The Pedagogical Pentagon: A Conceptual Framework for Artificial Pedagogy

Jordi Bieger,[1] Kristinn R. Thórisson[1,2] & Bas R. Steunebrink[3]

[1] Center for Analysis and Design of Intelligent Agents / School of Computer Science,
Reykjavik University, Menntavegur 1, 101 Reykjavik, Iceland
[2] Icelandic Institute for Intelligent Machines, Uranus, Menntavegur 1, 101 Reykjavik
[3] NNAISENSE, Switzerland
{jordi13,thorisson}@ru.is,bas@nnaisense.com

Abstract. Artificial intelligence (AI) and machine learning (ML) research has traditionally focused most energy on constructing systems that can learn from data and/or environment interactions. This paper considers the parallel science of teaching: Artificial Pedagogy (AP). Teaching provides us with a method—aside from programming—for imparting our knowledge to AI systems, and it facilitates cumulative, online learning – which is especially important in cases where the combinatorics of sub-tasks preclude enumeration or a-priori modeling, or where unforeseeable novelty is inherent and unavoidable in the learner's assignments. Teaching is a complex process not currently very well understood, and pedagogical theories proposed so far have exclusively targeted human learners. What is needed is a framework that relates the many facets of teaching, in a way that works for a range of learners including machines.

We present the *Pedagogical Pentagon*—a conceptual framework that identifies five core concepts of AP: learners, task-environments, testing, training and teaching. We describe these concepts, their interactions, and what we would need to know about them in the context of AP. The pentagon is meant to facilitate research in this complex new area by encouraging a structured and systematic approach organized around its five corners.

1 Introduction

Successful operation in any situation requires relevant knowledge.[1] which can either be innate or acquired through experience: nature vs. nurture. Here we are concerned with the *nurture* part of that equation. As a learner gets more capable of learning a broad range of tasks in a wide range of environments, and the ratio of acquired/required knowledge to innate knowledge increases, its

[*] This work was sponsored in part by the School of Computer Science at Reykjavik University and by a Centers of Excellence Grant from the Science & Technology Policy Council of Iceland.

[1] We use "knowledge" to refer to all kinds of knowledge, including beliefs (declarative), skills (procedural) and priorities (structural); cf. Section 3.2;

nurture becomes increasingly relevant. Research in artificial intelligence (AI) and machine learning (ML) has traditionally focused on the nature part. Systems are often thrown "in the deep end of the pool" where they must learn in a complex and often unhelpful task-environment, or from an unstructured pile of data, which greatly limits the range of tasks they can learn to tackle in practice.[2] By teaching—broadly defined as *"the intentional act of helping another system learn"*—we can overcome some of these limitations and greatly facilitate the learning process in general [2]. We suggest that in parallel to machine learning, a science of machine teaching—which we call *"Artificial Pedagogy"* (AP)—can provide many complementary benefits.

Aside from the initial programming of an AI system, teaching is the only way for us to impart our knowledge on it [1]. Teaching can often be more natural—e.g. if we cannot articulate our knowledge precisely enough to program/formalize it or if the AI's knowledge representation mechanism is opaque to us. Even more importantly, a hallmark of general intelligence is the ability to deal with new situations, including ones that were unforeseen by the AI's developers. We cannot program what we cannot anticipate, but teaching can be applied when it is needed and adapted to the requirements of any situation.

Cognitive architectures aspiring towards AGI often contain very little domain-specific knowledge to preserve their generality, and start their "life" in a baby-like state. Without knowledge, little more can be done than systematically (or randomly) exploring the state-action space, which becomes prohibitive as the complexity of targeted domains increases—*even if the learning system is very sophisticated.* Teaching can guide such systems towards salient stimuli or knowledge, or to provide it directly. As progress is made in AI/AGI research, the number of architectures capable of utilizing sophisticated teaching techniques is ever growing [6], making a general theory of teaching more desirable than ever.

Due to these benefits, many ML projects have developed methods for "helping their AI system learn", but so far this has mostly been done on an ad-hoc case-by-case basis. A general theory of AP could help us understand what works in which situations. Unfortunately, teaching is a highly nontrivial process that involves many moving parts. In the social sciences, similar efforts have entire research fields dedicated to them (i.e. pedagogy, educational science, developmental psychology, etc.), and we argue AP should be seen in a similar manner. In AP however, we cannot make the same assumptions that are warranted in the social sciences, because the (eventual) space of artificial minds is many times larger than the space of human minds. We cannot take concepts for granted, and must make an effort to define them explicitly and rigorously.

