.

Chapter 5

Results

The previous chapter presented the methods for predicting motives. This chapter will explore how the different features, machine learning methods and their parameters will affect the result of the predictions. The first part of this chapter presents the tried parameters for the machine learning methods. This is followed by an explanation of the used baseline and the results for different features and machine learning methods.

5.1 Methods and parameters

5.1.1 Kernel ridge regression

The different values tried for the parameters with kernel ridge regression can be seen in Table 5.1.

The s parameter was only used to scale the binary columns of the transformed nominal data features when both genres and contextual features were used. In other cases the step was skipped, since it would not have any effect. Dimensionality reduction was not used with genres features and thus the k parameter was not used.

5.1.2 Neural networks

Dimensionality reduction was not used with neural networks since a hidden layer can accomplish similar results. The binary columns were not scaled either because it would not significantly affect the trained neural networks model. [11]

The trained networks had either one or two hidden layers. In the case of one hidden layer, there were 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024 or 2048

Table 5.1: Grid search parameters for kernel ridge regression

Parameter	Values
All kernels	
s	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0
k	2, 4, 8, 16
α	0.000001, 0.00001, 0.0001, 0.001, 0.01
Polynomial and RBF kernel	
γ	0.01, 0.1, 1
Polynomial kernel only	
d	2, 4, 8
c_0	1, 10, 100, 1000

neurons in the hidden layer. In the case of two hidden layers, an identical hidden layer was added after the first one, thereby doubling the number of hidden neurons. In both cases, the number of output neurons was four – one for each motive – whereas the number of input neurons was decided by the dimension of the feature data.

5.2 Baseline

A baseline was used in this project to ensure that the results obtained by the other models can be considered meaningful. The baseline was chosen to be a trivial regressor that always predicted the mean values for the motives. Thus, the baseline will be the same for all data sets regardless of the selected features. The results of the baseline serves as a minimum requirement that the models needs to surpass in order to be useful. Any regressor performing worse than the baseline is considered useless since a trivial solution can already provide a better estimate. An acceptable model should however perform significantly better than the baseline. With leave-one-out cross-validation the test RMSE for the baseline was **0.233**.

5.3 Results using context

This section presents the results for predicting motives using contextual features. The methods used were kernel ridge regression with different kernels and dimensionality reductions as well as neural networks with one and two hidden layers.

Table 5.2 shows the RMSE test errors for different kernel dimensionality reduction combinations and Figure 5.1 visualizes the same information using box plots.

Table 5.2: The results for the best kernel ridge regression models using contextual features

method	parameters	μ RMSE	σ RMSE
Baseline		0.233	0.093
Linear kernel	$\alpha=0.01$	0.212	0.125
Linear kernel + NMF	$k=16.0, \alpha=0.01$	0.222	0.123
Linear kernel + PCA	$k=16.0, \alpha=0.01$	0.479	0.050
Linear kernel + Truncated SVD	$k=16.0, \alpha=0.01$	0.212	0.126
Polynomial kernel	$\gamma=1.0, d=2.0, c_0=1.0, \alpha=0.01$	0.198	0.123
Polynomial kernel + NMF	$k=16.0, \gamma=0.1, d=2.0, c_0=1.0, \alpha=0.01$	0.197	0.112
Polynomial kernel + PCA	$k=8.0, \gamma=0.1, d=8.0, c_0=1.0, \alpha=1e-05$	0.182	0.106
Polynomial kernel + Truncated SVD	$k=8.0, \gamma=1.0, d=4.0, c_0=1.0, \alpha=0.001$	0.194	0.111
RBF kernel	$\gamma=1.0, \alpha=0.01$	0.196	0.122
RBF kernel + NMF	$k=16.0, \gamma=0.1, \alpha=0.01$	0.196	0.112
RBF kernel + PCA	$k=8.0, \gamma=1.0, \alpha=0.0001$	0.181	0.110
RBF kernel + Truncated SVD	$k=8.0, \gamma=1.0, \alpha=0.0001$	0.195	0.116

For models with linear kernels the mean and median test errors were slightly lower than for the baseline except when PCA was used for dimensionality reduction. With PCA the best model with a linear kernel performs the worst of all models and has bad predictions for each test sample. For the other linear kernel methods the test errors vary more compared to the baseline, although they on average perform slightly better.

For polynomial and RBF kernels all the models perform quite similarly with mean RMSEs less than 0.2, which is better than the baseline and models with linear kernels. The variation of the test sample errors is however large compared to the baseline. For non-linear kernels, the variation is the biggest for models without dimensionality reduction and the smallest for models using PCA.

Figure 5.2 shows the test errors of neural networks with one hidden layer using contextual features and Table 5.3 shows the test errors and standard deviations for all neural networks using contextual features. With two and four hidden neurons the neural networks perform almost equally well compared to the baseline. When adding more neurons to the hidden layer the

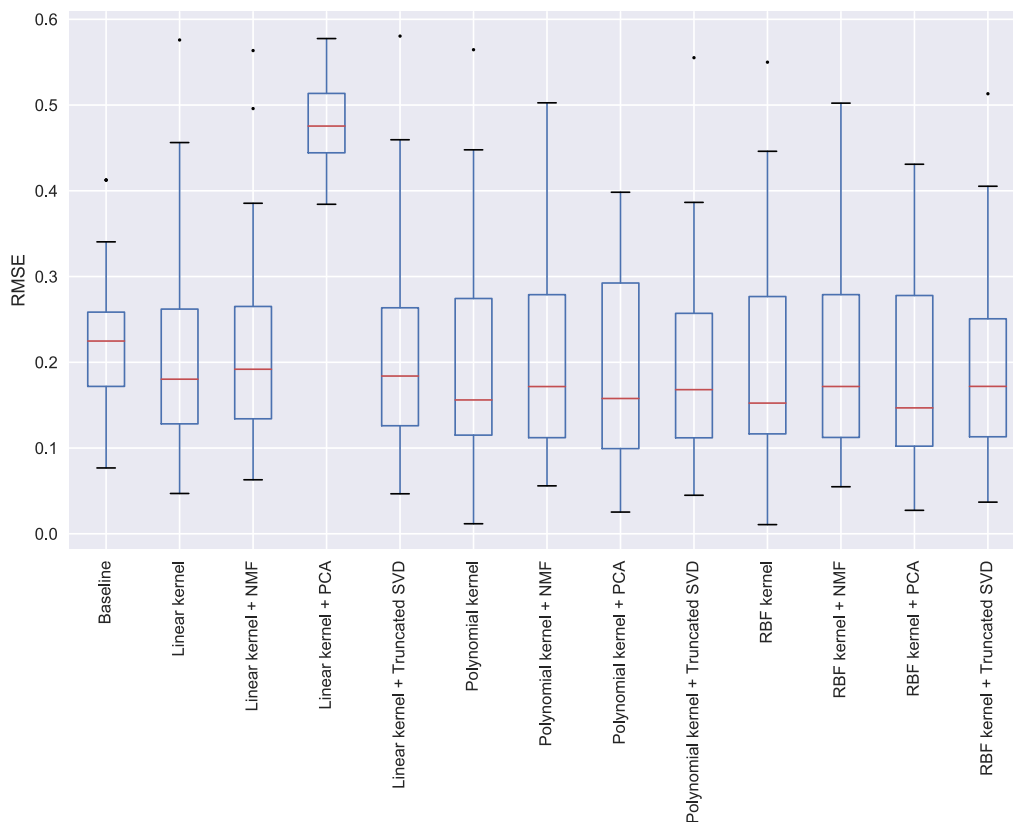


Figure 5.1: Root mean squared error (RMSE) distributions for kernel ridge regressions methods with different kernels and dimensionality reduction methods when using contextual features.

median and mean test error decreases until there are 64 hidden neurons. After 64 hidden neurons the median test error remain more or less the same. Interestingly, the variation of the test error appears to increase almost without an exception when adding more hidden neurons to the network. The mean test RMSE of the best one hidden layer neural network models is almost the same as for the kernel ridge regression counterparts, but the RMSE variation is slightly higher.

