Learning Pulse: a machine learning approach for predicting performance in self-regulated learning using multimodal data

Daniele Di Mitri
Daniele.Dimitri@ou.nl

Maren Scheffel
Maren.Scheffel@ou.nl

Hendrik Drachsler
Hendrik.Drachsler@ou.nl

Dirk Börner
Dirk.Boerner@ou.nl

Stefaan Ternier
Stefaan.Ternier@ou.nl

Marcus Specht
Marcus.Specht@ou.nl

Welten Institute, Research Centre for Learning, Teaching and Technology
Open University of the Netherlands
Valkenburgerweg 177, 6401 AT Heerlen

ABSTRACT
Learning Pulse explores whether using a machine learning approach on multimodal data such as heart rate, step count, weather condition and learning activity can be used to predict learning performance in self-regulated learning settings. An experiment was carried out lasting eight weeks involving PhD students as participants, each of them wearing a Fitbit HR wristband and having their application on their computer recorded during their learning and working activities throughout the day. A software infrastructure for collecting multimodal learning experiences was implemented. As part of this infrastructure a Data Processing Application was developed to pre-process, analyse and generate predictions to provide feedback to the users about their learning performance. Data from different sources were stored using the xAPI standard into a cloud-based Learning Record Store. The participants of the experiment were asked to rate their learning experience through an Activity Rating Tool indicating their perceived level of productivity, stress, challenge and abilities. These self-reported performance indicators were used as markers to train a Linear Mixed Effect Model to generate learner-specific predictions of the learning performance. We discuss the advantages and the limitations of the used approach, highlighting further development points.

Categories and Subject Descriptors
H.2.8 [Database Applications]: Data mining

Keywords
Learning Analytics, Biosensors, Wearable Enhanced Learning, Multimodal data, Machine Learning

1. INTRODUCTION
The permeation of digital technologies in learning is opening up interesting opportunities for educational research. Flipped classrooms, ubiquitous and mobile learning as other technology-enhanced paradigms of instruction are enabling new data-driven research practices. Mobile devices, social networks, online collaboration tools as well as other digital media are able to generate a digital ocean of data [9] which can be “explored” to find new patterns and insights. The opportunities that data opens up are unprecedented to educational researchers as they allow to analyse and understand aspects of learning and education which were difficult to grasp before.

The disruption lies primarily in how the evidence is gathered: “data collection is embedded, on-the-fly and ever-present” [5]. Collecting data is not enough to extract useful information: the data must be pre-processed, transformed, integrated with other sources, mined and interpreted. Reporting on historical raw data only does not bring, in most of the cases, added value to the final user. As Li points out [18] individuals are already exposed to so many data they risk to “draw” into data. What is instead more desirable is receiving support in-the-moment which can prescribe positive courses of action, especially for twenty-first century learners which need to orient themselves continuously in an ocean of information with very little guidance [13].

Machine learning and predictive modelling can play a major role in extracting high-level insights which can provide valuable support for learners. Such ability highly depends whether the attributes taken in consideration to describe the learning experiences (the Input space) are descriptive for the learning process, they carry enough information to be able to accurately predict a change in the learning performance (the Output space). The relation between these two dimensions is further described in section 3.1.

The standard data sources in the reviewed predictive applications are most of time Learning Management Systems (LMS) and the Student Information Systems. Looking only at clickstreams, keystrokes and LMS data alone gives a partial representation of the learning activity, which naturally occurs across several platforms [25]. Several authors have pointed out the need to explore data “beyond the LMS” [15] to be able to get more meaningful information of the learn-
ing process. We believe that an interesting alternative could be found in the Internet of Things (IoT) and sensor community. Schneider et al. [24] have listed 82 prototypes of sensors that can be applied for learning. The employment of IoT devices allows collecting real-time and multimodal data about the context of the learning experience.

These considerations have shaped the motivation for the Learning Pulse experiment. The challenges it seeks to answer are the following: (1) define a set of data sources “beyond the LMS”; (2) find an approach to couple multimodal data with individual learning performance; (3) design a system which collects and stores learning experience from different sensors in a cloud-based data store; (4) find a suitable data representation for machine learning; (5) identify a machine learning model for the collected multimodal data.

Learning Pulse’s main contribution to the Learning Analytics community consists in outlining the main steps for a new practice to design automated multimodal data collection to provide personalised feedback for learning with the ultimate aim to facilitate prediction and reflection, the two most relevant objectives of learning analytics [14]. This proposed practice borrows the modelling approach from the machine learning field and uses it to model, investigate and understand human learning.

