Deliverable 5.1

Project Title	Next-Generation Hybrid Broadcast Broadband
Project Acronym	HBB-NEXT
Call Identifier	FP7-ICT-2011-7
Starting Date	01.10.2011
End Date	31.03.2014
Contract no.	287848
Deliverable no.	5.1
Deliverable Name	ANALYSIS: Multi-User, Multimodal & Context Aware Value Added Services
Work package	5
Nature	Report
Dissemination	Public
Authors	Oskar van Deventer (TNO), Mark Gülbahar (IRT), Sebastian Schumann, Radovan Kadlic (ST), Gregor Rozinaj, Ivan Minarik (STUBA), Joost de Wit (TNO), Christian Überall, Christian Köbel (THM),
Contributors	Jennifer Müller (RBB), Jozef Bán, Marián Beniak, Matej Féder, Juraj Kačur, Anna Kondelová, Luboš Omelina, Miloš Oravec, Jarmila Pavlovičová, Ján Tóth, Martin Turi Nagy, Miloslav Valčo, Mário Varga, Matúš Vasek (STUBA)
Due Date	30.03.2012
Actual Delivery Date	12.04.2012

Table of Contents

1. General introduction

This document is the first deliverable of the HBB-Next project for Work Package 5 (WP5) on multi-user and context-aware personalisation. It provides an analysis of the relevant state of the art and the problem statements that will form the basis for further work in WP5.

The key terms in this document are „multi-user“, „multi-modal“, „context-awareness, „personalisation“ and “recommendation”. HBB-Next aims at an environment where multiple users share a single television screen and a single HBB-based television service. The fact that there are multiple users interacting with the service introduces several challenges.

First of all, how do the users interact with the multi-user service? Whereas other activities in HBB-NEXT consider tablet and smartphone devices for individual users to interact with the service, WP5 studies solutions that do not require separate devices, but that use voice control and gesture control instead. These voice-recognition and movement-recognition-based controls are referred to as „multimodal“ controls. A challenge is to capture and interpret the different commands and gestures provided by the different users.

Secondly, how can the service be tailored to the users present? Different users have different preferences. For example, content recommendation systems typically provide recommendations that are personalised for individual users. A challenge is to provide recommendations that are relevant for a group of television watchers.

Finally, how does everything come together? A challenge is to integrate context-awareness and personalisation into the system, and to create a business logic that makes service decision based on the detected inputs.

The tables 1.1 and 1.2 below provides the some relevant use cases from HBB-NEXT Deliverable D2.1 [HBB-NEXT_D2.1]. More details on those use cases are provided in that deliverable.

U.015	User activity tracking	The user can interact with the application without repeatedly identifying itself. The system knows who is giving command once everybody has been initially identified.	Appears in: Scenario 1, 3; Involved actors: A.002, A.004	Multi-modal interface for Multi-user Service Personalisation Engine
U.004	Rating content items	A user rates content	Appears in: Scenario 1, 2; Involved actors: A.001, A.002, A.004	Multi-modal interface for Multi-user Service Personalisation Engine??

Table 1.1: Use cases on multi-modal interface

U.010	Updating group profile	Intersection of all loaded profile preferences. Re-calculation when single user enters or leaves.	Appears in: Scenario 1, 2, 3; Involved actors: A.001, A.002, A.003, A.004	Content Recommendation system for Multi-user Service Personalisation Engine
U.036	User-Rights-Management	The overview of which viewer or groups of viewers have the right to view which content.	Appears in: Scenario 2, 3; Involved actors: A.001, A.004	Content recommendation system for multi-user service personalisation
U.003	Community-based recommendation	A group of people gets a recommendation based on all their interests	Appears in: Scenario 1, 2, 3; Involved actors: A.001, A.002, A.003, A.004	Content Recommendation system for Multi-user Service Personalisation Engine
U.034	Context-based recommendation	A recommendation based on the context of content which is currently being watched/used	Appears in: Scenario 2, 3; Involved actors: A.001, A.004	Content recommendation system for multi-user service personalisation
U.026	Personalised EPG.	A person uses the (personalised) EPG.	Appears in: Scenario 1, 2, 3; Involved actors: A.001, A.002, A.004	Content recommendation system for multi-user service personalisation

Table 1.2: Use cases on multi-user recommendation

The rest of this deliverable is outlined as follows, see also Figure 1.3:

- Section 2 studies the state of the art and problems for multi-modal control in a multi-user environment, looking specifically into multi-modal -identification (face recognition, voice analysis) and - controls (gesture recognition, speech recognition).
- Section 3 dives into multi-user content recommendations, starting with the basics of personalising content recommendations. The section looks into presentation and evaluation of content recommendation, and the creation of group recommendation. Also metadata issues, scalability and privacy are considered.
- Section 4 shows how things come together architecturally, introducing personalisation and context-awareness into the systems and their business logic.

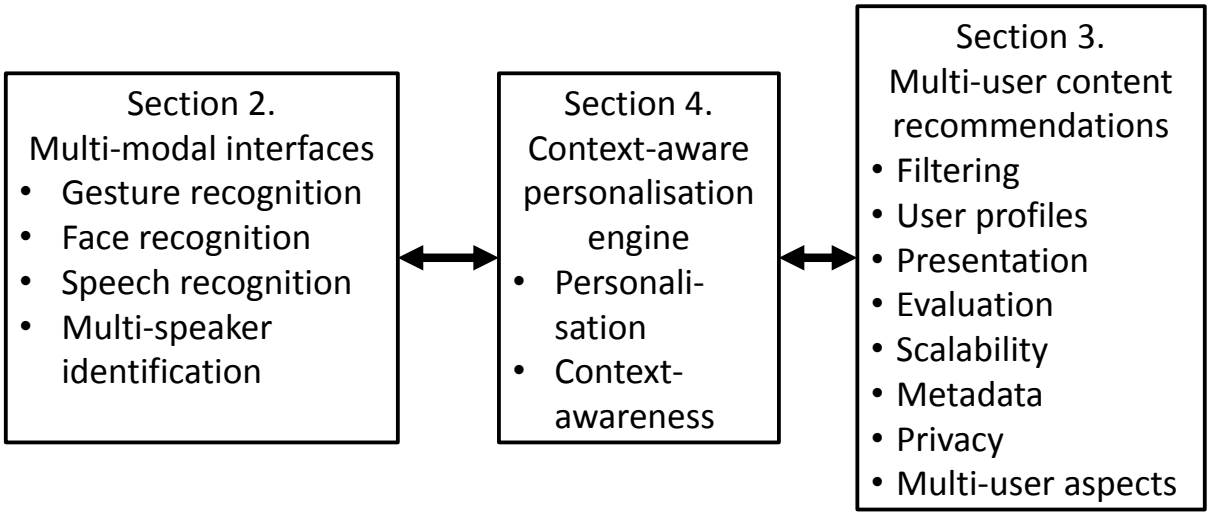


Figure 1.3: Sections of this document

2. Multimodal interface for user/group-aware personalisation in a multi-user environment

2.1. Outline

This section presents state of the art on multimodal interfaces for user/group-aware personalisation in a multi-user environment. The related use cases for multimodal control can be found in HBB-NEXT deliverable D2.1 [HBB-NEXT_D2.1]. Section 2.2 presents the problem statement. Section 2.3-2.6 analyse four input modes of multimodal systems:

1. gesture recognition,
2. face recognition,
3. speech recognition and
4. multi-speaker identification.

Section 2.7 deals with feedback from the multimodal system through the analysis of speech synthesis methods. Section 2.8 presents commercially available systems for multi-modal control. Section 2.9 presents the past performance on partner STUBA of the HBB-NEXT consortium.

2.2. Problem statement

Multi-modal interfaces enable users to interact with computer systems without requiring a keyboard, mouse, token or any other type of haptic device. Essentially, users use their body and voice to interact with the system. Multimodal interfaces can be used to answer the following two questions.

1. Who is that person or persons?
2. What does he/she/they want?

With the two types of input used, audio and image/video, four types of recognition are used in multimodal interfaces, see Table 2.2.1. The challenge is the accuracy in the user identification and the interpretation of the user intent, even in noisy and dynamic backgrounds.

	Who is that person(s)	***What does he/she want?***
Image/video	Face recognition	Gesture recognition
Audio	Multi-speaker identification	Speech recognition

Table 2.2.1: Four types of recognition in multi-modal interfaces

In order to provide more direct interaction with the user, the system can use synthesized speech in addition to the regular audio-visual feedback via the screen and system sounds. The challenge is the perceived quality of the user interaction, including ease of use.

2.3. Gesture recognition

Since the computers first came into the world of common people there has been much struggle with their operation. In fact, one of the greatest drawbacks in introducing computer technology to the ordinary life is their control. As computers and computer-based multimedia and entertainment systems find their way into our lives, a need rises to provide ways to command them which are more and more natural to human beings. Gestures are a suitable candidate to fulfil many operation tasks which have been previously done using (usually wired) peripherals, such as mouse or keyboard.

A gesture can be defined as expressive, meaningful body motion or position of a person's head, face, fingers, hands, arms or body meant to communicate meaningful information or interact with the environment [ChaIJC11]. In its complexity the gesture recognition requires knowledge in wide research area: computer vision and graphics, image processing, machine learning, bio-informatics and even psycholinguistics. Gesture recognition has wide range of application, including:

- Aids for the hearing impaired
- Sign language recognition
- Interaction with computers for very young children
- Driver alertness/drowsiness detection
- Navigating/manipulating virtual environments

- Communication in video conferences
- Distance learning, etc.

2.3.1. Gesture taxonomies

Various taxonomies have been suggested in the literature which deals with gestures from the psychological point of view. According to Kendon [KenSDC72] "autonomous gestures" can be distinguished from "gesticulation" where the prior is speech independent while the latter occurs in connection to speech. In his work, McNeill recognizes three different types of gestures: "iconic", "metaphoric" gestures and "beats".

For the purposes of Human-Computer Interaction (HCI) a taxonomy developed by Quek [QueVir94] appears to be most suitable. Each type of gestures has to be detected in a slightly different way as, according to different gesture taxonomy, the gestures can be of dynamic (i.e. move hand from starting position to ending position) or static (i.e. "thumb up") nature, or a combination of the two.

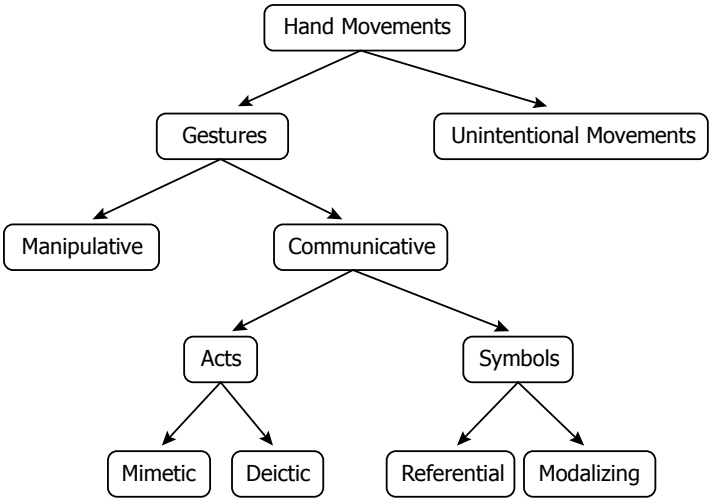


Figure 2.3.1.1: Taxonomy of Hand Gestures according to [QueVir94]

The key to gesture recognition is good feature extraction (or parameterization). Another problem is **Gesture spotting** - determining the start and end points of a meaningful gesture pattern from a continuous stream of input signals and, subsequently, segmenting the relevant gesture.

2.3.2. Feature extraction methods

2.3.2.1. Methods employing data gloves

These methods use sensors (mechanical or optical) attached to a piece of clothing (glove, bodysuit) which detect flexions of the respective body parts (fingers, arms, etc.) and convert them into electrical signals used to determine the body part's posture [HuaWAF95]. This approach may deliver precise results and is often used by professionals i.e. in film industry where movements of an actor are captured to animate computer-generated character.

In medicine, gestures are useful to manipulate fine tools during minimally invasive procedures. The drawback, however, is that person is required to wear special garments, often connected to the computer via plenty of cables, thus hindering the ease and naturalness of the user interaction.

2.3.2.2. Vision based methods

These methods are based on the way human beings perceive information about their surroundings, so they don't require persons to wear any special garment [MurJIT09]. The gesture detection and recognition is carried out based on image analysis of gesticulating person acquired using cameras or other sensors tracking the persons movement. This approach is much more convenient for ordinary people, but requires highly advanced algorithms to detect the gestures. It is also very difficult to achieve good results in an ever-changing environment though good results have been obtained in controlled environments [HuaWAF95].

We provide an overview of the most common methods employing the vision and image analysis principle to detect and recognize gestures.

2.3.2.3. Hidden Markov Models

HMM is a double stochastic process governed by both Markov chain with a finite number of states and a set of random functions, each associated with one state. In discrete time instants, the process is in one of the states and generates an observation symbol according to the random function corresponding to the current state. Figure 2.3.2.1. shows a five-state left-to-right HMM with the generated observations $o_{2\text{-}4}$.

Each transition between the states has a pair of probabilities, defined as follows: transition probability a_{xy}, which provides the probability for undergoing the transition, and output probability o_z, which defines the conditional probability of emitting an output symbol from a finite alphabet when given a state.

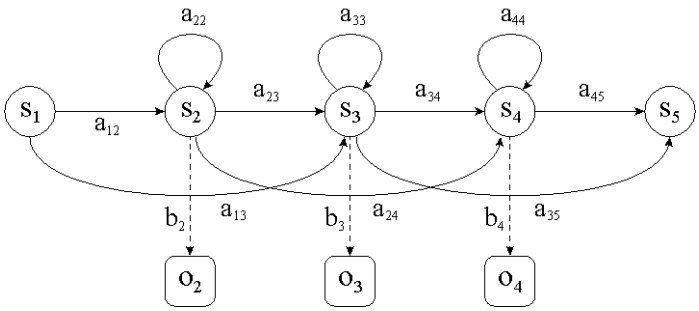


Figure 2.3.2.1: Five-state left-to-right HMM for gesture recognition

There are three key issues involved in the creation of the proper model, each of them solvable with the algorithm stated in the brackets:

1. Evaluation: determining the probability that the observed sequence was generated by the model (Forward–Backward algorithm)
2. Training or estimation: adjusting the model to maximize the probabilities (Baum–Welch algorithm)
3. Decoding: recovering the state sequence (Viterbi algorithm)

More on the algorithms can be found in [RabIEE89].

2.3.2.4. Particle Filtering

The idea behind particle filtering is to represent probability densities by set of samples. This approach results in the ability to represent a wide range of probability densities, allowing real-time estimation of nonlinear, non-Gaussian dynamic systems [MitSMC07]. The state of a tracked object at time t is described by a vector X_t, and the samples of observations $\{y_1, y_2, \ldots, y_t\}$ are represented by the vector Y_t. The posterior density $P(X_t|Y_t)$ and the observation density $P(Y_t|X_t)$ are often non-Gaussian.

Approximation of the probability density distribution by a weighted sample set

$$S_t = \{x_t^{(i)}, w_t^{(i)} \mid i = 1, \cdots, N_p\} \quad (2.1)$$

Each sample $x_t^{(i)}$ represents a hypothetical state of the object, and $w_t^{(i)}$ represents the corresponding discrete sampling probability of the sample $x_t^{(i)}$ such that

$$\sum_{i=1}^{N_p} w_t^{(i)} = 1. \quad (2.2)$$

The iterative evolution of the sample set is described by propagating each sample according to a system model. Each sample element in the set is weighted in terms of the observations, and N_p samples are drawn with replacement by choosing a particular sample with posterior probability $w_t^{(i)} = P(y_t | X_t = x_t^{(i)})$. In each step of iteration, the mean state (sample) of an object is estimated as

$$E(S_t) = \sum_{i=1}^{N_p} w_t^{(i)} x_t^{(i)}. \quad (2.3)$$

2.3.2.5. Condensation algorithm

The Condensation algorithm is proposed to deal with the problem of tracking rapid motion in clutter. Rather than attempting to fit a specific equation to the observed sensory data, it uses the N_p weighted samples to approximate the curve described by the observed data. When applied to tracking, each sample represents the *state* of the object being tracked, e.g., its velocity and location. Given such a randomly sampled state S_t at time t, a prediction of a new state S_{t+1} at time $t+1$ is made using a predictive model.

2.3.2.6. Finite State Machine Approach

In the FSM approach, a sequence of states in a spacio-temporal configuration space can model a gesture. The gesture is then recognized as a prototype trajectory from unsegmented continuous stream of sensor data which constitute an ensemble of trajectories consisting of points in the 2-D space.

The points represent sampled positions of head, hand, eyes, etc. The training of FSM is usually conducted off-line by providing plenty of possible examples of each gesture from which the parameters of each state in the FSM are derived. The recognition itself can be then performed in the real time using the trained FSM. Depending on the input data the recognizer decides whether or not to stay in the current FSM state. The gesture is considered recognized when the FSM reaches the final state [MitSMC07].

2.3.2.7. Soft Computing and Connectionist Approach

Soft computing is a consortium of methodologies that works synergistically and provides flexible information processing capability for handling real-life ambiguous situations. Its aim is to exploit the tolerance for imprecision, uncertainty, approximate reasoning, and partial truth in order to achieve tractability, robustness, and low-cost solutions. Soft computing tools, such as fuzzy sets, artificial neural networks (ANNs), genetic algorithms (GAs), and rough sets, hold promise in effectively handling these issues [MitSMC07].

The data sets describing the movement of the body or its part are examined using one of the aforementioned methods to determine which gesture has been performed with the highest probability.

2.4. Face recognition

Face detection and localization is the primary step in order to offer multi-user interface for multimedia systems. To distinguish between various individualities by their face characteristics, there is a crucial task to detect and localize their faces. There are several tasks which are similar in essence, but different in their results.

Face localization means determination of the position of one face in the image. There is assumption that image contains one face only. **Face tracking** provides information about face's position and motion in the sequence of images. **Face authentication** verifies identity of the user. **Face recognition** and identification provides evaluation of users identity compared to the database of potential users. **Facial expression recognition** helps to know user's emotional status and his intentions.

Detecting faces in images we have to pay attention to various factors which can make this task difficult and hard to implement. Shape and texture of a human face change based on expression, viewpoint and lighting conditions. Statistical models describing such flexible objects should reflect their flexibility – they should be able to change within similar limits [CooCom95]. Several factors affect face detection and localization process.

- **Pose** – result of face detection depends on mutual position of face and the camera. Face does not have to be fully visible, or can be occluded by other faces.
- **Presence of structural components** – beards, moustache, glasses etc. can cause confusion or can make user's identity identification difficult.
- **Face expression** – various emotional states results in different face appearance.
- **Image orientation** – there is a possibility of rotation of the image around the optical axis of the camera.
- **External conditions** – appearance of the face depends on camera's characteristics, illumination conditions etc.

2.4.1. Theoretical background

Face localisation is performed in many ways using variety of techniques. The most simple are various **histogram-based methods**. Histogram represents probability density function estimation *P(c|skin),* where *c* is a value of colour component and *skin* specifies the objection that the colour component value belongs to the skin colour. Usually, the histogram obtains large number of images pixel from large face image database [GarIee99] and [SobSig98]. This approach will be discussed later. Other probabilistic methods use **Bayesian classifier**. Theoretical background is given in [JonIjc02]. Bayes rule used for skin colour detection is used as follows:

$$P(skin \mid c) = \frac{P(c \mid skin).P(skin)}{P(c \mid skin).P(skin) + P(c \mid \neg skin).P(\neg skin)} \quad (2.4)$$

P(c/skin) and *P(c/¬skin)* are estimated from histograms for skin a non-skin regions. Probabilities *P(skin)* and *P(¬skin)* are also estimated from large amount of images. This approach combines probabilistic model and decision rules. Another example can be found in [YanMin09]. In combination with image refinement techniques can these algorithms provide satisfactory results by low computational costs (detection rate up to 95 %).

The **Gaussian Mixture Model** is an approach that combines probabilistic model and decision rules [DinIca10]. Skin colour model is described by the statistical model using number of Gaussian distributions. Model is defined as follows.

$$P(x) = \sum_{i=1}^{N} w_i \frac{1}{(2\pi)^{\frac{3}{2}} |\Sigma_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu_i)^T \sum_i^{-1} (x-\mu_i)} \quad (2.5)$$

Where x is three-dimensional RGB colour vector and i-th Gaussian affects distribution by its weight w_i. Number μ_i is mean value of the distribution and Σ_i is its diagonal covariance matrix. There are two of these models. One is used for skin region and one for non-skin region. Based on colour space used to create the model, effectiveness varies. Due to the low computational costs at the classification portion of the algorithm, this approach is suitable for real-time applications.

Statistical models can also represent shape of human face by a set of points. These points represent the boundaries and important landmarks of the objects. These points are labelled in a set of training images. The labelling can be done either manually or using some other tool. The correct labelling of the training set is very important and can severely influence the performance of the model. When applied to human faces the label points will likely mark such important points as the middle of the eyes and the edges and boundaries of eyes, mouths and noses [CooSpr05].

Next approaches are mainly classification-based algorithm. A simple way to classify image pixels to the two groups (skin and non-skin) is the Multi-Layer Perceptron (MLP). MLP is a feed-forward neural network used in regression and classification problems in many applications – not only in skin colour segmentation [KimIci10] In comparison for example with GMM, the MLP can produce more complex classification boundaries.

Neural networks and systems of machine learning give many methods. One of the newest is Adaboost-based algorithms (e.g., [ZuoSpi08]), support vector machines (e.g., [OsuIcc97]) or methods with use of eigenfaces (e.g., [MohIee09]).

Robust tool used in face detection is the **Gabor filter**. Theoretical background was given in [GabIns46]. Gabor filters are based on mechanism of human brain perception. The filter is described by

$$g(x, y) = s(x, y).w_r(x, y), \tag{2.6}$$

where

$$s(x, y) = \exp\left(j\left(2.\pi.(u_0.x + v_0.y) + P\right)\right) \tag{2.7}$$

$$w_r(x, y) = K.\exp\left(-\pi.\left(a^2.(x - x_0)_r^2 + b^2.(y - y_0)_r^2\right)\right). \tag{2.8}$$

It is a sine wave delimited by Gaussian envelope. Each filter is characteristic by its spatial frequency and orientation – those are two main component of human visual cortex perception. Thus, each filter tracks different characteristics of the image. These characteristics are processed by neural networks. One of the new applications is in [SurIcc11]. These algorithms are mostly too complex to be implemented in the real-time conditions but their detection rates are up to 95%. This variety of approaches and strategies offers many possibilities to combine them and gain the final face segmentation results.

Object-oriented approach brings Active shape and Active appearance models. **Active shape models (ASM)** is a method developed by Cootes at al. [CooCom95]. It uses a statistical model and fits it to new target images. First the model is placed at the center of the new image and its points are iteratively moved to better locations using the model parameters. Better locations are compared to the gray-level intensity changes in the neighbouring pixels of the target image. Threshold of the neighbourhood is defined based on the variance in the training images. Once the model has converged (either by not finding any better new locations or by exceeding the maximum allowed number of iterations), the ASM gives us a good approximation of important points in the new target image.

ASM can be used to recognize shapes of interest in target images and to compare the similarity of shapes from different images (both detection and recognition tasks). ASM was successfully used for a number of industrial and medical applications. The method resists noise and occlusion well in most cases, however especially with more complicated tasks such as human face recognition, incorrect approximations of the shape may occur [CooCom95]. Using a training set of only frontal facial images for example will perform well only on frontal target images and will fail recognize faces turned slightly sideways.

A training set of facial images that are too different from each other (to account for all possible poses for example) will result in ASM with low precision and will also perform poorly. The method is also computationally quite effective, while performing slightly better than other more complex models [CooCom95].

Active appearance models (AAM) is an extension of the ASM method that incorporates both the shape and texture of objects. A training set of images with important points labelled is once again used to produce a model of the average important shapes and shape variability. To build a statistical model of the texture images we first normalize the image using the shape model, to remove texture variation due the nearby landmarks of the object [CooSpr05] (see figure 2.4.1.1).

The shape-normalized image is used to select texture samples from important regions of the image as defined by the shape model. The samples can be further normalized using simple transformations to remove the effects of lighting. Principal component analysis (PCA) is applied to the samples to produce a statistical model that approximates each sample from each image in the training set [CooEur98].

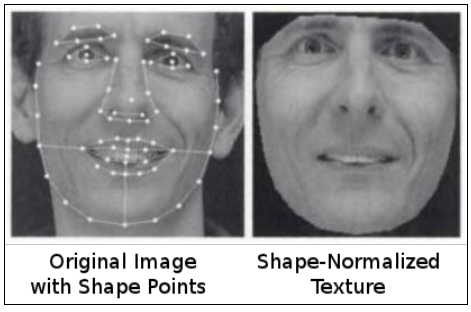


Figure 2.4.1.1: Labelled Shape and Normalized Texture [CooSpr05]

By adjusting the parameters of this model, we can produce a high number of new shape-normalized textures that differ from each other within the limits of the variability of the training set. Similarly as the points in ASM, these textures can be iteratively fitted to new images, to produce an accurate description of the objects of interest [CooEur98]. This approach can also be used to deform facial images in meaningful ways, for example changing the expression of the face (adding/removing a smile, frown, facial hair...). Textures generated using AAM can even be used to expand the size of the training set [CooEur98].

Shape-normalized textures can once again be used for detection and recognition tasks by comparing them to the shape-normalized textures of new target images. Since more information was extracted and used in the comparison, AAM is more robust than methods based solely on shapes and points. Though more complex than simple ASM AAM is still relatively computationally non-intensive and can be further improved by using some optimization and approximation during the fitting phase [CooEur98].

ASM and AAM techniques are presently used for a number of different applications. Face alignment is the task of correctly adjusting and fitting a face into an image. AAM is often used to perform this task [KimWor07]. Face detection and tracking means finding and keeping track of the position of a face in a series of images such as a video. Here AAM and ASM also perform well [IshCit02]. AAM can also be used for facial expression recognition [TanThi02]. Finally feature points extracted from AAM analysis can be used as a classifier for face comparison and traditional facial recognition tasks [EdwEur98].

ASM and AAM are helpful techniques that use flexible models to describe flexible objects of interest in two dimensional images. They use a set a pre-labelled training images to create adjustable models that can be fitted to new images to accurately describe the objects they contain. Their power lies in the fact that the flexibility of the models they use directly reflects the variation of the objects in the training images. AAM and ASM are used in a number of modern-day applications.

2.4.2. Team Expertise

STUBA's work within the field of Face Detection focuses on three major approaches. We use Skin Colour analysis, Gabor Wavelet filtrations and Active Shape model with Support Vector Machines (ASM and SVM). These approaches represent different aspect of the processed images. Skin Colour analysis takes and advantage of visual distinction regarding to the colour differences between regions. Gabor Wavelet extracts information about spatial frequency and orientation of the image signal. ASM and SVM incorporates both shape and texture models.

2.4.2.1. Skin Colour Analysis

In [BenIjs11] STUBA have proposed an approach for skin colour segmentation and facial features extraction which involves face detection and localization task. The proposed method is based on skin colour analysis which provides most reliable tool for distinguishing between skin colour region and background. This method can be also used for segmentation of hands and other parts of the human body which can be also used for multimodal interface such as hand gesture etc.

Facial features extraction is useful for image orientation determination and also for verification of the statement that skin colour region found by face detector is truly a face region.

For skin colour detection we use a histogram based method which tracks colour chrominance component distribution in YCbCr colour space. There is an assumption that skin colour is attached to the specific range of Cb and Cr components values. This colour distribution is independent of differences between various skin colour types. Face region is also described by the luminance component variation which has certain distribution.

The training set is obtained from the existing face recognition databases such as Feret etc.

2.4.2.2. Gabor Wavelet approach

Next, STUBA works on the Gabor wavelet approach to the face detection. We use Gabor filters for texture classification in [BanRad07] and [PavElm10]. Face image may be described as a texture which has its characteristic features.

We use bank of the Gabor filters to extract various feature of the image of the human face and based on this features we computer statistical characteristics of the face texture. This characteristic creates an input to the single layer perceptron neural network which performs classification whether considered image is or is not the image of the human face. This approach consists of many steps and provides complex information about images appearance and content. Using satisfactory hardware tools can provide promising results.

2.4.2.3. ASM and SVM utilization

In STUBA's paper [BesMgv09] we proposed the analysis of Active Appearance Models (AAMs) and Support Vector Machine (SVM) classifiers utilization in the case of human facial emotion and emotion intensity levels recognition problem. AAMs are known as a tool for statistical modelling of object shape/appearance or for precise object feature detection. In our case we examined their properties as a technique for feature extraction. We analysed the influence of various facial features data types (shape / texture / combined AAM parameter vectors) and size of facial images to final classification accuracy. Unlike the usual recognition algorithms evaluation approach (based on comparison of final classification accuracies) the proposed evaluation schema is independent of the testing set parameters such as number, age and gender of subjects or intensity of their emotions. Algorithms were tested and developed to run real time on PC-webcam testing setup. According to the general properties of the mentioned algorithms we can recommend them to for usage in mobile phones, tablets or TV environment.

2.5. Multi-speaker identification

Multi-speaker identification aims to identify possibly more speakers based on a recorded signal that may contain utterances of more individuals. This general task is a rather complex one so it can be separated into several categories based on additional refinements. If the speakers that may appear in given conversation/ record are known in prior, i.e. they were present in some sort of training phase (there exist their models, separation functions, or simply training data), the task resembles a single speaker identification problem where a ML segmentation can be performed.

However when the set of possible speakers is unknown then the techniques of speaker segmentation and clustering (diarization) must be used; however in this case the accurate time localization is not as eminent as in speech indexing or speech recognition tasks, where diarization systems are mostly used. It is expected that there is no serious overlap of utterances of different speakers (rather a dialog style conversation).

Then segmentation is based on acoustical similarities/ dissimilarities for adjacent speech frames, so after applying such an algorithm it is possible to estimate locations of speaker changes. Furthermore, by using a speaker clustering (merging parts that belong to a single speaker) it is possible to estimate the number of different speakers (given that the whole conversation -meeting is held in the environment with stable acoustic parameters).

2.5.1. Theoretical background

In order to solve a multiple speaker recognition problem one has to master a single speaker identification as it shares many techniques. Most of the multi- speaker identification systems (depend on the task) must employ feature extraction, segmentation, clustering and classification method. Thus in the following a brief overview of the most successful approaches is outlined.

Speech feature extraction methods

Proper speech features play an important role in identification process as well as in speech recognition task. For some more general properties of eligible speech features see section 2.6 Speech recognition of this paper and let us here describe in more details the design of the most important ones, i.e. MFCC and PLP as well as some approaches that detect and estimate the pitch period i.e. AMDF, YIN.

Both MFCC and PLP [HönINT05], [HerICA85] are similar in a way that the present some kind of modified spectra. They aim to extract and preserve spectral envelop in an audible resolution in a compact way. However they differ in a manner how they are constructed an in a psychoacoustic model so let us in the following outline the design for both of them.

MFCC

The speech signal is first modified by high pass filter so-called preemphasis filter to suppress the low pass character of the speech given by the lip radiation to the open space. Prior to the FFT computation a Hamming window is applied and the frequency in Hz is warped into the Mel scale to mimic the critical bands over different frequencies. Next, equally spaced triangular windows with 50% overlap are applied to simulate a filter bank. Finally the logarithm function and DCT transform are applied that produce a static feature frame. The logarithm not only acts as a tool to produce cepstrum (real one) but suppress the high-vale intensity in favour for low intensities as the human auditory system does. In addition, zero cepstral coefficients used as well to estimate the overall log energy.

PLP

The original process of PLP calculation follows these steps: calculation of FFT that is proceeded by Hamming windowing, frequency warping into Bark scale, smoothing the bark-scaled frequency spectra by a window simulating critical bands effect of the auditory system, sampling the smoothed bark spectrum in approximately 1 bark intervals to simulate the filter bank, equal loudness weighting of the sampled frequencies which approximates the hearing sensitivity, transformation of energies into loudness by powering each frequency magnitude to 0.33, calculating the linear prediction coefficients from the warped and modified spectra (all pole model of the speech production), finally cepstral LPC coefficients are derived from LPC as if the logarithm and the inverse FFT were calculated.