Figure 5.3 shows the test errors of neural networks with two hidden layer using contextual features. The results with two layers differ slightly from those with only one hidden layer. With 4–8 hidden neurons in the hidden layers the result is almost the same as for the baseline. The best two hidden layer model has almost the same mean as the best corresponding kernel ridge regression and one hidden layer models. The variation of the test error for the best model is however larger compared to the other methods.

Table 5.3: The results for neural network models using contextual features

hidden neurons	hidden layers	μ RMSE	σ RMSE
Baseline	0	0.233	0.093
2	1	0.231	0.094
4	1	0.231	0.093
8	1	0.226	0.093
16	1	0.223	0.098
32	1	0.212	0.115
64	1	0.197	0.126
128	1	0.197	0.136
256	1	0.210	0.143
512	1	0.195	0.142
1024	1	0.214	0.161
2048	1	0.218	0.168
4	2	0.228	0.094
8	2	0.227	0.097
16	2	0.227	0.107
32	2	0.196	0.132
64	2	0.198	0.140
128	2	0.200	0.149
256	2	0.209	0.161
512	2	0.220	0.169
1024	2	0.216	0.173
2048	2	0.223	0.156
4096	2	0.208	0.154

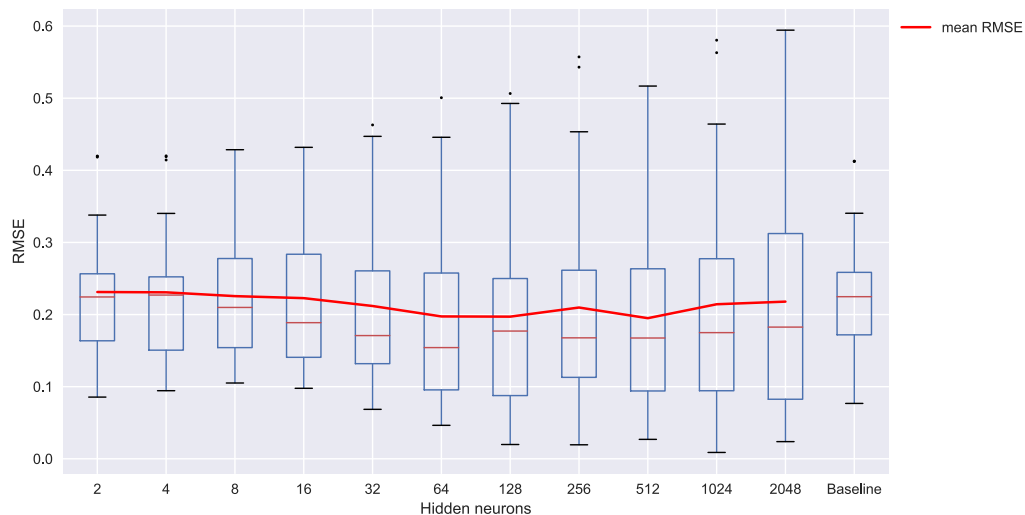


Figure 5.2: Root mean squared error (RMSE) distributions for neural networks with one hidden layer of different sizes when using contextual features.

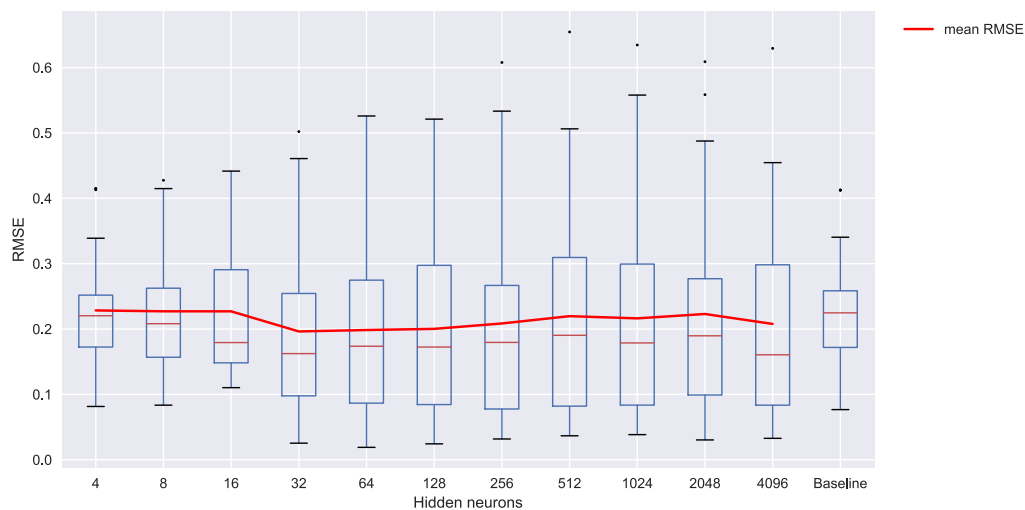


Figure 5.3: Root mean squared error (RMSE) distributions for neural networks with two hidden layers of different sizes when using contextual features.

5.4 Results using genres

This section presents the results for predicting motives using genres. The methods used were kernel ridge regression with different kernels as well as neural networks with one and two hidden layers. Predictions were performed using both genres separately and combined.

5.4.1 Genre A

This subsection presents the results for predicting motives solely with genre A.

The results of predicting motives with kernel ridge regression using only genre A are shown in Table 5.4 and Figure 5.4. All kernels perform better than the baseline with median test RMSE values under 0.1. The linear kernel has a few more test samples that gives bad predictions and therefore its mean test RMSE is higher than those of the polynomial and RBF kernel. The variation of the test RMSE also varies the most for the linear kernel, while the polynomial and RBF kernel have only marginally higher variation than the baseline.

Table 5.4: The results for the best kernel ridge regression models using genre A

method	parameters	μ RMSE	σ RMSE
Baseline		0.233	0.093
Linear kernel	$\alpha=0.01$	0.149	0.158
Polynomial kernel	$\gamma=0.01, d=4, c_0=1000, \alpha=0.01$	0.113	0.096
RBF kernel	$\gamma=0.01, \alpha=0.001$	0.114	0.098

The results of predicting motives with neural networks with one hidden layer using only genre A as features are shown in Figure 5.5 and Table 5.5 shows the test errors and standard deviations for all neural networks using genre A. All networks perform better than the baseline in terms of median and mean test RMSE. The network with the lowest median and mean RMSE has 16 hidden neurons and it has a marginally lower test RMSE than the best kernel ridge regression model. The variation is however larger for this model compared to both the baseline and the kernel ridge regression models with polynomial and RBF kernels.

