Recommendation System in an Integrated Digital Training Environment for the 5th Generation Air Force

Dirk Thijssen, Rik Bosma
Royal Netherlands Aerospace Centre (NLR)
Amsterdam
Dirk.Thijssen@nlr.nl, Rik.Bosma@nlr.nl

ABSTRACT

The Royal Netherlands Air Force (RNLAF) is currently transitioning to the 5th Generation Air Force. The transition involves changes in what is expected from personnel and will change the way of training. Therefore, the training strategy is moving towards a learning environment that is personalized, flexible and cost-efficient. Development of technology enables training organizations to gather learning data and train in various ways. However, integrating these new technologies make the future learning environment or ecosystem far more complex and introduce many other challenges. Currently, there is no framework on how to use learning analytics data to create a highly personalized learning environment. This paper describes a conceptual framework on how to create that mesh, how to determine the competency level of a learner and how to recommend the right learning tasks based on the learning needs. Next steps will include continued development and validation of the framework by application in various use cases.

ABOUT THE AUTHORS

Dirk Thijssen graduated from the Utrecht University as an Educational Scientist in 2020. Dirk has affinity with Data Science and Technology. He works as a Training Expert at the Royal Netherlands Aerospace Centre. His job is characterized by technology and innovation. He researches and applies the newest learning technologies to evolve training concepts, demonstrate their value and to improve training in aviation. His main focus is military aviation, specifically the Royal Netherlands Air Force. From time-to-time he sidesteps towards operational Concept, Development and Evaluation (CD&E)

Rik Bosma has a M.Sc. in Computer Science and worked on complex technical systems ranging from simulation environments to data science solutions. Rik has a passion for innovation, training and education in industrial and military environments. He works as R&D engineer at the Royal Netherlands Aerospace Center, where he researches and applies enabling technologies to create a future learning eco system for the Royal Netherlands Air Force.

Recommendation System in an Integrated Digital Training Environment for the 5th Generation Air Force

Dirk Thijssen, Rik Bosma
Royal Netherlands Aerospace Centre (NLR)
Amsterdam
Dirk.Thijssen@nlr.nl, Rik.Bosma@nlr.nl

INTRODUCTION

The Royal Netherlands Air Force is currently transitioning to the 5th Generation Air Force (Marchand, 2017; 2019). This is often regarded to new weapon systems (e.g. F-35, MQ-9 Reaper), but the workforce is not to be forgotten. The transition involves changes in what is expected from personnel and will change the way of training. Therefore, the training strategy is moving towards a learning environment that is personalized, flexible and cost-efficient. Advances in technology in the last decade make it possible to implement a distributed personalized training strategy. However, integrating these new technologies into a Learning EcoSystem is complex and introduces many challenges, including facilitating personalized training, collecting qualitative data and combining available data to create recommendations. An environment is needed that provides a mesh of a large variety of learning resources (e.g. learning content, experience data and technologies). This involves balanced training concepts (Walcutt & Schatz, 2019) as well as a complete infrastructure to form an Integrated Digital Training Environment.

The above challenges are examined in the IDTEAM (Integrated Digital Training Environment for Aircraft Maintenance) project funded by the Netherlands Ministry of Defense. One of the goals of this project is to develop a recommendation system to support an introduction course for aircraft maintenance technicians of the RNLAF. This paper describes the underlying recommendation framework and progressive scoring system, from both a technical and educational perspective. While this system was developed in the specific context of aircraft maintenance, it intends to be applicable to different contexts where performance tracking is relevant.

An ecosystem to support a recommendation framework relies on data. The Advanced Distributed Learning (ADL) Initiative established the Total Learning Architecture (TLA), which is a set of standards and specifications for capturing and managing such data in an ecosystem. These standards and specifications make up the core of an ecosystem. The recommendation system discussed in this paper is called an edge system, and uses data from this core to solve the problem of "how to recommend the next personalized learning experience to a learner". The solution focusses on a continuous method to monitor learning needs based on competencies, compare learning needs to activities and fit learning activities to learning needs.

A single integrated framework on how to quantify competencies based on a mesh of learning data, learning activity effects and flow of learning (Csikszentmihalyi, 1990) does not exist yet. In this paper a design of such an integrated framework, the Progressive Scoring System (PSS), is introduced. The first part goes further into the framework in regard to a Learning EcoSystem and the educational principles on which the system is based. The second part describes the steps taken by the recommender framework. The third part elaborates on these steps by describing the algorithms and dynamics of the system. At last, the fourth part discusses how the training concepts are integrated into the system and what is next for the future.