Pitch Period

Pitch period which is usually defined as the time between two consecutive opening instances of vocal folds indisputably conveys identification information, e.g. higher pitch is observed in females' population which is caused mainly by their physical differences to males. Except those physical aspects each individuals has a different speaking style that involves different modulation patterns. However, the pitch must be used with a care as there are problems in its estimation. Usually it is because the pitch varies so much (from 80Hz to even 400Hz) and it overlaps with the first formant. To tackle some of these problems many estimation techniques have been developed either in time or frequency.

The most common are autocorrelation ones (different modifications of clipping functions), and difference based methods like average magnitude difference function (AMDF). In [CheACA02] it was observed that AMDF outperforms the autocorrelation one as it is less sensitive to changes in magnitudes (the drop in error was from 10% to 1.95%). AMDF function is defined as:

$$AMDF\ (k) = \sum_{n=0}^{n<N} \left| x(n) - x(n+k) \right| \quad k \in < T_{0\min}, T_{0\max} > \tag{2.9}$$

In case of a period and its multiplications it exhibits low values (in ideal case zeros). Enhanced version of AMDF can be found in YIN method that is defined as:

$$YIN\ (k) = \frac{\sum_{n=0}^{n<N} \left(x(n) - x(n+k) \right)^2}{\frac{1}{k} \sum_{i=1}^{k} \sum_{n=0}^{n<N} \left(x(n) - x(n+i) \right)^2} \tag{2.10}$$

This method provides even more robust estimates with properly set absolute threshold for minimal values.

Classification Methods

Every recognition system must implement some kind of decision taking algorithm in the final stage. There are many of them ranging from theoretical concept presented by Bayes classifier to practically realizable ones like Gaussian Mixture Model (see GMM (Gaussian Mixture Model) below), Neural Networks (see NN (Neural Networks) below), decision trees, etc. In general these methods can be divided into two main groups: the so called generative methods that aim to model the feature space as close as possible like GMM, and discriminative methods like NN, SVM where the main focus is to separate among different classes instead of precise modelling of the feature space. Some of the most frequent classification systems used in connection with the text independent recognition systems are discussed below.

The optimal solution minimizing the overall mean cost is given by Bayes decision rule for multiclass problem which is given as follows:

$$R_k = \{x; \sum_{\substack{i=1 \\ i \neq k}}^{N} \left(\pi_i C(k/i) P(x/i)\right) \leq \sum_{\substack{i=1 \\ i \neq j}}^{N} \left(\pi_i C(j/i) P(x/i)\right);\ j = 1, \ldots, N \wedge j \neq k\} \quad (2.11)$$

.

Used symbols have the following meaning: R_k is the decision area in the feature space for the class *k*, *x* is the observed symbol (speech feature vector), π_i is the a priory probability of class *i*, *C(j/i)* is the cost of choosing class *i* even though it belongs to class *j*, *N* is the number of classes (speakers), and *P(x/i)* is the probability (likelihood) of observing vector *x* conditioned by the class *i*. The main obstacles for using this optimal solution are the unknown distributions *P(x/i)* and a priory probabilities π_i that must be properly modelled and estimated.

NN (Neural Networks)

Neural networks are mathematical models of related basic structures in a brain. There are more structures of neural networks, but the most widely used are multilayer perceptrons (MPL) and radial basis functions (RBF). Both possess the property of being universal approximators, thus they can approximate any continuous function with the arbitrary small error given certain constrains. Thus they can be designed and trained to provide the best separation possible using the training data. Furthermore they can generalize the decision function for unseen data samples as well. However the optimal number of neurons is not known and all the known training strategies do not guarantee to reach the global optima. Moreover NN are prone to the over fitting phenomenon thus cross validation must be performed during the training.

GMM (Gaussian Mixture Model)

Gaussian mixture model is a special HMM with only one state. It uses a set of multidimensional Gaussian distributions to describe the feature space in a probabilistic manner. It is assumed that a linear combination of Gaussian densities can approximate with arbitrary small error any continuous statistical distribution. Thus the main problem is to estimate the mean vectors, covariance matrices and weights of all Gaussian distribution. It is done by the well know EM algorithm that is a maximum likelihood (ML) estimator.

Again only a local maximum of ML is guaranteed and the number of Gaussian components is also not known and must be tested. Then the recognition may follow the optimal Bayes classification rule where each speaker (its distribution) is represented by GMM and via its output required likelihoods $P(x/class_i)$ can be obtained. However, one should keep in mind that the suboptimal estimation of GMM may prevent reaching the best results guaranteed by the Bayes rule.

SVM (Support Vector Machines)

Support vector machines are one of the latest techniques how to keep the structural risk low, i.e. to minimize empirical risk while keeping the expected risk bounded. In order to preserve expectation of the risk low only decision functions with low VC dimension are assumed. The empirical risk is minimized by placing the decision hyper-plane so that it preserves the larges possible margins between the two groups of samples that should be separated. This leads to the quadratic programming problem. In order to introduce some kind of nonlinear decision to non- linearly separable data the kernel trick is used via nonlinear transform to higher dimension space, where a proper hyper-plane is found. There are more kernel functions and the most used ones are: Gaussian, polynomial, etc.

In the case of a 2 class separation its usage is direct but in the case of multi class separation usually more classifiers are used each separating one group from the remaining ones.

KNN (K nearest Neighbours)

KNN belongs to a group of so called instance based learning algorithms**Error! Reference source not found.**. It judges a test sample according to its location to the training samples in the feature space. It is assumed that the closer a test sample to the referenced one is the more probable is the fact that these samples are from the same group. To suppress errors caused by a present noise or lack of training samples the robustness can be increased by exploring broader area, i.e. finding several closest samples to the test one. There exist more variations to this basic principle, e.g. weighted distance KNN or local linear regressions that are more general.

Even though this method is very simple it can perform very complex approximation of the true decision function, as all decisions are local. Its main drawback is computational complexity during test phase; however this is substantially suppressed by structuring the samples into search trees.

Speaker Segmentation

Speaker segmentation estimates the change point of speakers, may select also speech and non-speech parts. There are many approaches which can be categorize into single pass and multi pass (points are detected using more iterations, each time finer refinement is done) algorithms. Other common division is according to the used classification method: metric based, silence based and model based algorithm.

Metric Based

These algorithms use some kind of distance measure between neighbouring segments to assess the acoustic similarities; they are very widely used. If the distance overcomes some threshold (it may evolve in the time) a change point is declared. Distance can be some acoustic measure or more often probability one (likelihood). In this case some model of underling pdf must be assumed and estimated on a given segment, unlike the previous scenario. List and a short description of some most successful ones follow:

BIC (Bayes Information Criterion)

It is a likelihood measure used in connection with model selection that is adjusted by a penalization term in order to suppress the overfitting phenomenon. It is defined as follows [SchAOS78]:

$$bic(x_1 \ldots x_T, \lambda) = \log\left(p(x_1 \ldots x_T / \lambda)\right) - \frac{1}{2}\alpha N(\lambda)\log(T), \qquad (2.12)$$

Where $x_1, \ldots x_T$ are the observed feature vectors, $p(x_1, \ldots x_T / \lambda)$ is the likelihood of observing vectors at model λ, α is auxiliary (control) parameter set up in the design stage, and $N(\lambda)$ is the number of parameters that must be estimated while using model λ. Then, this value can be used for segmentation purpose as follows. Having two segments X and Y, construct and estimate separate models for X and Y and a mutual model XY (for both sets of data).

Then calculate which way of modelling is better in terms of *bic*, i.e. calculate the difference between them as:

$$\Delta bic(X,Y) = \log\left(p\left(\frac{XY}{\lambda_{XY}}\right)\right) - \log\left(p\left(\frac{X}{\lambda_X}\right)\right) - \log\left(p\left(\frac{Y}{\lambda_Y}\right)\right) - \alpha(N(\lambda_{XY}) - N(\lambda_X) - N(\lambda_Y))\log(T) \quad (2.13)$$

If it is better to model the segment by a single model, then probably there is no distinctive change, thus no speaker change point is detected and vice versa. This method requires computation of 3 models each time a comparison is made which is computationally expensive. There exist more modifications to this approach mainly by how to set the control parameter α.

KL (Kullback – Leibler Distance)

KL distance is calculated as follows:

$$kl(X,Y) = E\left\{\log\frac{P(x)}{P(y)}\right\}, \quad (2.14)$$

where *P(x)* and *P(y)* are probability distributions for *x* and *y* sets of data, *E* is an expectation operator. As it can be seen this is not a symmetrical measure. To make it symmetric KL_{sym} was designed as:

$$kl_{sym}(X,Y) = kl(X,Y) + KL(Y,X). \quad (2.15)$$

However in practical calculation there may appear problems as the exact, close formulae for arbitrary distributions may not exist (for single Gaussians it exists but for GMM no, so numerical methods must be used). If the distance falls under some threshold no change point is detected and vice versa.

GLR (Generalized Likelihood Ratio)

GLR considers two hypotheses: *H0:* both segments represents the same data and *H1:* they are chosen from different classes, thus they should be better modeled by separate models. GLR is defined by:

$$glr(X,Y) = \frac{H0}{H1} = \frac{P(X,Y/\lambda_{xy})}{P(X/\lambda_x)P(Y/\lambda_y)} \quad (2.16)$$

Usually the distance is calculated as $d_{glr}(X,Y)=-log(glr(X,Y))$ using segments of the same size. Again pdf function for all models must be estimated from the tested segments. There exist several modifications varying by setting up proper thresholds or penalization terms.

Other metrics exist, such as: DSD, Gish, Cross-BIC, etc. that can be found e.g. in [MirUPC06].

Silence and Decoder Based Segmentation

As it is more common for speaker change point to occur during pauses in speech, these approaches focus on detecting pauses, however the pause itself does not mean eminent change point. Thus points detected in this way must be further confirmed by other methods. Silence intervals are most frequently detected based on different sorts of energy or decoder (phoneme or word recognition system) based algorithms [KemSSP00].

Model Based Segmentation

These methods use models (mostly GMM) trained on the training data that were available in prior. These models are to cover different acoustic patterns like: noises, silence, different female and male utterances in various conditions, etc. Once having these models, ML segmentation according to these models is performed. Then the switching points between models also mark the change points for speakers (possibly).

Speaker Clustering

After the segmentation process the parts that contain the same speaker should be merged together in order to find out how many distinct speakers there were. There are two main approaches working in opposite directions. One category of algorithms performs the so called top down clustering (start with one or only few and splits them) while the other one makes bottom up (starts with abundant number of clusters that are iteratively merged). Other functional division can be according to the prior knowledge of the number of speakers (blind diarization- no prior knowledge). Final important criterion is whether the system works on or off line (has all the data available).

Bottom up Methods

These techniques are directly applicable to the output of segmentation systems, thus are most widely used. They merge similar clusters based on some distance measures that are in most cases the same as used for speaker segmentation tasks. They construct the so called distance matrix consisting of distances of all clusters to all and merge the closest pairs. The process goes on till some stopping criteria are met. In [RouICA06] a new distance based on KL was used in the case of GMM models defined as:

$$d(GMM_1, GMM_2) = \sum_{i=1}^{M_1} \pi_1(i) \min_{j \in M_2} KL\left(N_1(i), N_2(j)\right), \quad (2.17)$$

where GMM_1 is first cluster represented by a GMM model, $\pi_1(i)$ is the apriori probability of i-th mixture of the first model, KL is Kullback-Leibler distance, $N_1(i)$ is i-th Gaussian mixture of model 1, M_1 is the number of mixtures for model 1, and M_2 is the number of mixtures for model2.

Another approach [MorICS05] used different strategy by adapting overall GMM model obtained for the whole data to particular clusters via MAP criterion. Then adapted models (clusters) were compared by a new metrics based on KL2 distance, where only mean vectors were adapted while apriori probabilities of mixtures and covariance matrices were kept constant. Then the distance is defined as:

$$d(GMM_1, GMM_2) = \sqrt{\sum_{m=1}^{M} \sum_{d=1}^{D} \pi(i) \frac{\left(\mu_1^m(d) - \mu_2^m(d)\right)^2}{\delta_1^m(d)^2}}, \quad (2.18)$$

where $\mu_1^m(d)$ is d-th dimension and m-th mixture of a mean vector belonging to the first GMM model, $\delta_1^m(d)$ is standard deviation of m-th mixture in its d-th dimension for first GMM model, $\pi(i)$ is a-priori probability of i-th mixture, M is number of mixtures, and D is number of dimensions. Other approaches may be found e.g. in [MirUPC06].

Top down Methods

Compared to bottom up methods there are only few of top down methods. They start with one cluster and do the splitting in iterations. Thus they do not need speaker segmentation in prior. One method [AngICS04] uses MAP adaptation for new clusters, where the splitting

is performed according to the likelihood averaged over a window. As the stopping criterion the variation of likelihoods of all the models given the data is used. As there is no proof which of the two approaches is better some combinations of both of them have been designed.

2.5.2. Team Expertise background

STUBA's prime expertise is in single speaker recognition systems where it constructed several systems since 2006 based on different features and classification methods. The multi- speaker setting has not been regarded yet, so eligible segmentation and clustering methods will have to be implemented, adjusted and optimized together with the selected features and classification methods for a given application. Together with these proper voice activity detector might be necessary to design. However, STUBA has both theoretical and practical experience with construction of VAD algorithms.

2.6. Speech recognition

Speech signal is produced by human speech organs and is originally represented by air waves. It contains among other information lexical part that is crucial for the task of speech recognition. Lexical information is coded into the acoustic signal as sequence of acoustically different sounds. Each language contains its own set of basic sounds that are related to phonemes, e.g. for Slovak there are 51. Unfortunately phonemes uttered in a raw influence each other both in the time and in spectral domain resulting in acoustically different sounds. Furthermore, each phoneme even uttered in isolated way varies for different speakers (contains speaker specific information) as well as for the same individual. In addition there are additive and convolutional noises present in any real environment that make the situation even worse. Finally every language contains huge vocabulary usually of several hundreds of thousands of words each of which may exist in several forms (cases, times, etc.). Thus the situation is rather complex and computationally expensive. Many systems have evolved and thus some basic classifications are used: small, medium or large vocabulary systems, speaker dependent or speaker independent system, phoneme or word based system (sub-phoneme or phrases are also possible), continuous or isolated word (dictation) systems, etc.

For a couple of decades there has been a great effort spent to build and employ ASR systems in areas like information retrieval systems, dialog systems, etc., but only as the technology has evolved further other applications like dictation systems or even automatic transcription of natural speech [NouPIn05] are emerging. These advanced systems should be capable to operate on a real time base, must be speaker independent, reaching high accuracy and support dictionaries containing several hundreds of thousands of words. In the following some successful systems are outlined.

2.6.1. Theoretical background

The strict requirements on ASR systems mentioned above can be currently met by HMM models of tied context dependent (CD) phonemes with multiple Gaussian mixtures, which is a technique known from the 60ties. As this statistical concept is mathematically tractable it, unfortunately, doesn't completely reflect the physical underlying process. Therefore soon after its creation there have been lot of attempts to alleviate that. Nowadays the classical concept of HMM has evolved into areas like hybrid solutions with neural networks, utilisation of different than ML or MAP training strategies that minimize recognition errors by the means of corrective training, maximizing mutual information [HuaBHM09] or by constructing large margin HMMs [JiaITE07]. Furthermore, a few methods have been designed and tested aiming to suppress the first order Markovian restriction by e.g. explicitly modelling the time duration, splitting states into more complex structures, using double [CasITE07] or multilayer structures of HMM. Another vital issue is the robust and accurate feature extraction method. Again this matter is not fully solved and various popular features and techniques exist like: MFCC and CLPC coefficients, PLP features, TIFFING, ZCPA, RASTA filter, etc.

Even despite the huge variety of advanced solutions many of them are either not general enough or are rather impractical for the real-life employment. Thus most of the currently employed systems are based on continuous context independent (CI) or tied CD HMM models of phonemes with multiple Gaussian mixtures trained by ML or MAP criteria. As there is no analytical solution of this task, the training process must be an iterative one.

Unfortunately, there is no guarantee of reaching local maxima, thus lot of effort is paid to the training phase in which many stages are involved. As the overall result is sensitive to both speech feature selection and decision taking algorithm in the following main concepts are outlined revealing some mathematical background.

2.6.2. Speech feature extraction methods for speech recognition

One of the first steps in the design of an ASR system is to decide which feature extraction technique to use. At the beginning it should be noted that this task is not yet completely solved and a lot of effort is still going on in this area. The aim is to simulate the auditory system of humans, mathematically describe it, simplify for practical handling and optionally adapt it for a correct and simple use with the selected types of classification methods.

A good feature should be sensitive to differences in sounds that are perceived as different in humans and should be "deaf" to those which are unheeded by our auditory system. It was found [RabBFS93] that the following differences are audible: different location of formants in the spectra, different widths of formants and that the intensity of signals is perceived non-linearly. On the other hand, following aspects do not play a role in perceiving differences: overall tilt of the spectra like: $X(\omega)\omega\alpha$, where α is the tilt factor and $X(\omega)$ is the original spectra, filtering out frequencies laying under the first formant frequency, removing frequencies above the 3rd format frequency, and a narrow band stop filtering.

Furthermore, features should be insensitive to additive and convolutional noises or at least they should represent them in such a way that these distortions are easy to locate and suppress in the feature space. Finally, when using Continuous Density Hidden Markov Models (CDHMM) [XinASS05] it is required for the feasibility purposes that the elements of feature vectors should be linearly independent so that a single diagonal covariance matrix can be used. Unfortunately, there is no feature yet that would ideally incorporate all the requirements mentioned before.

Many basic speech features have been designed so far, but currently MFCC and PLP are the most widely used in CDHMM ASR systems. They both represent some kind of cepstra and thus are better in dealing with convolutional noises. However, it was reported that sometimes in lower SNRs they are outperformed by other methods, e.g. TIFFING.

Furthermore, the DCT transform applied in the last step of the computation process minimize the correlation between elements and thus justifies the usage of diagonal covariance matrices. Besides those static features it was soon discovered that the changes in the time represented by delta and acceleration parameters play an important role in modelling the evolution of speech. This is important when using HMMs as they lack the natural time duration modelling capability. Overall energy or zero cepstral coefficients with their derivations also carry valuable discriminative information thus most of the systems use them [MihITe08]. Furthermore, to take the full advantage of cepstral coefficients, usually a cepstral mean subtraction is applied in order to suppress possible distortions inflicted by various transmission channels or recording devices. At the end we shall not forget about the liftering of cepstra in order to emphasise its middle part so that the most relevant shapes of spectra for recognition purposes would be amplified. Well, this appealing option has no real meaning when using CDHMM and Gaussian mixtures with diagonal covariance matrices. In this case it is simply to show that the liftering operation would be completely cancelled out when computing Gaussian pdf.

Currently the most preferable acoustic features are MFCC, and PLP which are designed to capture positions and widths of formants that are acoustically perceivable. As these parameters are presented in more detailed way in section 2.5.1 Multi-speaker identification, please refer to it.

Additional features may include signal dynamic observed via delta and acceleration coefficients constructed over acoustic features showing their time evolution as well (possibly capturing the so called co articulation effects). They are defined as:

$$\Delta(n) = m\sum_{k=-L}^{L} kc(n+k) \quad and \quad \Delta\Delta(n) = m\sum_{k=-L}^{L} k\Delta(n+k) \quad , \tag{2.19}$$

where m is a normalizing constant and L is the time span. More sophisticated systems use the notion of supervectors (concatenated acoustic vectors over some time period) that are reduced by a linear transform usually to the original length of a single acoustic vector via dimension reducing methods while preserving essential information, most common are: PCA, LDA and HLDA. As the class separation is the prime focus LDA and HLDA are preferred.

2.6.2.1. HMM overview

Is a statistical modelling method for speech and more precisely for its parts (words, syllables, phonemes, sub phonemes, etc.). It is based on concept of Markov chain that makes it computationally very effective even though not reflecting time evolution of a genuine speech. Thus each model must be estimated using usually very large set of training examples that contain multiple recordings of the same word (its different realizations). HMM is defined by a-priori probabilities (π) of being in particular states at the begging, transition matrix (transition probabilities between states, a_{ij}) and probability distributions (likelihoods) of generating observation vectors in a given state, $P(x/s_i)$. These distributions are not known in prior but the most widely used ones are mixtures of multidimensional normal distributions (GMM). A 4 state left right model is depicted in Figure 2.6.2.1.1.

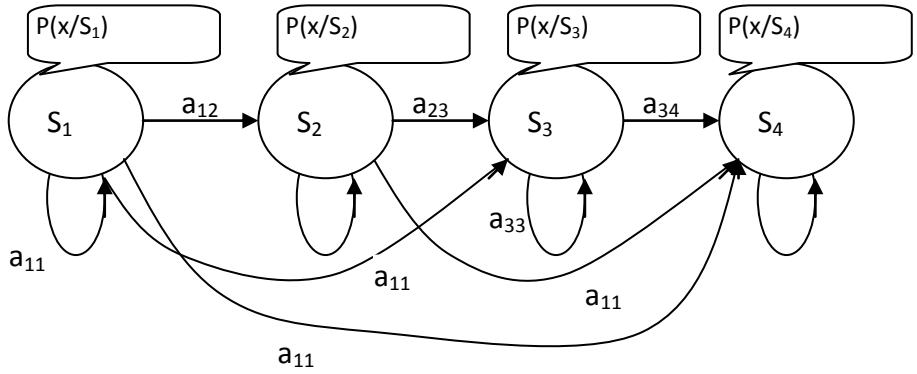


Figure 2.6.2.1.1: A 4-state left-to-right HMM model

Then the probability of observing string of feature vectors at the model λ is given by:

$$P(\mathbf{x}_1 \ldots \mathbf{x}_T \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i) \text{ where } \alpha_t(j) = [\sum_{i=1}^{N} \alpha_{t-1}(i) * a_{ij}] * P(\mathbf{x}_t / S_j) \quad j = 1, \ldots N$$
$$\text{and } \alpha_1(j) = \pi_j P(\mathbf{x}_1 / S_j) \qquad (2.20)$$

Then the recognition is done be choosing the HMM model *(λ)* with the highest probability. Great advantage of HMM models is the possibility of concatenating several models into a string thus utterance of any length can be built up using set of basic models. In practical situation only the path with maximal probability is calculated e.g. $P(x_1,S_{t1},x_2,S_{t2},\ldots x_T,S_{tT})$, and via backtracking the sequence of hidden states is determined (states can be related to sequence of words) using Viterbi algorithm.

The main problem with HMM is the estimation of their parameters. Usually it is done by maximizing the maximum likelihood $P(x_1,x_2,...x_T/\ \lambda)$ which leads to well-known method called Baum-Welch algorithm. However newer strategies like maximal mutual information (MMI), large margin HMM, and corrective training may be beneficial in some cases, especially when the HMM model constrains do not fit the underlying physical model. The MMI model estimation criterion is defined as follows:

$$\log(P(\mathbf{x}_1,...\mathbf{x}_T\ /\ \lambda_i)) - \sum_{j=1}^{P} \log(P(\mathbf{x}_1,...\mathbf{x}_T\ /\ \lambda_j)) \quad (2.21)$$

,

where λi is the correct HMM model according to the known transcription of the observation $(x_1,x_2,...x_T)$ and λ_j are all available models. This criterion aims to increase the overall separation gap among models.

Except training strategies the structure of HMM models is also important. Usually for speech units left - right models are used and the most common is probably the Bakis structure. It is commonly accepted that each acoustical state is modelled by 3 HMM states in order to capture beginning, middle and ending part of it. A basic HMM model can model phonemes, syllables, words or even whole phrases. It is a trade of between accuracy of modelling and computational feasibility and availability of the training data. Some frequent usually short words can be modelled by a single HMM model. A good balance is achieved by so called tied triphones models which are used by most of the employed systems. Whatever training criterion is used there are no close formulas for optimal parameters thus several iteration cycles must be applied in a controlled way. Unfortunately only local maxima are guaranteed to be found. Following the classification theory HMM models are also prone to be overfitted, thus a validation set must be used to detect and prevent this phenomenon. The model complexity, modelled speech units and training strategies depend on the particular application, and are mutually adjusted in order to achieve best trade-off between accuracy and robustness.

2.6.2.2. HMM variations and modifications

There exist tree types of classical HMM models: discrete, continuous, and semi continuous. They differ by the way the *P(x/Si)* is modelled. Discrete HMM uses VQ techniques so as to get only finite set of vectors for which it is possible to calculate its occurrence probability in each state, these are not use very often only in simple applications or in the presence of noises. Continuous models use predefined type of distribution which may not fit to the real one but can be a good generalization in the presence of limited data. Semi-continuous models model the whole space by continuous distributions but particular states contain only a-priori probabilities of these distributions. Very recently some most outstanding results were achieved by hybrid HMM models especially those connected with neural networks (however the idea is not so new). In such hybrid connection other classification techniques like SVM or NN are used instead of GMM modelling. SVM [KruPCP06] and NN [TreTNN03] are discriminative methods thus it is natural for them to separate between different states. Moreover it was show that for multilayer perceptron trained by minimizing mean square error or cross entropy, the network provides on its outputs MAP probabilities, i.e. $P(C_i/x)$, where C_i is a i-th class (may be phoneme, etc.). Thus it is quite natural to interconnect both techniques. Thus the training of HMM usually reduces to the estimation of transition and a-priori probabilities, while the distributions (classifiers) are learned in separate way using segmented data. In order to increase the accuracy it was show that hierarchical structures of NN are beneficial. At higher hierarchy only the most distinctive classes (broad class) of phones are discriminated and using these estimates finer separation is done.

2.6.3. Team Expertise background

As it can be seen speech recognition task is a complex one and usually requires lot of effort to implement and tune in a real system for particular environment and application. Thus the aim of STUBA research is to select the best features and possibly some additional preprocessing to suppress adverse effects of environment. Furthermore it involves a design of proper training method for robust and accurate HMM models, and selection and evaluation of more types of models and their structures. Overall accuracy and practical feasibility (real time) is at the main focus.

STUBA has been intensively involved in speech recognition domain since 2003. STUBA designed and evaluated several modifications to training procedures in order to get more accurate and robust models for Slovak language. Furthermore, tested and tuned several parameters in the training process and models' structures. STUBA also tried to optimize the decoding settings for some applications using finite state grammar. STUBA was involved in the project that created intelligent speech communication interface in Slovak, where services like whether forecast or train departures were implemented. Additionaly, speech recognition systems for Italian and Romanian languages were implemented using Slovak database with considerable results.

2.7. Speech synthesis

Although not directly part of an HBB-NEXT use cases, for purposes of multimodal control of HBB console, the menu navigation will be more comfortable with using visual and/or spoken commands on the side of the HBB console, too. This comfort cushiness denotes the spoken response of the HBB console. This leads to the design and implementation of multi-language speech synthesizer that can read loudly the menu commands, commands for navigation, etc.

The speech synthesis is a scientific discipline that deals with generation of artificial speech signal. Through the time, two basic demands arose that need to be fulfilled. The first demand is the intelligibility of the speech. It means the possibility to understand the meaning of the speech. First synthesizers that produced intelligible synthetic speech were formant synthesizers about 20 years ago. The second demand is the naturalness of the synthetic speech. It means that the synthetic speech is indistinguishable from the human speech. The speech is sensed as natural, when – in a given context – it is impossible to specify if the source is the computer or the human. The main cause for artificiality of the synthetic speech is the strong parameterization (used i.e. in formant synthesizers). To suppress the artificiality, the modern synthesizers use method of concatenation of speech segments that are cut from original speech recordings and for synthesis purposes, they are combined in the required order.

This concatenation synthesis is nowadays the most prevalent approach in speech synthesis. Its basis is the speech database containing the recorded speech segments that are chosen by the synthesis. The question is the choice of the size of the segments. Segments can be the phrases, words, syllables, diphones, phonemes, half-phonemes, HMM(Hidden Markov Model)-segments [DonCSL99], 5ms segments corresponding to the period of fundamental frequency [HirISC04] and other segments that are the combinations of the previous segments. The size of the segment influences the quality of the synthesized speech. The bigger is the size of the segment, the less segments are used for synthesis and the less synthetic concatenations is done (meaning less potential areas of quality degradation). The big size of segment brigs also the problems of extra-large database, because there are so many phrases, words or syllables, that their acquiring or storing in the computer is almost unsolvable.

The TTS (text-to-speech) synthesis usually consists of two levels – NLP (Natural Language Processing) and DSP (Digital Signal Processing) [BlaICC94],[DutJEE97]. Input to the DSP (low-level synthesis) are the parameters generated in the high-level synthesis – the NLP, which generates the parameters from the original input text. The DSP usually comprises two steps. The first step is the choice of the segments from the speech database; the second step is the postprocessing involving the additional adjusting of segments, which includes the prosody modification or discontinuities smoothing at the concatenation points.

2.7.1. Available solutions

In this part we will describe the state-of-the-art of the most known speech synthesis systems. There are more techniques for achieving speech synthesis. Many of them are used in entertainment production (games, animations etc.). For Example Animo Limited announced in 2007 the software application package development based on FineSpeech (its speech synthesis software) able to generate lines of dialogue according to user specifications [ANNWeb07]. In 2008 this application reached maturity after NEC Biglobe announced a web service that allows users phrase creation from voices of characters in Code Geass: Lelouch of the Rebellion R2 (game) [ANNWeb08].

The most important usability of TTS (Text-To-Speech) Synthesis is for disability and handicapped communication aids. This specific communication has become widely asked and used in Mass Transit in recent years, not only for handicapped people. It is more comfortable to listen to some information, than to see and read it. Well-known companies TalkingSigns and TextSpeakSystems designed TTS for Digital Signage for the Blind. This system uses standard speakers or radio receivers.

For speech synthesis are most used following types of speech synthesis methods. To each method is associated at least one example of speech synthesizer:

2.7.1.1. Formant Speech Synthesis

In formant synthesis, the synthesized speech output is created using additive synthesis and an acoustic model. Parameters as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech.

eSpeak is a compact open source software speech synthesizer designed for Linux, Windows and other platforms. This synthesizer is based on the formant synthesis approach, and provides many languages in a compressed size. eSpeak is used by Google Translate and some projects including Ubuntu.

2.7.1.2. Articulatory Speech Synthesis

Second approach uses the articulatory speech synthesis. Such synthesis is based on real-time generation of synthesized speech by rules. Synthesizer directly simulates the human vocal tract. Very necessary are the first three formant frequencies.

Gnuspeech is Extensible TTS computer software package that produces output speech with articulatory synthesis is called Gnuspeech. The quality is much better for English than for example for Chinese [GnuWeb10].

2.7.1.3. HMM-based Speech Synthesis

HMM-based synthesis is a synthesis method based on hidden Markov models. In this approach, the frequency spectrum that represents the vocal tract, fundamental frequency that represents the vocal source and duration describing prosody of speech are modelled simultaneously by HMMs. Speech waveforms are generated from HMMs themselves based on the maximum likelihood criterion.

HTS is HMM-based Speech Synthesis System (HTS) that has been developed by the HTS working group. The training part of HTS has been implemented as a modified version of HTK (Hidden Markov Model Toolkit).

2.7.1.4. Concatenative Speech Synthesis

As mentioned before, the concatenative synthesis is based on the concatenation of segments of recorded speech. In general, the concatenative synthesis produces the most natural-sounding synthesized speech, but the differences between natural variations in speech and the nature of the automated techniques for segmenting the waveforms can sometimes result in audible glitches in the output.

iFlyTek developed by Anlui USTC iFlyTek Co., Ltd (iFlyTek) refers in W3C paper adaptation of Speech Synthesis Markup Language for producing a mark-up language Chinese Speech Synthesis Markup Language (called CSSML). This language can include markup to clarify special pronunciation of different Chinese characters and allows adding some information about prosody [YanWeb05]. This synthesizer takes (as mentioned above) a corpus-based approach. It means that the biggest problem is to deal with unexpected phrases not matched with corpus. A lot of involved data is not published directly by iFlyTek but is able to see from the commercial products licensed by iFlyTek with their technology to Bieder'sSpeechPlus (one of more examples), where is 1.3 Gigabyte download, 1.2 Gigabytes of them is used for the highly-compressed data for a single Chinese voice. iFlyTek's synthesizer is able to synthesize mixed Chinese text with English text using the same voice (it means literally Chinese sentences containing some English words). English part of synthesis is considered as "average".

This corpus is heavily dependent on Chinese characters and it is not possible to synthesize directly from pinyin (the official system to transcribe Chinese characters into the Roman alphabet). Another corpus-based approach is used by Tsinghua University's SinoSonic, with Harabin voice data taking 800 Megabytes.

Another type of TTS synthesizers is Ekho. It is Chinese text-to-speech software, which supports Cantonese, Mandarin and Korean. This type concatenates sampled syllables. Ekho is able to speak English through Festival (general multi-lingual speech synthesis system) and supports Linux and Windows.