Our goal in presenting the Pedagogical Pentagon (see Figure 1a) is to provide something that our knowledge in this domain can be organized around, and to

[2] Note that the term "teaching" does not necessarily imply a mirroring of the human teacher-student setup—it is quite conceivable for an AI to have a built-in "automatic teaching mechanism". That would not, however, change the need for a theory of teaching. While teaching does not change the inherent capabilities of AI systems in principle, it allows them to reach more of their potential more efficiently.

facilitate structured and systematic research in this area. We take inspiration from e.g. Bloom's taxonomy of learning domains, which has been used as the basis for many educational programs for humans, by providing different learning targets to focus (or not focus) on [7]. While it is impossible to provide full theories of every concept involved in AP here, even such theories they existed, we hope to provide some ideas for how AP might be studied.

2 Background, Definitions & Concepts

To model the learning process, we consider the interaction of intelligent systems with various environments.[3] An *environment* is a perspective on the world, consisting of a set of variables with acceptable values, an initial state, and functions that describe how it changes over time [13]. Examples of possible environments include games, rooms, buildings, cities, countries and indeed the entire world. Intelligent systems can independently decide at which abstraction level they want to consider different parts of the world in different situations.

Intelligent systems continually receive inputs/observations from their environment and send outputs/actions back. Some of the system's inputs may be treated specially—e.g. as feedback or a reward signal, possibly provided by a teacher. Since intelligent action can only be called that if it is trying to achieve something, we model intelligent agents as imperfect optimizers of some (possibly unknown) real-valued objective function. *Tasks* are similarly defined by (possibly different) objective functions, as well as (possibly) *instructions* (i.e. knowledge provided at the start of the task or throughout its duration). Since tasks can only be defined w.r.t. some environment, we often refer to the combination of a task and its environment as a single unit: the task-environment.

In the AP setting, we have at least two different intelligent systems with the roles of "learner" and "teacher".[4] The teacher's *teaching task* is to change the learner's knowledge in some way (e.g. to make the learner understand something, or increase the learner's skill on some metric). The learner and the teacher each interact with their own view of the world (i.e. their own "environments") which are typically different, but overlapping to some degree. The learner will always exist in some form in the teacher's task-environment, and the teacher teaches by affecting the learner's. As we will see, there are many ways to do this, including full determination of the learner's environment, modification of existing environments, or simply by changing their own behavior (if the teacher is in the learner's environment this affects its dynamics from the learner's point of view).

[3] The formulation of an intelligent system (or agent) interacting with the world (or environment) is most commonly used in control theory and reinforcement learning. However, it is a fully general formulation, that also covers traditional cases of e.g. supervised and unsupervised learning. Here the environment simply presents a (training) datum at each time step, the agent responds with a classification or prediction, and—in the case of supervised learning—the environment replies with the target outcome or an error signal.

[4] Generally speaking, there could be multiple learners and teachers, but here we focus on the one-on-one situation.

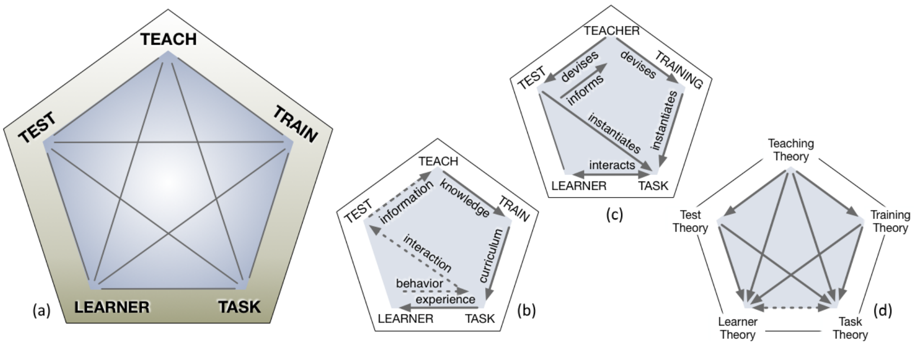

Fig. 1: (a) The Pedagogical Pentagon. (b) Information flow between processes. (c) Relations between systems. (d) Dependencies between theories.

An AP interaction is defined by a number of teachers interacting with their own environments who are given a *teaching task* that contains *learning objectives* for the involved learners as well a set of constraints (e.g. on budget, time, resources, allowed actions, etc.). Given (possibly incomplete and imperfect) information about the various aspects of an AP interaction, we want a theory of AP to give us predictions of what the teacher(s) would do, and more importantly, what they *should* do in order to optimize the objective function. For instance, if a chess teacher doesn't know the learner is deaf, we can predict he try to verbally explain things, realize this doesn't work, and switch to a different strategy—one that he perhaps should have used from the start, if he had known better.