Total Learning Architecture

The recommendation framework in Figure 1 is designed based off the Advanced Distributed Learning (ADL) Initiative Total Learning Architecture (TLA) data strategy. ADL provides standards and data strategies, business rules, government rules and policies for using data at an enterprise Department of Defense-level (Smith et al, 2022). The standards are formally established by the IEEE (Institute of Electrical and Electronics Engineers). The IEEE is a professional association that develops, defines, and reviews electronics and computer science standards.

This framework implements the xAPI (IEEE 9274.1) data standard. However, the standards for sharable competency definitions (IEEE 1484.20.3), learning activity metadata (IEEE 1484.12.1) and the learner profile (IEEE 2997) have been out of scope for this design, because the aim of DTP IDTEAM is to design a system that can be integrated within the existing learning environment of the Netherlands Ministry of Defense.

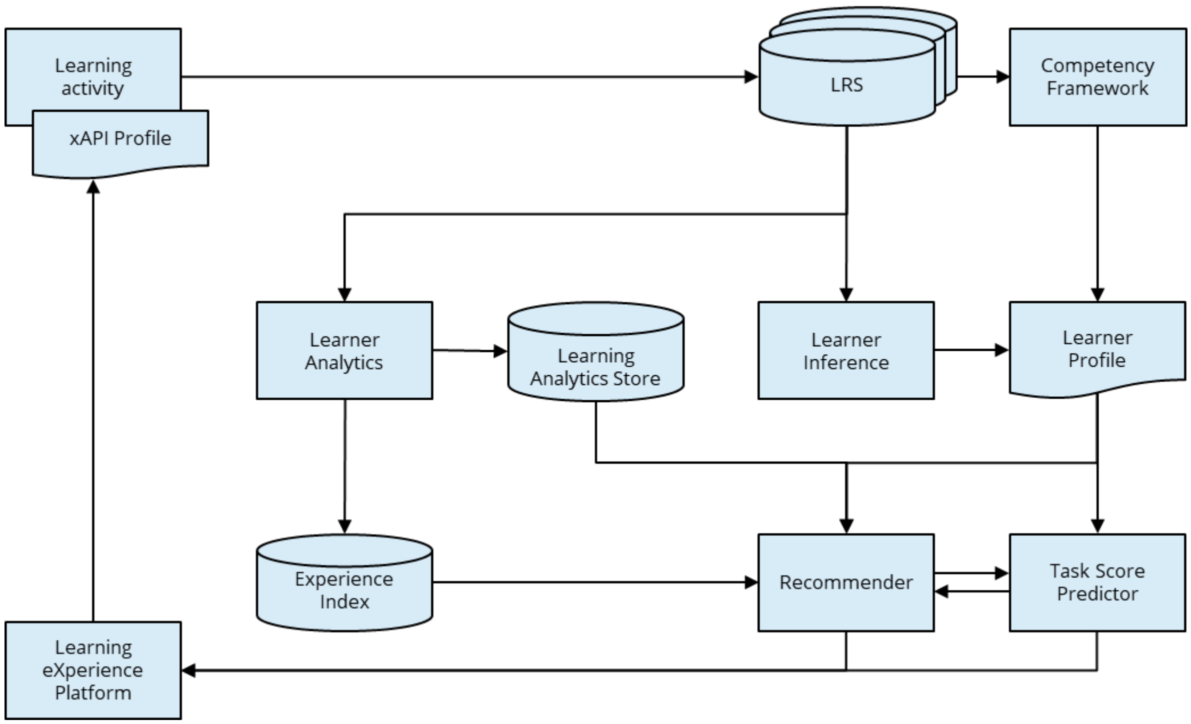

Figure 1. Reference architecture used by the recommendation framework

The TLA data strategy consists of multiple components, starting with a learning activity that is accompanied by a xAPI profile. A learning activity communicates learning data via the xAPI standard. What xAPI statements and in what order the statements are communicated is defined in a xAPI profile. The learning data is gathered in a learning record store (LRS) and further processed and filtered to other LRS instances.

Learning data from the LRS is used to perform two types of learning analytics. Learner analytics which is responsible for statistical analysis over a set of learners. Those results are stored in a learning analytics store. The second type is personal analytics, called learner inference. Learner inference provides the content to the learner profile.

The competency framework provides structure to the learner profile. A qualification profile is used, that consists of competencies that are defined by a set of performance indicators, to define this structure. The experience index is a collection of learning activities that are available in the Learning EcoSystem. These learning activities are being recommended to a learner. For each learning activity a difficulty level per competency is defined as well as a weight factor to define the influence of an experience on the competency.