2.7.1.5. MBROLA

Next algorithm to achieve synthesized speech is called MBROLA. It is software distributed without financial cost and only in digital form. The MBROLA project web page represent worldwide collaborative project and provides a lot of diphone databases for many spoken languages. Databases could be composed form diphones (as mentioned above) or triphones and most used are corpus databases because of their natural sounding.

2.7.2. Speech synthesis consortium background

STUBA has worked on several projects that involved development of speech synthesis systems. In the next paragraphs, the background of this work and STUBA experience will be described.

The main concern of STUBA's long-term work is the development of Slovak speech synthesizer. The principle of each synthesizer's core is the same: speech units are chosen from the speech database and put together to create desired outcome. To achieve natural synthesized speech the synthesizers should fulfil more complex tasks like pre-processing and post-processing. As experience has shown, it's better to implement each process independently and to create a module for each process. Depending on desired quality and/or availability of resources it can be chosen which module should contribute to the process of synthesis and vice versa which one can be turned off to save time or energy. The partial results / modules are integrated together to form the whole speech synthesis system. The next paragraph briefly mentions specific areas of research and their applications.

The core of the speech synthesis is based on concatenated synthesis, which means that outcome speech is created from smaller speech segments stored in a database. Naturally synthesized speech is, by definition, speech that is not recognizable from human speech. Creating this kind of speech is a difficult and complex task.

The first way how to reduce artificiality in synthesized speech was to create of database with units cut from recorded human speech. Also, the larger (and thus better quality) the database unit is, the more natural speech we get. The size units used in database depends on database volume - the larger the database, the larger unit in use: phonemes, diphones, words or group of words. Nevertheless human ear still detects an artificially merged signal that by design locally shows some inconsistencies [RozIWS11], [RybJDC11].

However, the goal of perfect synthesized speech requires more than good database. Text has to be analysed and prepared for synthesis, e.g. it has to be retyped into machine language, abbreviations and numbers are to be rewritten into full form and also Slovak language parameters like word class, gender, case and number are to be determined. Afterwards prosody of synthesized speech has to be modified to make the speech more fluent and natural [TurELM08].

The first module is devoted to design, characterization and implementation of process for an automatic creation of diphone speech database (our speech synthesizer uses diphone speech units). In order for the synthesizer to create a synthesized speech the database has to contain speech units and a descriptor file of database structure. The system is divided into several blocks; each of these has only a partial function. A block of recording speech corpus is designed to record the set of speaker's words forming the speech corpus. A block of phonetic transcription translates the written form of words to phonetic representation. A block of corpus segmentation uses a DTW method to determinate the time borders of speech units in corpus. The solution includes blocks of corpus analysis, selection of appropriate speech units and finally, a block of final database creation.

Further work, which falls into the category of speech synthesis, is devoted to coordinate work of basic building blocks of speech synthesis. Based on the input XML file, the program generates the corresponding audio recordings. In addition, the program creates an XML file that holds information about the borders of phonemes in a recording.

Synthesizer is able to work with multiple databases. It is possible to extend its functionality to work with databases in other formats. Synthesizer is programmed in the Java programming language.

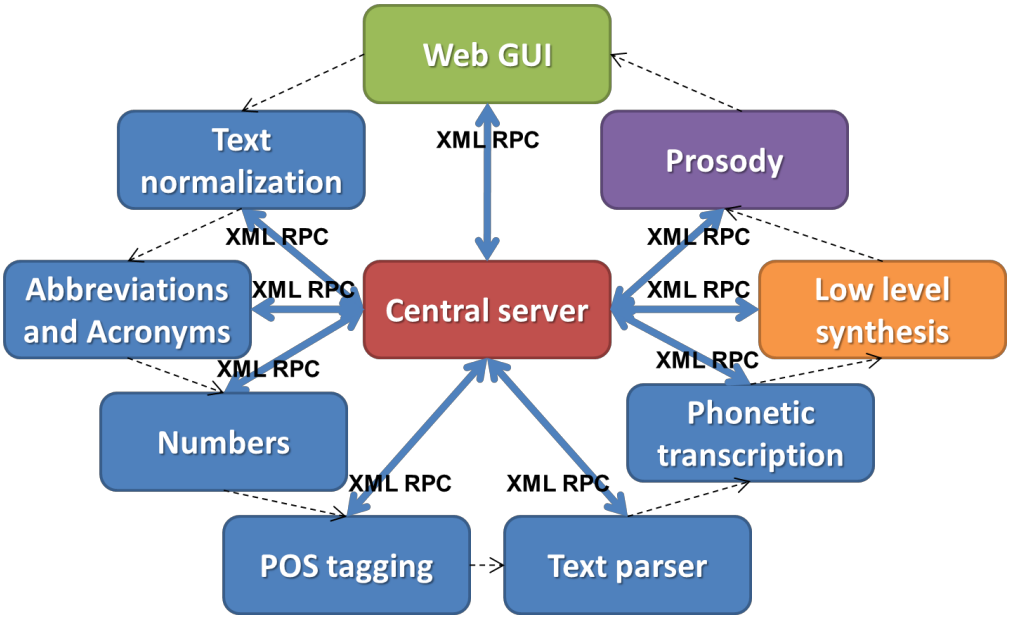


Figure 2.7.2.1: Modular speech synthesizer

The heart of modular synthesizer (Figure 2.7.2.1) is a control module, whose task is to receive requests for the synthesis from the web interface. Control module must perform data processing in the respective modules. The result is sent to the web interface. Communication between modules is ensured by XML RPC. The control module is independent from the type and number of connected modules. Condition for the proper functioning of the modular synthesizer is compliance with the agreed XML format. Control module is programmed in the Java programming language [RozRED11].

The basic assumption for creating LTS rules is that each phoneme has its own transcription into phonetic form. Our application for phonetic transcription is implemented in Python programming language and runs as a client - server concept. This application can generate SAMPA transcription of the text. Phonetic patterns are stored in the dictionary and LTS rules carry information in a binary structure of CART. Output data is presented in a clear XML format [VasELM11],[VasRED11].

Another large area of involvement for a speech synthesizer is abbreviations. Correct reading of abbreviations makes synthesized speech more natural. Reading of abbreviations is very individual and differs from person to person, thus making the task even more difficult. There are more possibilities how to read abbreviations: they can be replaced by words, spelled or pronounced if it's possible.

It is also important to consider how many times an abbreviation is placed in a text. If it's in an article more times, it can be replaced by whole words (full meaning of abbreviation) the first time and spelled later on, or it can be read in case it's readable. The module for abbreviations reading works as follows: a pre-processing phase looks for abbreviation in the synthesized text. In case of longer article is this text analysed and based on abbreviation count it is decided how to read it. If possible, the abbreviation is replaced by its glossary explanation. In case of multiple occurrences the first time it can be read in full meaning and later only spelled, or it can be mixed – spelled and read in full meaning. Another possibility how to increase the naturalism of abbreviation expansion is to simply read them as words. Of course not every expression can be pronounced fluently, e.g. expressions like „FIFA", „UNESCO" are comfortable to pronounce. In Slovak language a word is vocal if it contains vowels (a, e, i, o, u) or syllabical consonant (r, ŕ, l, and Í). In such case can a spelling be replaced by reading and the synthesizer chooses a method for reading as previously described. In case abbreviation is not in glossary, it is checked if it is vocal. In that case is read or spelled. In case it's not vocal, synthesizer spells it every time. The module for abbreviations reading is implemented in MS Visual Studio 2008 [TotELM11].

The words in Slovak language are inflected, e.g. they have different forms in different cases. Each word has a group of phonemes which are never changed, called a word core. The difference is in a suffix – the last three phonemes. In Slovak language words with the same suffix are grouped together to the word class. In speech synthesis process it is very important to determine the correct word class. It's part of the correct abbreviation reading and also the correct numbers reading. Determination of word class is done based solely on suffix. All 3-phonemes suffixes are stored in vocabulary. In case word class is ambiguously defined using this vocabulary, 4-phonemes suffix should be used.

Even if 4-phonemes are not sufficient, 5-phonemes suffix is used. Separate vocabularies for 4 and 5 phonemes suffixes are used. This module was implemented in Perl [RybJDC11].

The synthesized speech usually sounds artificial and unnatural. The main goal of post-processing is to modify prosody (e.g. to set frequency, energy and speech rate) in such a way that it is indistinguishable from human speech.

Slovak language has several types of intonation in sentences – for example three types of questions: question type Y –expected answer yes or no, type R – this question contains more sentences connected with word whether and type Z – all other types. For all other sentences there is only one type. In case a comma is detected, we differentiate these types of compound sentences: type V – the first sentence has upward melody and the second one decreasing melody, type O – all sentences have decreasing melody. Parameters such as fundamental frequency, speech rate, rhythm and accent also have rather large influence on intonation. Following methods are used to change of intonation in our module: change of fundamental frequency, change of speech rate, change of energy, change of prosody of whole sentence according sentence type. Prosody modification module is implemented in MS Visual Studio 2008[RozIWS11].

2.8. APIs for multimodal interfaces

2.8.1. Gesture recognition projects, overview

There are multiple on-going projects developing devices and algorithms mainly for the consumer market that aim to employ gesture recognition. Here is a short survey on the current status in the field of consumer electronics.

Microsoft Kinect for Xbox 360

Kinect is a peripheral device for the game and entertainment console Xbox 360 introduced in late 2010 by Microsoft Corporation. The device features an RGB camera, depth sensor and multi-array microphone which provide full-body 3D motion capture, facial recognition and voice recognition capabilities. The sensor provides recognition of majority of human body's joints, up to the level of wrists, and their position in the 3-D space in front of it.

The CSAIL LIS and Robot Locomotion Group presented [KinWeb11] an application based on open-source drivers where they demonstrated gesture recognition on the level of fingers.

As of February 2012, Microsoft is releasing an adjusted version of the sensor, Kinect for Windows. It is supposed to be optimized for shorter distances between the user and the sensor and, apart from the full body capture, body gesture capture should be possible for the upper body only [KinOfi].

See more on MS Kinect in the following paragraph.

Sony EyeToy for PlayStation 2

The EyeToy for Play Station 2 was the first device to bring gesture recognition as a means to control virtual environment. Richard Marks developed and first demonstrated the camera device in August 2002 which came to the market the next year. The EyeToy employs 320x240px sensor which produces 60 frames per second. There is a Play Station 3 version of the camera called PlayStation®Eye which provides higher, 640x480px resolution, or 120 frames per second video capture speed [EyeWik12].

Asus Xtion

The Asus Xtion family of sensors is in many ways similar to Microsoft's Kinect. However, there are some differences between the two. The Xtion sensor relies basically on the same infrared camera principle as the mentioned Kinect. The depth sensor captures IR light emitted from the emitter. Even though Xtion is from the beginning meant to work with PCs (Kinect is primarily designed for Xbox 360) its minimal capturing distance of 0.8m makes it more suitable for living room experience. There are two versions available: without (PRO) or with the RGB camera (PRO LIVE) which provides additional VGA (640x480 pixels) video output. The software development kit for Asus Xtion is based and relies on the OpenNI iniciative which attempts to certify and promotethe compatibility and interoperability of Natural Interaction (NI) devices, applications and middleware [OpenNI].

Samsung Smart TV 2nd generation

According to Boo-keun Yoon, new generation of Samsung Smart TVs will enable voice and gesture navigation through the TV menu with face recognition to personalize the user's profile in the TV and, i.e. to load the profile automatically based solely on the face detection. The aim of Samsung Electronics is to prospectively avoid the use of standard remote control (as revealed on CES 2012 [CroWeb12], [HarWeb12], [SamYou12]).

2.8.2. Gesture Recognition Projects, Microsoft Kinect in-depth description

The Kinect is currently the hardware that brings new possibilities of control and provides developers with the greatest opportunities for innovative programs.

This hardware is a device with two cameras that makes use of infra-red (IR) illumination to obtain depth data, colour images and microphone array. The IR is used as a distance ranging device much in the same way a camera autofocus works. It is claimed that the system can measure distance with accuracy in millimetres at 1m. The base resolution of depth camera is 320x240 pixels but by interpolation up to 640x480 and the colour image up to 1600x1200.

Figure 2.8.1.1 The Microsoft Kinect sensor

A custom chip processes the data to provide a depth field that is correlated with the colour image. That is the software can match each pixel with its approximate depth. The pre-processed data is fed to the machine via a USB interface in the form of a depth field map and a colour image. [KINWEB]

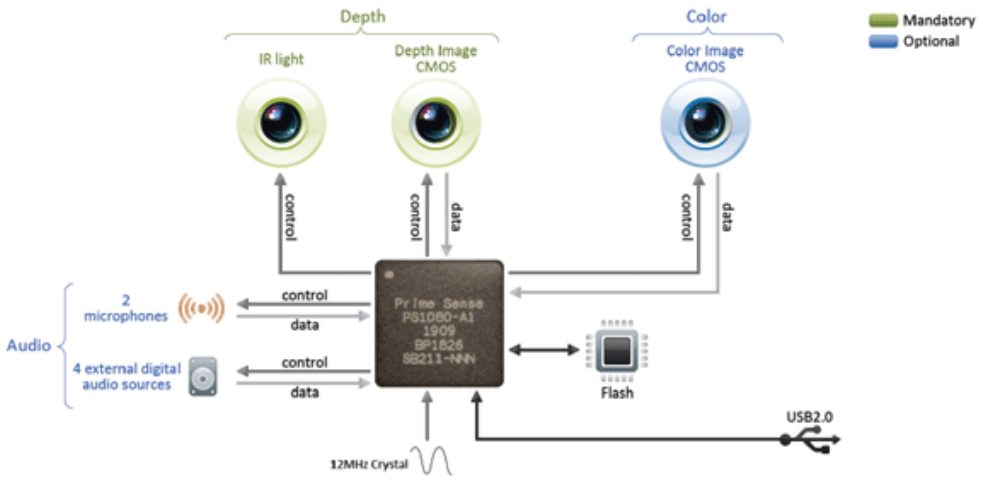


Figure 2 8.1.2: Scheme of major parts [KINWEB]

The depth map

The laser fires a pulse and then times how long it takes for the pulse to reflect off a surface. This is most definitely not how the system works. The accuracy and speed needed to implement so called "time of flight" distance measuring equipment is too much for a low cost device like the Kinect. Instead the Kinect uses a much simpler method that is equally effective over the distances we are concerned with called "structured light". The idea is simple. If you have a light source offset from a detector by a small distance then the projected spot of light is shifted according to the distance it is reflected back from. So by projecting fixed grid of dots onto a scene and measuring how much each one has shifted when viewed with a video camera you can work how far away each dot was reflected back from.

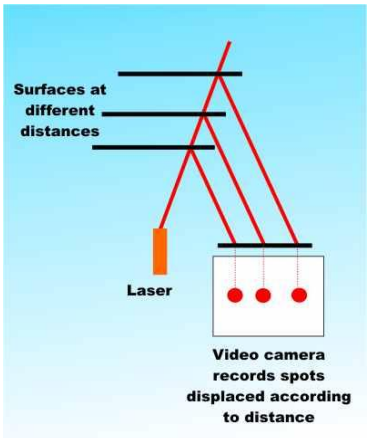


Figure 2.8.1.3: IR Laser [KINWEB]

The actual details are more complicated than this because the IR laser in the Kinect uses a hologram to project a random speckle pattern onto the scene. It then measures the offset of each of the points to generate an 11 bit depth map. A custom chip does the computation involved in converting the dot map into a depth map so that it can process the data at the standard frame rate. [KINWEB] After computing you give the map of distance of each pixel in depth image.

Microsoft SDK for Kinect

Microsoft SDK for Kinect is a package of libraries for developers. These libraries are addressed to programmers in C#, C++ or Visual Basic. Libraries help to program easier. Now it is able to recognize more than 12 joins of body. There are next new features in each next new version of SDK. Kinect with SDK will come in each department.

Tracking

It is a useful feature which is contained in Kinect sensor. It can automatically recognize human body in scene and determine joins of body. The features are all based on a simple formula:

$$f=d(x+u/d(x)) - d(x+v/d(x)), \quad (2.22)$$

where (u,v) are a pair of displacement vectors and d(c) is the depth i.e. distance from the Kinect of the pixel at x. This is a very simple feature it is simply the difference in depth to two pixels offset from the target pixel by u and v. The only complication is that the offset is scaled by the distance of the target pixel i.e. divided by d(x). This makes the offset depth independent and scales them with the apparent size of the body. [KINWEB]

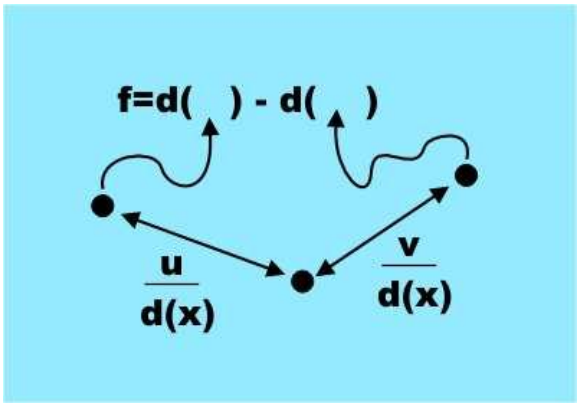


Figure 2.8.1.4: Principe of body tracking [KINWEB]

Face Recognition Projects and Standards

Face detection is always the part of the pre-processing in face recognition tasks. There are also many projects involving face detection as the crucial tool for face extraction from the image of the monitored environment.

2.8.2.1. Projects

One of the **projects** which use face detection for its essential part is 3D Face EU project for EU-citizens passport. Projects wants to develop system for biometric-enabled border control based on 2D face recognition technology. Its improvement is the 3D Face project is therefore focused on 3D face recognition technology research, including fusion with 2D face recognition technologies, and its application in secure environments. Actual systems use cameras to take a front picture of the face.

The resulting image is then processed by a recognition system. The first task is to filter the actual face from the image. This is not an easy task as a change of hairstyle, beard or glasses can severely disturb the extraction process.

There are also several projects in security system which involves face extraction from the public assemblies such as football match etc. One of these systems is used by Federal Bureau of Investigation for extracting and certifying faces of people present on the inauguration of the president of the United States. This leads us to the system which can extract various human individualities in image.

2.8.2.2. Standards

Standards in the field of face detection are mostly part of the biometric standards in which face detection is a first stage to the extraction of various biometric characteristics:

ISO/IEC 19794-5:2005 Information technology – Biometric interchange formats – Part 5: Face image data

ISO/IEC 19794-5:2005 specifies scene, photographic, digitization and format requirements for images of faces to be used in the context of both human verification and computer automated recognition. The approach to specifying scene and photographic requirements in this format is to carefully describe constraints on how a photograph should appear rather than to dictate how the photograph should be taken. The format is designed to allow for the specification of visible information discernible by an observer pertaining to the face, such as gender, pose and eye colour. The digital image format can be either ISO standard JPEG or JPEG2000. Finally, the 'best practice' appendices provide guidance on photo capture for travel documents and face recognition performance versus digital compression.

ANSI INCITS 385-2004 Information technology – Face recognition format for data interchange

This standard specifies definitions of photographic (environment, subject pose, focus, etc.) properties, digital image attributes and a face interchange format for relevant applications, including human examination and computer automated face recognition.

Table 1 mentions some projects funded under 7th FWP (Seventh Framework programme), which also deal with speech recognition and speaker identification.

Speech Recognition Projects and Standards

Projects Funded under 7th FWP (Seventh Framework Programme)

Project	Project details	Similar tasks
Distant-speech Interaction for Robust Home Applications (DIRHA)	Project Reference: 288121 Start Date: 2012-01-01 End Date: 2014-12-31 Project Status: Execution	speaker localization, acoustic echo cancellation, speech enhancement, acoustic event segmentation and classification, speech recognition, speaker identification (and verification)
Audio-VIsual Speech Processing for Interaction in Realistic Environments (AVISPIRE)	Project Reference: 247948 Start Date: 2009-10-01 End Date: 2013-03-31 Project Status: Execution	multi-party human interaction, speech recognition
Bayesian biometrics for forensics (BBFOR2)	Project Reference: 238803 Start Date: 2010-01-01 End Date: 2013-12-31 Project Status: Execution	speaker recognition
Trusted Biometrics under Spoofing Attacks (TABULA RASA)	Project Reference: 257289 Start Date: 2010-11-01 End Date: 2014-04-30 Project Status: Execution	speaker verification
Natural interaction with projected user interfaces (NIPUI)	Project Reference: 221125 Start Date: 2008-03-01 End Date: 2010-02-28 Project Status: Completed	speech recognition, directional speakers
Mobile Biometry (MOBIO)	Project Reference: 214324 Start Date: 2008-01-01 End Date: 2010-12-31 Project Status: Completed	voice authentication

Table 2.8.1.5: Projects funded under 7th FWP with similar tasks [CorWeb12]

Some standard organizations such as the European Telecommunications Standards Institute (ETSI), the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF), deal with speech recognition. Tables 2 and 3 mention some important standards.

Speech Standards

Organization	Standard No.	Standard Title
ETSI	ES 201 108	Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms
	ES 202 211	Distributed speech recognition; Extended front-end feature extraction algorithm; Compression algorithms; Back-end speech reconstruction algorithm
	ES 202 050	Distributed speech recognition; Advanced front-end feature extraction algorithm, Compression algorithms
	ES 202 212	Distributed speech recognition; Extended advanced front-end feature extraction algorithm; Compression algorithms; Back-end speech reconstruction algorithm
	TS 126 243	Digital cellular telecommunications system (Phase2+); Universal Mobile Telecommunications System (UMTSTM); ANSI C code for the fixed-point distributed speech recognition extended advanced front-end

Table 2.8.1.6: The European Telecommunications Standards Institute (ETSI) standards [EtsWeb12]

Organization	Standard	Standard description
W3C	VoiceXML (VXML)	a language for for creating audio dialogs that feature synthesized speech, digitized audio, recognition of spoken and DTMF key input, recording of spoken input, telephony, and mixed initiative conversations
	Speech Grammar Recognition Specification (SRGS)	a document language that can be used by developers to specify the words and patterns of words to be listened for by a speech recognizer or other grammar processor
	Semantic Interpretation for Speech Recognition (SISR)	a document format that represents annotations to grammar rules for extracting the semantic results from recognition

Organization	Standard	Standard description
	Pronunciation Lexicon Specification (PLS)	a representation of phonetic information for use in speech recognition and synthesis
	Speech Synthesis Markup Language (SSML)	a markup language for rendering a combination of prerecorded speech, synthetic speech, and music
	Call Control (CCXML)	a markup language to enable fine-grained control of speech (signal processing) resources and telephony resources to perform scenarios such as call screening, whisper call waiting, and call transfer
	State Chart XML (SCXML)	a markup language to simply and precisely represent the semantics of state machines
IETF	Media Resource Control Protocol Version 2 (MRCPv2)	allows client hosts to control media service resources such as speech synthesizers, recognizers, verifiers and identifiers residing in servers on the network

Table 2.8.1.7: The World Wide Web Consortium (W3C) [W3cWeb12] and The Internet Engineering Task Force (IETF) standards [IetWeb12]

Speech Synthesis Projects and Standards

There are multiple ongoing projects developing speech synthesis for variant languages. In the following tables is a comparison of speech synthesis programs and standards for speech synthesis with information that can be officially obtained.

Project Name	Creator(s)	First public release date	Latest stable version	Software license	Cost
Apple PlainTalk	Apple Inc.	1984	2007, October 26	Bundled with Mac OS X	Bundled
AT&T Natural Voices	AT&T Natural Voices		2008	Commercial	$295 - $995
Cepstral	Cepstral	-	-	Proprietary	$29+
eSpeak	Jonathan Duddington	2006, February 10	2011, April 25	GPLv3+	Free
Festival Speech Synthesis System	CSTR	-	2010, November	MIT-like license	Free

Project Name	Creator(s)	First public release date	Latest stable version	Software license	Cost
FreeTTS	Paul Lamere Philip Kwok, et al.	2001, December 14	2009, March 9	BSD	Free
IVONA TTS	IVONA Software	2005	-	Commercial	-
Kurzweil 1000 and Kurzweil 3000	Kurzweil Educational Systems, Inc.	1996	2005	Commercial	-
Loquendo	Loquendo	-	2011	Commercial	-
Nuance Vocalizer	Nuance Communications, Inc.	-	-	Proprietary	-
Praat	Paul Boersma David Weenink	-	2011, September 11	GPL	Free

Table 2.8.1.8 Speech synthesis projects

Project Name	Online demo	Available language(s)	Available voices(s)	Programming language	Operating system(s)
Apple PlainTalk	-	English (United States)	15+	-	Macintosh
AT&T Natural Voices	Yes	English (British), English (Indian), English (US), French, French (Canadian), German, Italian, Spanish (Latin American)	20	C++	Linux Windows
Cepstral	Yes	English (British), English (US), Italian, French (Canadian), German, Spanish (American), ...	25+	-	Mac OS X, Windows, i386-Linux, x86-64-Linux, Sparc-Solaris, i386-Solaris
eSpeak	Samples	English…	Several	C++	Linux, Windows, Mac OS X, RISC OS

Project Name	Online demo	Available language(s)	Available voices(s)	Programming language	Operating system(s)
Festival Speech Synthesis System	Yes	English…	Several	C++	Linux, Windows
FreeTTS	-	English…	Several	Java	Cross-platform
IVONA TTS	Yes	English (British), English (US), German, American Spanish, Castilian Spanish, French, Welsh, Welsh English, Polish, Romanian	26	C/C++	Windows, Linux
Loquendo	Yes	English (Australian), English (British), English (US), Castilian Spanish, Catalan, Valencian, Galician, French, German, Italian, Greek, Portuguese, Swedish, Dutch, Polish, Brazilian Portuguese, Mandarin Chinese, Mexican Spanish, Chilean, Argentinean, American Spanish, Canadian French, Turkish, Finnish, Russian, Danish, Norwegian, Arabic, Romanian	74	-	Windows, Linux
Nuance Vocalizer	Yes	English (Australian), English (British), English (US), Portuguese (Brazilian), French (Canadian), German, Spanish (Latin American)	10+	C/C++	-

Table 2.8.1.9: Speech synthesis projects (cont.)

Organization	Proposed standard	Purpose	Status
W3C Voice Browser Working Group	Semantic Interpretation Markup Language (SSML)	Specify the pronounciation, speaking rate, volume, pitch, voice and inflectons for speech synthesis engines	W3C Last Call Working Draft
W3C Voice Browser Working Group	Semantic Interpretation	Extraction and translation of words from text into a semantic representation	W3C Working Draft
W3C Multimodal Interaction Working Group	Extended Multimodal Annotation (EMMA)	Specify annotations to semantic representation from modality components	W3C Requirements
Internet Engineering Task Force (IETF)	SpeechSC Requirements for the distributed control of ASR, speaker verification and TTS Resources	Protocols for managing remote speech recognition, speaker identification and verification, and speech synthesis	Requirements and draft protocol evaluation has been published
European Telecommunications Standards Institute Aurora Project (ETSI)	Speech commands. Generic spoken command vocabulary for telephony devices and services	Spoken commands for communication devices and services in five European languages – English, French, German, Italian, Spanish	Approved standard

Table 2.8.1.10: Speech synthesis standards

2.9. Gaps analysis

2.9.1. Gesture recognition

The current methods of gesture recognition struggle with both hardware and software problems making enough room for more research in both hardware and software fields.

Almost no gesture recognition system nowadays is invariant to the surroundings and requires specific environmental conditions in order to be effective. The background of the scene must be static so that the only moving object the sensor acquires is the user's body. Furthermore, the devices, i.e. the Kinect, are as of yet capable of detecting two persons. More people in the field could fool the sensor with uncertainty of which persons to follow.

To avoid the scene illumination problem where low level of lighting could drive the sensor insensitive current sensors which exploit the 3D model-based detection methods are equipped with an infra-red light emitter and infra-red camera. This improvement greatly enriches image acquisition, however, at the cost of both higher price for the sensor and raised energy requirements. For the appearance based methods proper, sufficient and constant illumination is required for the sensor to be able to capture the surfaces' features. Colour of the light also plays an important part.

The current sensors have an operation distance ranging from approximately 0.8 cm up to 4 m. Considering the possible distance between the sensor and the person operating the device, maximum range of the sensor should be increased to improve the user experience in larger premises. This could be solved by applying current software technologies to higher definition image. It might however lead to the increase of computational complexity. Thus, sensors accommodating greater computational capacity should be employed or new and improved algorithms should be proposed which would be capable of processing an extended amount of data in the same time frame.

Yet other possibility is to devise algorithms which are more capable of gesture detection and recognition in the images provided by current sensory devices. Additionally, a slightly different approach might lead to a combination of various algorithms for gesture detection. The results could be used to receive a set of data sets to which statistical methods may be applied in order to decide the most probable of the known gestures.

Gesture recognition requires a database of possible gestures which is always limiting in terms of user's freedom to perform his/her own gestures. This problem could be solved either by teaching the end user gestures which may be used, or, vice versa, by employing teaching mechanism to adapt the system to the user. The system has to be robust enough to allow for imperfections in user's commands. On the other hand, it must only perform actions triggered by recognized gestures if recognition probability of the specific gesture was above required level.

2.9.2. Face detection

Efficiency of face detection is described by various parameters of the process. We use false rejection rate (FRR) and false acceptance rate (FAR) as a measure for accuracy of detection. FRR refers to the probability of misclassification of the images containing face and FAR indicates probability of misclassification of the images not containing face.

There are several ways how to improve results of the face detection task in order to provide more effective multi-user detection. One of the possible solutions is modification of image parameters such as brightness, colour distribution etc.

This leads us to the standard procedures such as spatial image filtration, histogram equalization or histogram specification.

In order to achieve better results of the face detection we propose to use the combination of existing approaches to the process with two or more stages. Each stage performs face detection independently and the result is obtained as a combination of the results of each process. Other option is sequential face detection performed by different methods. Each method eliminates more and more false face region candidates and to the end of the whole process the FAR will be minimalized. From the latency point of view, these algorithms are capable of real-time processing.

2.9.2.1. ASM, AAM: Gaps Analysis

Training Sets

ASM and AAM are both techniques that utilize a training set of images, where important points and regions were previously labeled by some other technique. This previous labeling severely impacts the performance of these methods including what kind of images (regarding pose and lighting for example) it applies to. Researchers that use ASM and AAM should discuss the training sets they use in more detail and also publish these labeled training sets for the benefit of future research. Large training sets with good quality of labeling should be available for images with a variety of conditions (lighting conditions, poses and expressions in the case of human faces).

Generating New Patterns

Both AAM and ASM can be used to generate new synthetic patterns [1]. These new patterns can be inserted into a database of known subjects to improve the performance of recognition methods. The generation itself usually requires more than one picture of the subject on which the generation was applied. ASM and AAM could be used to learn how generation of new patterns works on a small training set. In the case of human faces, rules for the generation of different expressions could be learned. These rules could be used to add or remove a smile or frown from a brand new image before using it for recognition tasks or adding it to a database of known patterns.

Multidisciplinary Approach

Both ASM and AAM use the notion of important points on a flexible object. Along with each point, the point's flexibility is stored, where the model's flexibility of the point should reflect the flexibility of the point in real life. The position and the flexibility of point however are only learned from the labelled training set of images used to construct the models. Intuitively, it is likely that the position and flexibility of these points reflects some natural phenomena that are being studied by other fields of science. In the case of a human face, important points and their flexibility are most likely determined by the properties of the human skull, muscles and skin. These are being thoroughly studied in the field of medical science.

It would be nice to see an expansion of the AAM and ASM methods that utilize the knowledge from other fields of science than statistics and computer science. A more multidisciplinary approach should yield better results in a broader variety of conditions.

2.9.3. Multi speaker recognition

As the multi speaker recognition problem is closely related to the single speaker recognition, most of the intensively studied areas for potential improvements are the same. Namely, the ever lasting search for optimal speech feature, channel normalization technique and classification method that would provide best possible results in connection with underlying processing stages. Thus for some brief overview on these open issues please see Gap analysis for speaker recognition.

Experiments on current diarization (who is speaking when) systems usually exhibit dearization errors rates (DER) ranging from 30% to 15% using headset microphones. Improvement of these rates is a tough task.

Even though, multi speaker problem encompasses single speaker one, the single speaker concept is extended to issues like: segmentation problem, clustering, overlapped speech detection, etc. As all these influence the overall accuracy, all mentioned stages must be considered in further refinement (as well as those shared with the single speaker recognition), namely:

Speaker segmentation

There are many techniques used for this task which are classified into more categories. For the distance based ones (most widely used) proper distance measures (to measure acoustical dissimilarities between adjacent segments) are vital. There is a search for a method, acoustical distance and optimal threshold setting to perform accurate and robust segmentation task using the shortest possible segments. Measures of various complexities are being suggested, tested, and compared as well as the manners how to adapt decision thresholds.