The role of "teacher" may be taken up by any entity or system, including e.g. school teachers, schools, specialist AI systems, AI system designers, or indeed us as AP practitioners. AP theory (and the Pedagogical Pentagon) can be applied fractally, on multiple levels of organization. For instance, a school could be seen as a "teacher", tasked with instilling certain kinds of knowledge in the children who go there. The school may pick out some high level methodologies (e.g. montessori), but for the most part it relies on employing human professors who interact with the children directly. These professors can be controlled to varying degrees (e.g. a curriculum could be provided or not), but ultimately they are themselves "teachers" (in the AP sense) with their own (limited) knowledge and capabilities that the school needs to take into account.

3 Conceptual Framework

In this paper we introduce a conceptual framework for studying AP in the form of the "Pedagogical Pentagon" (see Figure 1) which we believe outlines the five core concepts involved in AP: learning systems (learners), task-environments, evaluation (tests), knowledge acquisition (training), and teaching. Teaching consists, broadly speaking, of altering the training process of the learner, based on information about the learner and the task. *Learners* can have many different properties that influence how (well) they behave in various domains, what information they need and can use, and ultimately how they can and should be taught. Within one AP interaction, we see many different *task-environments*: one(s) for the teacher(s) to define what they can and should do, ones for which the learner needs to develop knowledge/skills, and ones in which the learner will be tested and trained. Proper teaching requires that the teacher has up-to-date knowledge of the learner, which can partially be provided a priori, but must otherwise be obtained through evaluation or *testing* of the learner as they interact with a task-environment and (hopefully) make progress on the learning objectives. Similarly, we want to have some idea of how interaction with a task-environment will *train* (or otherwise influence) the learner's knowledge. Finally, *teaching* can be done using different methods by utilizing knowledge of testing, training, task-environments and the learner in order to make sure the learner learns what is necessary within the constraints outlined by the teaching task.

The pentagon can be viewed on multiple levels. Figure 1b showcases the different goals of training—to imbue knowledge into the learner—and testing—to obtain information about the learner—by looking at the information flow between processes. Knowledge flows from the teaching to the training process to create a curriculum in the form of a task-environment that the learner experiences. And as the learner behaves in a task-environment, that interaction can be analyzed by a testing process to obtain information for teaching. Figure 1c views each corner as systems and specifies their relations. The teacher devises tests and trainings, which in turn instantiate task-environments that the learner interacts with. Figure 1d shows the hierarchical dependencies between theories: learners and task-environments can be analyzed in isolation or possibly together, training and testing use task-environments to instill/obtain knowledge into/about the learner, and teaching involves designing appropriate tests and training schemes.

All concepts can interact and constrain each other. For instance, any given task-environment imposes requirements on the learner (who must be able to perform the task), which in turn restricts the teaching methods we can use. Or if we want to use certain teaching methods, we must select or design a learning system that can make optimal use of them. Or if resources like time are limited, we might have to simultaneously use task-environments for testing and training.

Our ultimate goal is to develop a full theory of artificial pedagogy, in the same sense that we might want to develop a full theory of artificial intelligence or machine learning. The realization of this goal is naturally (vastly) beyond the scope of this paper. The Pedagogical Pentagon should be viewed as a conceptual framework around which the knowledge we obtain in this domain can be organized. By separating out different aspects of AP—each of which are deserving of their own comprehensive theories—and relating them to each other, we hope to make research in this domain more tractable and systematic/structured.

3.1 Tasks

The concept of a "task" is at the core of AI. We design AI systems to perform ranges of tasks, then we use related but possibly different tasks to train them, before using (often slightly different) tasks for evaluation. Yet, our understanding of the concept is mainly intuitive. We have argued before about the need for more rigorous task theories in AI, that aid us in the general analysis and construction of task-environments [13].

This is especially pressing in the context of artificial pedagogy, where many task-environments are often involved in a single pedagogical interaction. First, there is the task-environment for the teacher. This environment contains, among other things, the learner(s) and defines the actions and observations available to the teacher. The teaching task typically refers to the learning objective(s) as well as additional constraints on e.g. budget, time and other resources. The learning objectives are the objectives for the learner(s), which are typically to achieve some epistemic state (i.e. know or understand something), to alter preferences (e.g. in the case of inverse reinforcement or value learning), and/or to perform well in some range of task-environments. So secondly, we have the set of task-environments that the learner will interact with. In a pedagogical setting, these may either be created, influenced or utilized by the teacher. Here we can distinguish between task-environments meant for obtaining information about the learner (testing), meant for training the user, or both.