The recommender selects competencies and learning activities to train the selected competencies, which are provided to the task score predictor. This task score predictor calculates how well each activity suits the learner's needs. Based on these scores the recommender provides a subset of learning activities to the learner via a learner centric dashboard, called a learning experience platform (LXP). Regarding the LXP two elements have been considered. First, presenting a learner's current competency level scores. What abstraction level of performance metrics is actually useful for a learner and whether to show progress or score considering the motivation of a learner. Secondly, several user interface concepts to present learning activity recommendations have been considered. However, the LXP is beyond the scope of this recommendation framework. It can be considered as a way to effectively present recommendations to the learner.

Educational Principles
The problem described in this paper is technological as well as educational in nature. The technology is used to find a solution, but the technology has to behave in in a way that suits the way humans learn. Therefore, three training concepts are selected which are highly relevant for this problem statement, which are explained in this paragraph.

To create a personalized recommendation, on which learning activity a learner should do next, the recommender design could incorporate competency development (Stafford, 2019), flow of learning and the zone of proximal development (Vygotsky, 1978; Wertsch, 1984). The training need of individuals learner is very dynamic because it changes over time after every learning task. However, training is mostly executed based on a fixed syllabus with a fixed order and amount of learning tasks. The training need should be viewed from a competency perspective in which competencies can progress. Therefore, quality of the learning task execution should indicate if the training need is fulfilled instead of checking if a learning task is performed regardless of the result. The idea of progression is complemented by the concept of competency development by Stafford (2019). Competency-based learning is based on four principles: (1) a domain or job context is divided in specific parts which are called competencies; (2) consistent indicators are used to describe the proficiency level; (3) the learner shows he mastered competencies by the assessment of an execution; and (4) the learning process and learner are at the core of the system (Stafford, 2019).

The learner can be challenged in its zone of proximal development, which lies beyond that what the learner has mastered and can execute independently. It is the zone in which the learner can execute a task with support and guidance to eventually master it (Vygotsky, 1978; Wertsch, 1984). The flow of learning describes that the learner needs to experience a challenge that does not bore or overload the learner with anxiety as consequence. The flow of learning described learning activities which can be performed independently (Csikszentmihalyi, 1990). The alteration between learning experiences in the zone of proximal development as well as experiences in the flow of learning is beneficial for the learning process (Vygotsky, 1978; Wertsch, 1984; Csikszentmihalyi, 1990). It could even be argued that the zone of proximal development (Vygotsky, 1978) is located in the upper brown zone of Figure 2 in between the flow channel and excessive overload (Csikszentmihalyi, 1990).

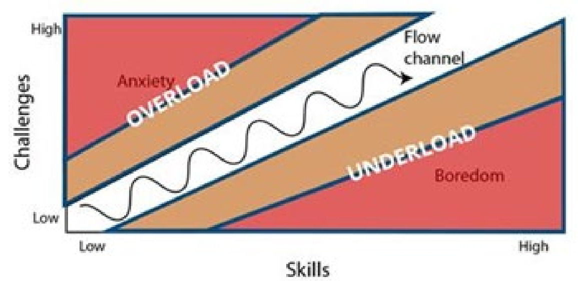

Figure 2. The flow of learning (Czikzentmihalyi, 1990)

RECOMMENDING LEARNING ACTIVITIES BASED ON FLOW

This paragraph conceptually describes the key concepts of the recommendation framework and the progressive scoring system.

Creating a recommendation
A personalized recommendation is made in four general steps (Figure 3). First, competencies requiring training are selected. Then, viable learning activities are selected and compared against the learner's current skillset. Based on this comparison, a prediction can be made how the learner performs on the viable learning activities. As a result, a selection of activities customized to the learner is produced.

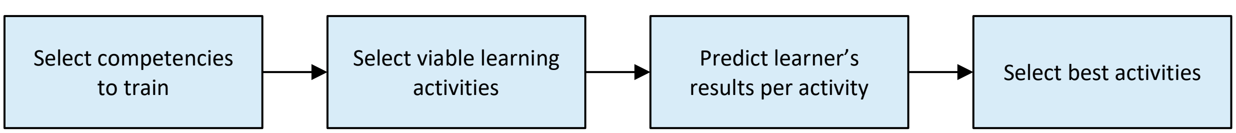

Figure 3. High level process of creating a recommendation

Quantifying and comparing
There are three main pieces of information that are used to create a learner recommendation:

1. Target state – Desired competency level of a learner that is represented in a qualification profile
2. Current state – Current competency level of learner that is represented in the learner profile

3. Learning activities – Training and education events designed to train a specific set of competencies

A learner's target state is a qualification profile which is composed of a set of competencies. The qualification profile can be multilayered with subcompetencies. A learner profile contains this qualification profile, and the same set of competencies. Where the target state contains the required level to become qualified, the learner profile contains the current level of the learner. Learning activities are designed to train the learner in this qualification profile's competencies. Even though an activity might train multiple competencies at once, the weight might differ per competency. Also, an activity is designed with a certain difficulty level in mind. Therefore, current and target competency levels are compared, as well as activities with a difficulty level and a weight per competency.

The Elo rating system is used as a basis ("Elo rating system", 2022) to quantify the competency levels of all three inputs. The Elo rating system is originating from chess and widely used in the gaming industry to match up players. The Elo rating system uses a relative scoring system and uses a deterministic prediction formula to predict the most likely score each player is going to get. Part of the Elo rating system is making a prediction by calculating an expected win rate, which can be determined when two opposing player's levels are known. In this case this is used to determine an expected win rate of a learner per learning activity. The expected win rate results in a chance that the learner performs the task successfully. A difficult learning task relative to the current competency level of the learner results in a low expected win rate for the learner. An easy learning task would result in a high expected win rate. The learner can gain in its competency level based on the final result relative to the expected win chance. The Elo rating system is slightly modified to accommodate for certain functionalities, and to make it less punishing when evaluating scores. This tailor-made Elo rating system for a total learning architecture is named the Progressive Scoring System (PSS).

Quantifying Flow

When comparing the competency score of a learner and an activity, an expected win rate is calculated. The expected win rate, R, is proposed as the value to quantify flow. The expected win rate can be seen as a difficulty score of an activity for this specific learner. When the win rate is too high, the learner might become bored by repetition and being left unchallenged. When the win rate is too low, the learner cannot comprehend the learning experience yet and might feel anxiety or frustration. This corresponds one on one with the flow channel over skill versus challenge. The flow value range determines the boundaries of too challenging and too simple relative to the learner's competency state. The recommendations take the flow into account by recommending activities within these boundaries. Based on the result of the activity the flow value is updated.

Progressive Scores

The bandwidth of a learner's flow is iteratively updated per activity and will adjust to the learner. It is needed to update the scores of all competencies over time which are progressive scores. Instead of giving traditional periodic or final scores to learners for every activity, the competency score of the learner is tracked and the effects of each activity on that competency score are determined and added. Competency scores of both the learner and learning activities adapt to conform to their actual values over time.

After completing a learning activity, the learner's competency scores are updated. Each activity can train multiple competencies, which competencies are trained per learning activity is defined. However, the emphasize is not on all the competencies and the influence of the activity on each competency vary in this respect. Therefore, a weight per competency for a learning activity which represents this influence is defined. The process of defining the weights is for now an educated guess by subject matter experts.

The score and weight of a learning activity are both initially given by instructors and training developers. Both values are updated after a number of learners have participated in the learning activity, based on the competency scores of all learners combined and their results.

RECOMMENDATION FRAMEWORK

Learner's Perspective

Seen from the perspective of a learner the journey starts at the LXP. After logging in, the LXP requests a recommendation from the recommender. The recommender requests the learner profile using the identity of the learner. Also, the qualification profile is requested and compared to the learner profile in order to select which competencies to train. To understand how the competencies are selected a bit of insight in the underlaying data models is required.

Data Models

In Figure 4 a simplified version of the data models, used to create a recommendation, is shown. A qualification profile is positioned that consists of competencies and has a name. Each competency has a name, a required level which is the minimal level to qualify, a desired level that represents the level the course aims for and a set of performance indicators that make up the qualification. Data in a qualification profile is treated as static data.

A learning activity is in place that looks identical to a qualification profile. However, the definition of a competency for an activity differs from the qualification profile competency. Per activity the qualification competencies and their difficulty level are linked to the qualification profile competencies. Also, a difficulty level for this competency is defined. At last, the influence of the activity toward a competency is defined. The difficulty level and influence of an activity are dynamic. After running learner analytics those values are updated as discussed at the end of this section.