Speaker clustering

Once having all potential change points selected it is necessary to merge those belonging to the same speaker. To do this, more competing, so-called bottom up methods are known and used. However no general suggestion working in all condition exists yet. On the other hand there is some competition between two different approaches, top down vs. bottom up class of methods which has no clear winner yet, even some hybrid solution exists.

Speaker overlap

This area is attracting more focus and more work has to be done. Some algorithms have been proposed and used in diarization systems however most of them did not bring great improvements. In the presence of more microphones signal localization techniques can be used to detect the overlap. In the case of single channel scenarios mixed (overlapped) speaker features exhibit different characteristics as those gathered over single speaker segments so they may be used for detection. This is also an open research area.

2.9.4. Speech recognition

Accuracy of automatic speech recognition (ASR) is influenced by many factors. Firstly the voices differ among population and to make it more complex there is also great speaker variability, depending on mood, health, actual mental condition etc. Second there are ubiquitous noises that can span great range of levels and characteristics, both in time and spectra. The most serious for speech recognition are additive and convolutional noises that require different suppression techniques. Next the peace of speech changes where also inner word variability is relevant. Furthermore, in each natural language there is a huge vocabulary that evolves in the time. Speech consists of concatenated sounds that exhibit different time and spectral characteristic. To make it more difficult there is known a sever coarticulation phenomenon meaning that neighboring sounds influence each other. Currently there are several dictation system covering vocabularies of several thousand words, but the greatest challenge is a transcription of natural conversation in a real time. Even in simple small vocabulary systems (like digit recognition) 100% accuracy has not been reported, e.g. recognition of isolated words achieves best success rates (about 95% without noise) and is now commonly used for example in mobile phones. Situation change dramatically if the noises are present and vocabulary increases, e.g. at SNR 15dB (sound of

fan) the accuracy may degrade by 30% compared to the baseline system. Also a very adverse effect has the distance from the microphone. Of course by increasing the vocabulary the accuracy degrades as well as there are more options to make a mistake between competing words. Finally a usable ASR system must be working in real time that limits the use of computationally demanding techniques and simpler or suboptimal solution must be used, e.g. using Viterbi algorithm and pruning, etc.

As it can be seen from this brief description it is a very complex issue consisting of many stages that includes: advanced signal processing techniques (denoising, equalizations, voice activity detection, etc.), speech feature extraction (accurate and robust, that contains phonetic information and have low inner class variation), speech modeling and classification methods. The overall performance depends on all stages that were just mentioned. Thus careful design (selected methods), their adjustment and evaluation must be performed for any application and environment. It is important to note that no universal solution exists.

However some suggestions and expected performance exist for given stages:

Feature extraction

This task is still not completely resolved and thus more competing methods exists which some of them are preferable to the others in some environments and settings. Most famous feature extraction techniques are Perceptual Linear Prediction (PLP) and Mel-Frequency Cepstral Coefficient (MFCC) that are designed to capture positions and widths of formants that are acoustically perceivable. The usage of dynamic features like delta and double delta coefficients have been shown very effective. Since recently supervectors are being constructed over longer periods of time and these are “optimally” (in linear sense) transform (PCA, LDA, HLDA, etc.) to fewer dimensions providing higher accuracies. However, the main issues are: which time span to consider, what classes of phonemes to use and what transform and how to apply it in a given system. The quality of recognition will depend on background noise (e.g. fan, TV set, background speech etc.).

Classification

The most used and successful methods for speech recognition are Hidden Markov Models (HMM), Neural Network (NN) and more recently, Support Vector Machine (SVM). However hybrid solutions like HMM & NN and HMM & SVM are often reported.

The advantage of HMM is that they can easily handle the variable length of speech sequences, but do not completely reflect the true nature of speech. NN achieves a good level of classification (in theory they have potential to do optimal mapping) for fixed length samples but do not have good abilities to model variable sequences. Thus their combination with HMM may bring improved results for wider spectrum of applications. SVM is a newer classification method aiming to find the larges margin between classified samples (thus it is robust). Usually it is used in higher dimension space by kernel methods that add also the required nonlinearity. It is suitable for limited number of samples and requires constant vector size. Therefore it is directly not very applicable for general speech signals. Thus it may be used with HMM. For real applications it usually achieves better generalization ability and classification accuracy.

As there are many options and setting that are application and environment specific it is always necessary to tune in every particular system.

2.9.5. Speech synthesis

The problems that need to be covered in the actual state of speech synthesis research can be divided into more categories. The categories interrelate with tasks that need to be solved in the project and are described below.

Incomprehensible speech. Speech segments are joined together by concatenate speech synthesis. Undesirable acoustic noise may occur in the joints of two speech segments. This noise negatively affects the resulting sound quality and speech intelligibility.

Unnatural speech. Naturalness in the synthetic speech is today one of the most intractable problems. Although speech synthesis systems have improved considerably over the last 20 years, they rarely sound entirely like human speakers. The naturalness is mainly connected with the good intelligibility of the speech, natural prosody, timing, melodic control and relationships between various prosodic parameters.

More natural synthesized speech can be achieved in some types of speech synthesis (i.e. corpus speech synthesis) but it's not always possible to use only this methods.

Mispronunciation. Phonetic transcription of text sometimes generates wrong word transcript. The result leads to mispronunciation. There are several types of mispronunciation. Some of them are hardly noticeable, but others are significant and have a negative effect on the impression, especially when they keep repeating.

Text preprocessing errors. The preprocessing of the text carried out multiple processes. Each one has its gaps. The module of numerals preprocessing can wrongly determine the grammatical form of numeral. The processing of abbreviations and acronyms could encounter the following problems: the given abbreviation or acronym is not in the database, the program spells abbreviation that is not appropriate to spell or not spell abbreviation that is appropriate to spell, abbreviation is readable and is not read correctly. There are also cases when the program incorrectly determines the grammatical form of abbreviation. If an error occurs in POS tagging, it can result indirectly as follows: Based on the POS analysis phrases shall be determined.

Then based on the determination of phrases the resulting speech prosody could be created incorrectly. The big gap is the ability to read tables, web pages and other complex structures.

Errors in prosody. Two main problems in speech prosody have to be examined – the prominence and phrasing. The prominence (stress, accent, focus) can be represented as layered and multidimensional for different domains (syllable, word, etc.). Phrasing involves both coherence in the form of specific combinations of existing accentual gestures and separate boundary gestures. The main role in the prosody plays the intonation. There are various models that represent it and they are in general based on phonetic model (like MOMEL or Tilt) or phonolological model (ToBI, INTSINT).

The errors in prosody mostly come out from wrong projection of intonation formula, too small speech database or poor prosody modification.

Multiplatform incompatibility. In the case of the multi-device usage, the speech synthesis as a part of multimodal system needs to work on devices based on various system platforms. Even though it is almost impossible to adapt the speech synthesis system to all possible platforms, it is possible to run the speech synthesis on the external server side and to ensure the proper streaming of the audio data to the clients.

Multilingual speech synthesis. As the European Union has 23 official languages, it is needed to implement the speech synthesis for majority of them. One of the approaches can be to use diphone database that will contain all diphones used in the mentioned languages. However, while the diphone synthesis achieves very comprehensible speech its major disadvantage is unnaturalness, which can only be solved with good post processing of synthesized speech. The second limitation is that for each language a specific set of synthesis rules is necessary. To design the multilingual database that will work for every language is not a simple task. In fact, there is still no speech synthesizer working with such kind of database. Those are common problems for all types of speech synthesis like HMM-based, concatenative or formant speech synthesis. The main approach in the multilingual problem is to design the rules how to easily record and set up the speech synthesis database and how to create rules for it, which will be the main goal in the project.

Unsupported module integration. The elements and modules of speech synthesizer are implemented in different programming languages. This fact causes many complications in the mutual communication, exchange of data, in the management of the system and in the integration process. The modular architecture allows us to set up the communication between modules based on different programming languages and platforms and will be based on XML-RPC protocol.

The learning system. Currently the absence of system that can learn can be regarded as a major gap, but also a great challenge. The user cannot make modifications to the internal databases used in the TTS. Learning mechanism should be designed and implemented for each module of speech synthesizer with various levels of learning. Implementation of the learning system can solve almost all problems listed above.

3. Context-aware and multi-user content recommendation

3.1. Outline

This section looks into context-aware and multi-user content recommendations. Section 3.2 presents some relevant use cases and it gives an impression of the sheer width of the relevant design space. Section 3.3 describes the basic types of content recommendation systems: context-based and collaborative filtering. Section 3.4 looks into user profiles, needed to make content recommendations. Section 3.5 analyses the presentation of content recommendations to users. Section 3.6 considers the evaluation of recommender systems. Section 3.7 touches upon scalability of recommender systems. Section 3.8 highlights meta-data issues, which are especially relevant for content-based filtering. Section 3.9 considers privacy aspects of recommendation systems and cryptographic solution directions. Section 3.10 makes the step from individual recommendations to content recommendations for groups of users.

3.2. Problem statement

The goal of task 5.3 of the HBB-NEXT project is to develop a Content Recommender Engine for context-aware personalized multi-user & multi-device-based content recommendation [DOW_HBB_NEXT]. Whereas recommender systems have been studied for quite some years [RicRSH10], HBB-NEXT introduces several new aspects.

- **Context awareness**: e.g. time of day, location, other users present, and the social relationship between the users.
- **Multi-user**: tailoring content recommendations to the combined or aggregated preferences of group of users.
- **Multi-device**: making the recommendations and/or their presentation dependent on the available device(s).

Figure 3.2.1 provides an illustration of a multi-user recommendation use case that is considered in HBB-NEXT. Here the family Weber [HBB-NEXT_D2.1] decides to watch a movie on demand and asks the system for a recommendation. The system recommends “Puss in Boots”, because it is a fun family movie, suited for 12 year old children.

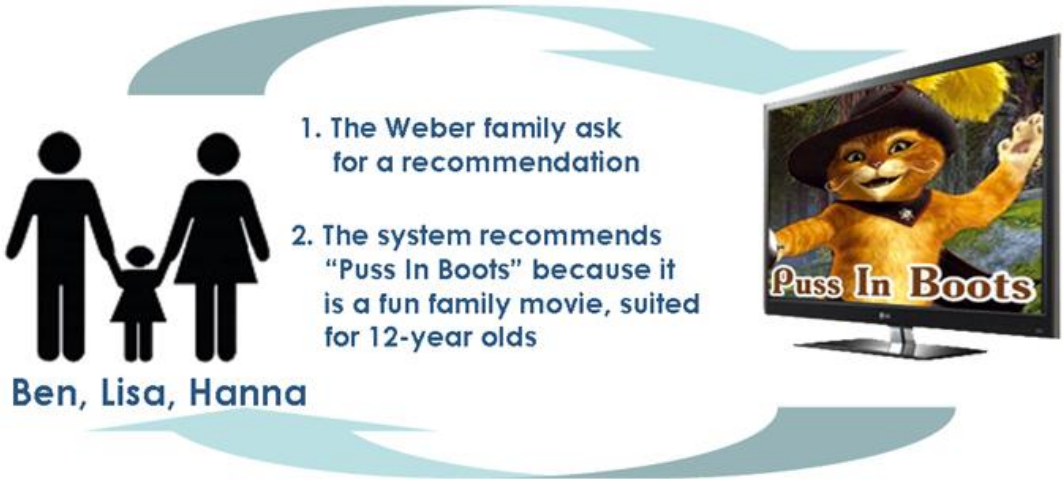


Figure 3.2.1: Illustration of a multi-user recommendation use case.

Even though the use case seems simple, much is happening "under the hood".

- The system knows which person the recommendation is for: Ben, Lisa and Hanna.
- The system may know characteristics of Ben, Lisa and Hanna individually, e.g. age.
- The system may know movie preferences of Ben, Lisa and Hanna individually.
- The system may know the family relationship between Ben, Lisa and Hanna.
- The system may know individual movie preferences in this group context.
- The system may have metadata about movies (content-based filtering).
- The system may have ratings by other users, groups or families (collaborative filtering).
- The system has algorithm(s) to generate recommendation.
- The system has algorithm(s) for combining profiles or recommendations to groups.
- The system may have algorithms to explain its recommendation(s) to the group.
- Perhaps the system also knows the time the recommended content is consumed, e.g. to adapt the length of the movie to the time left before the children have to go to bed

- Also, the system could consider if each of the group members has already watched the movie recently.

HBB-NEXT WP2 [HBB-NEXT_D2.1] distinguishes three usage scenarios, which are all for groups of users, emphasizing the multi-user aspect.

1. Family Sharing a House.
2. A Community of Like-Minded.
3. Friends Gathering to Watch a TV Event.

The scenarios contain use cases for community-based recommendations (e.g. “The BikeMe community has rated last years’ tour with 4.5 stars out of 5”), profile-based recommendation (e.g. “This item is already highlighted, because the system knows her favourites already”) and context-based recommendation (e.g. “The TV lists the tours organized next weekend in the range of 30 kilometres of his parents’ house”). Tables 3.3.2 and 3.3.3 provide the relevant text clippings and use cases from D2.1 [HBB-NEXT_D2.1].

Location	Text
Page 24, Early Evening	Mom, Dad, Grandma and Children enjoy Family TV: ... At 18:05 Marie (A.003) enters the living room just as the system recommends the regular animal documentary “Panda, Gorilla & Co” (PGC) based on the Weber’s group profile (U.005)....
Page36, last sentence of Friday 4th November 2011:	Requesting a Lift to the Tour: People picking up other bikers along the way; receive reputation points for this in the BikeMe community, which is also shown on their community profile (U.003, U.013)...
Page 41:	... The “SocialTV”-app asks Paul (U.038) whether he is willing to share his profile information with the EPG app running on Peter’s STB, and Paul accepts (U.038b) – now, the STB shows (U.029) a personalized EPG that matches both user profiles (U.005)...

Location	Text
Page 42	... Andrew arrives at Peter's house (U.009), and since Andrew is already known to Peter's STB (U.015, U.033) and also connected to it via his smartphone (0.028), Peter's STB detects and recognizes his? (U.022), and transfers the personalized EPG (which now reflects the likes of all three user profiles) to his smartphone (U.026, U.010)....

Table 3.3.2: Texts on content recommendations from [HBB-NEXT_D2.1].

ID	Title	Description	Notes	Linked enablers, if any
U.003	Community-based recommendation	A group of people gets a recommendation based on all their interests	Appears in: Scenario 1, 2, 3; Involved actors: A.001, A.002, A.003, A.004	Content Recommendation system for Multi-user Service Personalisation Engine
U.005	Profile-based recommendation	A user gets recommendation based on the interest stored in his profile	Appears in: Scenario 1, 2, 3; Involved actors: A.001, A.002, A.003, A.004	User Identity
U.026	Personalised EPG.	A person uses the (personalised) EPG.	Appears in: Scenario 1, 2, 3; Involved actors: A.001, A.002, A.004	Content recommendation system for multi-user service personalisation

Table 3.3.3: Use cases on content recommendations from [HBB-NEXT_D2.1].

3.3. Filtering types

Most research with regard to recommender systems has focused on recommendation algorithms, such as the social-based recommendation algorithms collaborative filtering [ShaPHF95], top-n-deviation [HerUIA00] and item-item filtering [LinIEE03] and the content-based recommendation algorithms case-based reasoning [SmyPTL00], genre learning [GorCAG04] and information filtering [PazMAL97].

Recommender systems have been applied in various types of applications, such as ecommerce websites, movie recommender systems, personalized electronic TV guides, and file downloads. Feedback from users is extremely important in recommender systems as feedback represent the user's interest and thus allows recommender systems to learn and provide better personalized recommendations.

However, almost all algorithm research and applications of recommender systems learn about the interests of the user via explicitly provided feedback of the user on objects using ratings. Only a few specialized recommender systems [AMA07] base their recommendations on implicitly acquired user data.

3.3.1. Content-based filtering

Content-based recommendation systems are to be understood as those where recommendations are made for a user based solely on a profile built up by analysing the content of items which that user has rated in the past. A well-known example of such a system is the TiVo Set-Top-Box [TIV09].

The content-based approach to recommendation has its roots in the information retrieval (IR) community, and employs many of the same techniques. Text documents (in the HBB-NEXT context: metadata elements describing pieces of A/V content) are recommended based on a comparison between their content and a user profile. Data structures for both of these are created using features extracted from the text of the documents. If the user liked a piece of content, weights for the elements within the metadata describing this piece can be added to the weights for the corresponding elements in the user profile, such as content genres, keywords, actor names etc.

This process is known as relevance feedback. As well as being simple and fast, it is empirically known to give improved results in a normal IR setting [BucSIR95].

Content-based

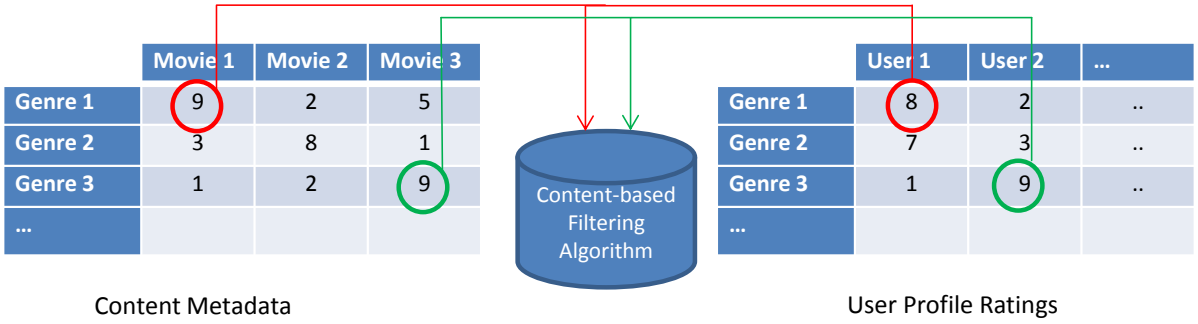


Figure 3.3.2.1: Content-based filtering.

A pure content-based system has several shortcomings.

Generally, only a very shallow analysis of the metadata can be supplied since content based filtering heavily relies on the quality of the given metadata. In most broadcast domains, the metadata being present for the description of content items is not “rich” enough to any useful feature extraction methods with current technology. Rich in this context means a lack of detailed, fine-granular metadata w.r.t. genres, actors, locations, technical information etc. This problem is further detailed in section 3.8.

Another problem which is well-known within this domain is that of over-specialization. Since the system only recommends items scoring highly against a user’s profile, the user is restricted to seeing items similar to those already rated. Often this is addressed by injecting a note of randomness.

Third, there is a general problem which is known by almost all kinds of recommendation systems—eliciting user feedback. The need to “feed” the recommender with ratings for items in order to improve the user profile and therefore the quality of the produced recommendations is a cumbersome task.

Especially for users new to a recommendation system, the "teaching phase" for an individual user profile is time-consuming (also known as the "cold start problem" [WikCSP11]).

With the pure content-based approach, a user's own ratings are the only factor influencing future performance, and there seems to be no way to reduce the quantity without also reducing performance – expect when making use of a "hybrid recommendation system" which combines content-based and collaborative filtering mechanisms.

3.3.2. Collaborative filtering

In contrast to the content based filtering, the collaborative filtering techniques use the data from a community for the creation of the recommendations. These recommendations can take the item-based and/or the user-based approach into account, see Figure 3.3.2.1. With the item-based approach given is a database of preferences for items by user. A new user is then matched against the database entries to discovers neighbours who have similar "taste" as the new user. Items that the neighbours like are then recommended to the new user who will probably also like these items. On deals with a m x n user/item data in form of a rating matrix. Each entry represents the preferences (rating) of a certain user to a certain item which is in our example a movie. The question mark (?) indicates for which item of the new user prediction is sought. The collaborative filtering algorithm utilize the entire user/item database to produce a prediction.

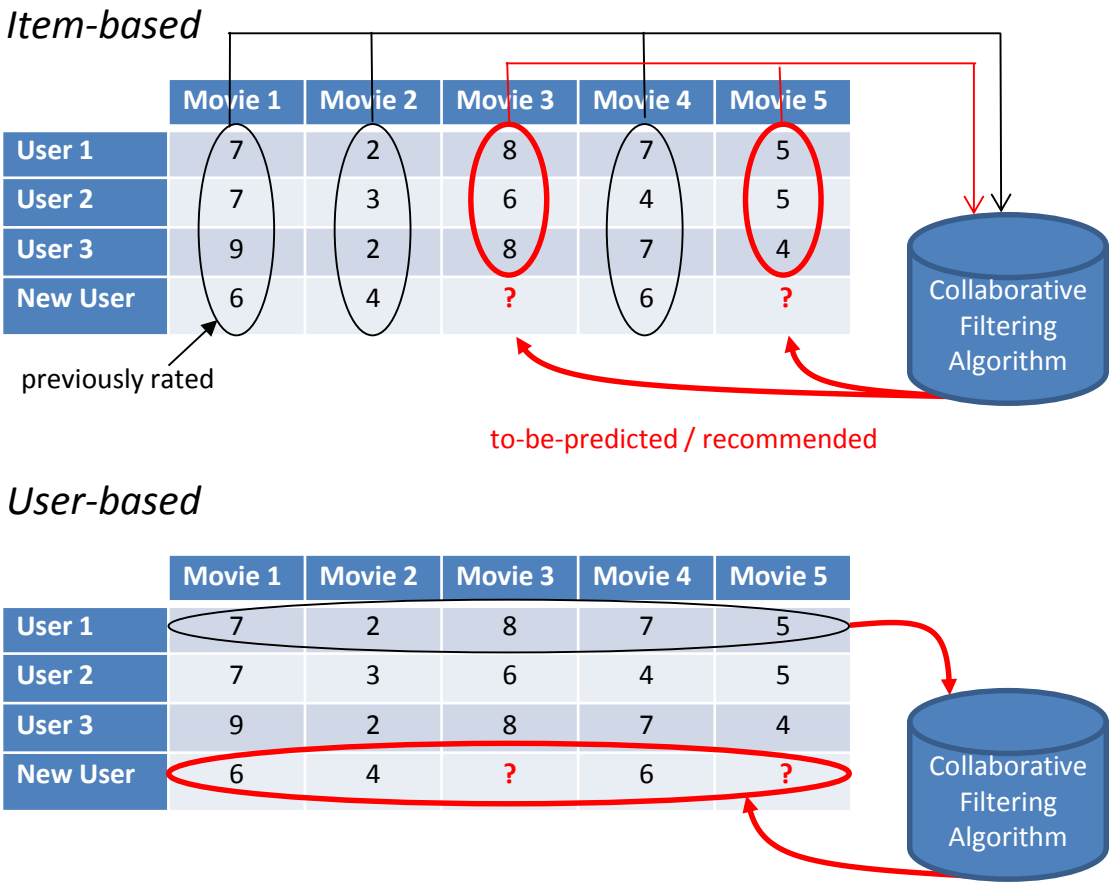


Figure 3.3.2.1: Collaborative filtering: item-based or user-based.

Sawar et al. [SawWWW01] uses several item-based collaborative-filtering algorithms. The authors use the Pearson-r Correlation, the Cosine Similarity, and the Adjusted Cosine Similarity for the calculation of the similarities. The authors use a dataset from MovieLens. They use this dataset for the creation of reduced user-item matrices. The publication evaluates its results by the usage of the Mean Absolute Error (MAE). The tests showed that the Adjusted Cosine Similarity creates lowest error. The authors only take the item-based approach into account, which calculates similarities between users.

Papegelis and Plexousakis [PapAai05] use the Pearson-r Correlation by considering the user-based and the item-based approach. The authors take ratings from explicit settings and implicitly created user profiles into account. The authors get the best results by the usage of the item-based approach.

Herlocker et al. [HerJIR02] uses the k-nearest neighbour approach. This approach finds the users/items, which are quite similar to the active user/item. They split the neighbourhood-based prediction approach into three components. The first component computes the similarities by using the Pearson-r Correlation, the Spearman Rank Correlation, and the Mean Squared Difference. The second component selects the neighbours. The third component rates the combination. The result of this paper is, that the Pearson-r Correlation and the Spearman Rank Correlation produces a lower error than the Mean Squared Difference. The authors recommend the Pearson-r Correlation for a recommendation system.

Martinez et al. [MarTCE09] propose a hybrid recommendation system that takes content-based recommendations and collaborative filtering algorithms into account. The authors also reduce the dimensions of the user-item matrix by using the Singular Value Decomposition (SVD). The calculation of the similarities is realized by using the Cosine Similarity.

The paper from Zhang et al. [ZhaCRS11] presents a regression procedure. It uses the matrix factorization. The authors present results by using the dataset from MovieLens and Yahoo Buzz. The proposed system is able to reduce the RMSE to 0.8777.

However, Netflix started a competition in 2006, which ends in 2009. The main goal of this competition was the reducing of the RMSE. The winner of the competition was able to reduce the RMSE to 0.8567.

Töscher et al. [TösKDD08] presents the results of the grand prize winners from 2009. The paper presents different researched approaches, such as the k-nearest neighbor approach in combination with correlation based algorithms, like the Pearson-r Correlation, the Spearman Rank Correlation, the Set Correlation, the MSE Correlation and the Ratio Correlation. In addition the authors use the several kinds of Singular Value Decomposition and matrix factorizations. The lowest RMSE was achieved by using the matrix factorization.

Wen [WenNET08] uses the Netflix dataset and introduces the reader into a k-nearest neighbor approach, which uses the Adjusted Cosine Similarity for the calculation. In addition the author uses an Item-Based EM algorithm and the Sparse SVD.

Besides these mentioned approaches, the author performs some postprocessing tricks, which shall reduce the RMSE as well. The author was able to reduce the RMSE to 0.8930 by using a blending of item-based EM and the Sparse SVD.

Basically the existing collaborative-filtering systems use algorithms for the calculation of similarities between users or items. Well-known algorithms, such as the Pearson-r Correlation, the Spearman Rank Correlation, the Cosine Similarity, or the Adjusted Cosine Similarity, calculate similarities between users or item. If the similarity between users is calculated, the user-based approach is used. The item-based approach calculates similarities between items. In order to improve the performance, the SVD is used as well. The SVD is able to calculate the similarities between users or items in parallel. This approach reduces a m-dimensional matrix into n-dimensions too.

Normally these kinds of recommendation systems consider large dataset. The mentioned existing approaches use datasets from MovieLens or Netflix, which contains a large number of entries. Since we are focusing on group recommendations within a smaller environment, we have to evaluate the usefulness of the existing approaches by using small number of entries.

3.4. User profiles

Recommender systems are completely dependent on the information they gather and store in order to calculate personal recommendations. This information is primarily about the items they recommend and the users that will receive the recommendations. The data that is used by a recommender system can be really diverse and primarily depends on the recommendation technique that is used by the system. For instance, in collaborative filtering, a user profile consists of the ratings provided by the user for some of the items in the catalogue [MclSIG04, ZieWWW05]. Other techniques require much more information of the users or items. In a demographic recommender system, socio-demographic characteristics such as gender, age, profession and level of education are used while others use extensive metadata, ontological descriptions [RicRSH10a] or social relations between users [BonBTT06] for example.

3.4.1. Implicit and explicit feedback

When creating a user profile, the information can either be collected explicitly or implicitly. When profile information is gathered implicitly, it is often derived from the user's actions/activities. Observing the items that a user viewed (and for how long) in an on-line store is an example of implicit feedback.

3.4.2. User profiles in TNO's Personal Recommendation Engine Framework

The Personal Recommendation Engine Framework (PREF) developed at TNO has a quite generic data model. The core of the data model consists of the three most important entities that arise in every recommender system: users, items and the ratings provided by users with respect to items. Users and items are very basic and consist primarily of a unique identifier and a list of user/item characteristics. An ItemRating relates a user to an item and associates a utility score with it. This score is in the domain [-1, 1], where -1 indicates that the user really hates the item and 1 that he really likes it. The user profile may be cold started by explicitly asking the user to rate some items.

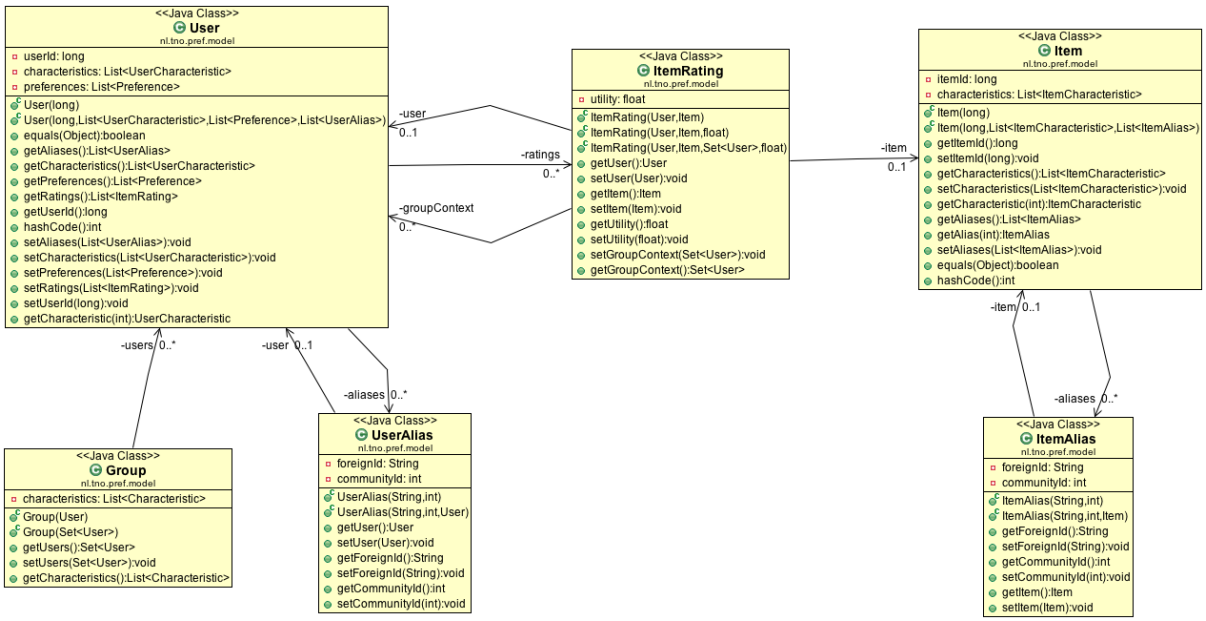


Figure 3.4.1: The core of the PREF data model

The data model supports the use of aliases. These aliases are used to map the internally used users and items onto an ID that is used in the outside world by a certain community. This way two different communities can use different identifiers to reference to the same user or item.

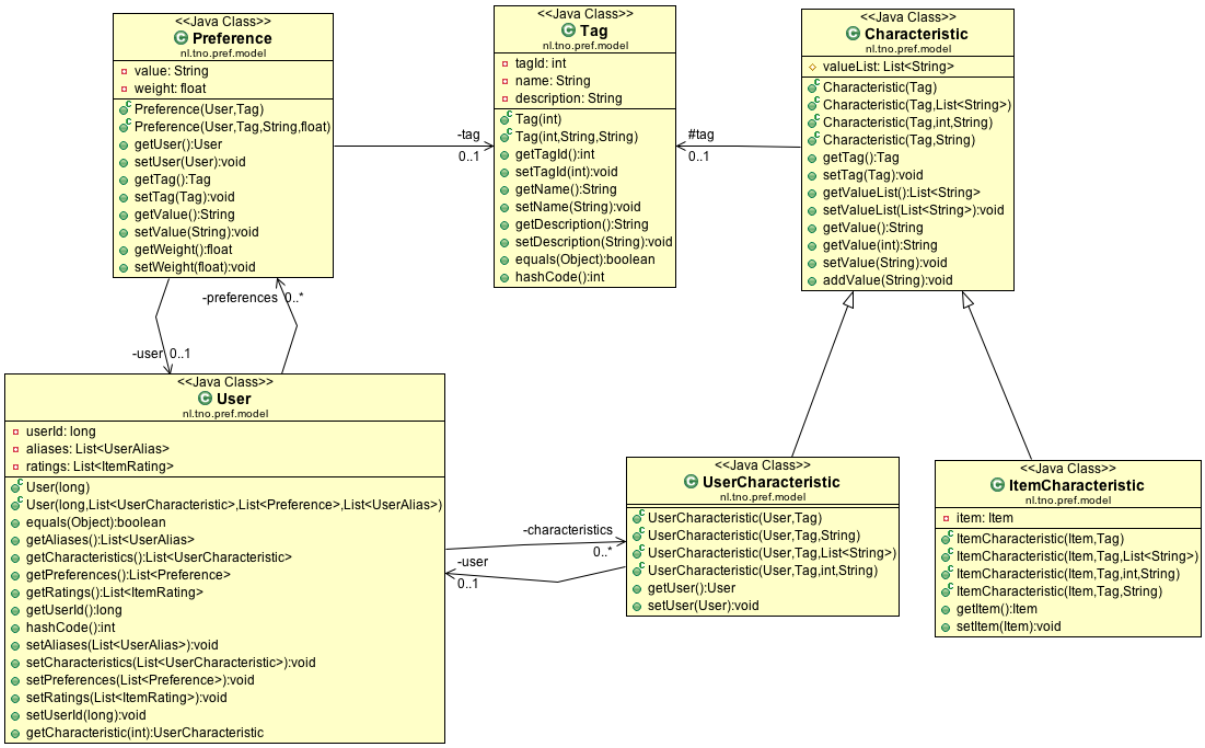


Figure 3.4.2: Characteristics and preferences in the PREF data model

The PREF has a very flexible and generic way to store information about users and items. It contains a Characteristic class that represents a single piece of information about a user or item. This piece of information can be a single value (i.e. “30”) or a list of values (i.e. [“joost.dewit@tno.nl”, “joost@gmail.com”]). This value on itself does not have a clear meaning yet. To indicate the meaning of a Characteristic, a Tag is added. The tag adds the semantics to a Characteristic. When the tag “age” is added to the value “30” as part of a UserCharacteristic of user A for example, it becomes clear that user A is 30 years old. The same principle is used to store user preferences. When a user generally likes action movies for example, it could have a Preference with Tag “genre”, value “action” and weight “0.9”.