2. RELATED WORK

Learning Pulse belongs to the cluster of Predictive Learning Analytics applications. The scope of this sub-field in Learning Analytics was framed by the American research institute Educause with a manifesto [10] reporting some example applications, including Purdue’s Signals [1] or the Student Success System (S3) by Desire To Learn (D2L) [12]. These applications rely solely on LMS data for predicting academic outcomes or student drop-outs. Learning Pulse goes beyond those Predictive Analytics Applications by using multimodal data from sensors to investigate the learning process.

The field of multimodal data was given more prominence in the last Conference Learning Analytics and Knowledge (LAK16) with the workshop Cross-LAK: learning analytics across physical and digital spaces [21]. The concept behind Learning Pulse was presented at the Cross-LAK workshop [8]. In this workshop, several topics were touched: data synchronisation [11], technology orchestration [20] or face to face collaboration settings [30].

With a mission similar to Learning Pulse, a data challenge workshop on Multimodal Learning Analytics (MLA16) took place at LAK16 for investigating learning happening on the physical or virtual world through multimodal data including speech, writing, sketching, facial expressions, hand gestures, object manipulation, tool use, artifact building.

Finally, there has been a paper by Pijeira Diaz et al. [22] who used a multimodal data for Computer Supported Collaborative Learning in a school setting. Although not focused on using machine learning, the link made with psychophysiology theory introduce a novel research question, i.e. the possibility to infer psychological states including cognitive, emotional and behavioural phenomena from physiological responses such as sweat regulation, heart beat or breath [4].

3. METHOD

The background exposed in the previous chapter has led to the formulation of an overarching research question:

How can we store, model and analyse multimodal data to predict performance in human learning? ((RQ-MAIN))

This main research question leads to three sub-questions:

(RQ1) Which architecture allows the collection and storage of multimodal data in a scalable and efficient way?

(RQ2) What is the best way to model multimodal data to apply supervise machine learning techniques?

(RQ3) Which machine learning model is able to produce learner specific predictions on multimodal data?

To further investigate these research questions, we designed the Learning Pulse experiment that involved nine PhD students as participants and generated a multimodal dataset of approximately ten thousands records.

3.1 Approach

While frameworks already exist for standard within-the-LMS Predictive Learning Analytics, e.g. the PAR Framework [27], there are no structured approaches to treat beyond-the-LMS’ data in the context of multimodal data. For this reason, in this work, a novel approach for predictive applications inspired by machine learning is proposed. The objective is to learn statistical models out of the learning experiences and outcomes. Using a mathematical formalism that corresponds to learning a function \( f \) in the equation \( y = f(X) \), where \( X \) is a vector containing the attributes of one learning experience which work as the input of the function and, \( y \) is a particular learning outcome.

By using such an approach, three elements need to be further clarified: (1) the scope of investigation (the learning context); (2) the attributes encompassed by multimodal data (the Input space); (3) the learning performance object of the predictions (the Output space).

3.1.1 Learning context

The learning context investigated is self-regulated learning (SRL) which is defined as “the active process whereby learners set goals for their learning and monitor, regulate, and control their cognition, motivation, and behaviour, guided and constrained by their goals and the contextual features of the environment” [23]. Self-regulated learners are able to monitor their learning activity by defining strategic goals and that drive them not only to academic success, but lead to an increased motivation and personal satisfaction [31]. There is an overarching difference between self-regulated and non-self-regulated learners: the former are generally more engaged with their learning activities and desire to improve their learning performance [9]. On the contrary, the latter are less experienced, they do not perceive the relevance of their learning program and for this reason need to be followed closer by a tutor.

3.1.2 Input space

Learning is a complex human process and its success depends on several endogenous (e.g. psychological states) and
exogenous factors (e.g. learning contexts). Defining the Input space consists of selecting the relevant attributes of the learning process and structuring them into a correct data representation. This modelling task is non-trivial: according to Wong [29] modern “seamless” learning encompasses up to ten different dimensions. In this project, two of them are of main interest: Space and Time. The Input space can be imagined as the sequence of events happening throughout the learning time across digital and physical environments as shown on the left of figure 1.

Learning in a digital space means “mediated by a digital medium” i.e. by technological devices like laptops, smartphones or tablets. Digital learning data are easier to collect as most of the digital tools leave traces of their use. On the contrary, learning happening in the physical space refers to the learning not mediated by digital technology, like ‘reading a book’ or ‘discussing with a peer’. Although the line between Digital and Physical gets blurred with the pervasiveness of technology, the bulk of the learning activities still happens offline and should be “projected into data” through a sensor based approach to be able to take advantage of those moments.