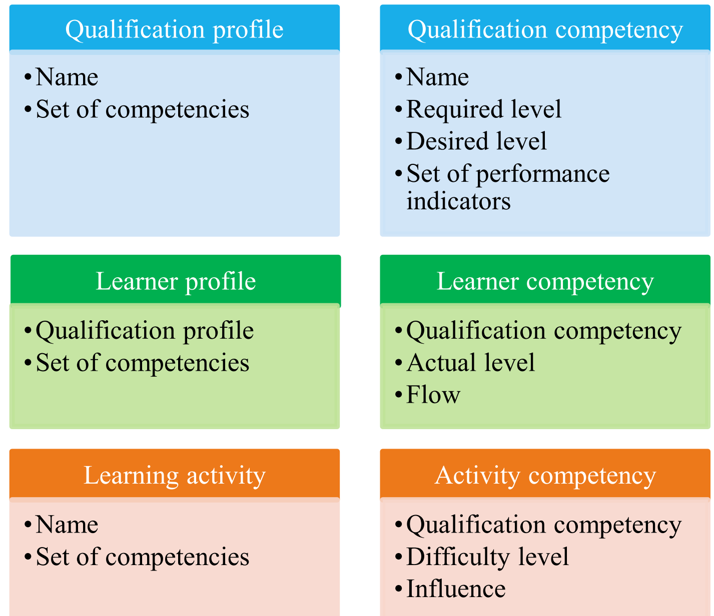

Figure 4. Simplification of data models used in the recommender framework

Creating a Recommendation

The data models allow the creation of a recommendation, by the process depicted in Figure 3. The first step is to select the competencies to train.

Quantifying Competencies

To be able to select competencies, they need to be quantified. The Elo-scores are used throughout the learning process or training course, which are defined as relative measure of skill level within the system it is calculated. At the start of the course the Elo-scores are provided with an initial value based on the phase of training (e.g. initial, advanced training). The required and desired level are also provided. The progress made by performing learning tasks and the amount of score gained is dependent on the result relative to the difficulty of the learning task.

Another important value is K, which is the maximum possible progression per learning tasks. This value is determined based on the delta between the initial and desired level. The delta is divided by the amount of learning tasks that trains the competency plus a constant factor. The factor is needed to make it practically possible to reach the required and desired levels. The K is different for every competency, because the amount of learning tasks can differ.

$$K = (desired\ level - initial\ level)/(amount\ of\ learning\ tasks + factor) \qquad (1)$$

The last variable that is needed to complete the quantification of competencies is the expected win chance of the learner. Based on the competency states of the learner relative to difficulty levels of the learning tasks, a prediction can be made of the win chance of the learner ($R_{learner}$). When all components are put together, a learner's competency can be quantified and progression can be tracked over time in a single value as is shown in the second formula below.

$$R_{learner} = 1 \,/\, (1 + 10^{((lc.actualLevel - ac.difficultyLevel)/400)}) \qquad (2)$$

$$New\ actual\ level = actual\ level + K \cdot (result - R_{learner}) \qquad (3)$$

The learner profile shows the competency levels of the learner. After every learning task the competency levels are updated accordingly. It is defined which competencies are trained on which level within a learning task.

Selecting Competencies to Train

To select the competencies to train, the learner competencies are compared to the qualification levels. To compare those, the difference (delta) of the actual level of a learner, the required and desired levels are determined.

Functionally speaking, a negative number for the delta means the level is reached. A positive number means training is needed to acquire either the required or desired level. A higher positive value corresponds with a larger training gap.

After the deltas are calculated for all learner competencies, the competencies are sorted by descending the positive required level deltas. Followed by the desired level deltas in descending order as well. However, for the desired values the competencies already present in the sorted list are skipped and negative deltas are included.
This way of sorting results in always recommending to train the competency a learner lacks in. To counter this, one might consider different sorting strategies.

Selecting Activities to Consider

Either the entire sorted list of competencies or a subset can be used to make a first selection of learning activities. The activities that train these competencies are retrieved from the experience index. The resulting list of activities is than filtered by prerequisites. If the learner has repeatedly failed to complete an activity then the activity is removed from the list as well. After these filter steps the result is a list of activities potentially viable to train the selected competencies.

Predict Learner's Results Per Activity

The list of selected competencies is sent to the task score predictor. The task score predictor calculates a fit between the learner's competencies and the competencies in the learning activity, using the fit of the expected win rate formula.