The results of predicting motives with neural networks with two hidden layers using only genre A as features are shown in Figure 5.6. Similarly to the networks with one hidden layer, the networks with two hidden layers all

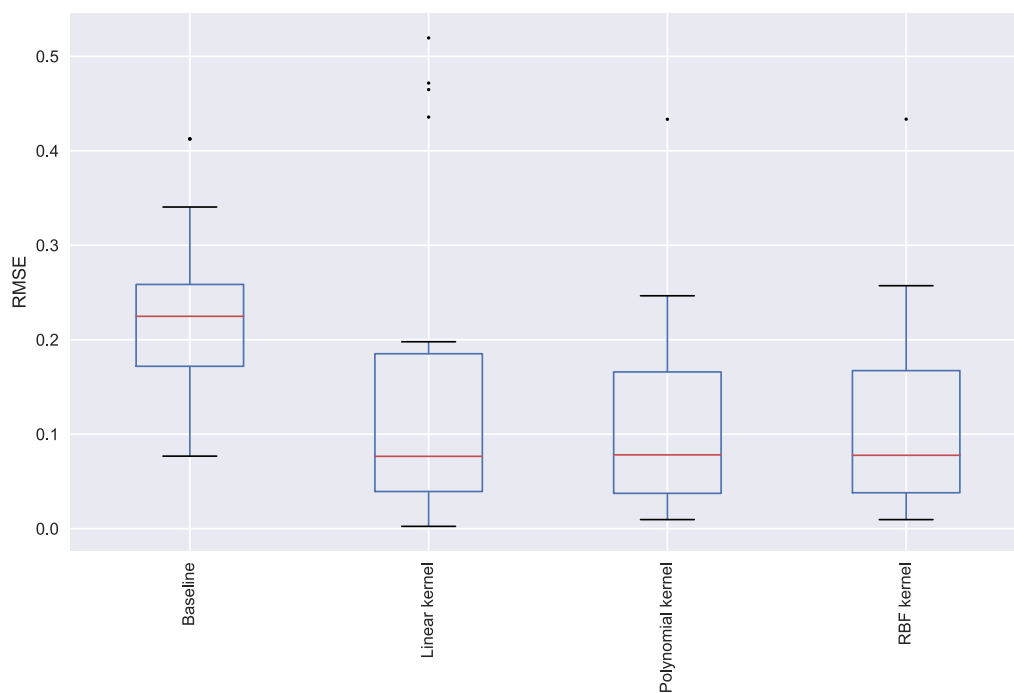


Figure 5.4: Root mean squared error (RMSE) distributions for kernel ridge regressions methods with different kernels and dimensionality reduction methods when using genre A as feature.

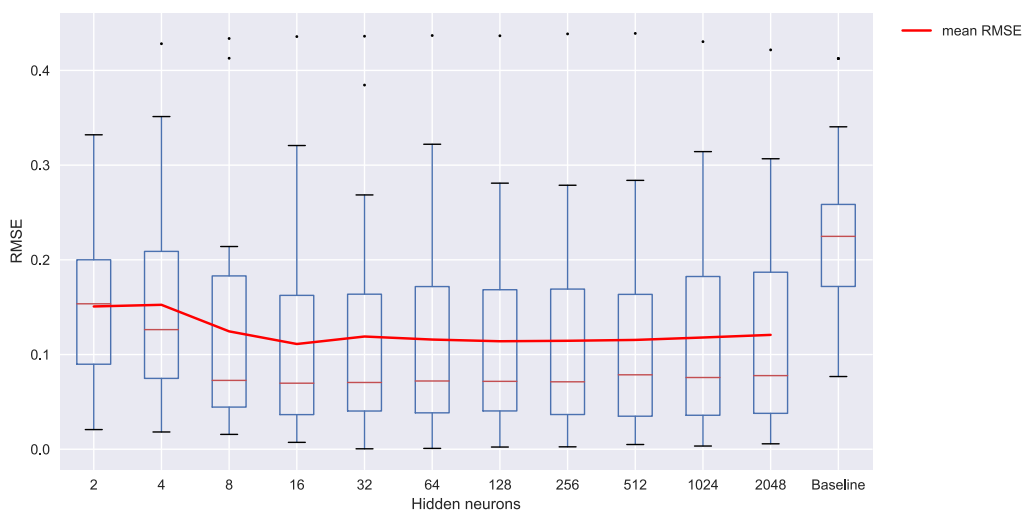


Figure 5.5: Root mean squared error (RMSE) distributions for neural networks with one hidden layer of different sizes when using genre A.

Table 5.5: The results for neural network models using genre A

hidden neurons	hidden layers	μ RMSE	σ RMSE
Baseline	0	0.233	0.093
2	1	0.151	0.079
4	1	0.153	0.107
8	1	0.124	0.111
16	1	0.111	0.105
32	1	0.119	0.114
64	1	0.116	0.107
128	1	0.114	0.103
256	1	0.115	0.104
512	1	0.115	0.104
1024	1	0.118	0.105
2048	1	0.121	0.105
4	2	0.163	0.089
8	2	0.104	0.081
16	2	0.111	0.094
32	2	0.116	0.105
64	2	0.117	0.109
128	2	0.117	0.103
256	2	0.117	0.107
512	2	0.119	0.100
1024	2	0.122	0.104
2048	2	0.123	0.098
4096	2	0.125	0.094

perform better than the baseline when comparing the median and mean test RMSE. The best network has 8 hidden neurons, or 4 hidden neurons per hidden layer and its predictions are the most accurate model using genre A as a feature.

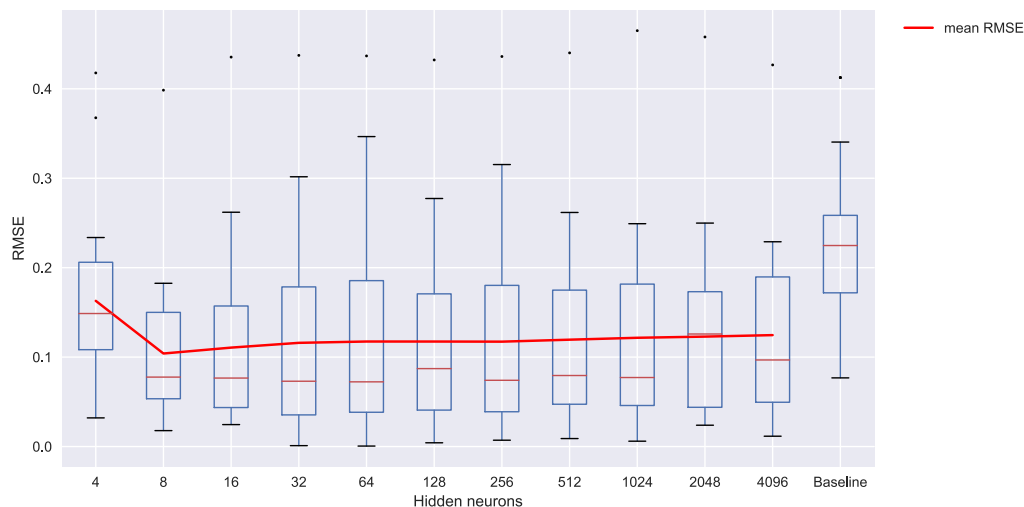


Figure 5.6: Root mean squared error (RMSE) distributions for neural networks with two hidden layers of different sizes when using genre A.

5.4.2 Genre B

This subsection presents the results for predicting motives solely with genre B.

The results of predicting motives with kernel ridge regression using only genre B are shown in Table 5.6 and Figure 5.7. The kernel ridge regression model with the polynomial and RBF kernel perform a little better than their counterparts using genre A. Their mean test RMSE is clearly lower than the one of the baseline and their test RMSE values also vary slightly less than those of the baseline. The linear kernel on the other hand only performs slightly better than the baseline and its test RMSE values also vary much more.