Time is also a relevant dimension: the data-driven approach works best whenever the data collection becomes continuous and unobtrusive for the learner. This requirement inevitably limits the scope of investigation only to tangible events whose values are easy to measure over time. If on the one hand, this constraint makes data collection easier as there is no need to employ time-consuming surveys and questionnaires, on the other hand, this approach does not make it possible to directly capture psychological states which manifest during the learning.

Besides spanning across physical and digital space, the Input space of Learning Pulse can be grouped into three layers as shown in figure 1: those are 1) Body encompassing physiological responses and physical activity, 2) Learning Activities 3) and Learning Context.

Figure 1: Bi-spatial and three-layered Input Space

3.1.3 Output space

The Output space of the prediction models corresponds to the range of possible learning performances. These outputs are crucial for the machine learning algorithms to distinguish between successful learning moments from the unsuccessful ones. As self-regulated learners decide on their own learning goals and required learning activities, we need performance indicators which go beyond common course grades.

An interesting approach to measure learning productivity is the concept of Flow theorised by the Hungarian psychologist Csikszentmihalyi. The Flow is a mental state of operation that individuals experience whenever they are immersed in a state of energised focus, enjoyment and full involvement with their current activity. Being in the Flow means feeling in complete absorption with the current activity and being fed by intrinsic motivation rather than extrinsic rewards [6].

In the model theorised by Csikszentmihalyi depicted in figure 2, the Flow naturally occurs whenever there is a balance between the level of difficulty of the task (the challenge level is high) and the level of preparation of the individual for the given activity (the abilities are high).

Figure 2: Csikszentmihalyi’s Flow model

To measure the Flow we applied experience sampling [17]: the participants reported about their self-perceived learning performance. As self-assessment is strictly subjective it has the advantage to be exclusively based on the learner’s personal feelings. If carefully designed, self-assessment can lead to models tailored on personal dispositions. This brings clear advantage in the context of self-regulated learning: what is perceived as good (or productive, stressful etc.) is classified as such, meaning that what is good is only what the learner thinks is good.

3.2 Participants and Tasks

The experiment took place at the Welten Institute of the Open University of the Netherlands involving nine doctoral students as participants, five males and four females, aged between 25 and 35 with a background in different disciplines including computer science, psychology and learning science. PhD students are good self-regulated learners, as they are generally experienced learners and have strong engagement and motivation with their tasks.

All participants were provided with a Fitbit HR wristband and installed the tracking software on their laptops. As sensitive data were collected, every participant signed an informed consent. In addition, to ensure their privacy, their personal data were anonymised making use of the alias ARLearn plus an ID between 1 and 9.

The experimental task requested from the study participants was to continue their typical research activity throughout the day: the only additional action consisted in rating their learning activity every working hour between 7AM and 7PM (for the amount of hours they worked) through the Activity Rating Tool (described in sec. 3.4.1).

The actual experiment lasted for eight weeks and consisted of three phases: 0) Pre-test, 1) Training and 2) Validation. Phase 0: Pre-test. System infrastructure was tested in all its functionalities. A presentation was rolled out to intro-
duce the experimental setting and the study’s rationale to the participants. Participants were instructed to set-up the data collection software on their laptop as well as the fitness wristband.

**Phase 1: Training.** The first phase of the experiment lasted three weeks and consisted of the rating collection: participants have rated their activities hourly. The only visualisation they could see at that point were the ratings during that day. The first phase was named training because the collected data and ratings were necessary to train the predictive models.

**Phase 2: Validation.** After two weeks of break, the second phase started lasting for another two weeks. In the Validation phase, the activity rating collection continued in a Learner Dashboard visualisation. The second phase was called Validation as its purpose was to compare the predicted Performance indicators with the actual rated ones and to determine the prediction error.

### 3.3 Data sources

#### 3.3.1 Biosensors

The physiological responses and physical activity (Biosensor data for short) in this study are represented by heart rate and step count respectively. The approach used to track these “bodily changes” consisted in making use of wearable sensors. The decision of the most suitable wearable tracker was dictated by following criteria: 1) heart rate tracking sensor; 2) price per single device; 3) accuracy and reliability of the measurements; 4) comfort and unobtrusiveness; 5) openness of the APIs and data for analysis.

The choice converged to Fitbit Charge HR\(^2\): standing out on the cost-quality trade off, Fitbit HR complied with all the requirements, in particular by offering open access to the collected data through the Fitbit API. Such way of accessing data was beneficial on the one hand, as the software application developed for the project had to communicate exclusively with the Fitbit cloud datastore - while being agnostic to sensor trackers and their interfaces. The downside on the other hand was the dependence to the API specifications: the maximum level of detail available was a heart rate value update every five seconds and step count update every minute.