As shown in the Pseudocode 1 below, the fit is determined per activity. This fit is determined by calculating a weighted fit per competency of the activity. However, instead of competencies any representation of skills, like performance indicators, can be used in this algorithm, as long as a difficulty level can be assigned.
Once the task score predictor has determined all fits, it sends back those results to the recommender component.

```
for each activity
{
        //Define expected win rate R for activity
        aRsum = 0
        //Calculate expected win rate for each competency in activity
        for each activityCompetency (ac) in activityCompetencies
        {
                learnerCompetency (lc) = learnerProfile.get(activityCompetency)

                //Calculate activity competency win rate
                acR = 1 / (1 + 10^((ac.difficultyLevel – lc.actualLevel)/400)

                //Calculate fit of activity competency to learner's competency flow
                ΔcR = |random flow value from interval - acR|

                //Add influence/weight to competency fit
                ω = ac.influence
                aRsum = ∑ΔcR · (1 − ω)
        }
        //Finally, normalize to get overall activity fit
        acR = aRsum / activityCompetencies.count
```

Pseudocode 1. Determine activity's fit to leaner's flow per competency

Selecting Activities to Recommend
From the list of viable activities, a selection of activities is made based on the influence on preselected competencies. However, for the final selection the overall fit of an activity is determined and prioritized, rather than the preselected competencies.

To select activities to recommend, several strategies can be deployed. First, the most straight forward option would be to sort the list of activities based on their fit with the learner's capacities. This represents picking the most optimal overall learning curve with respect to pushing a learner as fast as possible through a training course. Second, a subset of activities with the most influence on one of the three selected competencies can be selected, resulting in three lists, one per competency respectively. Each of these lists are than sorted based on their fit. Based on the competency a recommendation is provided. Therefore, a focus is on optimizing the learning curve for specific competencies. A third strategy could be to involve a learner's history, where based on the previous used learning activity the most influential competencies are selected and compared to the activities processed by the task score predictor. Then sort those activities based on their fit. This approach favors the quality of a learning curve over fast progress. Ideally, an instructional model employing a mixture of these strategies is used to pick a subset of learning activities to present to the learner in the learning experience platform.

Updating Dynamic Values
Once a leaner partially or fully completes a learning activity, the learner analytics and learner inference components start evaluating the dynamic values. Dynamic values as used in the recommender framework are: the actual level and flow of a learner's competency and the difficulty level and influence of an activity competency. The learner analytics component is responsible for evaluating the activity competency's difficulty level and influence. To evaluate the learner competency's actual level and flow the learner inference component is employed.

Learner Competency Evaluation
To update the leaner's actual level the Elo-rating equation is used:

$$New\ actual\ level = actual\ level + K \cdot (result – R_{learner}) \quad (4)$$

K is the maximal level change that can be applied. The $result$ is either 0 or 1, and represents whether the activity has been failed or passed. And $R_{learner}$ is the expected win rate of the learner for the activity.

A flow value is determined which symbolizes the central point of the flow channel, which is initially set on an expected win chance of 45%. The easy and hard bound of a learner's flow bandwidth is created by adding and subtracting 15% to/from the flow value. This creates a flow channel from 30%-win chance to a 60%-win chance. Two variants of a formula are used to evaluate the flow value F, depending on the $result$ of the learning activity. In case of a pass the following formula is applied:

$$F_{new} = F_{current} + \min(0, R_{learner} - F_{current}) \cdot factor \quad (5)$$

The variable $factor$ is introduced to control the reward and punishment of the flow value and is relative to the size of the fail or pass (shown in table 1). In case of a fail the following formula is applied:

$$F_{new} = F_{current} + \max(0, R_{learner} - F_{current}) \cdot factor \quad (6)$$

$$FAIL\ \leq\ 0.5\ >\ PASS \quad (7)$$

Table 1. The reward and punishment factor of the flow value relative to the size of the fail or pass

$result$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$factor$	1.0	0.8	0.6	0.4	0.2	0.2	0.4	0.6	0.8	1.0

The flow value expressed in F_{new} is the central point of an interval that represents the zone in which is searched for a suitable learning activity. The flow value indicates the expected win rate of the learner that fits the flow perfectly. An interval is created by adding and subtracting a deviation (*D*) to/from the flow value. This creates an easy and hard bound. The next learning activity is found in between these boundaries to recommend an appropriate learning activity.

$$F_{easy} = F_{new} + D \qquad\qquad F_{hard} = F_{new} - D \tag{8}$$

The effect of the formulas for F_{new} has on the flow is as follows; when $R_{learner}$ is a lower value than $F_{current}$, meaning it is challenging and the learner passes the activity the F_{new} gets lower, thus providing more challenging activities in the future. However, when an activity is passed where $R_{learner}$ is higher than $F_{current}$ the value remains unchanged. In case a learner fails an activity, the exact opposite happens. Thus, the flow value is only changed when an easier challenge is failed. The easy and hard bound moves automatically along with the changes in F_{new}. The size of the change is relative to the results as is shown in table 1.