Table 5.6: The results for the best kernel ridge regression models using genre B

method	parameters	μ RMSE	σ RMSE
Baseline		0.233	0.093
Linear kernel	$\alpha=0.01$	0.190	0.203
Polynomial kernel	$\gamma=0.01, d=8, c_0=10, \alpha=1e-06$	0.109	0.090
RBF kernel	$\gamma=0.01, \alpha=0.001$	0.109	0.091

The results of predicting motives with neural networks with one hidden layer using only genre B as features are shown in Figure 5.8 and Table 5.7

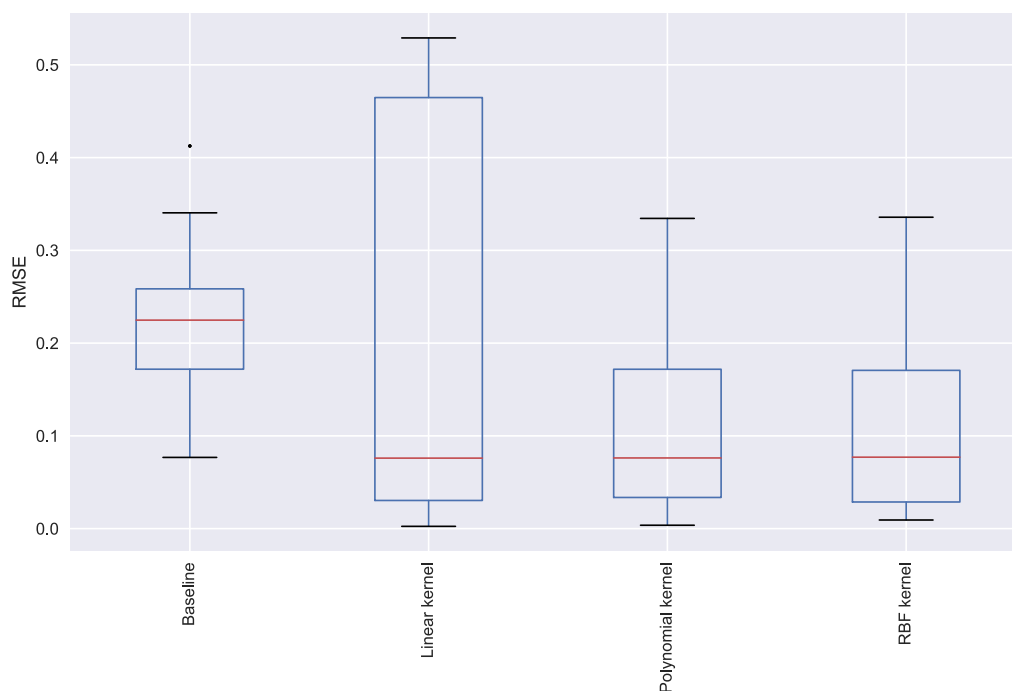


Figure 5.7: Root mean squared error (RMSE) distributions for kernel ridge regressions methods with different kernels and dimensionality reduction methods when using genre B as feature.

shows the test errors and standard deviations for all neural networks using genre B. The best model with one hidden layer has 32 hidden neurons and a mean RMSE of 0.111 which is marginally higher than for the best kernel ridge regressions using the same features. The variation of the results is also larger for all neural networks models.

The results of predicting motives with neural networks with two hidden layers using only genre B as features are shown in Figure 5.9. The best model with two hidden layers has 32 hidden neurons, or 16 per hidden layer and a mean RMSE of 0.111 which is the same as for the best model with one hidden layer. The variation of the RMSE values for the best two hidden layer model is however lower than for the best one hidden layer network and kernel ridge regression model.

Table 5.7: The results for neural network models using genre B

hidden neurons	hidden layers	μ RMSE	σ RMSE
Baseline	0	0.233	0.093
2	1	0.161	0.094
4	1	0.134	0.093
8	1	0.124	0.097
16	1	0.115	0.100
32	1	0.111	0.103
64	1	0.118	0.101
128	1	0.119	0.101
256	1	0.120	0.105
512	1	0.124	0.111
1024	1	0.126	0.109
2048	1	0.126	0.110
4	2	0.163	0.104
8	2	0.132	0.094
16	2	0.127	0.119
32	2	0.111	0.088
64	2	0.129	0.119
128	2	0.121	0.105
256	2	0.118	0.096
512	2	0.128	0.102
1024	2	0.132	0.108
2048	2	0.125	0.102
4096	2	0.126	0.097

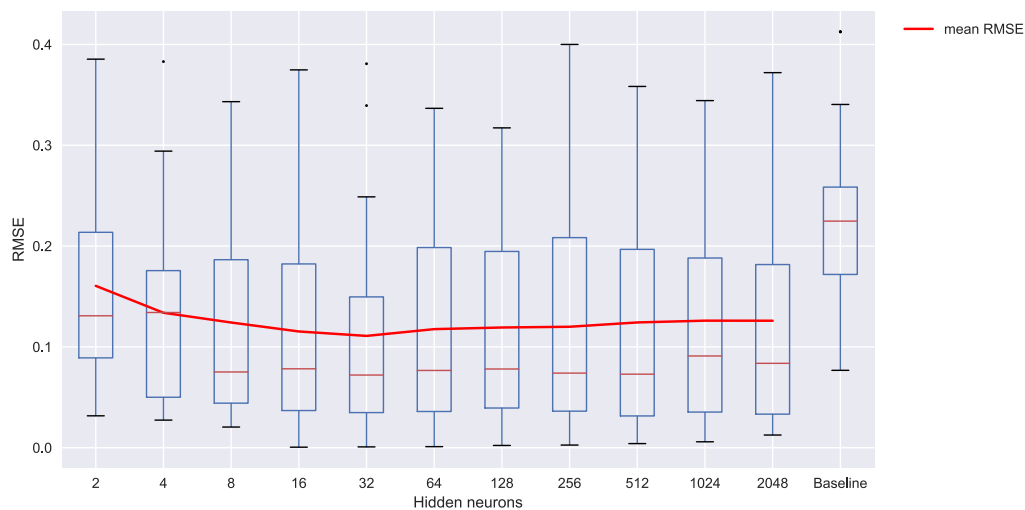


Figure 5.8: Root mean squared error (RMSE) distributions for neural networks with one hidden layer of different sizes when using genre B.

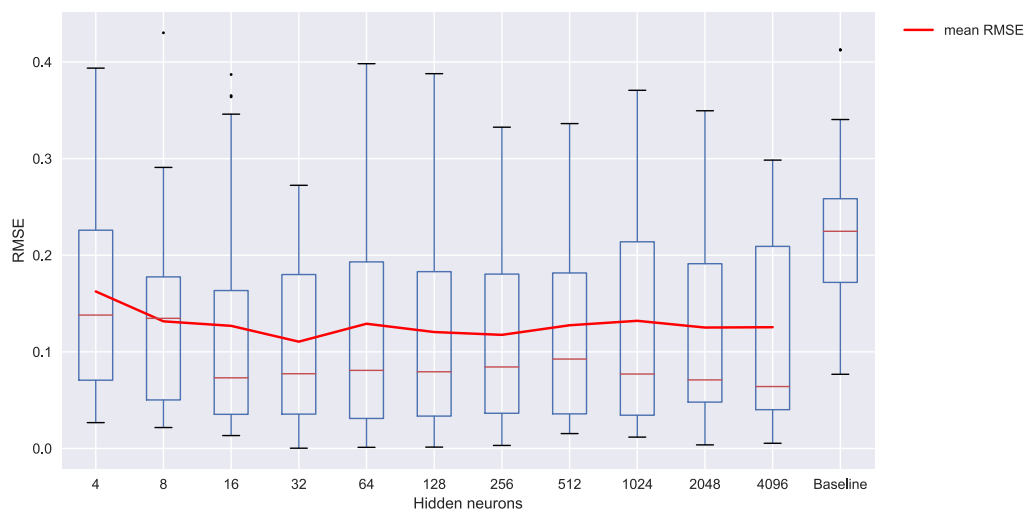


Figure 5.9: Root mean squared error (RMSE) distributions for neural networks with two hidden layers of different sizes when using genre B.