It is relevant to point out the difference of the heart rate and step count signals: while the heart rate values are a continuous time-series, also called fixed event, the number of steps per minute is a random event as it represents a voluntary human activity and not an involuntary process as the heart beat. The value of step count at one time point is not dependent on the previous ones (i.e. is random) while the heart rate value at time \(t\) surely depends on the value at time \(t-1\).

#### 3.3.2 Learning Activities

To monitor self-directed learning we decided to track PhD students’ activities on their laptops, being those the main learning medium in which they perform their PhD activities. Given the variety of learning tasks executed by the participants during the experiment, the actual learning happens across different platforms including software applications, websites, web tools. To capture and represent this heterogeneous complex of digital activities a software tracking tool was installed on the working laptop of the participants. The idea is that the use of a particular software or application adds up a valuable piece of information to consider when abstracting the learning process.

The tool chosen to monitor working efficiency was RescueTime, a time management software tool. RescueTime stores every five minutes (maximum level of detail allowed by its API specifications) into a proprietary cloud database an array containing the applications in use by the learner, weighted by their duration in seconds. Each activity in one interval has an activity ID and duration in seconds. The duration ranges between 1 and 300 (max seconds in five minutes), as the zero valued entries are the applications not used in an interval.

Given the diversity of research topics and learning tasks there is a high intersubject difference on the set of applications used during the learning experience; apart from a few common applications, the majority of applications used are very sparse. To mitigate this problem applications were grouped into categories by hand. The name of the categories chosen were: 1) Browsing, 2) Communicate and Schedule, 3) Develop and Code, 4) Write and Compose, 5) Read and Consume, 6) Reference Tools, 7) Utilities, 8) Miscellaneous, 9)Internal Open Universiteit, 10) Sound and Music.

In figure 3, the distribution of the applications is compared with their categories. The height of the bars represents the number of executions that application had during the experiment, which equals to the presence of that application in one of the five-minute intervals. While in the left-hand chart the long tail effect due to the sparsity is quite noticeable, on the right hand side that does not appear.

![Figure 3: Plots showing the number of executions per Applications (left), per Application category (right).](https://www.fitbit.com/chargehr)

#### 3.3.3 Performance indicators

The indicators used in Learning Pulse are four: Stress, Productivity, Challenge and Abilities. The four indicators were collected with the following questions.

1. **Stress:** how stressful was the main activity in this time frame?
2. **Productivity:** how productive was the main activity in this time frame?
3. **Challenge:** how challenging was the main activity in this time frame?
4. **Abilities:** how prepared did you feel in the main activity in this time frame?

Each participant had to rate each of these indicators retroactively with respect to the main activity performed in the
time frame being rated. The participants were expected to answer these questions at the end of every working hour from 7AM to 7PM using for each of them a slider in the Activity Rating Tool described in section 3.4.1 which translated the rating into an integer ranging from 0 to 100.

The Flow

The Flow is operationalised through a single numerical indicator calculated based on the Challenge and Abilities indicators, as indicated by formula 1. $i$ identifies a specific learner, while $j$ references a specific time frame. $F_{ij}$ is the Flow score for the learner $i^{th}$ at the time frame $j^{th}$, $A_{ij}$ and $C_{ij}$ is the level of Abilities and Challenge rated by the learner $i^{th}$ at the time frame $j^{th}$.

$$F_{ij} = (1 - |A_{ij} - C_{ij}|) + \frac{|A_{ij} + C_{ij}|}{2}$$  \hspace{1cm} (1)

Figure 4 plots the ratings of all the participants throughout the whole experiment in a two-dimensional space, where the x-axis are the level of Abilities and the y-axis is the level of Challenge. Both indicators are expressed as percentages. The dots in the scatter plot are coloured depending on their Flow-value calculated with the formula 1.

The colour scale used for the Flow goes from red over yellow to green recalling the metaphor of a traffic light: high Flow values are green, medium ones are yellow and low Flow values are red. The plot visualises how the formula 1 works. The Flow is higher if two conditions apply: 1) the difference between Abilities and Challenge is small, meaning they are close to line $x = y$; 2) the mean between Abilities and Challenge is close to one, meaning the observation falls into the top-right corner of the plot, which corresponds to the Flow zone, as in the original definition of Flow (see figure 2).

Besides the four questions also the Activity Type was sampled along with the GPS coordinates. The Activity Type was a categorical integer representing the following labels 1) Reading, 2) Writing, 3) Meeting, 4) Communicating, 5) Other.

The rationale behind this labelling was to have a hint on the nature of the main learning task executed during that time frame. Finally, the GPS coordinates consisted of two floating points which are the latitude and longitude of the location where the rating was submitted with the Activity Rating Tool.