3.5. Presentation of the recommendations

Presentation of recommendations is essential for the user to understand and use the recommendation. “Computer says watch this” is not an effective way to have users follow recommendations.

Amazon explains its item-based collaborative-filtering recommendations as “Customers Who Bought This Item Also Bought [the following books]”. YouTube explains its content-based recommendations as “[we recommended you this video clip]because you watched [this other video clip]” [YouTube]. Recommendation may be even more complex to explain in case of multiple users, especially when privacy of users is involved. Especially teenagers are sensitive to keep preferences secret if these are not be in line with their peers. Tintarev et al [TinRSH11] gives an overview of the area of explanations in recommender systems. They define the seven aims for recommendation presentation, see Table 3.5.1

Aim	Definition
Transparency	Explain how the system works
Scrutability	Allow users to tell the system it is wrong
Trust	Increase users’ confidence in the system
Effectiveness	Help users make good decisions
Persuasiveness	Convince users to try or buy
Efficiency	Help users make decisions faster
Satisfaction	Increase the ease of use or enjoyment

Table 3.5.1: Explanatory criteria and their definitions [TinRSH11]

3.6. Evaluation/validation of the recommendation

Previous sections presented the most commonly used recommendation techniques. Each technique has adherents that claim it to be an improvement over some other techniques, given a particular purpose. However, no common notion of recommender system quality exists and therefore comparing different systems is difficult. Many researchers recognised this problem and several contributions that try to solve it have been presented. This chapter will present the evaluation metrics that are used to evaluate recommender systems.

3.6.1. Accuracy

From the countless number of dimensions you could measure, accuracy is by far the most adopted [HerACM04], [AdoIEE05], [SchSIG02], [SchECR05], [MclSIG04], [ZieACM05], [ChoIEE07], [Bla07], [ZhaSIG07], [CleSIG07]. An accuracy metric empirically measures to what extent a ranking, as predicted by a recommender system, differs from the actual ranking provided by the user. Accuracy metrics can measure how well a user's ratings can be reproduced by the recommender system, but also how well a user's ranked list is predicted. To illustrate this subtle difference an example is shown in Table 3.6.1. On the left the recommender system (RS column) tried to reproduce the true user ranking (User column), while on the right it tried to reproduce the user's rating for each of the items A to F.

	Ranking		Rating	
Item	User	RS	User	RS
A	1	1	5	5
B	2	5	4	3
D	3	4	4	4
G	4	6	4	2
E	5	3	3	5
C	6	2	2	5
F	7	7	2	2

Table 3.6.1: Ranking and rating example

Herlocker et al. [HerACM04] identified three classes of accuracy metrics:

- Predictive accuracy metrics. Predictive accuracy metrics measure to what extent a recommender system can predict ratings of users. These metrics are especially useful for systems that display the predicted ratings to their users. Since rated items have an order, predictive accuracy metrics can also be used to measure a system's ability to rank items. Metrics that are commonly used to measure predictive accuracy are (variations of) Mean Absolute Error (MAE), Mean Square Error (MSE) and Mean User Gain (MUG).
- Classification accuracy metrics. These metrics measure to what extent a RS is able to correctly classify items as interesting or not. The defect's magnitude with respect to the user's true rating and the rating predicted by the RS is ignored in these metrics. The two dominant measures for evaluation are precision and recall. Other metrics include the F measure, mean average precision (MAP) and the receiver operator characteristic (ROC).
- Rank accuracy metrics. These metrics evaluate a recommender system's ability to recommend an ordered list of items to a user, assuming that the order of the items on the list is important. Rank accuracy metrics penalise the recommender for not producing the right order of items. To accomplish this, correlation metrics such as Spearman's rank correlation and Kendall's Tau are typically used. Other common metrics are Half-life Utility Metric, normalised distance-based performance measure (NDPM) and relative edit distance (RED).

3.6.2. Coverage

Besides accuracy there are a number of other dimensions that can be measured [ZieACM05], [HerACM04]. One of the dimensions that is mentioned in the literature is coverage. Coverage measures the percentage of items for which the recommender system can make predictions or recommendations. A recommender system cannot always generate a prediction since there might be insufficient data. When an item has never been rated before an item-based prediction technique cannot predict a particular user's rating for that item for example.

There are two types of coverage identified by Herlocker et al. [HerACM04], called prediction coverage and catalogue coverage. Prediction coverage is a measure for the percentage of items that the recommender system can form predictions for. Catalog coverage on the other side is a measure for the percentage of items that is ever recommended to any user. A higher coverage means that the RS is capable to support decision making in more situations. Coverage can be measured by taking a random sample of user/item pairs from the data set and then ask the RS to provide recommendations for all of them. Both predictive and catalog coverage can then be estimated.

Like precision and recall cannot be taken on their own, coverage cannot be considered independently from accuracy. A recommender system can possibly achieve high coverage by making spurious predictions, but this has its repercussion on accuracy.

According to Herlocker et al. [HerACM04] it might be more useful to compute coverage only over the items that a user is actually interested in. When a RS cannot form a recommendation for an item that the user is not interested in anyway then it is considered not much of a problem.

3.6.3. Confidence

Recommendations made by a RS can be characterised along two dimensions: strength and confidence. Strength indicates how much the user will like the recommended item. Confidence indicates how sure the RS is about the accuracy of the recommendation. Both dimensions are often conflated, resulting in the assumption that the higher an item's predicted rating, the better a user will like it. Extremely high predictions, however, are often based on a small amount of user ratings (i.e. high strength, low confidence). As the number of ratings grows the prediction will usually regress to the mean (i.e. low strength, high confidence).

Recommender systems have different approaches to deal with confidence. Most systems do not display items with a confidence level that is below a certain threshold. Another approach is to display the confidence of a recommendation to the user.

Herlocker et al. investigated the latter approach and found out that good confidence displays achieved significant improvement in decision making. However, many recommender systems do not include some notion of confidence.

3.6.4. Diversity

Diversity of the items in a recommendation list is another aspect that might be important to users and therefore must be measured [McN06]. A user who bought a book of ”The Lord of the Rings” on Amazon for example, might receive recommendations for all other books of Tolkien. Although the user probably likes all of these books (which results in a high accuracy) he might not be completely satisfied.

According to McNee [McN06] difference in list diversity will have a large effect on the usefulness of recommendation lists and therefore McNee, Ziegler and others [McN06], [ZieACM05] presented the intra-list similarity metric.

3.6.5. Learning rate

Many recommender systems incorporate learning algorithms that gradually become better in recommending items. Collaborative filtering algorithms are likely to perform better when more rating information is available for example. Content based filtering algorithms that learn what attributes are important to a specific user also need feedback in order to learn each attribute’s weight. Learning rate is a measure for how quickly an algorithm can produce ”good” recommendations. The metric for quality is usually accuracy. Some recommendation algorithms only need a small number of ratings to produce sufficiently good recommendations while others might need extensive training before they reach the same level of quality. Learning rates can be described by some asymptotic (and thus non-linear) function since the quality of the recommendations cannot increase indefinitely. An example of a graph in which the MAE is plotted against the number of ratings provided by the users is shown in Figure 3.6.5.1.

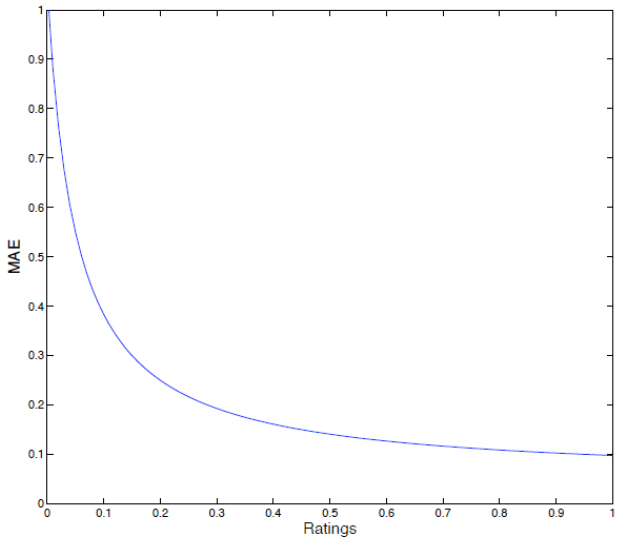


Figure 3.6.5.1: Example of learning rate graph in which the MAE is plotted against the number of ratings

Herlocker et al. identified three different types of learning rates: overall learning rate, per item learning rate and per user learning rate.

The overall learning rate is defined as cumulative quality over the number of ratings provided by all users for all items, while per item and per user learning rate only take into account the ratings provided for a single item or by a specific user respectively.

Learning rate is a strong indicator for a recommender system's capability to cope with the cold start problem. But despite the fact that the cold start problem is widely recognised by researchers the evaluation of a recommender system's learning rate has not been extensively covered in the literature and no specific metrics exist.

3.6.6. Novelty and serendipity

A recommender system can produce highly accurate recommendations, have a reasonable good coverage and still does not satisfy a user. Suppose for example that an EPG with recommendation functionality recommends the news to every user. Although this probably results in high accuracy it has no added value to the user, since he already knew the news and would probably have viewed in anyway [TerHCI01]. However, a recommendation that is obvious to one user, might be new for another.

An expert on recommender system evaluation probably finds a recommendation of Herlocker's "Evaluating collaborative filtering recommender systems" obvious, while a user who is new to this field is dying to find such a paper.

Novelty and serendipity are two (closely related) dimensions for non-obviousness, as identified by Herlocker et al. [HerACM04]. Serendipity is the experience of discovering an unexpected and fortuitous item. This definition contains a notion of unexpectedness, which is the novelty dimension. Novelty and serendipity metrics thus measure the non-obviousness of recommendations and penalize "cherry-picking". The literature does not describe a metric that can be used to measure novelty, nor serendipity.

3.6.7. User satisfaction

User satisfaction is a somewhat vague and soft aspect and therefore it is difficult to measure. In the context of this research user satisfaction is defined as the extent in which a user is supported in coping with the information overload problem.

The dimensions described in the above sections will probably support and/or inhibit user satisfaction to some extent. In order to be able to determine the effect of these dimensions on user satisfaction, user satisfaction itself must also be measured.

Herlocker et al. presented a number of dimensions along which user satisfaction evaluation methods can be classified [HerACM04]. These dimensions are listed next.

- Explicit vs. implicit. The evaluation method can explicitly ask how satisfied the user is, or it can be observed in some way. Implicit evaluation methods require assumptions in order to translate the observations into a measure for user satisfaction, like an increase in sales implies greater user satisfaction for example.
- Laboratory studies vs. field studies. Lab studies are studies in a controlled environment while field studies take place in the user's own real context.
- Outcome vs. process. The study can only focus on the outcome but it can also focus on the process that originates to it.

- Short term vs. long term. Short term user evaluations might miss results that only become apparent after a certain amount of time. The evolvability of a user's taste is an example of a phenomenon that can only be investigated in a long term user evaluation.

Studies that investigated user satisfaction with respect to recommender systems are scarce and studies that focus on user satisfaction with respect to recommendations are even rarer. Many studies that focus on user satisfaction are primarily concerned with human computer interaction and do not focus on the recommendation list's properties.

3.7. Scalability

Scalability of recommender systems is very important. A solution that works fine when tested off-line on relatively small data sets may become inefficient or even totally inapplicable on very large datasets. [RicRSH11a]. Note that the scalability has two aspects, viz. the scalability of the content metadata and user profiles databases, and the scalability of the recommender algorithms.

Database scalability is a well-known problem with well-known solutions, including caching, replication, partitioning, Bigtable/HBase/Hypertable,

Dynamo/Dynomite/Voldemort, Cassandra [EllOSC09]. The goal is to scale the cost of the database at most proportionally with the numbers of queries per seconds and with the number of records stored. Scalability of recommender algorithms is a relatively new area of research about which discussion is missing in the current literature since it has been mostly

investigated by practitioners [RicRSH11a]. Some research has been done in this area, see [SarCIT02], [SarIST05], [GeoCDM05], [HanESA04], [PapNCS05], [XieDES04]. Collaborative filtering algorithms are known to be relatively well scalable, as the neighbourhood ("users like me", "items like this") can remain small to achieve an acceptable accuracy and it can be updated sporadically in an off-line process.

3.8. Metadata issues

As already shortly mentioned in chapter 3.3, content-based recommendation systems make use of content descriptions (metadata) and profiles of users. Potential matches for content items of interest for a given user are being identified based on a comparison between their content and a user profile. Data structures for both of these are created using features extracted from the text of the documents. If the user liked a piece of content, weights for the elements within the metadata describing this piece can be added to the weights for the corresponding elements in the user profile, such as content genres, keywords, actor names etc. This process is known as relevance feedback.

A pure content-based system has several shortcomings, where in this chapter, focus is given to the fact that content-based recommenders heavily rely on the quality and quantity of metadata describing a piece of content.

In most broadcast domains, transmission of metadata automatically means greater bandwidth consumption, which results in additional costs. As known from DVB (-S / -C / -T) networks, bandwidth is expensive, and is aimed at to be kept as low as possible – by reduction of video encoding quality, as well as by reduction of additional bandwidth to be spent for binary coded metadata.

Therefore, the amount and quality of metadata being present for the description of content items is not “rich” enough to any useful feature extraction methods with current technology. Rich in this context means a lack of detailed, fine-granular metadata w.r.t. elements that are made use of by a content-based recommendation system, such as:

- Content genres (“comedy”, “action”, “sports” etc.)
- Actors and their roles (“ ’Johnny Depp’ playing ‘Captain Jack Sparrow’ “ etc.)
- Additional credits (“ ‘Gore Verbinski’ as ‘Director’ “ etc.)
- Depicted and / or production locations (“ ‘Bequia’, ‘St. Vincent’, ‘Grenadines’ “, “Walt Disney Pictures” etc.)
- Technical information etc.

Content-based recommenders do not only benefit from such kind of information, they rely on it. Additionally, they in most cases depend on clearly interpretable fields from which they can pick such information, the well-structured form of the metadata being given (e.g. within the TV-Anytime [TVA] or MPEG7 [MP7] metadata formats). For a purely content-based recommender, it is essential to have the highest possible amount of structured high-quality metadata, so they can trust of combinations of elements, in order to reflect those within the user profile.

One example a recommender could show as a so-called "reasoning" for a given recommendation is: "User Andrew likes 'Arnold Schwarzenegger' performing in movies of the genre 'Action', but he does not like 'Arnold Schwarzenegger' performing in movies of the genre 'Comedy' - therefore, 'Terminator 2' and 'The Expendables' will be highly recommended to him, but 'Jingle all the way' and 'Twins' won't be".

In order to be able to reflect such a conclusion within the user profile and when it comes to the prediction of the potential match with a new metadata item, the necessary metadata elements:

- Need to be present (quantity & quality)
- Need to be accessible (structured data, not just flow-text)
- Need to be correct (quality, multi-lingual)

Some projects [XBM] try to solve the lack of rich metadata by the implementation of so-called "scrapers" [XBS], which provide access to open on-line metadata sources such as IMDB [IMD], and automatically retrieve metadata about movies, TV series etc. They try to perform a simple title-match in order to find matching entries of metadata at the open metadata sources, which in most cases works. However, there are currently no implementations that combine the rich metadata retrieval with recommendation systems.

3.9. Privacy, anonimized recommendations

An important issue with recommendation systems is the privacy aspect [RamICC01], [SchDMK01]. Namely, in order to compute recommendations, personal information of users has to be processed to be able to fine-tune to the personal preferences and wishes of users. The more sensitive the personal information, the larger the privacy problem.

There are several approaches to tackle the privacy problem within recommendation systems, the preferred one depending on the actual setting and requirements. We list a couple of ideas and techniques that are useful here. They can be divided in two main categories:

1. Data hiding:The recommender server is not allowed to learn personal data.
2. Identity hiding:The recommender server is not allowed to learn user identities.

A general disadvantage is that when actual products have to be delivered or paid, some information about the person is likely to leak.

Finally, this section looks into privacy-preserving group recommendation technologies.

3.9.1. Data hiding

One approach is to use cryptographic techniques like encryption to encrypt user data and have the central server compute the recommendations in the encrypted domain [ErkSSP11], [ErkCAS11], [CanSSP02]. The advantage is that personal data are guaranteed to remain unknown to the central server.

The disadvantage is that extra computational and communication resources are needed since computing in the encrypted domain is more intensive than in the unencrypted one.

Another way of hiding user data is to use perturbation. Random values are added to user data such that when the data is aggregated, the randomized parts cancel out [PolSAC05]. This type of privacy introduces a trade-off between privacy and accuracy. Furthermore, a certain amount of data leakage is unavoidable [BerCRS].

A more esoteric and less common approach is based on distributed aggregation of off-line profiles [ShoCRS09]. Users are supposed to have two types of profiles, namely off-line and online. While users keep their off-line profile locally, the server stores the online one. These two profiles are regularly synchronized. The proposed method assumes that the users communicate over other media too such as mobile phone or e-mail. As the previous approach, it suffers from the trade-off between privacy and accuracy.

3.9.2. Identity hiding

Instead of hiding the data one could also hide the identity of the person that supplies the data. Different anonymization and pseudonym based techniques are available. One disadvantage is that the recommender server could still obtain the identity by analysing the traffic of data, or combining it with other sources of personal data.

The Open-ID platform [OpenID] enables a user to communicate with a recommender server through an identity server. Only the identity server will know the identity of the user and the server will be ignorant. Similar single sign-on solutions are possible with Shibboleth [Shib] or Windows CardSpace [CardSpace]. The advantage of such systems is that it becomes very difficult for a service provider to combine personal information over different domains because users will have different pseudo-IDs for each domain, which mitigates these kinds of privacy leaks.

General techniques for anonymous browsing like TOR [TOR] could also be used for anonymous recommendations. Its disadvantage is that the server is not able to combine previous visits to enhance the quality of the recommendation. A similar solution is provided through P3P [P3P] which attempts to automate the enforcement of privacy by negotiating policies that describe the privacy requirements in detail.

Anonymous credential mechanisms like Idemix [CamCCS02] and U-Prove [BraBIP00] can be used to anonymously prove personal statements towards a server like “I’m over 21 years”, but these seem less applicable to recommender systems because information is required that has to be processed by the server.

3.9.3. Privacy and group recommendations

In this section we describe a couple of ideas [VeuPAT11b] for recommending content to a group of users and additionally show how privacy can be preserved [VeuPAT11a]. We use collaborative filtering for computing the recommendations as described in section 3.3. Items are rated by users (and groups), a rating being denoted by R(i,m) for user i and item m. A subset of densely rated items is used to determine correlations between two users (or groups), the remaining items are used for producing recommendations.

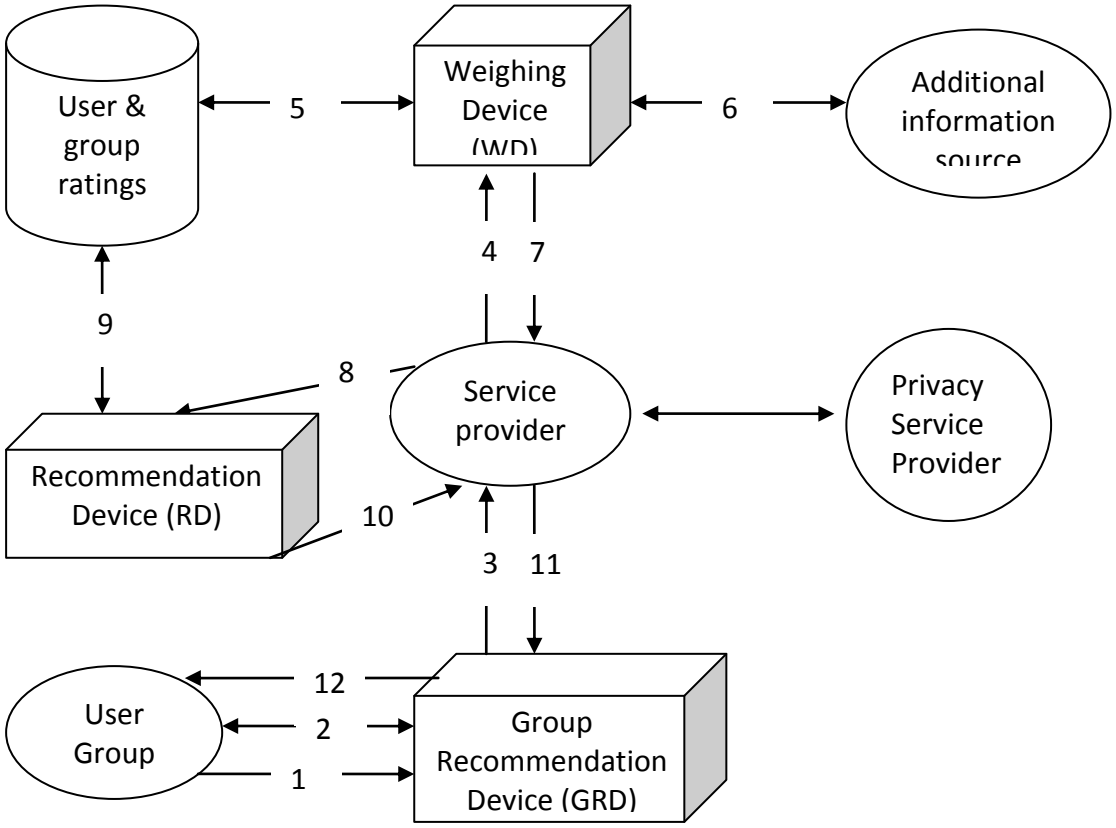


Figure 3.9.3.1: Group recommendation system

- We assume each user has ratings stored in a central database, but this not necessarily holds for group ratings. To produce recommendations for groups, Figure 3.9.3.1 describes 12 consecutive steps:
- A group requests a recommendation for certain content by notifying a certain device called Group Recommendation Device (GRD).

- The group and the GRD exchange information about the group and its members. Which information should be exchanged will depend on the actual system but could contain user identities, social information about the users or the group like age, gender, friends or colleagues, etc., anything that could be important for their recommendation.
- The GDR uploads the request and the additional information to the Service Provider. The role of the service provider is to generate the recommendation.
- The service provider redirects the information to a device called Weighing Device (WD). The role of the WD is to produce initial group ratings by carefully weighing the user ratings.
- The WD checks whether ratings of the requesting group are available in the database of ratings.
- If no initial group ratings are found, they are generated by WD from the user ratings by giving each group member a suitable weight. Extra information from external databases could be used here to construct suitable weights. The extra information entered at step 2 will play a role here.
- The WD sends the group ratings to the service provider.
- The service provider sends the group ratings to the Recommendation Device (RD). The role of the RD is to compute recommendations through collaborative filtering techniques.
- The RD finds similar users/groups by maximizing the similarity measure and computes the recommendation as an average of item ratings of the most similar users/groups.
- The recommendation for the group is sent to the service provider.
- The service provider sends the recommendation to the GDR.
- The GDR presents the recommendation to the group of users.

The basis of the system described above is similar to a collaborative filtering type user recommendation system, where a group is considered as a (new) user. Alternatively, one could extend the current similarity measures between users to similarity values between groups. The correlation C between groups {A, B, C} and {D, E, F} could e.g. be computed as:

$$C(\{A,B,C\}, \{D,E,F\}) = \max\{C(ABC, DEF), C(ABC, DFE), C(ABC, EDF), C(ABC, EFD), C(ABC, FDE), C(ABC, FED)\},$$

where for example $C(ABC, DEF) = C(A, D) + C(B, E) + C(C, F)$.

So the correlation between two groups is actually defined as an optimal matching of the individual users in the group. This is quite different from considering the group as an average of the individual users.

Another way to extend user similarity values is to first define an ordering of the users in a group and then compute the group similarity as the sum of the similarities between the pairs of ordered users of the two groups.

To have the group recommendations computed in a privacy preserving way, similar techniques could be used previous sub-sections. To generalize techniques based on homomorphic encryption, we would need a Privacy Service Provider (PSP) as depicted in Figure 3.9.3.1. The general idea is that ratings are encrypted using a homomorphic encryption system, the PSP holding the private key that enables decryption. Homomorphic encryption has the nice property that easy linear operations can be performed in the encrypted domain. So many operations can be processed by the service provider himself, but e.g. for comparing different (encrypted) similarity measures, the service provider would have to cooperate with the PSP in a cryptographic protocol.

Such protocols are described in the field of secure two-party computation, where the inputs of both parties (the decryption key for the PSP and the encrypted user ratings for the service provider) are guaranteed to remain privately known to one party.

Therefore, in our privacy preserving variant of a group recommendation system, the same operations will eventually be processed by the service provider as in the non-privacy-preserving system, occasionally assisted by a PSP. The main difference is that all operations will be performed with encrypted data, so both service provider and PSP will never learn any user ratings or other personal data. The advantage is obviously that the system preserves the privacy of the users, its disadvantage is that much more communication and computation is required by the service provider and the PSP.

3.10. Group recommendations

Recommender systems are typically used to recommend items to individual users. In many domains this limitation is quite logical. There is no need for Amazon for example to recommend its books to a group of customers since they won't read the book collectively. Books, as well as many other content items are usually consumed by one individual person. However, there are plenty of items that people like to consume as a group. Examples are music, board games, holidays, movies, TV programs and so on [ConSCW01]. The challenge recommender system researchers and designers typically faced is how to decide what items from a large collection of candidates would be most satisfying to an individual user. Since the appearance of the first papers on recommender systems in the mid-1990s a lot of progress has been made on this. The multi-user aspect in recommender systems, however, has not been investigated as thoroughly although it is recognized to be an important aspect by many researchers [ConSCW01], [QuiTAI10], [JamRTG07]. Most work on group recommendations is dated after 2006.

The following subsections describe group formation and properties; aggregation strategies to combine individual recommendation lists into a single group recommendation list; uncertainty in group recommendations; and two existing operational group recommendation systems.

3.10.1. Group formation and properties

O'Connor et. al. [ConSCW01] identify that aggregation functions that appear to work for recommendations to small groups might not work as well for large groups. Therefore the groups size is an aspect that has to be taken into account when recommending items to a group of users.

3.10.2. Aggregation strategies

In one way or another, recommending items to a group of people involves merging the individual preferences of each of the group members. The way these individual preferences are brought together in order to come up with the recommendations is called an aggregation strategy [CamUMU09]. According to the literature, either the user preferences or the recommendations for the individual users can be aggregated [ConSCW01], [CamUMU09]. Another approach is to construct a single preference model for a group of people [QuiTAI10]. When the user preferences are merged, a kind of pseudo user's profile is created. The pseudo-user's profile can be created manually by the different users in the group (i.e. they decide together that they like action movies for example) or it can be created by merging each individual profile. This profile is then used as if it were a regular user's profile. The recommendations that are generated based on this profile are presented to the group. In the second approach, the recommender system generates recommendations for each of the individual users in the group, based on their profile. The resulting recommendations are then merged into a single list and presented to the group.

In either case, the actual merging of user specific preferences into a single group preference can be done in many ways. A very straight forward strategy is to take the average of the individual user preferences. This approach is illustrated in Table 3.10.2.1.

	A	B	C	D	E	F	G	H	I	J
Peter	10	4	3	6	10	9	6	8	10	8
Jane	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6

Table 3.10.2.1: Example preferences for three users

Judith Masthoff presented eleven aggregation strategies that are inspired by Social Choice Theory [MasRSH11], [MasUMU04]. These strategies are explained in Table 3.10.2.2.

Strategy	Description	Example
Plurality voting	Uses 'first-past-the-post' voting repetitively, the item with the most votes is chosen	A is chosen first, as it has the highest rating for the majority of the group, followed by E (which has the highest rating when A is excluded)
Average	Averages the individual ratings per item	The group rating of B is (4+9+5)/3 = 6
Multiplicative	Multiplies the individual ratings per item	The group rating of B is 4*9*5 = 180
Borda count	Counts points from item's rankings in each individuals' (predicted) preference list. The bottom item gets 0 points, the next 1, etc.	A's group rating is 17, namely 0 (last rank for Jane) + 9 (first for Mary) + 8 (shared top 3 for Peter)
Copeland rule	Counts how often an item beats other items (using majority voting) minus the times it looses	F's group rating is 5, as F beats 7 items (B,C,D,G,H,I,J) and loses from 2 (A,E)
Approval voting	Counts the individuals with ratings for the item above a certain threshold (e.g. 6)	B's group rating is 1 and F's is 3
Least misery	Takes the minimum of the individual ratings per item	B's group rating is 4 since it's the smallest of {4,9,5}
Most pleasure	Takes the maximum of the individual ratings per item	B's group rating is 9 since it's the largest of {4,9,5}
Average without misery	Averages individual ratings, after excluding items with individual ratings below a certain threshold (e.g. 4)	J's group rating is (8+8+6)/3 = 7.3, while A is excluded because Jane hates it
Fairness	Items are ranked as if individuals are choosing them in turn	Item E may be chosen first (highest for Peter), followed by F (highest for Jane) and A (highest for Mary)
Most respected person	Uses the rating of the most respected / authoritarian individual	When Jane is considered to be the most respected person, A's group rating would be 1

Table 3.10.2.2: Existing aggregation strategies

Note that (besides the last strategy) none of the strategies takes the composition of the group into account. Each user in the group is treated equally independent of the user's social status or role in the group.

3.10.3. Uncertainty in group recommendations

The previous examples do not take uncertainty into account. They assume that the inputs of the aggregation strategies are precise and that the strategies compute precise outputs. De Campos et. al. [CamUMU09] states that this assumption does not necessarily hold, especially when user preferences are considered to be determined by automatic means (i.e. implicit feedback). In these cases, a probability distribution over candidate ratings might be used to express user likelihoods. Furthermore, the automatic aggregation of user preferences results in some degree of uncertainty.

De Campos [CamUMU09] proposed a generic model, based on the Bayesian network formalism for modelling the above mentioned uncertainties. They focused on the case where the process of combining the individual preferences is known and then extended this to the situation in which it is unknown. In the latter case, two situations are distinguished:

- Total ignorance. It is completely unknown how the group combines the information.
- Learn from previous ratings. Although it is not known exactly how a group combines the information, the history of group ratings is known.

Experimental results demonstrate that by taking uncertainty into account at the individual level when aggregating, better predictions for the groups can be obtained.

3.10.4. Existing systems for group recommendations

Up till now (2012), only a limited number of group recommendations are in operation. This section describes two of those: PolyLens and MusicFX.

PolyLens

PolyLens [ConSCW01] is an operational recommendation engine that recommends items to groups. It is developed by the GroupLens research lab in the Department of Computer Science and Engineering at the University of Minnesota. The GroupLens lab has an extensive track record with respect to recommendation engine research. They investigate many different aspects related to recommender systems such as evaluation, group recommendation and collaborative filtering algorithms.

PolyLens is built as an extension to the MovieLens recommender system. MovieLens is a free movie recommender with over 80,000 users and their ratings of over 3,500 movies.