5.4.3 Both genres

The results of predicting motives with kernel ridge regression using both genres are shown in Table 5.8 and Figure 5.10. When using both genres in the prediction the results are quite similar to the results of using only one genre. The kernel ridge regression models using polynomial and RBF kernels have almost the same mean test RMSE as when using only genre A or genre B. The RMSE variation is also just a little bit lower compared to when using only one genre. The linear kernel model does not seem to benefit from two genres either since its mean RMSE is 0.150. Overall, using two genres appears to predict roughly as well as when using only one genre.

Table 5.8: The results for the best kernel ridge regression models using both genres as features.

method	parameters	μ RMSE	σ RMSE
Baseline		0.233	0.093
Linear kernel	$\alpha=1e-06$	0.150	0.146
Polynomial kernel	$\gamma=0.01, d=2, c_0=1, \alpha=0.01$	0.110	0.088
RBF kernel	$\gamma=0.01, \alpha=0.01$	0.110	0.088

The results of predicting motives with neural networks with one hidden layer using both genre A and genre B as features are shown in Figure 5.11 and Table 5.9 shows the test errors and standard deviations for all neural networks using both genres. The lowest mean RMSE is 0.109 for the network with 16 hidden neurons, which is almost the same as the as for kernel ridge regression methods using polynomial and RBF kernels. The RMSE values however vary more for the neural network models compared to the kernel ridge regression models.

The results of predicting motives with neural networks with two hidden layers using both genre A and genre B as features are shown in Figure 5.12. With two hidden layers the lowest mean RMSE is 0.105 for the model with 16 hidden neurons or 8 hidden neurons per hidden layer. This is the lowest test error for models using both genres as features. The standard deviation of the RMSE for this model is also almost the same as for the baseline.

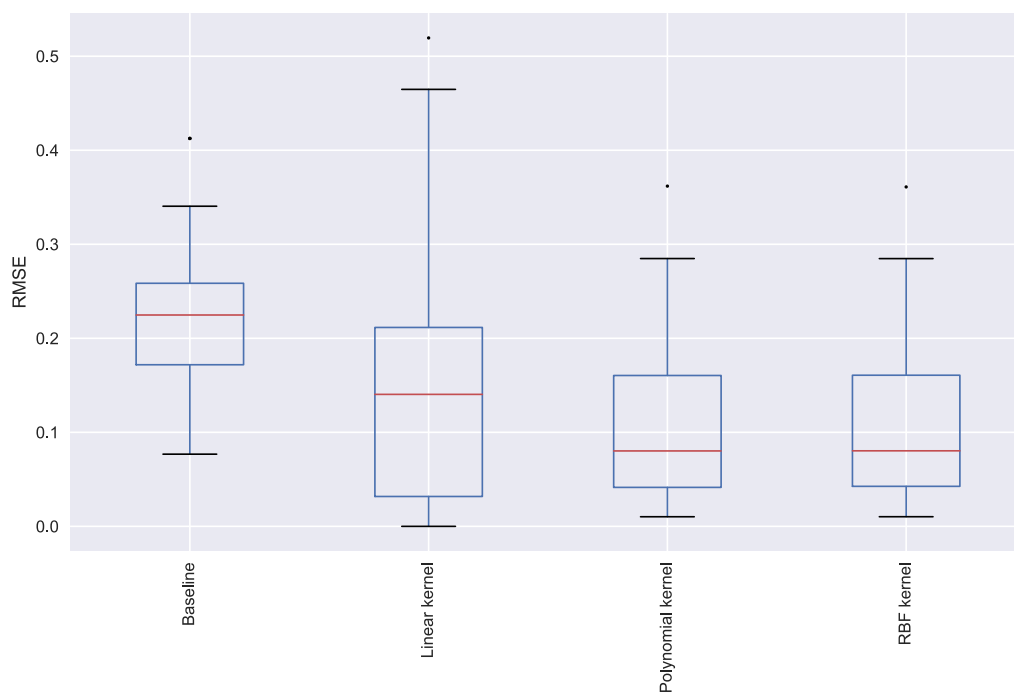


Figure 5.10: Root mean squared error (RMSE) distributions for kernel ridge regressions methods with different kernels and dimensionality reduction methods when using both genres as features.

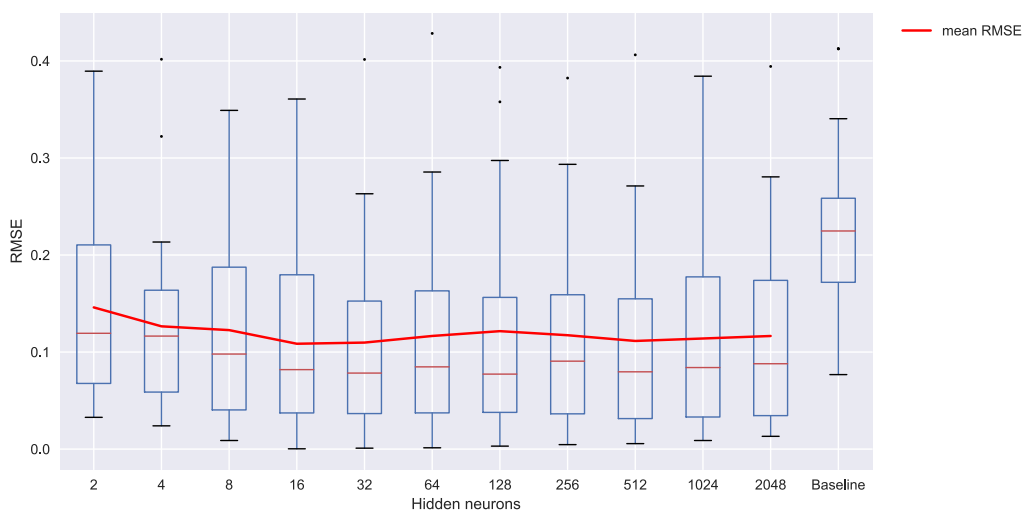


Figure 5.11: Root mean squared error (RMSE) distributions for neural networks with one hidden layer of different sizes when using both genres.

Table 5.9: The results for neural network models using both genres

hidden neurons	hidden layers	μ RMSE	σ RMSE
Baseline	0	0.233	0.093
2	1	0.146	0.095
4	1	0.127	0.092
8	1	0.123	0.100
16	1	0.109	0.098
32	1	0.110	0.100
64	1	0.117	0.105
128	1	0.122	0.109
256	1	0.117	0.099
512	1	0.111	0.099
1024	1	0.114	0.096
2048	1	0.117	0.098
4	2	0.147	0.105
8	2	0.136	0.124
16	2	0.105	0.096
32	2	0.110	0.101
64	2	0.118	0.101
128	2	0.112	0.098
256	2	0.112	0.094
512	2	0.114	0.094
1024	2	0.113	0.095
2048	2	0.112	0.101
4096	2	0.113	0.087

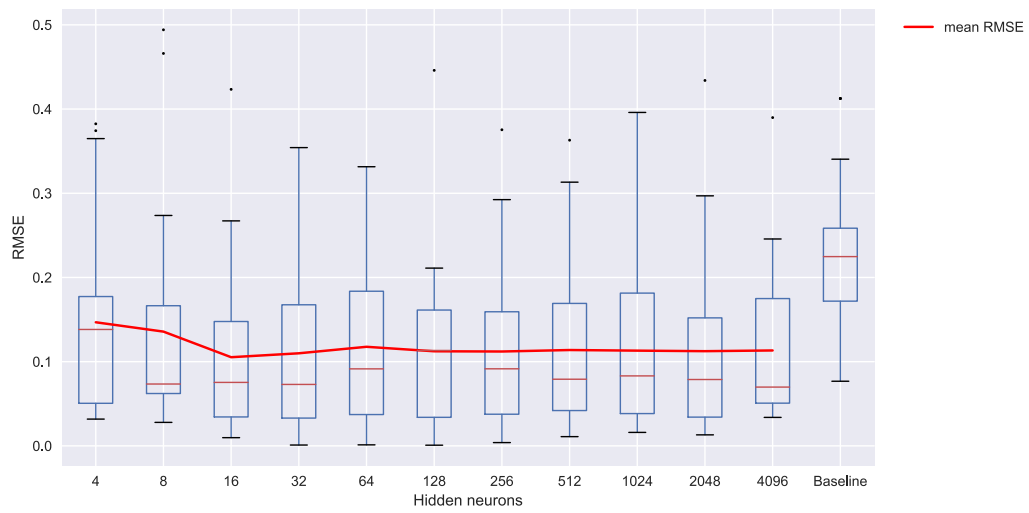


Figure 5.12: Root mean squared error (RMSE) distributions for neural networks with two hidden layers of different sizes when using both genres.