Figure 5 shows the ratings of the four indicators of one participant during one day of the experiment, as well as the calculated Flow indicator. The background colours represent the different activity types, as the legend visually indicates.

3.3.4 Environmental context

The third data source is made up by the surrounding context of learning as the environment might also have an impact on the final learning outcomes. The ideal solution would be to track information about the indoor surrounding environment, such as measuring the light intensity, humidity and heat inside the office, thus combining these with the information about the weather.

Given the lack of adequate sensors to employ in the office environment, only the outdoor weather conditions were monitored. For each participant, the GPS coordinates were stored that allowed to call the weather data API through the online service OpenWeatherMap and to store weather data specific to the location from where each participant was operating. The weather API was called automatically every ten minutes for each of the nine participants. The attributes extracted from these statements were 1) Temperature, 2) Pressure, 3) Precipitation, 4) Weather Type, with the first three being floating points while the latter is a categorical integer.

3.4 Architecture

Combining different Data Sources into a central data store and processing them in real time is not a trivial task. Figure 6 presents a transversal view of the system architecture which is divided into three layers.

At the top level, the Application Layer groups all the services that the end-user interfaces with including the Fitbit wristband and the RescueTime application here referred as Third Party Sensors. The Activity Rating Tool (ART) belongs to the same level.

The middle level is the Controllers Layer which gathers the back-end components of the Applications. In this layer,
as figure 6 shows, the software is running on two server infrastructures: the Cloud and the Virtual Machine. Not reported here are the controllers of the Third Party Sensors and the Learner Dashboard as the System Architecture described here is agnostic towards their implementation. On the Cloud side, there are the Learning Pulse Server, a scripting software responsible for importing data from different APIs and storing them into the Learning Record Store. In addition, also running on the Cloud, there is the server software of the Activity Rating Tool which connects the client user interface with the database. The scripting software running on the Virtual Machine is the Data Processing Server, which as the name indicates, implements the post-processing operations including data transformation, model fitting and predictions.

The lowest level is the Data layer. While the Third Party Services use their own APIs which receive regular queries by the importers of the Learning Pulse Server, the main datastore is the Learning Record Store. Consisting of a Fact Table and a Big Query Index, the Learning Record Store is the cloud-based database which collects the data about the learning experience of all participants. It also runs on the Cloud infrastructure and is further described in section 3.4.2.

Even though they are not directly part of the Learning Record Store, also the results of the Data Processing server are pushed into a datastore which is also shown in the Data Layer. This datastore is developed with a non-relational database and collects the predictions (also referred as forecasts) and the transformed representation of the historical data, namely the learning experience data in the Learning Record Store opportune processed and transformed. Finally, the Data Processing Server makes use of further persistent data, as for example the Learners’ Models, which are stored locally, reused constantly and regenerated once a day.

3.4.1 Activity Rating Tool

Responsible for collecting the participants’ ratings about their learning experience, designed and developed as a scalable web application, the Activity Rating Tool runs App Engine using webapp2 lightweight Python web framework. While the back-end was written in pure Python, the front-end uses Bootstrap.

The interface of the tool was designed to be as intuitive as possible and with the aim to make the rating action quick and easy for the participants considering they needed to use it several times a day. Figure 7 shows two screenshots of the application’s main page; on left-hand side, it shows the list of all the past time frames between 7AM and the hour previous to the current. To rate a time frame the form shown on the right-hand side of figure 7 opened. There users are asked to

---

3http://getbootstrap.com/
select the Activity Type through five different icons; below, users can input the rating for the four indicators through four sliders, differently coloured for each indicator. Once the desired values are chosen, the sliders translate the position of the slide into an integer between 0 and 100. To prioritise straightforwardness and to avoid information overload, the guiding questions were hidden into a help tool-tip at the right-hand side of the sliders.

Once the participant pressed “Submit” the time frame turned green coloured in the time frame list. The participant could also delete ratings or resubmit in case of errors. Additionally, a Daily Rating Plot is shown just before the “Submit” button which shows the past ratings recorded that day with the purpose of reminding the participant their previous ratings that day in order to support a coherent overall rating.

### 3.4.2 Learning Pulse Server

The Learning Pulse Server is the script component responsible for pulling the data from the third party APIs and transforming them into learning records and handing out their identifiers. The learning records are first stored into the Fact Table by assigning a UUID (Universally Unique Identifiers). The Learning Pulse Server script and the Fact Table were implemented as application and data store in the Cloud, which allowed to balance the load of data on a distributed architecture for scalability purposes. From the Fact Table, the data were synchronised into a Query Index, implemented with a scalable non-relational database, which contrarily to the Fact Table, allowed to query the distributed learning statements with SQL language. The synchronisation between the Fact Table and the Query Index happens using a queue, such that no learning record could get lost.