Some assumptions were made in designing PolyLens. Groups, for example, are assumed to be persistent. Users explicitly create a group which will be stored for later use. Which groups exist and what users are part of a certain group is not known to other users. Another assumption is that the groups are fairly small (typically two or three users) and that many groups exist. Because groups are small, an aggregation strategy is used where the group's rating is the minimum of the individual members' ratings (the least misery strategy in Table 3.10.2.2). The system does not recommend movies to the group when one of the group members had already rated (and presumably seen) it. The PolyLens recommender uses an algorithm that aggregates the recommendation lists of the group members. According to O'Connor et. al. this has several advantages over the aggregation of user ratings into a single group profile. The main advantage is that these merging strategies are able to present results that can be directly related to the results that would be seen by individual group members. This means that the results are relatively easy to explain (e.g., "the system believes that three of you would like it a lot, but two of you wouldn't like it at all"). However, O'Connor notes that group recommendations based on merging recommendation lists are less likely to identify unexpected, serendipitous items.

MusicFX

MusicFX [MccSCW98] was a kind of group recommender system that adjusts the selection of music playing in a fitness centre based on the musical preferences of the people working out at a given time. Users explicitly rated different musical genres on a 5-point Likert scale by login in to the system. Using a NFC badge system, MusicFX determined who was working out at any given time so it could adjust the radio station according to their preferences. Users were not able to control the system directly. MusicFX was deployed at an office building at Accenture Technology Park in Northbrook, USA from November 1997 through January, 2002.

Unlike PolyLens, MusicFX aggregates the preferences of the people working out to create a single group preference score. These scores are then used in a weighted random selection.

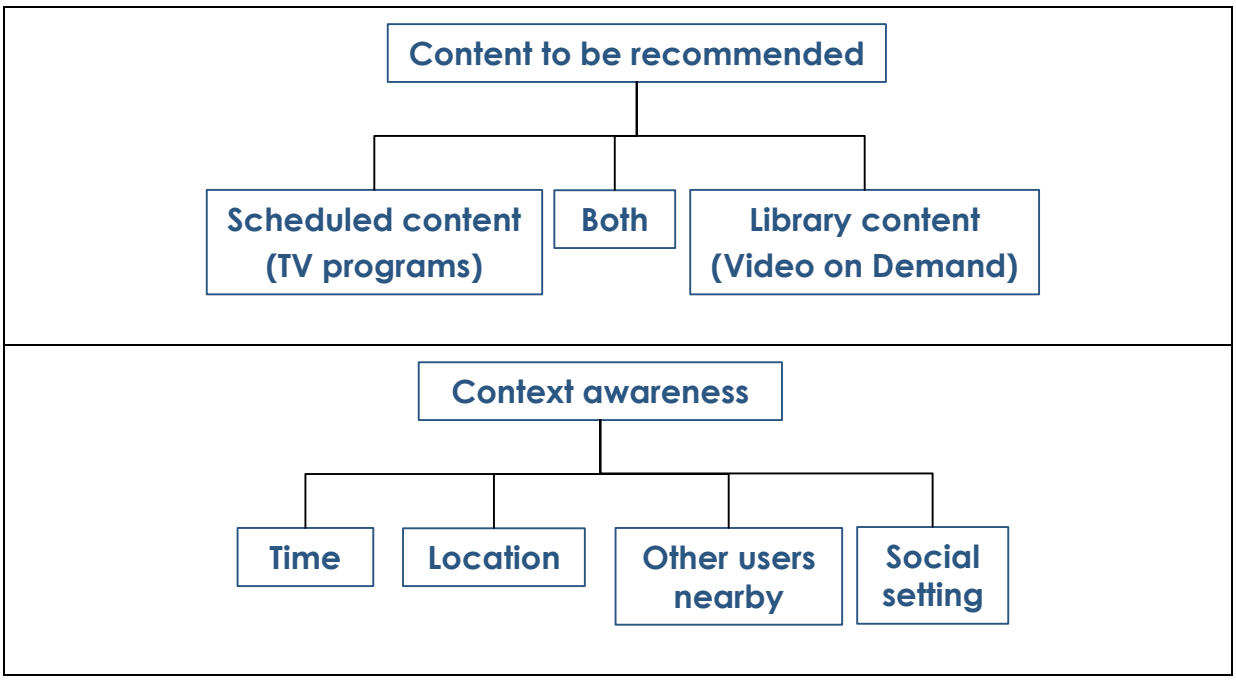
Every time a member leaves or arrives or when the maximum station playtime is exceeded the random selection is repeated. A later (simulated) version of the system used a multi-agent market-based economy to allocate influence within the environment. Inhabitants that liked the music being played payed the inhabitants who did not like that music. Over time, people who were continually subjected to music they did not like could accrue enough capital to pay for music they would like.

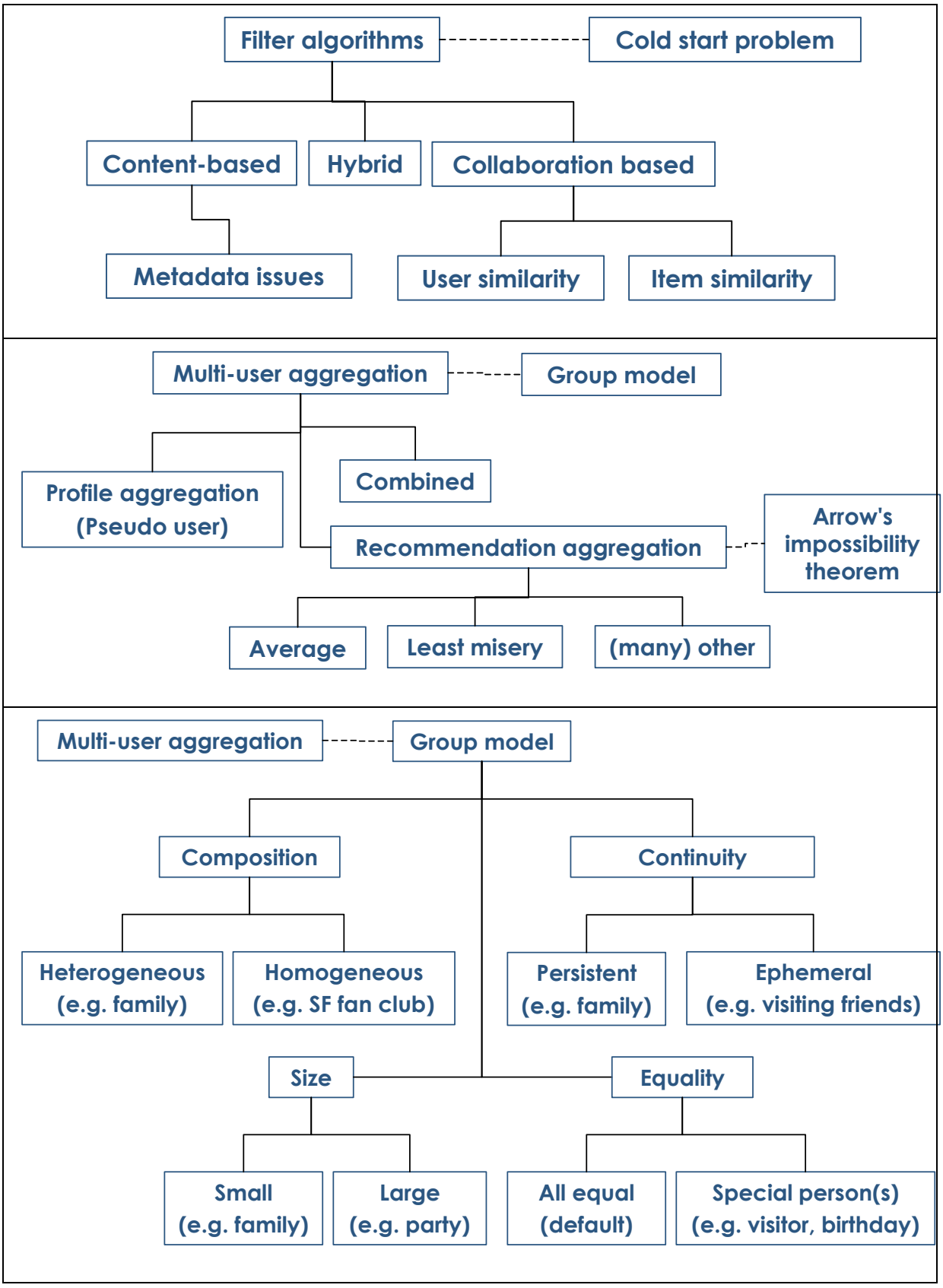
3.11. Conclusions Context-aware and multi-user content recommendation

When studying the literature on recommender systems, the first thing that is noticed is the sheer width and depth of the design space and the nearly infinite number of aspects that could be taken into account.

Whereas the scenarios and associated use cases provide ample features that could be supported, some major limiting decisions will need to be taken on supported features in order to handle the complexity of the task at hand. Also, design choices will need to be taken into account, e.g. balancing the scalability (computational complexity) versus the quality of the algorithms.

Figure 3.11.1 provides an illustration of feature and design choices identified in this section.




Filter algorithms
Cold start problem
Content-based
Hybrid
Collaboration based
Metadata issues
User similarity
Item similarity
Multi-user aggregation
Group model
Profile aggregation
(Pseudo user)
Combined
Recommendation aggregation
Arrow's
impossibility
theorem
Average
Least misery
(many) other
Multi-user aggregation
Group model
Composition
Continuity
Heterogeneous
(e.g. family)
Homogeneous
(e.g. SF fan club)
Persistent
(e.g. family)
Ephemeral
(e.g. visiting friends)
Size
Equality
Small
(e.g. family)
Large
(e.g. party)
All equal
(default)
Special person(s)
(e.g. visitor, birthday)

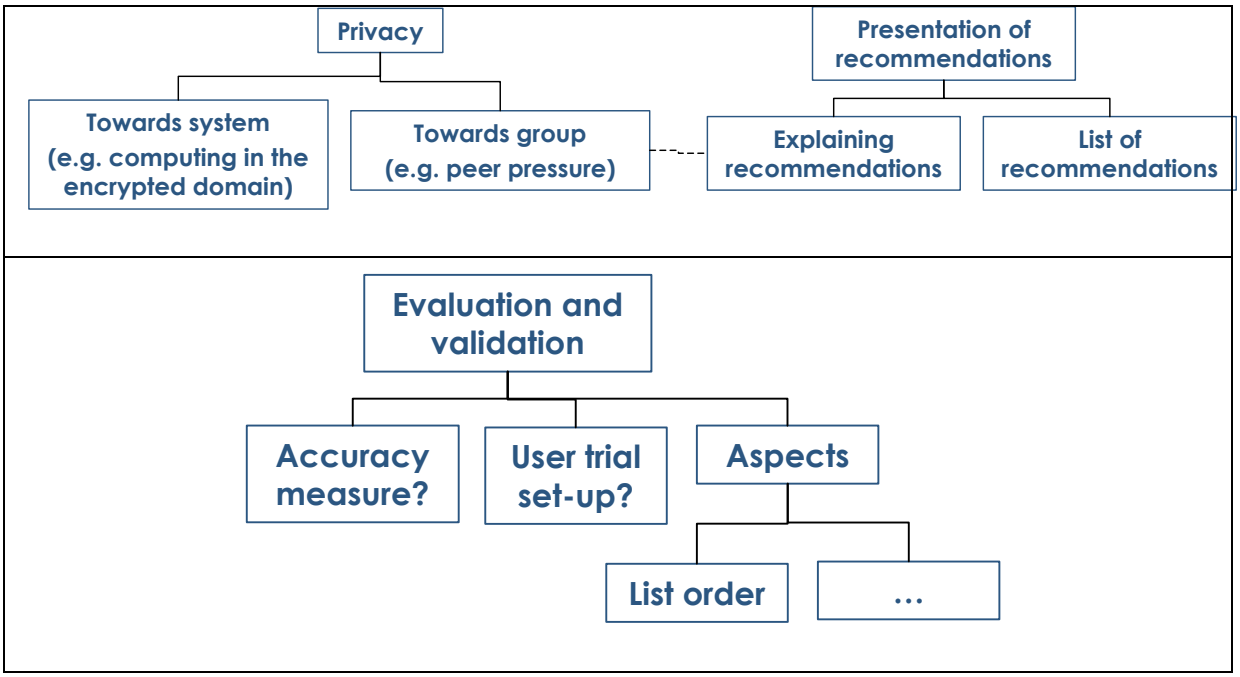


Figure 3.11.1: illustration of the width and depth of the design space.

4. Personalization and context awareness

4.1. Outline

This chapter discusses the state of the art of context-aware personalization. The 'personalization engine' is a module that is planned to be implemented within this work package. The module should re-use some enablers from WP3: Identity and profile management (IdM/PM). This enabler helps to identity the people that are interacting with the HBB-NEXT environment as well as their devices and services.

One of the enablers that WP5 plans to provide is personalization. Personalization in the context of HBB-NEXT means the delivery of a service **in a tailored manner** to a **particular** user or group of users. The tailoring process should be performed by the WP5 enabler based on the knowledge it has about the environment and based on the capabilities of the particular service (i.e. context).

The components for personalization are illustrated in Figure 4.1.1 which shows that anonymous persons become known to the system with the help of identity and profile management. Personalization can then process all this information (e.g. interest, connected networks and their information, preferences, device information) and prepare a filter that helps target the delivery of the service and thus enhance the overall experience

- from a user perspective (the user has a better experience)
- from a service perspective (the service receives processed information based on the context and does not have to care about it individually).

Figure 4.1.1: Components of personalization

Figure 4.1.2 shows that almost any commonly designed service has the potential to be enhanced by personalization and thus appear differently. In other words, one service which offers numbers of information in layout, can present only specific information, relevant for user in specific (personalized) layout.

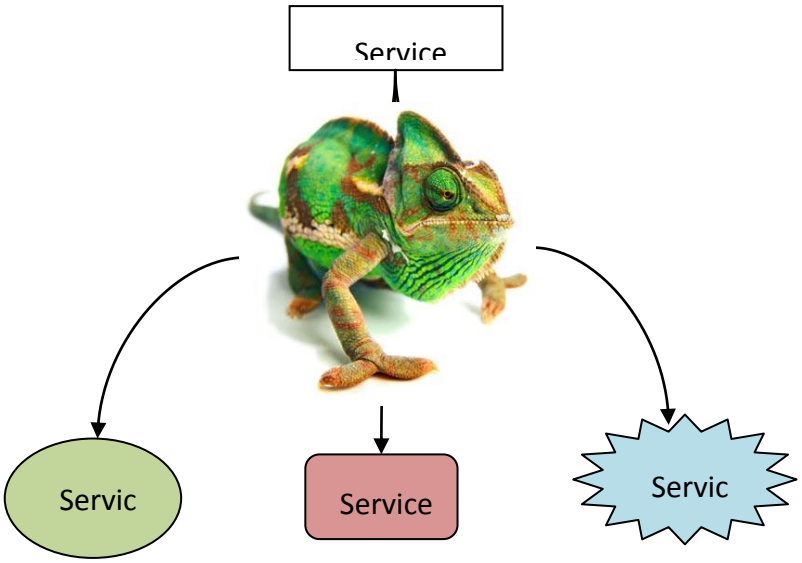


Figure 4.1.2: Principle of personalization

The definition of context as proposed by Dey in [DeyGIT00]: Context is ”any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves”. The most widely used example of context is the location of a user; but as can be seen from the definition any piece of information that describes a situation (…) that is relevant is considered context. For the HBB-NEXT example, this is both the presence situation (i.e. location), as well as the other present users.

Personalization based on context awareness is to be understood as the dynamic adaptation of systems according to newly or changed context information that is being retrieved and correctly interpreted. Elements that could be adapted automatically within a hybrid TV scenario are:

- content filtering, based on profiles of identified users
- adjusting of peripherals, e.g. a webcam adapting its zoom once someone enters the viewport
- playback control, e.g. automatic pausing of a broadcast playback once someone stands up from the couch or the telephone rings
- second-screen functionality support, once some user being present in front of the TV turns on a tablet or smart phone
- available applications and their respective behavior
- the appearance of the front-end (e.g. color, language)
- preferences on how to interact with the system (e.g. speech, gesture, remote).

Besides processing the gathered information about the persons that currently interact with the system, the personalization can be based as well on information how the users are related to each other (information that is provided as well by WP3).

Summarizing, the following description of context-aware personalization elaborates

- the state of the art of personalization
- the state of the art of systems' context-awareness
- the state of the art of context-aware personalization.

The conclusion of this chapter sums up the gaps that are discovered while analyzing the state of the art and define work items for the next deliverable within the work package.

4.2. Problem statement

Personalization will be a main part of the deliverables that are going to be evaluated as task in WP5. Task 5.1 forms the base analyzing the multi-user paradox (see also section 3.2) that may arise w.r.t. personalization & context awareness on the one hand and multiple simultaneous users and multimodal interaction on the other. As stated in [DOW_HBB_NEXT] personalization and context-awareness are related to an individual user's profile, preferences, and context. This task shall now study HBB service consumption as a group experience with shared consumption and control of content and applications. Deliverable D2.1 contains several use cases, which focus on personalization: U.003, U.010, U.034, and U.036.

Nowadays, most efforts in IPTV personalization are put to "targeted advertisements". This personalization is driven by commercial targets, not by the user needs. Currently, television systems do not modify the delivered content at all or perform the modification (personalization) based on a single available profile (set-top-box profile) only. This profile is based only on a set of properties filled by the user through the device.

There is no system, which is continuously analyzing the user and his environment, and gathering information from many sources to set up the television environment according the particular user's needs.

With the support of WP3 (identification of users, storage and availability of user profiles), the interacting users are known and can be targeted based on their personal preference.

Personalization as such is nothing new: A personalized environment already surrounds users. Personalization occurs nowadays in many different ways:

- Websites usually deliver their content personalized. Users can set their preferred languages, themes, privacy settings, security settings, etc. Although the set of options is limited per se, each user will see the website in his preferred way – according to his personal preference.
- Services such as Netflix [NETFLIX] have basic recommendation engines for the users of the system. Users can rate movies and based on their rating, they have a certain set of recommended movies. As this set varies per user, it is delivered personalized.
- Cars have systems to recognize the driver according to the key they use [AUDI]. Based on the previously defined profile, seat and steering wheel are arranged in the way the customer set it, radio stations are set to predefined values, and also other parameters are loaded as the driver previously saved them in his profile. Based on this profile information, the driver uses the car in a personalized way.

All examples show that personalization can be performed as soon as the personalizing system knows information about the user (identification and preferences). The information must be relevant, that is, the system has to be able to deliver the service in a personalized manner based on them. Not all stored information has to be relevant for all services. Personalization can happen only based on some parameters, supposing the personalization engine is aware of which parameters to use for which service.

A sample of extended personalization is the recent change in how Google will process the data it has about users and their (inter)-actions using Google products.

So far, Gmail will personalize advertisements based on the screened mails and the user profile built from this information. However, information from this profile has so far not been used to personalize other Google products. From March, 1st 2012 on, Google will use information from all Google products in order to personalize its product. A video recommendation on YouTube can then be based on searched terms, text within exchanged mails, or pictures uploaded to Picasa.

Context-based applications do exist as well: The iOS application Color (www.color.com [Color]) published in 2011 utilizes the context information "location: to form virtual communities. People simply share pictures (see Figure 4.2.1) and sets/communities are created as hoc based on a common location. Unknown people (see Figure 4.2.2) become all part of a community for a certain time – dynamically (see Figure 4.2.3).

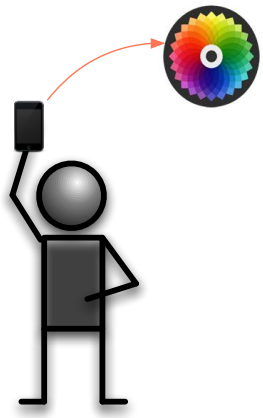


Figure 4.2.1: Single user uploading images to Colour

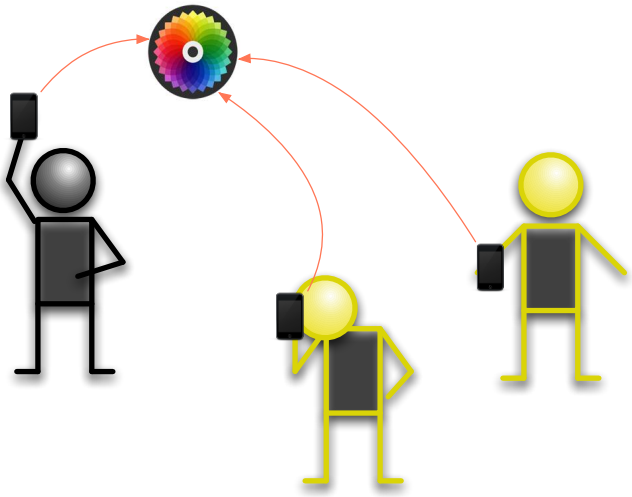


Figure 4.2.2: Multiple users in the same location uploading pictures

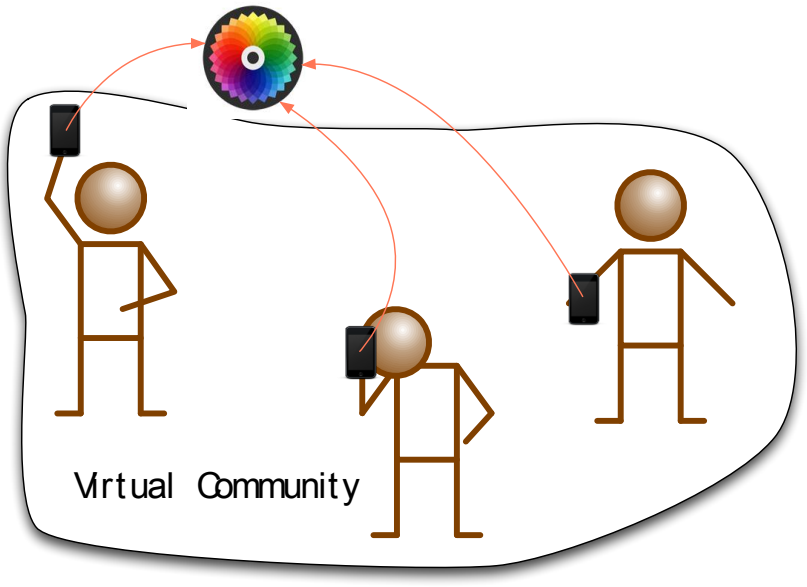


Figure 4.2.3: Virtual dynamic community based on location is created

A less abstract sample for a personalization engine can be easily constructed from current social networks implementations: Current social networks may aggregate and assign context to information, but this is done in a fairly rudimentary manner. The user's stream and published information is distributed based on friends preferences and static settings (e.g. group to share with), but not automatically directed to the right people based on language, current location, or other parameters that could be used to do so.

A sample where personalization could increase the user experience is depicted in Figure 4.2.4. It shows that on Facebook, information is usually shared for all friends. Publishing to selected friends requires a pre-selection, and is rather not obvious. On Google+ sharing with circles is more obvious, however, also here a manual pre-selections has to take place. For publishing information in German and Slovak, two status updates have to be posted, explicitly send to two different pre-assigned circles.

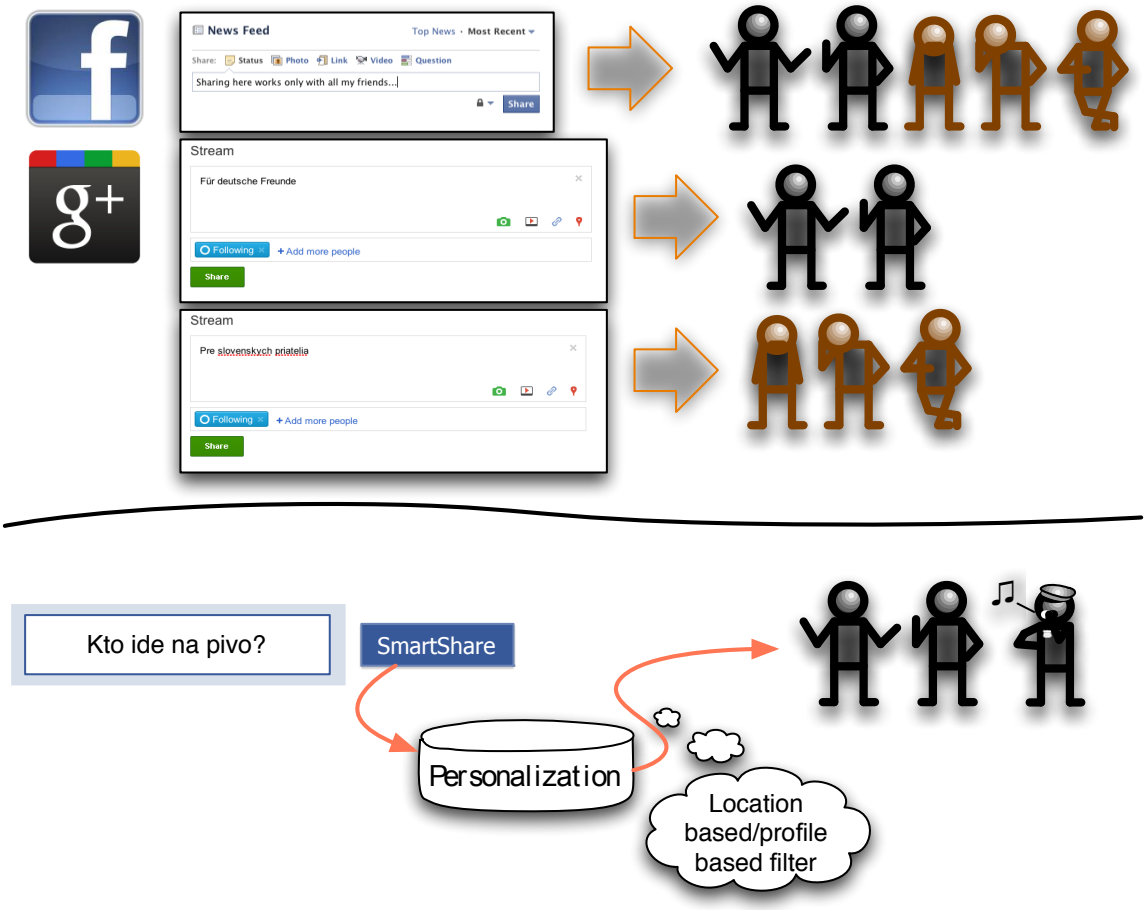


Figure 4.2.4: Potential evolution of personalization

Personalization as illustrated in Figure 4.2.4 could happen in a way that

- Either the language is recognized or the location determined
- Only relevant people based on the determined context receive the information

The engine could be helpful for both sending and receiving party. Even when the user would publish all in English, deep context awareness would still allow to send, e.g., requests for a drink sent in Bratislava only to people there, and when in Berlin vice versa. This can happen automatically and thus improve the user experience.

4.3. Personalization

Personalization can either be implicit (i.e. based on collected and interpreted information and user behavior) or explicit (i.e. based on parameters set by the user). This is similar to feedback discussed in section 3.4.1. Personalization can take place in the backend (i.e. modify service behavior/output) or frontend (i.e. modify service appearance, design, etc.). Research has been done with regard to personalization, notably profiles play an important role in many solutions. Also standard bodies discuss user profile management in the regard of personalization (see [BarIJI08]). It is important to note that personalization is discussed in WP5 and often in connection to recommendation, but the enabler that is to be developed shall act as independent service. The target of personalization within HBB-NEXT is to aggregate several profile sources and to provide a personalized profile for various services. Even yet undefined applications shall be able to make use of the personalization engine without prior explicit definition. Goals and gaps towards current implementations are summarized at the end of this chapter. Multi-user recommendation will involve also both systems: Personalization will provide profiles to a multi-user recommendation algorithm within the personalization engine and provide a calculated multi-user recommendation profile to the recommendation engine.

To define the WP5 task personalization (and thus the definition within the HBB-NEXT project) better and to narrow down state of the art aspects of personalization, some theorems and axioms shall be defined in the following.

Theorem

- The following axioms are provided within the context of HBB-NEXT WP5 - Personalization.

Profile axioms

- Profiles within the context of "HBB-NEXT WP5 - Personalization" contain parameterized information, which can be used to enable personalization for users and groups.
- The parameters (attributes as well as values) may vary depending on whether it is a user profile or a group profile.

- Groups are formed by users.
- Group profiles can be formed by user profiles.
- Certain user profile information is static for the user (e.g., age). Static per definition as it does not change depending on the context.
- Certain user profile information is dynamic. Dynamic per definition as it changes depending on the context.
- The user profiles may contain multiple "dynamic" parameters of the same value with different attributes depending on the context.
- Group profile information can be dynamically composed by static and dynamic user profile information of all group members.

While current personalization systems are fulfilling many of the above mentioned profile axioms, axiom 6.-8. are not so commonly implemented. A sample of the implementation of axioms 6.+7. is the Mac OSX network settings manager. One can define multiple locations and based on the chosen location, network parameters are set. A partial sample is the automatic forwarding of google.com to the relevant localized version of the website depending on the HTTP request's country of origin. The website changes depending on the context (i.e. the environment (country) within which it is displayed), however, not per user. Once the user is logged in, his personal settings take preference.

The usage of group profiles that are compiled by the profiles of each group member (axiom 8) is currently not done often. In computer systems, group policies have usually been defined in order to group parameters for users and let them inherit settings based on the group they are in (e.g. Microsoft uses group policies as in [MSGroup] to define domain policies valid for all domain members). The plan of personalization in the context of this work package is the exact opposite: A group profile is built, not by inheriting user parameters, but by aggregating and interpreting individual parameters from each group member's profile towards a single parameter in the group profile that is valid for the whole group (or at least the major part of it).

Service axioms

- Many services do not personalize attributes differently for users or groups, they personalize differently based on certain parameterized profile information they have available. All services that personalize based on single user profile attributes should be able to personalize based on group profile attribute if the attributes and value ranges are equivalent. Group values can be determined by various ways (some are described in section 3.10.2).
- The parameters that are used for personalization can contain an indication if it is user or group profile information. Services that differ in personalizing for users and groups can make use of the indication and interpret the profile information accordingly.
- Services process concrete information.
    - User profiles contain abstract information.
    - Group profiles contain abstract information as they are built from user profiles.
- Personalization is the interface between service and profile management that interprets profile parameters.

The problem with regard to tasks for the personalization engine can be derived from the service axioms and shall be discussed further in the following section. Section 3.9.3 describes the interaction with the group recommendations. A personalization engine could handle certain functionality that is needed for this.

- Build group model
- Aggregate user profile information and provide group profile

It is up to further elaboration where the functionality of actual calculation of group parameters should be located (whether the personalization engine does that by itself or provides an interface to a third party with the collected profile information for final "calculation"). As mentioned before, the personalization (calculation of group parameter) should happen in the personalization engine, while the recommendation based on these parameters should happen in the recommendation engine. It has to be studied whether the functions can be split accordingly.

[WeiDim08] discussed personalization for digital multimedia content. Content attribute mapping as well as explicit and implicit profiling are discussed (cmp. sec. 1). in that document.

4.4. Personalization engines

The personalization engine is a component that helps personalizing service information by aggregating user profiles (multi-user capability) and taking the context into account.

The personalization engine interacts strongly with the identity management and profile management (both discussed in WP3). It shall provide capabilities to services (e.g., recommendation) to deliver a personalized service experience that is not restricted to a single user (in that case, simple profile information could be used instead). For single users as well as multiple users, the experience is enhanced further by taking context information into account (e.g., location, environment variables (e.g. language, noise)).

With regard to HBB-NEXT, WP5 has the requirement on the following responsibilities shared by the modules:

IdM

- Contains all users
- Contains relation between user and device
- Contains information about active users

Profile Management

- Contains services of the user
- Contains user profiles
- "Dumb" with regard to profile information (no interpretation)

Personalization Engine

- Collects user profiles from PM and builds group profile from it
- Interprets group profile and provides personalized profile
- “Smart” with regard to profile information (interpretation will take place within the personalization engine
- Interprets context

Profile analyzer (may be part of personalization engine, to be analyzed)

- Passively collecting information from user traffic analyzes
- Passively collecting information from user actions
- Puts information to PM

Figure 4.4.1 shows a potential interaction of the personalization engine.