5.5 Results using context and genres

This section presents the results for predicting motives using both contextual features and genres. The methods used were kernel ridge regression with different kernels and dimensionality reductions as well as neural networks with one and two hidden layers.

The results of predicting motives with kernel ridge regression using contextual features combined with both genres are shown in Table 5.10 and Figure 5.13. All kernel dimensionality reduction combinations except the linear kernel with PCA perform better than the baseline. The polynomial and RBF kernels again perform better than the linear kernel. The best results with kernel ridge regression were obtained with the polynomial and RBF kernel using truncated SVD dimensionality reduction which have almost identical results. The standard deviation for these models are also 0.079 and 0.080 respectively, which is less than the standard deviation of the baseline.

Table 5.10: The results for the best kernel ridge regression models using contextual features and both genres

method	parameters	μ RMSE	σ RMSE
Baseline		0.233	0.093
Linear kernel	$s=0.1, \alpha=0.01$	0.132	0.079
Linear kernel + NMF	$s=0.2, k=8.0, \alpha=1e-06$	0.130	0.082
Linear kernel + PCA	$s=0.1, k=8.0, \alpha=1e-06$	0.446	0.054
Linear kernel + Truncated SVD	$s=0.1, k=16.0, \alpha=0.001$	0.129	0.078
Polynomial kernel	$s=0.3, \gamma=0.01, d=2.0, c_0=10.0, \alpha=0.01$	0.112	0.082
Polynomial kernel + NMF	$s=0.3, k=8.0, \gamma=0.01, d=8.0, c_0=1.0, \alpha=0.001$	0.107	0.080
Polynomial kernel + PCA	$s=0.2, k=8.0, \gamma=0.01, d=4.0, c_0=100.0, \alpha=0.01$	0.105	0.079
Polynomial kernel + Truncated SVD	$s=0.3, k=8.0, \gamma=0.01, d=8.0, c_0=1.0, \alpha=0.001$	0.104	0.079
RBF kernel	$s=0.3, \gamma=0.01, \alpha=0.001$	0.112	0.082
RBF kernel + NMF	$s=0.3, k=8.0, \gamma=0.01, \alpha=0.001$	0.111	0.076
RBF kernel + PCA	$s=0.2, k=8.0, \gamma=1.0, \alpha=0.0001$	0.110	0.072
RBF kernel + Truncated SVD	$s=0.3, k=8.0, \gamma=0.01, \alpha=0.0001$	0.104	0.080

The results of predicting motives with neural networks with one hidden layer using contextual features combined with both genres are shown in Figure 5.14 and Table 5.11 shows the test errors and standard deviations for all neural networks using all available features. The lowest mean RMSE with one hidden layer is 0.097 for the network with 8 hidden neurons, which is the lowest mean RMSE recorded in this project. The standard deviation for this model is 0.091 which is also slightly lower than the standard deviation of the baseline.

The results of predicting motives with neural networks with two hidden layers using contextual features combined with both genres are shown in Figure 5.15. The lowest mean RMSE is 0.100 for the neural network with 16 hidden neurons or 8 hidden neurons per hidden layer. This is not as good as

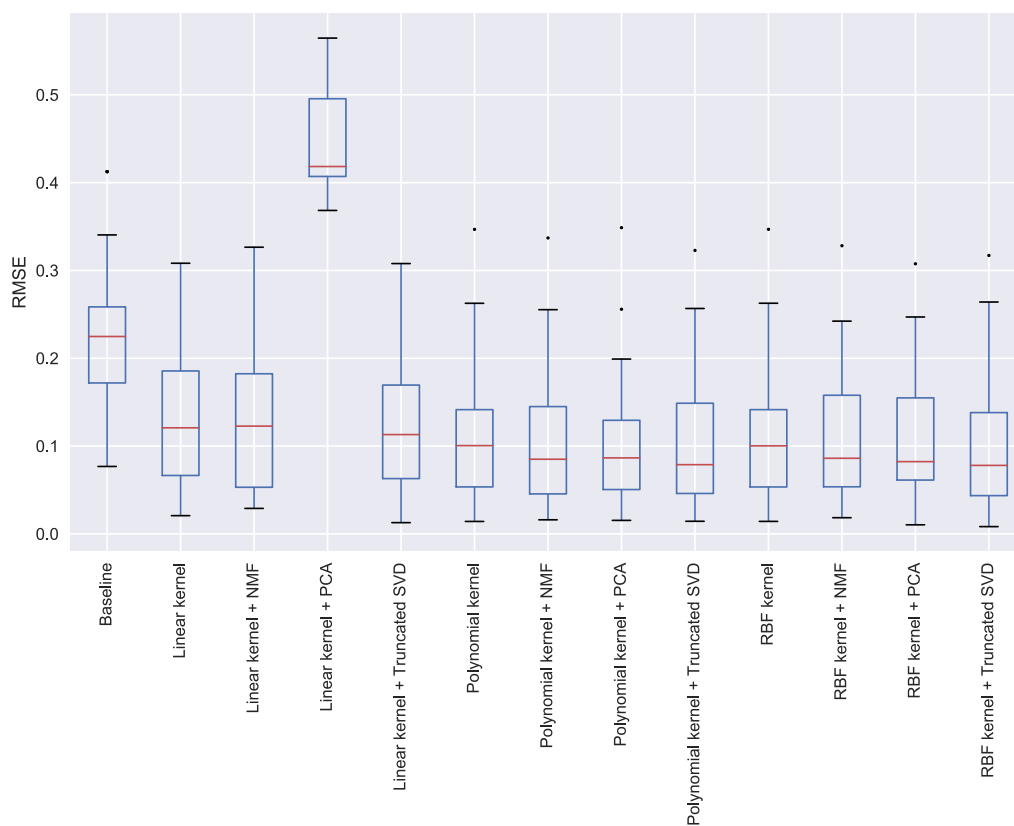


Figure 5.13: Root mean squared error (RMSE) distributions for kernel ridge regressions methods with different kernels and dimensionality reduction methods when using contextual features and both genres.

the best result for the neural network with one hidden layer, but it is better than the best kernel ridge regression methods.