While the Learning Pulse Server is the application script responsible for pushing and pulling the learning records, the Fact Table and the Query Index together form the LRS. Implementing the LRS with a cloud-based solution allowed to achieve properties such as (1) high availability: the LRS could be reached at any time, with respect to the privileges of the client; (2) high scalability: although the size of the data collected was about 1 Gigabyte the number of learning statements could easily scale up tens or even hundreds of times more; (3) high reliability: the cloud infrastructure chosen provided performance and security.

### 3.4.3 Experience API

The chosen data format for the learning records was the Experience API (or xAPI) data standard, an open source API language through which systems send learning information to the LRS. XAPI is a RESTful web service, with a flexible standard which aims at interoperability across systems. The XAPI standard has the format actor-verb-object and are generated and exchanged in a JSON format, opportunely validated by and stored in the LRS. The main advantage of using xAPI is interoperability: learning data from any system or resource can be captured and eventually queried by the third party authenticated services. For each event captured in Learning Pulse, an xAPI statement template was designed following the Dutch xAPI specification for learning activities [2].

[A list of the statements can be found here](http://bit.ly/DutchXAPIreg)

### 3.5 Data processing

After being stored in the LRS, learning records were processed, transformed and mined in order to generate predictions to be shown to the learners. Data collection and Data processing can be seen as two legs which walk side by side, complementing each other’s role. The data processing software was named Data Processing Application (DPA) and its main responsibilities consisted in (1) fetching the data from the Learning Record Store; (2) transforming the new data by time resampling and features extraction; (3) learning and exploiting different regression models; and (4) storing the results of the regression.

The DPA needed to run continuously on a server always-on without the need of human interaction. Other important requirements for the DPA were the possible integration with other software components (e.g., interfacing with the LRS) and availability of statistical and Machine Learning tools. The final choice converged on using Python as the main programming environment, mainly because of its flexibility and wide support for data analysis.

**Figure 8: The data processing workflow**

For the Data Processing Server, namely the computer infrastructure which hosted the DPA, cloud options were considered including popular cloud IaaS solutions. For financial reasons, the choice directed towards an in-house server solution constituting of a Virtual Machine running an OpenSuse Linux distribution.

The diagram in figure 8 shows the data processing workflow, a close-up of the system architecture shown in section 3.4. The figure is divided into three layers: the controllers, the data and the visualisations.

#### 3.5.1 Data fetching

A cron-job on the Virtual Machine activated the scheduler every ten minutes, every working day, from 7AM to 7PM. The main task of the scheduler was to query the Learning Record Store and to realise whether new intervals could be formed based on the learning records retrieved. In order to be valid, the learning intervals have to be completed for Biosensor, Activity and Weather data. If any of these data are not available, the execution of the Data Processing Application is interrupted and postponed to the next round. To connect to the Learning Record Store, the DPA uses Pandas’ Big Query connector. This interface can authenticate

---


6The source code of the Data Processing Application is available at [https://github.com/WELTEN/learning-pulse-python-app](https://github.com/WELTEN/learning-pulse-python-app)
the client (the DPA Python script) to the Big Query service, submit a query and fetch the results that are returned into a data frame, the popular data format for structuring tabular data in Pandas.

3.5.2 Multi-instance representation

Each data source had its own frequency of data generation: the ratings were submitted every hour, the heart rate was updated every five seconds, the step count every minute, the activities every five minutes and the weather every ten minutes. That resulted in the so-called relational representation as for each participant a different number of relations corresponded with all the other entities depending on how frequent their values were updated. Relational representations are not ideal for machine learning as the input space which needs to be examined can become very broad [7].

The problem was therefore translated into a multi-instance representation where each training sample is a fixed length time interval. The interval length is determined by how frequently the labels i.e. the ratings, are updated. As the ratings here equal the working hours (say 8 hours), if multiplied by the experiment days (say 15), that would result in the best-case scenario of 120 samples for each participant, which is too small in size for a training set. To overcome this problem the compromise was found selecting 5 minutes long intervals. This decision, however, triggered another problem, what to do with those attributes that are updated more or less frequently. The approach used was different for each entity. Ratings, which are updated hourly, were linearly interpolated; the step count, which is updated every minute, was aggregated with a sum function; the weather, which was updated every 10 minutes, was copied backwards; the activities came already with a five minutes frequency, therefore no action was required. Finally, to represent a five minutes heart rate signal into one or more features, the best solution was to use different aggregate functions, namely: 1) the minimum of the signal, 2) the maximum, 3) the mean, 4) the standard deviation and 5) the average change - i.e. the mean of the absolute value of the difference between two consequent data points. This naive approach consists in plugging in several different features and letting the machine learning algorithm decide which ones are the most influential on predicting the output. It is, however, useful to point out that more sophisticated techniques for feature extraction from the heart rate exist, such as the Heart Rate Variability [28] or the Sample Entropy.