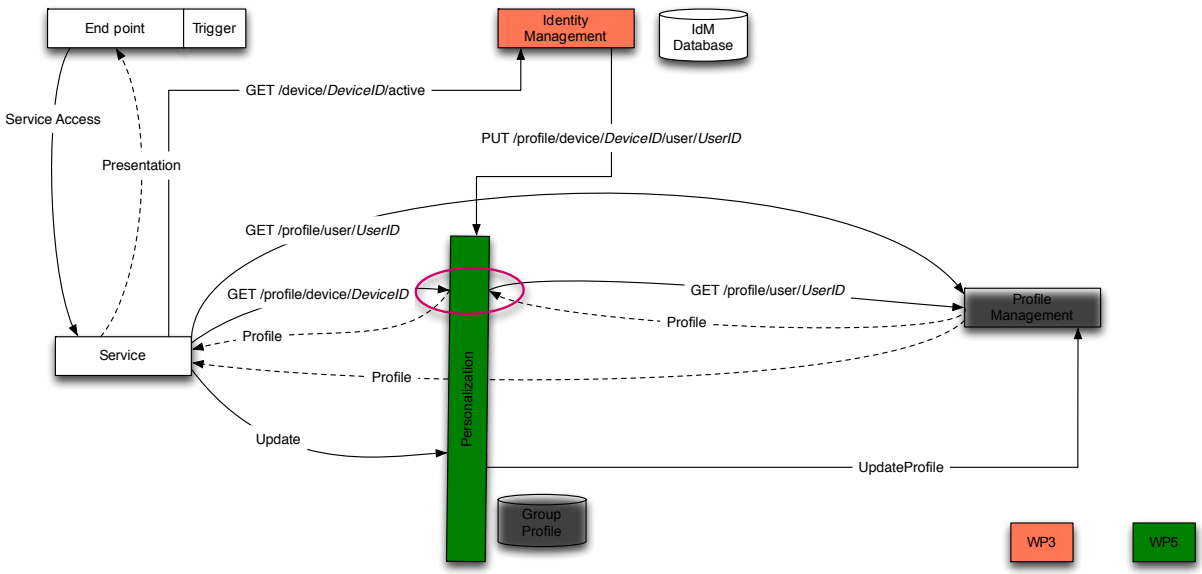


Figure 4.4.1: Personalization interacting with a service and WP3 components

The personalization engine should interface the identity management as well as the profile management. It should provide an interface by which the personalized user profile can be retrieved.

Searching available sources, no implementation of a personalization engine as described within this section has been implemented yet. There are certain solutions available in the area of eCommerce (e.g. ATG Adaptive Scenario, CNET Intelligent, Coremetrics Intelligent. Within the ePerSpace – IST Integrated Project some research and outputs have been produced as well discusses also personalization engine in a similar way this WP plans to do. The gaps in section 4.6 discuss some limitations the project encountered and the plan to overcome them within HBB-NEXT.

4.4.1. IPTV Personalization

Personalization for IPTV can be widely used for recommendation purposes (commercial benefits) and for increasing of user experience (service attractiveness benefits). There is a huge amount of parameters dependent on personal preferences, e.g., genre, language, actors, location (News).

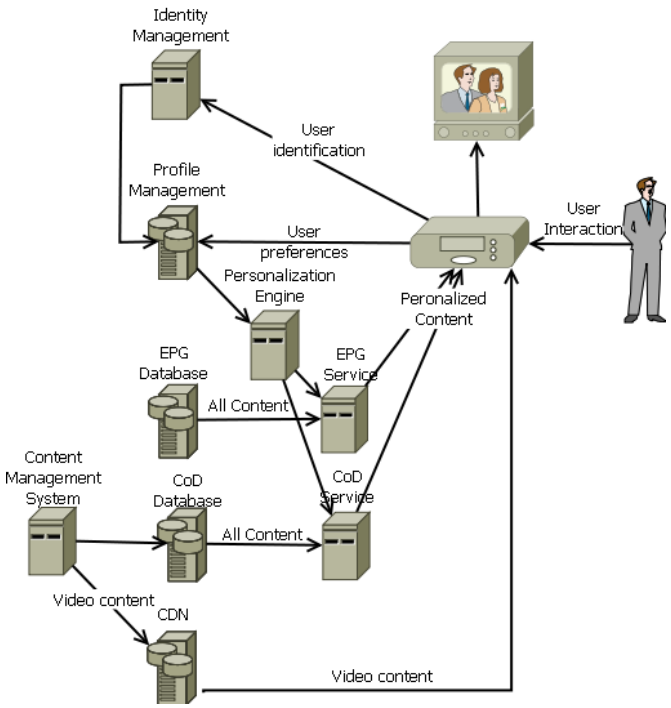


Figure 4.4.1.1: Service interconnection

The service must contain functions for personalization of its content or presentation layer. This function includes a client for profile collection and adaptation (Figure 4.4.1.1). The service is responsible for personalization in its context.

There are two main streams, which can be personalized in TV:

- Content filtering – shows only preferred content
- Content parameters – shows content in preferred way.

Group mechanism is also very useful for TV, because in contrast with mobile or PC, TV is mostly not only personal. TV is in the center of many houses living rooms and is used by all family members and by guests from external environments.

In comparison with "self setup" personalization, profile management offers information to device without disturbing users.

Example: Imagine that a college from work is coming for visit. You do not know, that he loves Sci-Fi genre, your TV will get this information from PM and fortunately because you also loves Sci-Fi offers you movie from this genre.

4.4.2. Sample tasks

This section contains information on which the theory of personalization should be built upon. The EPG is used as sample service, a concrete sample is provided. Afterwards, abstract steps are derived from it.

4.4.2.1. Sample task description (single user personalization)

Scenario

1. User opens EPG
2. EPG displays all channels
3. User can select channel where he would like to jump or program for additional service (information, recording, etc.)

Steps (w/o personalization)

As a sample, a state of the art scenario without personalization is described.

1. User opens EPG
    1.1. User selects EPG on end device
    1.2. EPG service receives trigger with request for displaying content
2. EPG displays all channels
    2.1. EPG retrieves content
    2.2. EPG processes content for presentation
    2.3. EPG sends prepared content back to device
3. User can select channel where he would like to jump or program for additional service (information, recording, etc.)
    3.1. EPG instructs set-top-box for channel change or add. service

Steps (w/ personalization)

Based on the above-mentioned state of the art task sample, the following steps sketch a scenario with applied personalization (in a way HBB-NEXT could provide Elements relevant to personalization are highlighted.

1. User opens EPG
    1.1. User selects EPG on end device
    1.2. EPG service receives trigger with request for displaying content and **source of request** (device)
2. EPG displays personalized channels
    2.1. EPG retrieves **relevant profile parameters of source of request** (interest, channel order, language, device capabilities)
    2.2. EPG retrieves content

2.3. EPG **processes personalized content** for presentation

2.4. EPG sends prepared content back to device

3. User can select channel where he would like to jump or program for additional service (information, recording, etc.)

3.1. EPG instructs set-top-box for channel change or add. service

3.2. EPG provides **feedback about action for personalization**

4.4.2.2. Sample task description (group profile personalization)

Scenario

1. User started watching channel or video
2. Channel is displayed and audio plays
3. Another person is coming.
4. User can change audio language.

Steps (w/o personalization)

As a sample, a state of the art scenario without personalization is described.

4. User is watching TV

1.1. User starts video playback

1.2. Another person is coming

1.3. User changes audio settings to English

2. STB gets information from server what languages are available for content and plays

2.1. STB checks group of languages from server. Preferred language is chosen according information from device profile and requested from server.

2.2. STB continues playing

2.3. STB requests another language stream from server and plays.

3. User is watching TV, and switches language

    3.1. Video is played in language device prefers. (e.g. French),

    3.2. Video continues playing

    3.3. Audio language has changed

Steps (w/ personalization)

Based on the above-mentioned state of the art task sample, the service is dynamically personalized without user interaction.

1. User is watching TV

    1.1. User starts video playback

    1.2. Another person is coming

    1.3. User does not need to changes audio settings to English

2. STB gets information from server what languages are available for content

    2.1. STB checks group of languages from server. STB gets information from personalization what group profile is used. STB chooses the language according this information.

    2.2. Face recognition engine informs about group change. Group profile is updated and STB is informed about new profile.

    2.3. STB requests another language stream from server and plays.

3. User is watching TV, and language is automatically switched

    3.1. Video is played in language user prefers. (e.g. French)

    3.2. Video continues playing

    3.3. Audio language has been changed automatically.

4.4.2.3. Sample for a recommendation service

Figure 4.4.2.1 displays a sample flow how a personalization engine can generally work. Figure 4.4.2.2 depicts a block diagram how a personalization can enable context aware personalized recommendation.

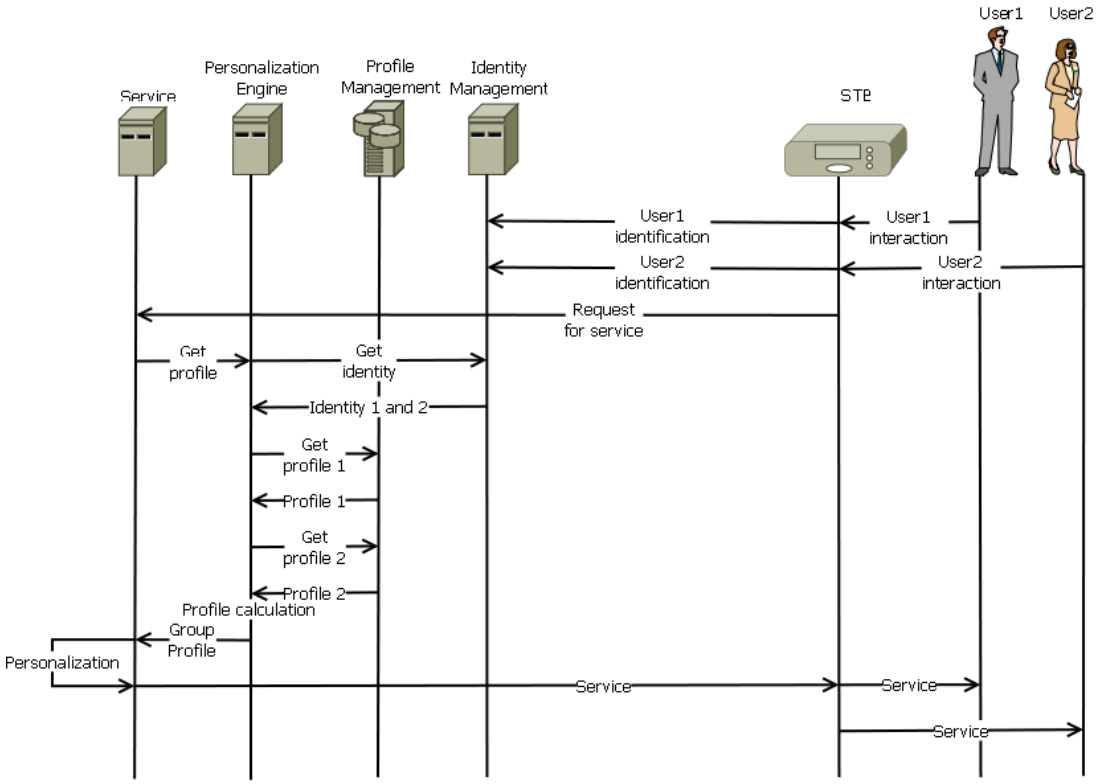


Figure 4.4.2.1: High-level interaction scenario for delivering context aware multi-user personalization (flow)

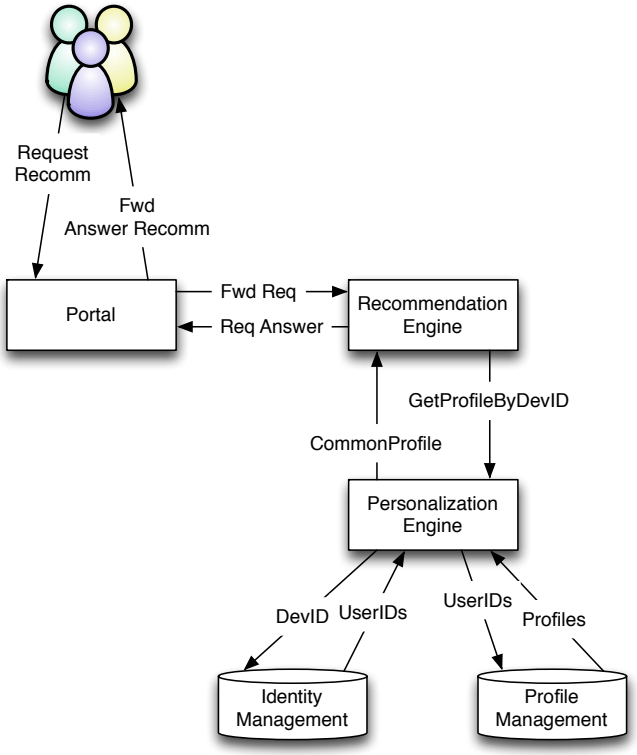


Figure 4.4.2.2: High-level interaction scenario for delivering context aware multi-user personalization (blocks)

The sample depicted above in general and in particular consists of the following high-level steps:

- User access recommendation portal
- Portal should provide recommendation
- Portal knows the device, asks Personalization module for profile info
- Personalization module has a group profile, as it has been informed before by IdM that these users are online on the device
- Personalization module answers with computed group profile
- Recommendation module recommends acc. that profile
- Recommendation module sends feedback to personalization module
- Group profile is updated
- User profiles are updated

Currently there is no television system that could provide the above-mentioned sample. From a high level perspective, any system that supports general personalization nowadays could be modified to perform the above mentioned tasks (as long as service axiom 1 (described in Section 4.1) applies): A personalization engine could be placed between the profile storage and recommendation system and provide parameters as if they were user parameters – with the difference that it modified requested parameters (assuming it knows from which users) and personalized the profile with respective algorithms. Section 4.6 provides an outlook on gaps in the area of personalization and concludes on how the HBB-NEXT project can overcome certain drawbacks in that area.

4.5. Context awareness

In the past computers were devices located in fairly stable settings, and were operated by single users sitting at a desk [DouPUC04]. These systems were comprised of computers, screens, keyboards and mice. In 1991, Weiser described a vision of how computers systems were to be used in the future [WeiSCA91]. Research on this topic went by several names: Ubiquitous Computing [WeiSCA91], Context-Aware Computing [DeyHCI01], Pervasive Computing [ArkISJ91], Embodied Interaction [DouMIT01], etc. Whatever the name, the fundamental difference in the new way computer systems were used and envisioned, was that now they resided in many different settings, and that these computer systems should designed to be responsive to aspects of these settings [DouPUC04].

In order to work toward this vision, researchers, designers and application developers needed to get a grip on the concept of context. Many publications illustrate how difficult it is to accurately describe and define context. Dervin described context as “The Unruly Beast” [DerMTT97], and further on says: “there is no term that is more often used, less often defined, and when defined, defined so variously as context”.

Given the difficulty of defining context as a concept, how are we to understand this difficult concept of context, let alone develop new systems that are supposed to respond to this difficult-to-define concept? Dervin states that the many academic works on context are actually located on a continuum between two scientific philosophies: the positivist science and the qualitative, humanistic science or critical/cultural science [DerMTT97].

Another explanation on the different views on context by Dourish describes the first as the representational view on context, the latter as the interactional view on context [DouPUC04]: the representational view on context is concerned with how to capture, model, and represent context: "the idea that context consists of a set of features of the environment surrounding generic activities, and that these features can be encoded and made available to a software system alongside an encoding of the activity itself". The way of looking at this problem, stems from the positivist approach in social sciences, in which social phenomena are studied, and after which theories are formulated and simplified models are constructed that capture underlying patterns. For the interactional view the central concern is "how and why, in the course of their interactions, do people achieve and maintain a mutual understanding of the context for their actions." [DouPUC04]. This is a phenomenological view of the issue: mostly subjective and qualitative in nature. The difference between these views is essential. Whereas most engineering disciplines utilize a positivist approach, HCI utilizes a phenomenological approach.

Representational view	Interactional view
Context is a form of information	Context is a relational property
Context is delineable	The scope of contextual features is defined dynamically
Context is stable	Context is an occasioned property
Context and activity are separable	Context arises from the activity

Table 4.5.1: Approaches to context and context-awareness. Table from [DouPUC04]

In the domain of HCI, several frameworks have been developed which correspond with the interactional view on context. In the following paragraphs, we will briefly describe these frameworks: Activity Theory [NarMIT97], Situated Action [SucCUP87], Distributed Cognition [floESP91], and the Locales Framework [FitCHI96].

In **Activity Theory** the activity defines the context [NarMIT97] [LeoACP87]. An activity consists of a subject (the person or group performing the activity), an object (the motivation for that activity), and operation (the way an activity is carried out.

Furthermore, artefacts and environment mediate the activity. The activity is central to this theory's view on context. The object is directly connected to the activity: "actions are goal-directed processes that must be undertaken to fulfil the object" [NarMIT97] ("object is here understood in the sense of "objective"). Actions are conscious, and different actions may be undertaken to meet the same goal. The objects can change in the course of an activity, emphasizing the dynamic nature of context, but they do not change on a moment-by-moment basis. There is some stability over time. Changes in objects are not trivial; they can change the nature of an activity fundamentally. "What takes place in an activity system composed of object, actions, and operation, is the context" [NarMIT97]. Context is both internal to people (objects and goals) and external to people (people, artefacts and specific settings). An operation is the way an action is actually carried out, and an operation becomes routinized and unconscious with practice. Constituents of activity are not fixed, but can dynamically changes as conditions changes. Object remains fixed, but goals, actions, and operations changes as conditions change.

Artefacts (instruments, signs, language, and machines mediate activity and are created by people to control their own behaviour ("computer-mediated activity").

Situated Action [SucCUP87] strongly emphasizes the dynamics of the situation, in contrast to planned behaviour (routine, predictable) that was given the most attention in traditional cognitive sciences [NarMIT97]. Dourish and Button [DouHCI98] clarify though that this does not imply that plans do not exist: "plans are one of a range of resources which guide the moment-by-moment organization of an activity, rather than laying out a sequence of work which is then blindly interpreted". This aspect was clearly observed in the user study on how people watch video and TV on various devices in the home, carried out and described in D2.2: certain households planned more of the TV programmes they would watch, whereas other household would watch what would be broadcasted. The important thing here is that people actually are actually situated on an entire spectrum between planned and spontaneous behaviour, and people also move on this spectrum. In Situated Action a researcher studies one situation in great detail; very little effort is made to make comparisons across situations, because it considers each situation as unique. In addition, this theory is critical of how people explain their actions afterwards in an interview.

It is seen as a mere rationalization of people's activities after the facts, and doesn't play a role in how and why their activities evolved on a moment-by-moment base. Therefore, there are some methodological implications: detailed observation is preferred (what people do) over interviews (what people say).

Flor and Hutchins describe **Distributed Cognition** as "a new branch of cognitive science devoted to the study of: the representation of knowledge both inside the heads of individuals and in the world…; the propagation of knowledge between different individuals and artefacts…; and the transformation which external structures undergo when operated on by individuals and artefacts… by studying cognitive phenomena in this fashion it is hoped that an understanding of how intelligence is manifested at the systems level, as opposed to the individual cognitive level, will be obtained" [floESP91]. The main goal of distributed cognition is to understand how groups achieve certain goals instead of individuals.

The classic example is how an entire flight crew can be seen as one group achieving their goal: reaching their destination. Therefore, it focuses on how people coordinate and negotiate their action, and on how artefacts mediate the process of achieving their common goal.

The purpose of the **Locales Framework** is to understand the nature of social activity and work, and how a locale (place) can support these activities [FitCHI96]. The key point here is that a locale is not per se associated with a location or physical space. Locales arise out of social activities. The living room can be considered a locale when people are watching TV in the evening; but also a group of people having a pick nick together at a certain time listening to a portable radio can considered a locale. The Locales Framework contains a number of concepts, some of which are very relevant to HBB-NEXT. The first concept is that of individual views: at any given moment a person can be part of several different social worlds. This also implies that these people have their own view in each one of these social worlds. To illustrate this aspect: as our user study in D2.2 showed: people switch their attention from the 1st screen (TV) to the 2nd screen (laptop, smartphone or tablet) sometimes, they actually move from the locale of the living room, to the (virtual) locale of a Skype conversation of Facebook chat with a friend. Then, there is the concept of mutuality: "identifying the mutual communicative processes through which awareness is achieved.

It considers how spaces and things are made available and accessible to people and how people and their activities are made available and accessible to others through the spaces and things they use as locale." And another concept is interaction trajectory: "is concerned with actions and courses of actions, identifying the dynamic and temporal aspects of the living social world and its interactions within and across locales—past, present and future." This concept illustrates the focus on how activities evolve on a moment-by-moment basis, which is also found in the Situated Action framework.

To summarize the above frameworks: in the interactional view the activity is central to context awareness, and more importance is given to the situated aspect, or how context is mostly a dynamic concept. One of the goals associated with the representational view is how contextual information can be gathered beforehand and subsequently used to make predictions about the users' situation.

The key point about the interactional view is that making such predictions beforehand is very difficult to achieve: a lot depends on what will happen during the action. The mentioned frameworks have an important flaw: they are overly complex in order to be used [RogIST04] because they require a profound knowledge of the respective theoretical concepts and a significant amount of time to apply for practitioners. Therefore, these frameworks are difficult to use in industry.

Finally, we discuss a more recent **Proxemic Interactions** framework, based on the term "proxemics" [HalDOU66], an area of study in anthropology, which "identifies the culturally dependent ways in which people use interpersonal distance to understand and mediate their interactions with other people". For HCI the most relevant aspects are the four zones indicating how people experience interpersonal distance:

1. Intimate (< 1.5 feet)
2. Personal (1.5 - 4 feet)
3. Social (4 - 12 feet)
4. Public (12 - 25 feet)

These zones can be interpreted as follows: the smaller the distance, the more intimate the personal interaction. The Proxemic Interactions framework has applied the concept of proxemics to context aware computing or ubiquitous computing. Their main critique on context aware computing, meaning the current devices, is that they are still very much blind to each other, and to other non-computational elements in the room such as the people, semi-fixed elements (such as chairs which can be configured) and fixed elements (such as the location of the door). With their framework they intend to take these elements into account as well. What makes this framework interesting for HBB-NEXT is that the framework is illustrated via applications in closer proximity to the TV (touchscreens) but also with media devices used from the couch. In addition, the framework is comprised of a number of elements, which are clearly described: distance, orientation, movement, identity and location. Location for example, is an element that has also been recognized by our user study in HBB-NEXT deliverable D2.1 [HBB-NEXT_D2.1]: there are a number of locations in the home that have particular characteristics.

The living room is described as comfortable, lean-back, cosy, and social and mostly used for watching TV in social setting, but also to make longer Skype calls (2nd screen). In our study the dining table was also used for watching TV 1st screen, but also for 2nd screen activities (working, surfing) and other activities with the 1st as background companion (i.e. knitting). The Proxemic Ubicomp framework therefore, might be very useful input for the context-aware aspect in HBB-NEXT.

4.6. Conclusions Personalization and context awareness

With regard to personalization, some gaps will be listed with regard to current implementations for delivering a personalized television experience. The plans outline how WP5 will overcome drawbacks in state of the art solutions and provide a useful enabler for the future of hybrid television and beyond.

Gap 1

Location of personalization engine and interaction with other services needs to be defined. This is particularly important as it is considered service enabler rather than a simple function.

Plan

Place personalization engine and define its APIs in a way it can generically interact with services.

Gap 2

Context-based profiling is not adequately developed for proper and advanced personalization.

Plan

Analyse how profiles can be made adaptive and universal based on the context. Design models and prototype implementations for how profiling can support personalization taking the users context into account. Find the proper location for storing and processing the context-based profile.

Gap 3

Lack of dynamic parameters in profiles. Dynamic parameters are changed over time. Yet unclear are means how to collect them (feedback), where to store them, and at most how to inform a service that a dynamic parameter has changed.

Plan

Analyze parameters, which are changed over time. Plan how to store them and mostly how to inform a service that a dynamic parameter has changed. Analyze feedback mechanisms from services to update parameters for single users and for groups. Define process and function entities, which will cover functionality for data collection, storage and processing.

Gap 4

Data is currently not aggregated but rather processed independently from services.

Plan

Define data aggregation functionality, which must be distributed over the services. These services can then provide feedback to the profile management. The following work will define interfaces for submitting such information.

Gap 5

When/where profile info is calculated. Now there is no functional entity, which will process the profile.

Plan

Define functional entities. First entity will calculate new profile. Other entities will act if related environment has changes. Then this information will be evaluated if changes have influence to service profile.

Gap 6

Profile information interchange mechanisms.

Plan

Analyze automatic subscription mechanisms. Dynamic information and service interaction does require mechanisms beyond single profile fetching. Information can be updated in the mean time, the context can be changed and all that needs to be communicated within the HBB-NEXT framework.

Gap 7

Make single user systems usable for groups

Plan

Automatically personalize single user systems w/ profile value calculation acc. single users' parameters. Build interfaces for universal data input of multiple users.

A large amount of context middleware has been developed, ranging from simple toolkits for plugging together and reusing context sources and processing components [SalCHF99] to more complete context management frameworks that provide homogeneous access to a large number of application and services [FloMCS05], [GroPCC05], [BauMWC06], [KraCOM06], abstracting from aspects like distribution and data representation of the underlying sources of context information.

Gap 8

Context frameworks typically target relatively homogeneous systems that allow the integration of simple context sources, not the integration of complete context frameworks

Plan

This significant challenge will be addressed in order to provide a context management service that can provide the context information from and for all service infrastructures relevant for interactive hybrid applications and services. The context information shall be made available and processable for delivering context-relevant personalization.

Gap 9

Absence of mapping functionalities for accessing context information as well as the mapping of context models and representations from and to different technology domains (here: broadcast, broadband and mobile domains).

Plan

WP5 will define structured model to define the context and process it in the personalization engine.

Finally, the representational view on context, which mainly focuses on modelling context, is somewhat limited, since context is actually a very dynamic concept. People's actions can change on a moment-be-moment basis. Another problem is that computer systems cannot always, and probably should not, know everything what the user knows. Therefore, there will always be some errors in predications made by a system.

The implications for HBB-NEXT then are the following:

1. Systems that try to model and represent contextual information should take into account that this context is dynamic, and therefore needs to be updated regularly. The more difficult thing here is, knowing which elements are relevant at which moment in time. The Proxemic Ubicomp framework might provide a number of useful elements that can be taken into account for HBB-NEXT. Also, the user study in D2.2 gives some insight in how users today use TV and video applications at home in everyday life on different devices.

2. Attention should be paid to situations where the context-aware system makes a mistake or an inaccurate assessment. The impact thereof can be quite high, in case private information is involved for example. Therefore, context-aware systems should allow for easy, fast correction by the user, and should learn from these corrections. In addition, systems should be fairly conservative in the actions they make, certainly for the high-risk actions.

3. In line with the above implications, a third implication has also been formulated in literature: instead of merely relying on the system to perform context-aware interpretations (not that this effort should cease off course, they will always get better given developments in Artificial Intelligence and Data Mining), it would be useful to provide the user with certain information allowing the user to make these context-aware interpretations: “allowing the user to make continual determinations of the potential consequences of their actions, and their opportunities to reconfigure or realign the technologies through which they are conducting their actions”. This requires an architecture that allows for tailorability and adaptation by the user. Important in HBB-NEXT then is information on the social context: “notifying the user of incoming events and situation changes”.

5. Conclusion

According to the HBB-NEXT Description of Work [HBB-NEXT_D2.1], the objective of WP5 is to develop a Service Personalisation Engine (D5.5.2, D5.6.2), with multi-user recommendations, multimodal interaction and context-awareness. Task 5.1 (resulting in the D5.1 deliverable that you are now reading) makes a state-of-the-art analysis of existing solutions, challenges and technology gaps. Task 5.2 focuses on the front-end functionality (D5.5.2) developing solutions for user-to-system interaction involving a multimodal user-interface in a multi–user environment. Task 5.3 augments these solutions with back-end functionality (D5.6.2), specifically context-aware personalised multi-user content recommendation. The work will build on previous work from other projects, such as iNEM4U and the partners' strong expertise in this area.

Section 1 presented use cases on multi-user, multimodal & context aware value added services, and explained how the remaining sections link together, as illustrated in Figure 5.1.

Section 2, input to HBB-NEXT Task 5.2, presented state of the art on multimodal interfaces, viz. four multi-modal inputs (face recognition, gesture recognition, speech recognition and multi-speaker identification) and one multi-modal output (speech synthesis). The main challenge beyond the SoTA will not be the improvement of individual multi-modal interfaces, but their combination and integration into the HBB-NEXT architecture as a whole, and the design of generic future-proof plug&play API's (e.g. open to support a mindreading interface and odour feedback, both of which are currently out of the HBB-NEXT scope).

Section 3, input to HBB-NEXT Task 5.3, presented state of the art on content recommendation, including filtering types, user profiles, presentation, evaluation, scalability, metadata, privacy and multi-user aspects. The main challenge beyond the state of the art is the width of design space combining context-awareness, multi-user filtering, content-based filtering (metadata merge from multiple sources) and collaborative filtering.

Section 4, as input to of HBB-NEXT Task 5.2 and linking to Task 5.3, presented approaches to personalisation and achieving context awareness. The main challenge beyond the state of

the art will be modelling the context from many sources (monitoring) and actively reacting on it in a personalised way such that the group of users together benefit from it.

Here, the representational view on context, which mainly focuses on modelling context, is somewhat limited, since context is actually a very dynamic concept. People's actions can change on a moment-be-moment basis. Another problem is that computer systems cannot always, and probably should not, know everything what the user knows. The HBB-NEXT system design should consider this.

6. References

[AdoIEE05] Gediminas Adomavicius and Alexander Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Transactions on Knowledge and Data Engineering 17, pages 734–749, 2005.

[AMA07] Amazon.com, Inc.: http://www.amazon.com. May 2007.

[Amazon] Amazon.com.

[AngICS04] Anguera, X. and Hernando, J.: 2004a, Evolutive speaker segmentation using a repository system, Proc. International Conference on Speech and Language Processing, Jeju Island, Korea.

[ANNWeb07] "Speech Synthesis Software for Anime Announced". Anime News Network.2007-05-02. http://www.animenewsnetwork.com/news/2007-05-02/speech-synthesis-software. Retrieved 2010-02-17.

[ANNWeb08] "Code Geass Speech Synthesizer Service Offered in Japan". Anime News Network.2008-09-09. http://www.animenewsnetwork.com/news/2008-09-09/code-geass-voice-synthesis-service-offered-in-japan. Retrieved 2010-02-17.

[ArkISJ91] Ark, W.S., Selker, T. (1991).A lok at human interaction with pervasive computers. IBM Systems Journal.

[AUDI] Audi Glossary, Driver's seat with memory function - http://www.audi.com/com/brand/en/tools/advice/glossary

[BanRad07] Bandzi, P., Oravec, M., Pavlovičová, J.: New Statistics for Texture Classification Based on Gabor Filters.In: Radioengineering. - ISSN 1210-2512. - Vol. 16, No. 3 (2007), s. 133-137.

[BauMWC06] M. Bauer, R. L. Olsen, et al., "Context Management Framework for MAGNET Beyond", Workshop on Capturing Context and Context Aware Systems and

Platforms, IST Mobile and Wireless Communications summit, Myconos, Greece, 2006.

[BarIJI08] G. Bartolomeo, et al., "Personalization and User Profile Management", ETSI STF 342, International Journal of Interactive Mobile Technologies (iJIM), Vol 2, No 4, 2008.

[BenIjs11] Beniak, M., Pavlovicová, J. and Oravec, M. (2011) '3D chrominance histogram based face localisation', Int. J. Signal and Imaging Systems Engineering, Vol. 4, No. 1, pp.3–12.

[BerCRS] S. Berkovsky, Y. Eytani, T. Kuflik, and F. Ricci. Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems, pages 9–16, New York, NY, USA.

[BesMgv09] Beszédeš,M., Culverhouse,P., Oravec,M.: Facial Emotion Classification Using Active Appearance Model and Support Vector Machine Classifier, M&GV (Machine Graphics & Vision), Institute of Computer Science, Polish Academy of Sciences, ISSN 1230-0535, Vol. 18, No. 1, 2009 (pp. 21-46).

[Bla07] Marcel Blattner, Alexander Hunziker, and Paolo Laureti, When are recommender systems useful?, September 2007.

[BlaICC94] Black, A. W., Taylor, P. "CHATR: a generic speech synthesis system". In Proc. of the International Conference on Computational Linguistics, Kyoto, Japan. 1994.

[BonBTT06] P. Bonhard and M. A. Sasse. "Knowing me, knowing you" - Using Profiles and Social Networking to Improve Recommender Systems. BT Technology Journal, Vol 24 No 3, July 2006.