Table 5.11: The results for neural network models using contextual features and both genres

hidden neurons	hidden layers	μ RMSE	σ RMSE
Baseline	0	0.233	0.093
2	1	0.128	0.083
4	1	0.104	0.078
8	1	0.097	0.091
16	1	0.107	0.103
32	1	0.101	0.090
64	1	0.113	0.106
128	1	0.112	0.092
256	1	0.120	0.106
512	1	0.131	0.110
1024	1	0.133	0.115
2048	1	0.141	0.118
4	2	0.110	0.072
8	2	0.115	0.095
16	2	0.100	0.088
32	2	0.117	0.105
64	2	0.116	0.097
128	2	0.128	0.115
256	2	0.130	0.108
512	2	0.134	0.114
1024	2	0.133	0.118
2048	2	0.128	0.108
4096	2	0.135	0.104

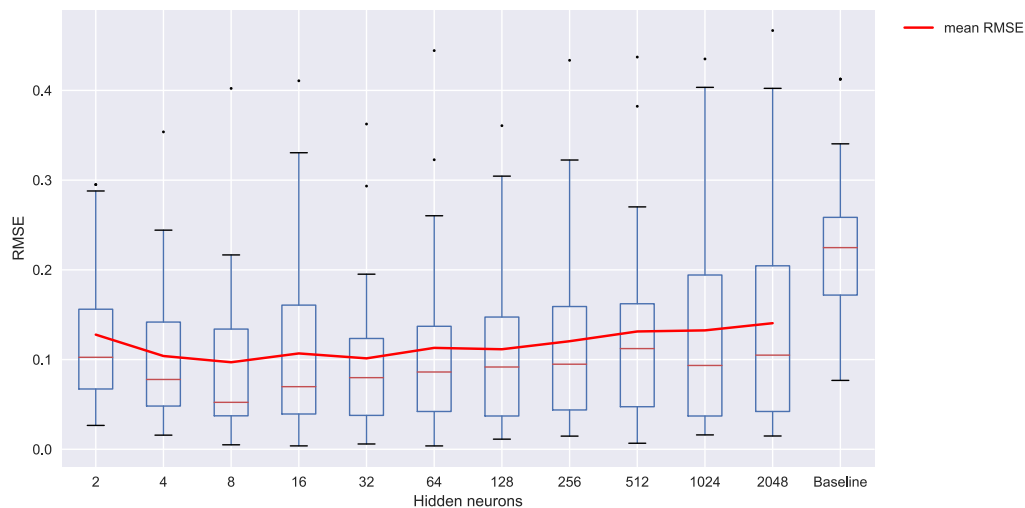


Figure 5.14: Root mean squared error (RMSE) distributions for neural networks with one hidden layer of different sizes when using contextual features and both genres.

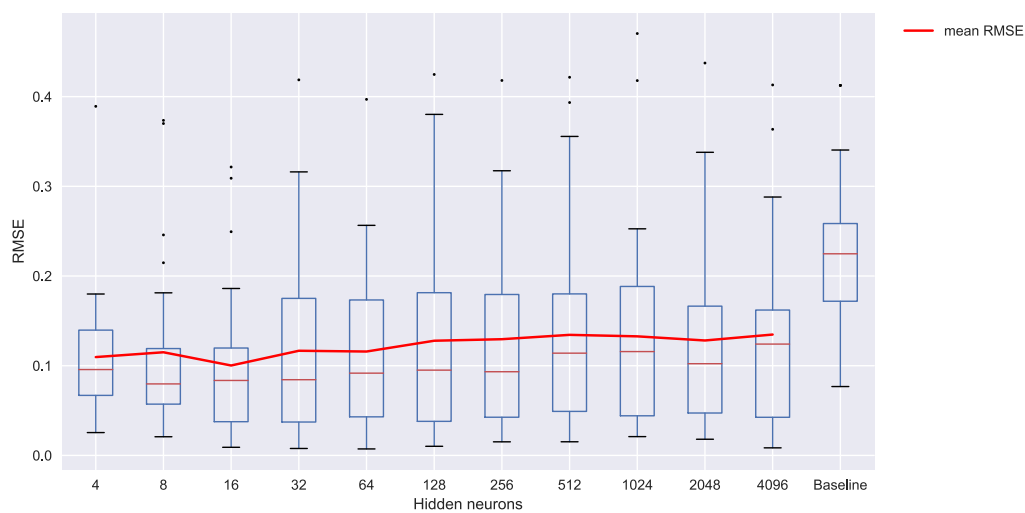


Figure 5.15: Root mean squared error (RMSE) distributions for neural networks with two hidden layers of different sizes when using contextual features and both genres.

5.6 Summary of results

This section summaries the results of the previous sections to enable easy comparison of the different results.

Figure 5.16 shows heat map of the best results obtained with each data set and method. From the heat map it can be seen that contextual features alone can only provide slightly better predictions than the baseline. The genres alone or together provide clearly better results compared to contextual features with the exception of kernel ridge regression with the linear kernel. Combining the contextual features with the genres only improves the results slightly compared to the genres. In general the best neural network model appear to predict motives slightly better than its kernel ridge regression counterpart. Kernel ridge regression with the linear kernel always performs worse than the other methods and especially bad in combination with PCA.



Figure 5.16: The heat map shows how different machine learning methods and features manage to predict motives.

Chapter 6

Discussion

This thesis studied how well motives can be predicted for VOD content using different types of data and machine learning methods. This chapter answers the research questions set in the introduction and discusses how the results can be generalized. The chapter also compares the result to previous research, inspects the biggest limitations that apply to motive prediction and looks at what can be studied in the future.

6.1 Predicting motives

The first research question was “Can viewing motives be predicted for content with contextual consumption data and genres?”. The results show that the best model was able to predict motives with a mean RMSE of 0.097. This is a significant improvement over the mean RMSE of the baseline which was 0.233.

The first research question also contained the subquestions “How good is the prediction when using (a) only contextual data, (b) only genre data (c) both contextual and genre data?”. It turned out that when only using contextual data the performance of the predictor was at best a bit better than the baseline with a mean RMSE of 0.181.

When using only genre data the lowest mean RMSE values were 0.104, 0.111 and 0.109 for genre A only, genre B only and both genres combined respectively. Compared to contextual data, genres perform clearly better. The different genres perform approximately equally well, which is a sign that they contain a lot of overlapping information. Surprisingly, only using genre A alone results in a better predictor than using both genre A and B together. This could be explained by the model not finding the best features from the combined genres and therefore ending up with an average between the results

of the of the individual genres. The results however differ very little, so it could also just be due to noise in the results.

The best model was, however, obtained by using both contextual and genre data. The lowest mean RMSE of 0.097 is a small improvement to the mean RMSE values of the genre predictors. This shows that there is some advantage of using both features for predicting motives, although genres appear to explain most of the result. This could indicate that the connection between context and motives is not very strong, although the viewing context is able to explain something that cannot be explained with genres alone.

The second research question was “Which machine learning method suits best for predicting motives?”. In most cases the performance of the best kernel ridge regression model was very close to the performance of the best neural network model, stating that there was no major difference between the performance of the different methods. One could however argue that neural networks suit slightly better since the activation function of the output layer can scale the motive values between zero and one. In fact, the best performing model was a neural network with 8 hidden neuron in one hidden layer.

The parameters of the machine learning methods also affected the outcome of the predictors. For kernel ridge regression the polynomial kernel and RBF kernel performed the best with almost identical results, whereas the linear kernel always performed worse. The dimensionality reduction appeared to improve the kernel ridge regression models slightly. In most cases, PCA and truncated SVD worked the best. An interesting exception was however the combination of PCA and the linear kernel, which worsened the result drastically. This could be due to the centering and principal components failing to preserve some linear relations with the motives. For neural networks the number of hidden neurons and the number of hidden layers did not affect the results considerably. In nearly all experiments the optimal number of hidden neurons was either 8 or 16 regardless of the number of hidden layers.

6.2 Generalization of the result

The data for this project was gathered from the Yle VOD platform which contains a wide range of programs. Since genres play an important role in the trained model, it could potentially be used for predicting motives for another VOD platform. This would however require the same features that have been used in this project. It is worth to mention that the motives are strongly coupled with the content, so if a platform only offers a limited type of content then the motives to use the service will likely also be limited. For

instance, if most of the content is very entertaining the motives to watch the contents will also mostly be about entertainment. Therefore, predicting motives for the content may not even make sense.

Other types of online services that have same type of data of their content could potentially use the same techniques to predict motives. Similarly to the VOD services, the content should cover a wide range with different consumption motives.