3.5.3 Data storing

Similarly to the data collection, also the data processing had to be the same. In order not to repeat the processing step of the same data multiple times, it was convenient to store the results of the transformation in a permanent data store, to be able to retrieve it when necessary. To do so a Big Query table was created called History: the name was used to differentiate the transformed historical data with the forecast about the future, whose table is called Forecasts. The Big Query was preferred over other solutions since the LRS was developed with the same technology. In addition, Pandas offers an easy Big Query interface, which allows to push and pull data easily from the Cloud Database.

3.6 Regression approach

As the collected data were longitudinal, the fixed effects showed stochastic behaviour implying that the observations were highly dependent on one another. In formal terms, this means that observing the behaviour of one participant at time $t$, the output variable $y_t$ is described by the equation $y_t = \alpha + \beta X_t + \epsilon_t$. The dependence among the samples means that given a later observation at time $t + 1$, the covariance $\text{cov}(\epsilon_t, \epsilon_{t+1}) \neq 0$ with $t \neq t + 1$.

As the samples were intercorrelated it was not possible to employ common regression models, as most of these techniques assume that the residuals are independent and identically distributed normal random variables. Treating correlated data as if they were independent can yield wrong $p$-values and incorrect confidence intervals. To overcome this problem the approach chosen was to Linear Mixed Effect Models (LMEM).

LMEM relax the dependency constraint of the data and they can both treat data of mixed nature, including fixed and random effects, plus they describe the variations of the response variables with respect to the predictor variables with coefficients that can vary for each subject [19]. In formal terms, the LMEM as described by [16] consist in a $n_i$-dimensional vector $y$ for the $i$-th subject:

$$ y_i = X_i \beta + Z_i \gamma_i + \epsilon_i, i = 1, ..., M; \gamma_i \sim N(0, \Sigma) $$

- $n_i$ is the number of samples for subject $i$
- $Y$ is a $n_i$ dimensional vector of response variables
- $X$ is a $n_i \times k_{fe}$ dimensional matrix of fixed effects coefficients
- $\beta$ is a $k_{fe}$-dimensional vector of fixed effects slopes
- $Z$ is a $n_i \times k_{re}$ dimensional matrix of random effects coefficients
- $\gamma$ is a $k_{re}$-dimensional random vector with mean zero and covariance matrix; each subject gets its own independent $\gamma$
- $\epsilon$ is a $n_i$-dimensional within-subject error with mean $0$ and variance $\Sigma^{2e}$ with a spherical Gaussian distribution.

4. ANALYSIS AND RESULTS

At the end of the experimental phase, the transformed dataset presented the following characteristics: a total of 9410 five-minute learning samples, counting for all nine participants. The biggest sample size was $ARLearn5$ with 1725 samples, while the one with the smallest number of samples was $ARLearn4$ with 514. There were 29 attributes in total.

As a single-output LMEM implementation was chosen, five different models were learnt each of them having as response variable one of the five performance indicators (Abilities, Challenge, Productivity, Stress and Flow). The models were initialised with the following parameters:

- Fixed Effects: timeframe, latitude, longitude, weatherConditionId, pressure, temp, humidity, hr_min, hr_avg, hr_mean, hr_std, hr_max
- Random Effects: Browsing, Communicate_Schedule, Develop_Code, Internal_OU, Miscellaneous, Read_Consume, Reference, Sound_Music, Utilities, Write_Compose, Steps

As the way of rating of each participant was different, the predicted values were normalised with respect to the learner-specific historical min and max using the following formula.
\[ x_{\text{new}} = \frac{(x_{\text{max}} - x_{\text{min}}) \cdot x_i + x_{\text{min}}}{100} \]

For the evaluation of the predicted results we used R-squared, a statistical measurement which scores how close the data are to the regression line and outputs a number from 0 and 1 which measures the goodness-of-fit of the model. The results obtained were the following: Stress: 0.32, Challenge: 0.22, Flow score: 0.16, Abilities: 0.08, Productivity: 0.05.