Exploration and exploitation of flow bounds

When calculating a fit with the learner's competency, as shown in Pseudocode 1, either an explore or exploit strategy can be used. To balance this out the proposition is to plot a normal distribution with an integral value of 1 between the easy and hard bound. Then use the integral of the normal distribution given a random number between 0 and 1 to get a flow value which is used to find a next learning activity. This favors values close to the average flow value but may also push the upper and lower boundaries from time to time.

Activity Competency Evaluation

Each learning activity presents a challenge for a learner, which can be broken down to competencies that are being trained by the activity. After a significant number of leaners participated in the activity the results can be used to evaluate each difficulty level and influence. By using the result of all leaners as input to calculate the expected win rate of the activity's competency is calculated, which is then used to calculate a new level score using the same formula as is used for updating a learner's competency level.

There are situations where unaccounted complexity factors are introduced. Therefore, it is not recommended to use an automated system to update the activity competency values. Rather, it is proposed to let an expert decide whether to update values based on data driven suggestions by the system.

DISCUSSION

The Progressive Scoring System creates a competency level overview with the proficiencies of the learner such that the Recommender can provide suggestions for the next training activity. This section discusses to which extent the principles competency development (Stafford, 2019), flow of learning (Csikszentmihalyi, 1990), and the zone of proximal development (Vygotsky, 1978; Wertsch, 1984) are implemented in the PSS and Recommender and where we see a need for future development and research topics are discussed.

The training need of an individual learner is highly dynamic. If the progression of learning needs to be tracked in order to reveil the training need, a data structure is needed to express the progression. The Progressive Scoring System uses the competency framework (Stafford, 2019) to structure data and to define what can be learned in a learning activity. In the end, all the data of the Learning EcoSystem is shaped along the structure provided by the competency framework. It empowers the Recommender system to detect the training need.

The PSS and Recommender system use databases filled with learning tasks, competencies, difficulty levels and so forth to feed the system. These databases are dependent on the human input by instructors and training experts. If the organization does not adopt a competency-based approach, then chances are that the data is too general and hard to structure to provide insights that are useful for training. Besides adopting a competency-based approach, it is beneficial to use competencies and performance indicators that are specifically developed for the job context that is trained.

The Recommender system considers the zone of proximal development and flow of learning with its flow value, the initial easy bound of 60%-win chance and hard bound of 30%-win chance. Besides that, the Progressive Scoring System quantifies the progress made by a learner based on its result relative to the difficulty level of the learning activity. Therefore, the system is capable to provide learning experiences that fit either the zone of proximal development or the flow of learning. The alternation between both is beneficial for learning, which is enabled by the normal distribution in the recommendation process (Vygotsky, 1978; Wertsch, 1984; Csikszentmihalyi, 1990). This alteration could even be expanded with the construct of self-efficacy, which is essential for successful learning (Zimmerman, 2000). However, the calculation of the probability (win chance) is binary, either win or lose. This originating from the original Elo score system, but training is not as binary as win or lose. It is necessary to research calculations methods that suit training better than a binary method.

The current way of depecting a learner's training status is by listing its scores over time. This method only presents snapshots of the competency level of training. The Progressive Scoring System overcomes the 'snapshot-problem' by expressing the competency level with one score. The PSS considers the character of relativity regarding the learner's competency level at a given time and the difficulty level of the learning task. Therefore, the system is able to progressively quantify competency levels. However, this system is different from what is used today and can come across as 'black box'-process. It is important to explain the system to stakeholders and provide time to adapt.

In the effort to conform the Progressive Scoring System and Recommender system with the three training concepts (Vygotsky, 1978; Wertsch, 1984; Csikszentmihalyi, 1990, Stafford, 2019), there are also concepts missed or not yet implemented. Firstly, the training concepts of skill decay. Whenever somebody learns, there is also competence decay whenever a competency is not used or trained for a period of time (Wang et al., 2013; Vlasblom et al., 2020). Initial training often exists of repetition, but especially in continuation training competencies can be untouched for long periods of time. In the PSS as well as the Recommender system the concept of competence decay could be implemented. The competency scores from the PSS could for example decay over time according to a model. These models are thoroughly investigated and developed (Jastrzembski et al., 2009). The Recommender system could implement rules in its selection process in order to train competencies in time to prevent or mitigate competency decay.