[BraBIP00] Brands, S. Rethinking Public Key Infrastructures and Digital Certificates; Building in Privacy, 1st ed. MIT Press, 2000. ISBN 0-262-02491-8.

[BucSIR95] Buckley, C., and Salton, G. Optimization of relevance feedback weights. In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Seattle, July 1995).

[CamCCS02] Camenisch, J., and Herreweghen, E. V. Design and implementation of the Idemix anonymous credential system. In ACM Conference on Computer and Communications Security (2002), V. Atluri, Ed., ACM Press, pp. 21–30.

[CamUMU09] Luis M. de Campos, Juan M. Fernández-Luna, Juan F. Huete • Miguel A. Rueda-Morales: Managing uncertainty in group recommending processes, Springer, User Model User-Adap Inter (2009) 19:207–242, DOI 10.1007/s11257-008-9061-1.

[CanSSP02] J. F. Canny. Collaborative filtering with privacy. In IEEE Symposium on Security and Privacy, pages 45–57, 2002.

[CardSpace] Windows CardSpace, http://en.wikipedia.org/wiki/Windows_CardSpace.

[CasITE07] Casar, M. &Fonllosa, J. (2007). Double layer architectures for automatic speech recognition using HMM, in book Robust Speech recognition and understanding, I-Tech education and publishing, ISBN 978-3-902613-08-0, Croatia, Jun, 2007.

[ChaIJC11] AnkitChaudhary, J. L. Raheja, Karen Das, Sonia Raheja, "Intelligent Approaches to interact with Machines using Hand Gesture Recognition in Natural way: A Survey," International Journal of Computer Science & Engineering Survey, vol. 2, no. 1, pp. 122-133, Feb 2011.

[CheACA02] A. Cheveigne, H. Kawahara, “YIN, a fundamental frequency estimator for speech and music”, J. Acoust. Soc. Am., Acoustical society of America, Vol. 111, No. 4, 2002.

[ChoIEE07] Jinhyung Cho, Kwiseok Kwon, and Yongtae Park, Collaborative filtering using dual information sources, IEEE Intelligent Systems 22 (2007), no. 3, 30–38.

[CleSIG07] M. Clements, A. P. de Vries, J. A. Pouwelse, J. Wang, and M. J. T. Reinders, Evaluation of Neighbourhood Selection Methods in Decentralized Recommendation Systems, 2007, SIGIR 2007 Workshop, pp. 38–45.

[Color] http://www.color.com.

[ConSCW01] Mark O'Connor, Dan Cosley, Joseph A. Konstan, John Riedl: PolyLens: A recommender system for groups of users, Proceedings of the Seventh European Conference on Computer Supported Cooperative Work, ECSCW 2001.

[CooCom95] T.F. Cootes, D. Cooper, C.J. Taylor and J. Graham, "Active Shape Models - Their Training and Application."Computer Vision and Image Understanding. Vol. 61, No. 1, pp. 38-59, Jan. 1995.

[CooEur98] T.F.Cootes, G.J. Edwards and C.J.Taylor. "Active Appearance Models", in Proc. European Conference on Computer Vision 1998 (H.Burkhardt& B. Neumann Ed.s). Vol. 2, pp. 484-498, Springer, 1998.

[CooPAM01] T. F. Cootes, G. J. Edwards, C. J. Taylor, "Active Appearance Models", IEEE PAMI, Vol.23, No.6, pp.681-685, Department of Medical Bio-Physics, University of Manchester, Oxford Road, Manchester M139PT, England, 2001.

[CooSpr05] T. F. Cootes, C. J. Taylor, H. Kang, V. Petrović, "Modeling Facial Shape and Appearance", Imaging Science and Biomedical Engineering, University of Manchester, England, Handbook of Face recognition, chapter 3, pp.39-63, Springer, 2005.

[CorWeb12] European Commission CORDIS (Community Research and Development Information Service).[Online]. Available: http://cordis.europa.eu/.[Accessed 12 1 2012].

[CroWeb12] J. Crook, "Samsung's Smart TV Can "Listen, See, And Do" No Evil," 9 1 2012.[Online]. Available: http://techcrunch.com/2012/01/09/samsungs-smart-tv-can-listen-see-and-do-no-evil/.[Accessed 11 1 2012].

[DerMTT97] Dervin, B. (1997). Given a Context by Any Other Name: Methodological Tools for Taming the Unruly Beast. In P. Vakkari, R. Savolainen, & B. Dervin (Eds.à, Information seeking in context (pp. 13-38). London: Taylor Graham.

[DeyGIT00] Dey, A.K., Providing architectural support for building context-aware applications, in Computer science, 2000, Georgia Institute of Technology. pp. 240.

[DeyHCI01] Dey, A.K., Abowd, G.D., Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Human-Computer Interaction, Vol. 16, No. 2, p97-166.

[DinIca10] Ding, B., Shi, R., Liu, Z. and Zhang, Z (2010) 'Human object segmentation using Gaussian mixture model and graph cuts' Audio Language and Image Processing (ICALIP), 2010 International Conference on , Vol., No., pp.787-790.

[DonCSL99] Donovan, R. E., Woodland, P. "A hidden Markov-model based trainable speech synthesizer". Computer Speech and Language, 13(3):223-241, 1999.

[DouHCI98] Dourish, P., Button, G. (1998). On technomethodology: ofundational relationships between ethnomethodology and system design. Human-Computer Interaction, Vol. 13, No. 4, p395-432.

[DouMIT01] Dourish, P. (2001). Where the action is: the foundations of embodied interaction. MIT Press.

[DouPUC04] Dourish, P. (2004). What we talk about when we talk about context. Personal Ubiquitous Computing, Vol. 8, p19-30.

[DOW_HBB_NEXT] HBB-NEXT consortium, "HBB-NEXT Division of Work", FP7 project no. 287848, 12 August 2011.

[DutJEE97] Dutoit, T. "High-Quality Text-to-Speech Synthesis: an Overview". Journal of Electrical & Electronics Engineering, Australia: Special Issue on Speech Recognition and Synthesis, vol. 17, n°1, pp. 25-37. 1997.

[EdwEur98] G.J. Edwards, T.F. Cootes and C.J. Taylor, "Face Recognition Using Active Appearance Models", in Proc. European Conference on Computer Vision 1998 (H.Burkhardt& B. Neumann Ed.s). Vol. 2, pp. 581-695, Springer, 1998.

[EllOSC09] Jonathan Ellis: What Every Developer Should Know About Database Scalability, Open Source Convention, OSCON 2009.

[ErkCAS11] Erkin, Z., T. Veugen, and R. L. Lagendijk, "Generating Private Recommendations in a Social Trust Network", The International Conference on Computational Aspects of Social Networks (CASoN 2011), Salamanca, Spain, IEEE, 2011.

[ErkSSP11] Erkin, Z., M. Beye, T. Veugen, and R. L. Lagendijk, "Efficiently Computing Private Recommendations", International Conference on Acoustic, Speech and Signal Processing-ICASSP , Prag, Czech Republic, pp. 5864--5867, May/2011.

[EtsWeb12] The European Telecommunications Standards Institute (ETSI) standards.[Online]. Available: http://www.etsi.org/.[Accessed 12 1 2012].

[EyeWik12] „EyeToy,“[Online]. Available: http://en.wikipedia.org/wiki/EyeToy.[Cit. 11-01-2012].

[FitCHI96] Fitzpatrick, G., Mansfield, T., and Kaplan, S.M. (1996). The Locales Framework: Exploring foundations for collaboration support. Computer-Human Interaction, pp. 34-41.

[floESP91] Flor, N., Hutchins, E. (1991). Analyzing districted cognition in software teams: a case study of team programming during perfective software maintenance. In J. Koenemann-Belliveau et al., eds., Proceedings of the Fourth Annual Workshop on Empirical Studies of Programmers (pp. 36-39). Norwood, N.J.: Ablex Publishing.

[FloMCS05] P. Floréen, M. Przybilski, P. Nurmi, J. Koolwaaij, A. Tarlano, M. Wagner, M. Luther, F. Bataille, M. Boussard, B. Mrohs, S. Lau, "Towards a Context Management Framework for MobiLife", IST Mobile & Communications Summit, 2005, Dresden, Germany.

[GabIns46] Gabor, D. (1946) 'Theory of communications', Inst. Elect. Eng., Vol. 93, pp.429–457.

[GarIee99] Garcia, C. and Tziritas, G. (1999) 'Face detection using quantized skin color regions merging and wavelet packet analysis', IEEE Trans. Multimedia, Vol. 1, No. 3, pp.264–277.

[GeoCDM05] George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. In: Proceedings of the 5th IEEE Conference on Data Mining (ICDM), pp. 625–628. IEEE Computer Society, Los Alamitos, CA, USA (2005).

[GnuWeb10] "gnuspeech" http://www.gnu.org/software/gnuspeech/ Retrieved 2010-02-17.

[GorCAG04] Goren-Bar, D. & Glinansky, O. (2004). FIT-recommending TV programs to family members. Computers and Graphics, 28, 149-156.

[GroPCC05] M. Grossmann, M. Bauer, N. Hönle, U.-P. Käppeler, D. Nicklas, T. Schwarz, "Efficiently Managing Context Information for Large-Scale Scenarios", Third IEEE International Conference on Pervasive Computing and Communications (PerCom) 2005., pp. 331 - 340.

[HalDOU66] Hall, E.T. (1966). The Hidden Dimension, Doubleday, New York.

[HanESA04] Han, P., Xie, B., Yang, F., Sheng, R.: A scalable p2p recommender system based on distributed collaborative filtering. Expert systems with applications (2004).

[HarWeb12] D. Hardawar, "Samsung's Smart TV evolves with motion, voice controls and upgrade slots," 9 1 2012.[Online]. Available: http://venturebeat.com/2012/01/09/samsungs-smart-tv-evolves-at-ces/.[Accessed 11 1 2012].

[HBB-NEXT_D2.1] Sven Glaser, Bettina Heidkamp, Jennifer Müller et al, HBB-NEXT Usage Scenarios and Use Cases, Deliverable D2.1 of the FP7 HBB-NEXT project, 1 October 2011.

[HerACM04] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl,Evaluating collaborative filtering recommender systems, ACM Trans. Inf. Syst., pages , 5–53, 2004.

[HerICA85] H. Hermansky, B. A. Hanson, H. Wakita, "Perceptually based linear predictive analysis of speech," Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85., 1985.

[HerJIR02] J. Herlocker, J. Konstan, and J. Riedl. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. Information Retrieval, 5(4), pages 287-310, 2002.

[HerUIA00] Herlocker, J. (2000). Understanding and Improving Automated Collaborative Filtering Systems. University of Minnesota, Minnesota.

[HirISC04] Hirai, T., Tenpaku, S. "Using 5ms Segments In Concatenative Speech Synthesis", 5th ISCA Speech Synthesis Workshop, pp. 37-42, Carnegie Mellon University, Pittsburgh. 2004.

[HönINT05] F. Hönig, G. Stemmer, Ch. Hacker, and F. Brugnara, "Revising Perceptual linear Prediction (PLP)," Proceedings of INTERSPEECH 2005, pp. 2997-3000, Lisbon, Portugal, Sept., 2005.

[HuaBHM09] Huang, X.; Ariki, Y. & Jack, M. (1990). Hidden Markov Models for Speech Recognition, Edinburg university press, 1990.

[HuaWAF95] Thomas S. Huang; Vladimir I. Pavlović, "Hand Gesture Modeling, Analysis, and Synthesis," in IEEE International Workshop on Automatic Face and Gesture Recognition, Zürich, Switzerland, 1995.

[IetWeb12] The Internet Engineering Task Force (IETF) standards.[Online]. Available: http://www.ietf.org/.[Accessed 12 1 2012].

[IMD] IMDB Web-page: http://www.imdb.com.

[IshCit02] T. Ishikawa, I. Matthews, S. Baker, “Efficient Image Alignment with Outlier Rejection”, CMU-RI-TR-02-27, pp 1-24, Citeseer, 2002.

[JamRTG07] Anthony Jameson: Recommendation to Groups, Springer-Verlag Berlin, Heidelberg ©2007, ISBN: 978-3-540-72078-2.

[JiaITE07] Jiang, H. & Li X. (2007) A general approximation-optimization approach to large margin estimation of HMMs, in book Robust Speech recognition and understanding, I-Tech education and publishing, ISBN 978-3-902613-08-0, Croatia, Jun, 2007.

[JonIjc02] Jones, M.J. and Regh, M. (2002) 'Statistical color models with application to skin detection', International Journal of Computer Vision, Vol. 46, No. 1, pp.81–96.

[KemSSP00] Kemp, T.; Schmidt, M.; Westphal, M.; Waibel, A.; , "Strategies for automatic segmentation of audio data,". ICASSP '00. in Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, 2000, vol.3, no., pp.1423-1426 vol.3, 2000

[KenSDC72] Kendon, "Some Relationships between body motion and speech.," in Studies in Dyadic Communication, A. W. Siegman and B. Pope, Eds., New York, Pergamon Press, 1972.

[KimIci10] Kim, S.-H.and Lee, H.-S. (2010) 'Face/Non-face Classification Method Based on Partial Face Classifier Using LDA and MLP' Computer and Information Science (ICIS), 2010 IEEE/ACIS 9th International Conference on , Vol., No., pp.234-239.

[KimWor07] D. Kim, J. Kim, S. Cho, Y. Jang, S. Chung, B. Kim, "Progressive AAM Based Robust Face Alignment", World Academy of Science, Engineering and Technology, 2007.

[KinOfi] Craig Eisler, "Starting February 1, 2012: Use the Power of Kinect for Windows to Change the World"[Online]. Available: http://blogs.msdn.com/b/kinectforwindows/archive/2012/01/09/kinect-for-windows-commercial-program-announced.aspx[Cit. 11.1.2012].

[KINWEB] Harry Fairhead,"All about Kinect",[Online]. Available: http://www.i-programmer.info/babbages-bag/2003-kinect-the-technology-.html.[Cit. 10.01. 2012].

[KinWeb11] „Finger Movements can be detected by Kinect," Kinect Hacks, 21. 02. 2011.[Online]. Available: http://www.kinecthacks.com/finger-movements-can-be-detected-by-kinect/.[Cit. 11.01. 2012].

[KraCOM06] H. van Kranenburg, M.S. Bargh, S. Iacob, A. Peddemors, "A Context Management Framework for Supporting Context Aware Distributed Applications In: IEEE Communications Magazine, vol. 44, nr. 8, 2006, pp. 67-74, IEEE.

[KruPCP06] S. E. Kruger, M. Schaffoner, M. Katz, E. Andelic, and A. Wendemuth, "Mixture of support vector machines for HMM based speech recognition," in Proc. 18th Int. Conf. Pattern Recognit., Hong Kong, China, 2006, pp. 326–329.

[LeoACP87] Leont'ev, A. N., Hall, M. J. (1987). Activity, Consciousness, and Personality. Englewood Cliffs, NJ: Prentice-Hall.

[LinIEE03] Linden, G., Smith, B., and York, J. (2003). Amazon.com Recommendation: Item-to-Item Collaborative Filtering. IEEE Internet Computing, January-February, 76-80.

[MarTCE09] A.B.B. Martinez ABB, J. Arias, A.F. Vilas, J.G. Duque, M.L. Nores. What's on tv tonight? an efficient and effective personalized recommender system of tv programs. IEEE Transactions on Consumer Electronics, Volume 55(1), pages 286-294, 2009.

[MasRSH11] Judith Masthoff: Group Recommender Systems: Combining individual models, Chapter 21 of Ricci et al, Recommender Systems Handbook, 2011.

[MasUMU04] Judith Masthoff: Group modelling - Selecting a sequence of television items to suit a group of viewers, User Modeling and User-Adapted Interaction pp.37-85, 2004, Kluwer Academic Publishers.

[MccSCW98] Joseph F. McCarthy , Theodore D. Anagnost: MusicFX: An Arbiter of Group Preferences for Computer Supported Collaborative Workouts, Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW '98).

[MclSIG04] Matthew R. Mclaughlin and Jonathan L. Herlocker, A collaborative filtering algorithm and evaluation metric that accurately model the user experience, In Proceedings of the 27th annual international conference on Research and development in information retrieval. (New York, NY, USA), ACM Press, 2004, pp. 329–336.

[McN06] Sean Michael McNee, Meeting user information needs in recommender systems, Ph.D. thesis, University of Minnesota, June 2006.

[MihITe08] Mihelič, F., & Žibert, J., Speech recognition: Technologies and applications. Vienna: I-Tech Education and Publishing, 2008. Available online: http://www.scribd.com/doc/18020731/Speech-Recognition

[MirUPC06] X. A. Miro, "Robust Speaker Diarization for Meetings," PhD thesis, UniversitatPolitecnica de Catalunya, Barcelona, 2006.

[MitSMC07] S. Mitra and T. Acharya, "Gesture Recognition: A Survey," IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, vol. 37, no. 3, pp. 311-324, May 2007.

[MohIee09] Mohan, A.R. and Sudha, N. (2009) 'Fast face detection using boosted eigenfaces', IEEE Symposium on Industrial Electronics & Applications, Kuala Lumpur, Malaysia, Vol. 2, pp.1002–1006.

[MorDip10] A. Mordelová, Testovanie zrozumiteľnosti slovenského posunkového jazyka vo videokonferencii, Diploma Thesis, Bratislava, 2010.

[MorICS05] Moraru, D., Ben,M. and Gravier, G.: 2005, "Experiments on speaker tracking and segmentation in radio broadcast news," Proc. International Conference on Speech and Language Processing, Lisbon, Portugal.

[MP7] MPEG7 metadata standard: http://mpeg.chiariglione.org/standards/mpeg-7/.

[MSGroup] Microsoft - Understanding the Group Policy Feature Set, http://technet.microsoft.com/en-us/library/bb742376.aspx

[MurJIT09] G. R. S. Murthy, R. S. Jadon, "A Review of Vision Based Hand Gestures Recognition," International Journal of Information Technology and Knowledge Management, vol. 2, no. 2, pp. 405-410, July-December 2009.

[NarMIT97] Nardi, B. (1997). Studying context: a comparison of activity theory, situated action models, and distributed cognition. Context and consciousness: activity theory and human-computer interaction, MIT Press, p35-52.

[NETFLIX] Netflix – http://www.netflix.com.

[NouPIn05] Nouza, J.; Zdansky, J.; David, P.; Cerva, P.; Kolorenc, J. &Nejedlova, D. (2005).Fully Automated System for Czech Spoken Broadcast Transcription with Very Large (300K+) Lexicon. Proceedings of Interspeech 2005, pp. 1681-1684, ISSN 1018-4074, Lisboa, Portugal, September, 2005,.

[OpenID] Open-ID, specifications, http://openid.net/developers/specs/.

[OpenNI] [Online] http://www.openni.org/.

[OsuIcc97] Osuna, E., Freund, R. and Girosi, F. (1997) 'Training support vector machines: an application to face detection'. Proc. of Int. Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, pp.130–137.

[P3P] Platform for Privacy Preferences (P3P) Project , http://www.w3.org/P3P/, W3C.

[PapAai05] M. Papegelis and D. Plexousakis. Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents. Engineering Applications of Artificial Intelligence, Volume 8, pages 781-789, 2005.

[PapNCS05] Papagelis, M., Rousidis, I., Plexousakis, D., Theoharopoulos, E.: Incremental collaborative filtering for highly-scalable recommendation algorithms. In: M.S.

Hacid, N.V. Murray, Z.W. Ras, S. Tsumoto (eds.) ISMIS, Lecture Notes in Computer Science, vol. 3488, pp. 553–561. Springer (2005).

[PavElm10] Pavlovičová, J., Oravec, M., Osadský, M.: An Application of Gabor Filters for Texture Classification. In: Proceedings ELMAR-2010 : 52nd International Symposium ELMAR-2010. Zadar, Croatia, 15.-17.9.2010. - Zadar : Croatian Society Electronics in Marine, 2010. - ISBN 978-953-7044-11-4. - S. 23-25.

[PazMAL97] Pazzani, M. J. & Billsus, D. (1997). Learning and revising user profile: The identification of interesting web sites. Machine Learning, 27, 313-331.

[PolSAC05] H. Polat and W. Du. SVD-based collaborative filtering with privacy. In SAC '05: Proceedings of the 2005 ACM symposium on Applied computing, pages 791–795, New York, NY, USA, 2005. ACM Press.

[QueVir94] F. K. H.Quek, "Toward a Vision-Based Hand Gesture Interface," in Virtual Reality Software and Technology Conference, Singapore, 1994.

[QuiTAI10] Quijano-Sánchez, L., Recio-García, J.A., Díaz-Agudo, B.: Personality and Social Trust in Group Recommendations, 22nd IEEE International Conference on Tools with Artificial Intelligence (ICTAI), 2010.

[RabIEE89] Rabiner, L. R. "A tutorial on hidden markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, 1989, pp. 257–286.

[RabBFS93] Rabiner, L. & Juan, B. (1993). Fundamentals of speech recognition, ISBN 0-13-015157-2, Prentice Hall PTR, New Yersey.

[RamICC01] Ramakrishnan, N., B. J. Keller, B. J. Mirza, A. Y. Grama, and G. Karypis. Privacy Risks in Recommender Systems. IEEE Internet Computing, 5(6):54-62, 2001.

[RicRSH10] Ricci, F., Rokach, L., Shapira, B., Kantor, P. (Eds.), Recommender Systems Handbook, Springer Science+Business Media, Dordrecht, Netherlands, 2010.

[RicRSH10a] Francesco Ricci, Lior Rokach, Bracha Shapira and Paul B. Kantor. Recommender Systems Handbook. ISBN 978-0-387-85819-7, Springer New York, 2010.

[RicRSH11a] Francesco Ricci, Lior Rokach and Bracha Shapira: Chapter 1 - Introduction to Recommender Systems, in Ricci et al, Recommender Systems Handbook, Springer, 2011.

[RogIST04] Rogers, Y. (2004). New theoretical approaches for HCI. Annual Review of Information Science and Technology, No. 38.

[RouICA06] Rougui, J., Rziza, M., Aboutajdine, D., Gelgon, M. and Martinez, J.: 2006, "Fast incremental clustering of gaussian mixture speaker models for scaling up retrieval in on-line broadcast," Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Toulouse,France.

[RozIWS11] Rozinaj, Gregor - Rybarova, Renata: "Learning Methods for Speech Synthesizer" In: IWSSIP 2011 18th International Conference on Systems, Signals and Image Processing June 16-18, 2011. Sarajevo, Bosnia and Herzegovina.

[RozRED11] Rozinaj, Gregor - Grenčík, Robert - Hajdu, Lukas - Hlavatý, Mario., - Hluzin, Martin - Hollý, Patrik: "SIP Protocol Based Intelligent Speech Communication Interface" In: 5th International Workshop on Multimedia and Signal Processing - Redžúr 2011, Bratislava, Slovak Republic. 12 May 2011. - Bratislava : STU v Bratislave FEI, 2011. - pp. 61-64, ISBN 978-80-227-3506-3.

[RybJDC11] Rybarova, Renata - Rozinaj, Gregor: "Intelligent Speech Synthesizer" In: International Journal of Digital Content Technology and its Applications (JDCTA), published by AICIT (Advanced Institute of Convergence Information Technology), (accepted), (8 pages), ISSN : 2233-9310 (Online), ISSN : 1975-9339.

[SalCHF99] D. Salber, A. Dey, G. Abowd, "The Context Toolkit: Aiding the Development of Context-Enabled Applications", Proceedings of the Conference in Human Factory in Computing Systems (CHI), 1999, pp. 434-441.

[SamYou12] SamsungTomorrow, "[CES 2012] Full Version: Samsung Press Conference," SamsungTomorrow, 10-01-2012.[Online]. Available: http://www.youtube.com/watch?v=PSp3fY_4bqY.[Accessed 11 1 2012].

[SarCIT02] Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Incremental singular value decomposition algorithms for highly scalable recommender systems. In: Proceedings of the 5th International Conference in Computers and Information Technology (2002).

[SarIST05] Sarwar, B.M., Konstan, J.A., Riedl, J.: Distributed recommender systems for internet commerce. In: M. Khosrow-Pour (ed.) Encyclopedia of Information Science and Technology (II), pp. 907–911. Idea Group (2005).

[SawWWW01] B. Sawar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web (WWW '01), pages 285-295, New York, NY, 2001. ACM.

[SchAOS78] Schwarz, G.: 1978, Estimating the dimension of a model, The Annals of Statistics 6, 461–464.

[SchDMK01] Schafer, J. B., J. A. Konstan, and J. Riedl. E-commerce recommendation applications. Data Mining and Knowledge Discovery, 5(1/2):115-153, 2001.

[SchECR05] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock, Croc: A new evaluation criterion for recommender systems: World wide web electronic commerce, security and privacy (guest editors: Mary ellen zurko and amy greenwald), Electronic Commerce Research 5 (2005), no. 1, 51+.

[SchSIG02] A. Schein, A. Popescul, L. Ungar, and D. Pennock, Methods and metrics for cold-start recommendations, Proceedings of the 25th Annual International

ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR2002), 2002, pp. 253–260.

[ShaPHF95] Shardanand, U. & Maes, P. (1995). Social information filtering: algorithms for automated "Word of Mouth". In Proceedings of Human factors in computing systems 1995 (pp. 210- 17). New York: ACM Press.

[Shib] Shibboleth, http://shibboleth.internet2.edu/.

[ShoCRS09] R. Shokri, P. Pedarsani, G. Theodorakopoulos, and J.-P. Hubaux. Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. In RecSys '09: Proceedings of the third ACM conference on Recommender systems, pages 157–164, New York, NY, USA, 2009. ACM.

[SmyPTL00] Smyth, B. & Cotter, P. (2000). A personalised TV listings service for the digital TV age. Knowledge-Based Systems, 13, 53-59.

[SobSig98] Sobottka, K. and Pitas, I. (1998) 'A novel method for automatic face segmentation, facial feature extraction and tracking', Signal Processing: Image Comm., Vol. 12, No. 3, pp.263–281.

[SucCUP87] Suchman, L. (1987). Plans and situated actions: the problem of human-machine communication. Cambridge University Press.

[SurIcc11] Suri, P.K., Walia, E. and Verma, E.A. (2011) 'Novel face detection using Gabor filter bank with variable threshold' Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on, vol., no., pp.715-720.

[TanThi02] F. Tang, B. Deng, "Facial Expression Recognition using AAM and Local Facial Features", 978-0-7695-2875-5, Third International Conference on Natural Computation, 2007.

[TerHCI01] L. Terveen and W. Hill, Beyond recommender systems: Helping people help each other, HCI in the New Millennium (J. Carroll, ed.), Addison Wesley, 2001.

[TinRSH11] Nava Tintarev and Judith Masthoff: Designing and Evaluating Explanations for Recommender Systems, Chapter 15 of Ricci et al, Recommender Systems Handbook, Springer, 2011.

[TIV09] TiVo recommendation system: http://www.tivo.com.

[TOR] TOR, http://en.wikipedia.org/wiki/Tor_(anonymity_network).

[TösKDD08] Andreas Töscher, Michael Jahrer, and Robert Legenstein. Improved neighborhood-based algorithms for large-scale recommender systems. In Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netix Prize Competition, NETFLIX '08, pages 4:1-4:6, New York, NY, USA, 2008. ACM.

[TotELM11] Tóth, Ján - Kondelová, Anna - Rozinaj, Gregor: "Natural Language Processing of Abbreviations." In: Proceedings ELMAR-2011 : 53rd International Symposium ELMAR-2011,14-16 September 2011, Zadar, Croatia. - Zadar : Croatian Society Electronics in Marine, 2011. - ISBN 978-953-7044-12-1. - S. 225-228.

[TreTNN03] E. Trentin and M. Gori, “Robust combination of neural networks and hidden Markov models for speech recognition,” IEEE Trans. Neural Netw.,vol. 14, no. 6, pp. 1519–1531, Nov. 2003.

[TurELM08] Turi Nagy, Martin - Rozinaj, Gregor - CepkoJozef: "System for Prosodic Modification of Corpus Synthetized Slovak Speech", In: 50th International Symposium ELMAR-2008 focused on Mobile Multimedia, 10-12 September 2008, Zadar, Croatia.

[TurELM10] Turi Nagy, Martin – Rozinaj, Gregor: "Compression of a Slovak Speech Database Using Harmonic, Noise and Transient Model", Proc. of 52th International Symposium ELMAR-2010 focused on Mobile Multimedia, 15-17 September 2010, Zadar, Croatia.

[TVA] TV-Anytime metadata standard: http://tech.ebu.ch/tvanytime.

[VasELM11] Vasek, Matúš - Rozinaj, Gregor - Rybárová, Renáta: "Training Database Preparation for LTS Rules Training Process with Wagon." In: Proceedings ELMAR-2011 : 53rd International Symposium ELMAR-2011,14-16 September 2011, Zadar, Croatia. - Zadar : Croatian Society Electronics in Marine, 2011. - ISBN 978-953-7044-12-1. - S. 221-224.

[VasRED11] Vasek, Matus - Rozinaj, Gregor: "LTS Letter-specific Tree Rules" In: 5th International Workshop on Multimedia and Signal Processing - Redžúr 2011, Bratislava, Slovak Republic. 12 May 2011. - Bratislava : STU v Bratislave FEI, 2011. - pp. 85-88, ISBN 978-80-227-3506-3.

[VeuPAT11a] Thijs Veugen, Privacy friendly group recommendations, European patent application 11187404.6, TNO, 1 November 2011.

[VeuPAT11b] Thijs Veugen, Oskar van Deventer, Ray van Brandenburg, Weighted group recommendations, European patent application 11187406.1, TNO, 1 November 2011.

[W3cWeb12] The World Wide Web Consortium (W3C) standards.[Online]. Available: http://www.w3.org/.[Accessed 12 1 2012].

[WeiDim08] D. Weiss, et al., "A User Profile-based Personalization System for Digital Multimedia Content", 3rd International Conference on Digital Interactive Media in Entertainment and Arts, 2008, Athens, Greece.

[WeiSCA91] Weiser, M. (1991). The computer for the 21st century. Scientific American, Vol. 265, No. 3, p94-104.

[WenNET08] Z. Wen Z Recommendation system based on collaborative filtering, 2008.

[WikCSP11] Cold start problem of automated data modeling systems: http://en.wikipedia.org/wiki/Cold_start.

[XBM] XBMC Web page: http://xbmc.org.

[XBS] Scrapers in XBMC: http://wiki.xbmc.org/?title=Scrapers.

[XieDES04] Xie, B., Han, P., Yang, F., Shen, R.: An efficient neighbor searching scheme of distributed collaborative filtering on p2p overlay network. Database and Expert Systems Applications pp. 141–150 (2004).

[XinASS05] Xinwei Li; Hui Jiang; Chaojun Liu; , "Large margin HMMs for speech recognition,". in Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005, (ICASSP '05), vol.5, no., pp. v/513- v/516 Vol. 5, 18-23 March 2005

[YanMin09] Yan, X. and Xiao-Wei, Ch. (2009) 'Multiple Faces Detection Through Facial Features and Modified Bayesian Classifier' Multimedia Information Networking and Security, 2009. MINES '09. International Conference on , Vol.1, No., pp.73-77.

[YanWeb05] Yan Jun, Yin Bo, Wu xiaoru, Wang Ren-hua, Liu Qingfeng: "Overview of Chinese Speech Synthesis Markup Language" http://www.w3.org/2005/08/SSML/Papers/iFLYTech.pdf Retrieved 2010-02-17.

[YouTube] YouTube.com.

[ZhaCRS11] Liang Zhang, Deepak Agarwal, and Bee C. Chen. Generalizing matrix factorization through flexible regression priors. In Proceedings of the fifth ACM conference on Recommender systems, RecSys '11, pages 13-20. ACM, 2011.

[ZhaSIG07] Yi Zhang and Jonathan Koren, Efficient bayesian hierarchical user modeling for recommendation system, SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (New York, NY, USA), ACM, 2007, pp. 47–54.

[ZieACM05] Cai-Nicolas Ziegler, Sean M. Mcnee, Joseph A. Konstan, and Georg Lausen, Improving recommendation lists through topic diversification, WWW '05: Proceedings of the 14th international conference on World Wide Web (New York, NY, USA), ACM Press, 2005, pp. 22–32.

[ZieWWW05] Cai N. Ziegler, Sean Mcnee, Joseph Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In Proceedings of the 14th International World Wide Web Conference. ACM Press, 2005.

[ZuoSpi08] Zuo, F. and de With, P.H.N. (2008) 'Facial feature extraction by cascade of model-based algorithms', Signal Processing: Image Communication, Vol. 23, No. 3, pp.194–211.