5. DISCUSSION

The first question (RQ1) focused on the best architectural setup to process multimodal data. The answer found to the question was satisfactory as architecture design discussed in section 3.4 was capable of: (1) importing a great number of learning statements from the sensors and their APIs; (2) feeding the statements into a cloud-based LRS avoiding collisions among them and information loss; (3) combining the statements with the self reports regularly provided by the learners; (4) programmatically transforming the learning statements by extracting relevant attributes and by resampling into uniform intervals; (5) fitting the predictive model on historical observations and saving for the reuse with the newer observations and (6) saving the predictions in a separate store to be able to compare with the actual values. On the other hand, the architectural design had some limitations. First of all, it exhibited a real-time syncing issue: the data synchronisation with the wearable trackers was slower than expected; in the best case scenario, the data about the heart rate and the steps were available in the LRS only 15 to 20 minutes later. Secondly, the Data Processing Server hosting the Data Processing Application was poor in performance: the weak processing power slowed down the data processing and that resulted in long job cycles.

The second research question (RQ2) was concerned with finding the best way to model multimodal data suitable for machine learning. The solution found was to treat the problem using a Multiple Instance Representation as detailed in section 3.5.2, i.e. using a tabular representation where each row represents a five minute learning interval and each column a different attribute. This representation helped to overcome the problems derived from the relational nature of the collected data. Additionally, third party APIs influenced a lot the type of data that is possible to be retrieved from the sensors. An example is the Fitbit Charge HR, whose API only allows to get values of the heart rate every five seconds and no inter-beat distance. This scarcity of available data did not allow to calculate useful measurements on the heart rate, like the Heart Rate Variability which has been proven to be a good predictor for workload stress [26].

The third research question (RQ3) asked which machine learning model for regression is best suited for the heterogeneous type of data. The solution discussed in section 3.6 consisted in using the Linear Mixed Effect Models as they allow (1) taking into account data specific to each learner; (2) distinguishing between fixed and random effects; (3) taking categorical data into account. Despite LMEM being the appropriate model for the intended task, the R-squared evaluation test yielded poor prediction accuracies for the five outputs. One possible reason might be the sparsity of random effects, especially those that refer to the least used activity categories (whose distribution is shown in figure 3). We observed that while adding up sparse attributes (random effects) as predictors decreases the prediction accuracy, fixed effects improve the general accuracy.

The answers to the three sub research questions provide an answer to the main research question (RQ-MAIN): a way to store, model and analyse multimodal data was successfully found. Nevertheless the limited significance of the prediction results does not allow us to assert that accurate and learner-specific predictions can be generated. This might have been caused by: 1) the combination of multimodal data selected in the experiment; 2) no clear learning task to be executed, high variance of the learning context explored; 3) sparse random effects were still too many as opposed to fixed effects.

6. CONCLUSIONS

This paper described Learning Pulse, an exploratory study whose aim was to use predictive modelling to generate timely predictions about learners’ performance during self-regulated learning by collecting multimodal data about their body, activity and context. Although the prediction accuracy with the data sources and experimental setup chosen in Learning Pulse led to modest results, all the research questions have been answered positively and have lead towards new insights on the storing, modelling and processing multimodal data.

We raise some of the unsolved challenges that can be considered a research agenda for future work in the field of Predictive Learning Analytics with “beyond-LMS” multimodal data. The ones identified are: 1) the number of self-reports vs unobtrusiveness; 2) the homogeneity of the learning task specifications; 3) the approach to model random effects; 4) alternative machine learning techniques.

There is a clear trade-off between the frequency of self-reports and the seamlessness of the data collection. The number of self-reports cannot be increased without worsening the quality of the learning process observed. On the other side, having a high number of labels is essential to make supervised machine learning work correctly.

In addition, a more robust way of modelling random effects must be found. The found solution to group them manually into categories is not scalabile. Learning is inevitably made up by random effects, i.e. by voluntary and unpredictable actions taken by the learners. The sequence of such events is also important and must be taken into account with appropriate models.

As an alternative to supervised learning techniques, also unsupervised methods can be investigated, as with those methods fine graining the data into small intervals does not generate problems with matching the corresponding labels also the amount of labels is no longer needed.

Regarding the experimental setup, it would be best to have a set of coherent learning tasks that the participants of the experiment need to accomplish, contrarily to as it was done in Learning Pulse, where the participants had completely different tasks, topics and working rhythms. It would be also useful to have a baseline group of participants, which do not have access to the visualisations while another group does have access; that would allow to see the difference of performance, whether there is an actual increase.

To conclude, Learning Pulse set the first steps towards a new and exciting research direction, the design and the development of predictive learning analytics systems exploiting multimodal data about the learners, their contexts and their
activities with the aim to predict their current learning state and thus being able to generate timely feedback for learning support.

Acknowledgements

The Learning Pulse project was partially funded by the Learning Analytics Community Exchange (LACE) project (grant no. 619424).

References