Secondly, the recommender searches for competencies that are behind in development and recommends activities that train these competencies. Competencies are mostly isolated and less interconnected in aircraft maintenance and therefore this selection strategy could work. However, in other contexts such as military pilots, the competency framework is more hierarchical and interconnected because it is a dynamic environment that requires continuous control. The search strategy cannot yet cope with the interconnection within a competency framework.

Thirdly, a prerequisite of the Progressive Scoring System is that learning tasks are given a difficulty level for each relevant competency. The fixed difficulty level gives a good indication, but in reality, difficulty is more variable and learning tasks can turn out differently than expected in individual sessions. The PSS described previously does not yet consider this variety, this could be considered by for example subsequent calculation of complexity initiated by the instructor.

This framework is only at a design stage and further validation is needed, more specifically:

- The system sets a key role for the competency framework, but the requirements a competency framework should adhere to are unknown. It should be considered how many competencies are in the framework and how layered it is. There could be a balance between explainability and degree of detail with the amount of learning tasks. A statistical analysis could provide insights in the added value of extra competencies or layers to the framework in order to find an optimum.
- The formulas in the framework consists of several factors and initial values. A statistical analysis could provide insights in how to optimize these values. The data could be gathered from a use case. The behavior of learner could also be simulated to enlarge the dataset and understand the behavior of the system.
- The experience index is a cornerstone of the system by indicating the difficulty and weight of competencies in regard to the learning task. These parameters can be determined by an educated guess by experts. However, it would be more reliable and valid to use a methodical approach. These methods are not yet developed, but could make use of the current work regarding task analyses in training and education.

Many new research questions are rising with the emergence of the concept Learning EcoSystem. The recommendation framework described in the present paper kicks off the discussion in this regard. An initial concept is constructed to utilize educational principles for the construction of algorithms in a Learning EcoSystem next steps for research and development are identified in order to enhance human performance in the 5th Generation Air Force.

ACKNOWLEDGEMENTS

Project N18/031 'Integrated Digital Training Environment for Aircraft Maintenance (IDTEAM)' – funded by MoD NLD

REFERENCES

Csikszentmihalyi, M. (1990). Flow--the psychology of optimal experience. New York: *Harper & Row*.

Elo rating system. (2022, 1 June). In *Wikipedia*. https://en.wikipedia.org/wiki/Elo_rating_system

Jastrzembski, T.S., Gluck, K.A., Rodgers, S.M., TechNet, A., & Krusmark, M. (2009). *The Predictive Performance Optimizer: Mathematical Modeling for Performance Prediction*.

Marchand, A. (2017, August 7) *5e generatie – wat houdt het in?* https://magazines.defensie.nl/vliegendehollander/2017/07/06_5e-generatie

Marchand, A. (2019, May 12) *5 vragen over de 5e generatie luchtmacht.* https://magazines.defensie.nl/vliegendehollander/2019/12/05_5e_generatie

Smith, B., Johnson, A., Hayden, T., & Tolk, F. (2022) *Total Learning Architecture – Functional Requirements Document*. Orlando, Florida: Advanced Distributed Learning Initiative

Stafford, M. (2019) Competency-based Learning. In J.J. Walcutt, & S. Schatz (Red.), *Modernizing Learning – Building the Future Learning Ecosystem* (pp. 3 – 4). Washington, D.C., United States: Government Publishing Office.

Vlasblom, J.I.D., Pennings, J.J.M., Van der Pal, J., & Oprins, E.A.P.B. (2020) *Competence retention in safety-critical professions: A systematic literature review*. Educational Research Review, Volume 30, 100330, ISSN 1747-938X, https://doi.org/10.1016/j.edurev.2020.100330.

Vygotsky, L.S. (1978). Mind in society. Cambridge, MA: *Harvard University Press*.

Walcutt, J.J. & Schatz, S. (2019), *Modernizing Learning – Building the Future Learning Ecosystem*. Washington, D.C., United States: Government Publishing Office.

Wang, X., Day, E.A., Kowollik, V., Schuelke, M.J., & Hughes, M.G. (2013). Factors influencing knowledge and skill decay. In W. Arthur, E. A. Day, W. Bennett, A. M. Portrey (Eds), *Individual and team skill decay: The science and implications for practice* (pp. 258-281). New York, NY: Routledge

Wertsch, J. V. (1984). The zone of proximal development: Some conceptual issues. *New Directions for Child and Adolescent Development*, *1984*(23), 7-18.

Zimmerman, B. J. (2000). Self-efficacy: An essential motive to learn. *Contemporary educational psychology*, *25*(1), 82-91.