6.3 Comparison to previous research

Motives has been studied extensively in media, but the research usually limits to finding certain key motives for using a type of media or genre. Previous studies by Papacharissi and Mendelson [19] on motives for watching reality TV and Rubin and Perse [23] on motives for watching soap operas do suggest there is a strong link between viewing motives and the genre.

The only known study that has studied prediction of motives is by Kärkkäinen [16] who explored how motives can be predicted with machine learning for users at specific time blocks using the users' read articles, watched videos and used devices. The results showed, however, that most of the motives were predicted with the same accuracy as the baselines which predicted the most common motive. The motives that performed better than the baseline were information seeking and enjoying or relaxing. Kärkkäinen also found in the survey result that the motives depended strongly on the time of the day.

In this project, the motive was combined with the content and thereby motive distributions were predicted for content instead of users and time blocks. The results for predicting motives with only contextual data were not too good, which contradicts with Kärkkäinen's result that motives depend strongly on the time of the day. This could be because due to people answering surveys differently than they actually act or due to the content also including articles. The different statement questions used could also be a reason behind the contradiction.

6.4 Limitations

The biggest limitation for predicting motives was the access to training data and the difficulty to collect it. In this project the only way to collect motives information was through surveys, which limited the number of answers. Another aspect that limited the training set size was that the predictions were made on aggregated data. This also meant that hundreds of responses per

program were required to get reliable motive distributions. To receive enough responses for a program, it needed to have enough registered viewers, which meant that only the most popular programs could be used in the surveys. Since the surveyed users and the data came from only one platform, the final training set became very small. This made training and evaluating of the predictors a challenging task.

Another limitation was the contextual features used for predicting. The problem with the contextual features was that it takes a few days for them to accumulate and that they change over time. Since the contextual features essentially describe the distribution between different devices and times, they are also unreliable for programs with very few views.

The representativeness of the survey respondents can also be seen as a limitation in this project. For the first survey 68% of the respondents were female and many of them were teenage girls. It can however be argued that different programs attract different demographics and that for the surveyed programs the demographics may not be skewed.

6.5 Future work

To avoid the biggest limitation in this project which was the small training set, information about motives should be collected differently. One option would be to ask some users for their reason to watch it before playing the content. This could drastically increase the training set size since data could be collected from any users and not just a small subset.

In the future, predicting motives could be attempted with different features for the content. It would be interesting to see how well motives could be predicted using a item-user matrix. Features extracted from the video and audio of the could also be useful in predicting motives. These experiments would still suffer from the same limitations faced in this project.

Another interesting topic would be using content from multiple VOD services to predict motives. It could however be difficult to gather and combine data from all the different services, not to mention gaining access to the data of the services.

Chapter 7

Summary

The user motives are valuable information for the service provider since it can inform why users choose to watch content. Additionally, the information can be used to determine the diversity of the provided content. The goal of this thesis was to explore if viewing motives can be predicted for programs on a VOD service and to evaluate how different features and machine learning methods affect the result.

In previous work, Kärkkäinen explored how to predict motives for users during time blocks (morning, afternoon, early evening, late evening) using read articles, viewed programs and used devices. The accuracy in the study was however mostly the same level as the used baseline which predicted the most common option.

In contrast to previous work, this thesis studied the viewing motive distributions for programs using contextual features and genres. The contextual features describe how the views of the program were distributed over different time slots, weekdays and weekends as well as different devices. This data was fetched from the VOD services analytics database. The motives for the predicted programs were gathered from active registered users using two surveys. In the surveys, the users were asked about their motives for watching a selected set of programs.

The motives for programs were predicted using kernel ridge regression and neural networks. Different dimensionality reduction methods were also tried with kernel ridge regression and contextual features.

The first research question in this thesis was if viewing motives can be predicted for content with contextual consumption data and genres. Of the different trained models, the best result for predicting motives was acquired with a neural network with 8 hidden neurons in one hidden layer. This network had a mean RMSE 0.097 in the cross-validation which is a significant improvement over the mean RMSE of the baseline, which was 0.233. This

shows that useful predictions can be made with low accuracy.

The second research question in this thesis was which machine learning method suits best for predicting motives. In the tests, kernel ridge regression models performed nearly as well as neural networks, and the dimensionality reduction methods improved the results slightly when used. The two genres used as features were able to predict motives better than the contextual features. The best result was however obtained by combining all the features.

In future studies motive prediction could be attempted with an item-user matrix or features extracted from video or audio. Another thing that could be attempted is motive prediction for content on multiple different VOD platforms.

Bibliography

- [1] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.
- [2] Gediminas Adomavicius and Alexander Tuzhilin. *Context-aware recommender systems*. Springer, 2011.
- [3] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2 edition, 2010.
- [4] Christopher M Bishop. *Pattern recognition and machine learning*. Springer, 2006.
- [5] Beverly A Bondad-Brown, Ronald E Rice, and Katy E Pearce. Influences on tv viewing and online user-shared video use: Demographics, generations, contextual age, media use, motivations, and audience activity. *Journal of Broadcasting & Electronic Media*, 56(4):471–493, 2012.
- [6] Kenneth Burke. *A grammar of motives*. Univ of California Press, 1969.
- [7] Anind K. Dey and Gregory D. Abowd. Towards a better understanding of context and context-awareness. In *Handheld and ubiquitous computing*, pages 304–307. Springer, 1999.
- [8] Paul Dourish. What we talk about when we talk about context. *Personal and ubiquitous computing*, 8(1):19–30, 2004.
- [9] Seymour Geisser. *Predictive inference*. Chapman and Hall/CRC, 1 edition, 1993.
- [10] Gary Hanson and Paul Haridakis. Youtube users watching and sharing the news: A uses and gratifications approach. *Journal of Electronic Publishing*, 11(3), 2008.
- [11] Simon Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall, 2 edition, 2004.

- [12] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University, 2003. URL <https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>.
- [13] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Springer, 2013.
- [14] Ian T Jolliffe. A note on the use of principal components in regression. *Applied Statistics*, pages 300–303, 1982.
- [15] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.
- [16] Kimmo Kärkkäinen. Predicting demographics and motives of website users. Master’s thesis, Aalto University, Finland, 2016.
- [17] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.
- [18] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [19] Zizi Papacharissi and Andrew L Mendelson. An exploratory study of reality appeal: Uses and gratifications of reality tv shows. *Journal of Broadcasting & Electronic Media*, 51(2):355–370, 2007.
- [20] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.
- [21] Matthew Pittman and Kim Sheehan. Sprinting a media marathon: Uses and gratifications of binge-watching television through netflix. *First Monday*, 20(10), 2015.
- [22] Alan M Rubin. Television uses and gratifications: The interactions of viewing patterns and motivations. *Journal of Broadcasting & Electronic Media*, 27(1):37–51, 1983.
- [23] Alan M Rubin and Elizabeth M Perse. Audience activity and soap opera involvement a uses and effects investigation. *Human Communication Research*, 14(2):246–268, 1987.

- [24] Alberto Ruiz and Pedro E López-de Teruel. Nonlinear kernel-based statistical pattern analysis. *IEEE Transactions on Neural Networks*, 12(1):16–32, 2001.
- [25] Giovanni Seni and John F Elder. Ensemble methods in data mining: improving accuracy through combining predictions. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 2(1):1–126, 2010.
- [26] Angus Stevenson and Christine A. Lindberg. *The new oxford American dictionary*. Oxford University Press New York, 3 edition, 2010.
- [27] Gilbert W Stewart. On the early history of the singular value decomposition. *SIAM review*, 35(4):551–566, 1993.
- [28] Gilbert Strang. *Introduction to linear algebra*. Wellesley-Cambridge Press, 4 edition, 2009.
- [29] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.