Despite these interactions with other corners in the Pedagogical Pentagon, we believe that task theory can also be studied in relative isolation. A task theory should provide a method for representing tasks and environments in a way that facilitates their analysis and construction [13]. A more specific list of desiderata includes abilities to compare tasks, to create abstractions, concretizations and decompositions, to characterize tasks in terms of various (emergent) measures and provided instructions, to estimate resources necessary for task completion, and to construct new tasks based on combination, variation and specifications. Different AI research scenarios will make use of different aspects of task theory, but it seems that a good teacher would potentially use everything.

3.2 Learners

The ultimate goal of any teaching interaction is to help another learning system—the learner—learn something. Naturally, the way in which any system learns—as well as how to optimize this process—depends on the specifics of that system. A 'learner theory" would parallel the above mentioned "task theory" in that it should allow us to analyze, define, characterize, categorize and compare learning systems. Many partial attempts at comparison and categorization have been made (cf. [6] for a recent overview), but we are not aware of any rigorous and comprehensive treatment of all aspects of learning systems.

From a teaching (and learning) perspective, it's important to distinguish between structure and content. By "structure" we mean aspects of the system that remain relatively constant throughout the learning interaction like the architecture / algorithm(s) and the body. By "content" we mean knowledge, of which various kinds exist, including declarative knowledge or beliefs (e.g. "the capital of France is Paris" or "yesterday I felt good"), procedural knowledge or skills (e.g. knowing how to ride a bike), and structural knowledge or priorities (e.g. feeling that avoiding a predator is more important than eating now).

Structure properties include the kinds of memory (e.g. procedural, episodic, and/or semantic), reasoning (e.g. inductive, deductive, counterfactual and/or analogical), and learning (e.g. supervised, unsupervised and/or reinforcement) mechanisms the learner has, as well as their capacity and how they operate. These properties are important for AP, because there is no sense in explaining something by analogy if the learner can't reason by analogy, or providing affective feedback if there are no reinforcement learning mechanisms. Knowledge properties are much more fluid, and can often be the subject of the teaching task—e.g. to make the learner understand/know something, be good at something, or want something. Since this knowledge is likely to refer to or model the environment, the chosen representation should be compared to the representation mechanism used for task-environments. From these properties, other—often measurable—properties emerge, such as performance (in different situations), adaptivity, robustness and understanding [5].

The relationship between the learner and testing corners of the Pedagogical Pentagon is that for the learner, we are primarily interested in "what" properties it has and how they are defined, whereas testing is primarily concerned with "how" this information can then be obtained (approximately) *from* a specific instance of a learner [5]. As such, we could come up with formal definitions of properties we care about (e.g. intelligence [8]), without worrying about whether they can be measured directly. Learner theory lets us consider the "insides" of a hypothetical learner directly, while testing provides an "outside" view based on observed behavior. Similarly, most aspects of the learner can be analyzed and defined without making reference to the exact way in which knowledge (and consequently emergent properties) change as the learner interacts with some task-environment (i.e. training). Some aspects of learners could also be studied without a theory of task-environments, but this is not always the case. For instance, to estimate the (changing) level of complexity and variety that a learner can handle, we need a task theory to provide measures of complexity and variety of task-environments.

3.3 Testing

To teach well, the teacher has to know the student. While some aspects of the learner may be known a priori, others must be obtained by the teacher interactively (e.g. progress towards the learning objectives). We define "testing" generally as the empirical means through which an observer obtains information about another system by systematically observing its behavior as it interacts with its task-environments [5]. Specifically, testing is meant to obtain information about the structural, epistemic and emergent properties of learners described in Section 3.2. Testing can be done for different purposes: e.g. to ensure that a learner has good-enough performance on a range of tasks, to identify strengths and weaknesses for an AI designer to improve or an adversary to exploit, or to ensure that a learner has understood a certain concept so that we can trust it will use it correctly in the future. A "Test Theory" for growing recursive self-improvers may first and foremost be concerned with gauging levels of understanding in service of such confidence-building [12]. In the context of AP, our primary concern is to let a teacher obtain information about limitations, strengths and preferences of the learner, and to measure progress with respect to the learning objectives. A test theory should allow us to extract information about a learner from its behavior in a task-environment, predict what kind of information we could obtain in a given task-environment, and help to construct (or alter) task-environments to obtain desired information using minimal resources.

There are many different ways of AI evaluation [3,4,9], but we are not aware of any theory that covers all kinds of information extraction. Information can be extracted by sporadic evaluations (e.g. like school tests) or continual observation (e.g. like a sports coach does), it can be over or covert, and it can be done using many different tests (e.g. multiple-choice vs. open questions vs. a project). Designing tests is subject to real-world constraints such as malleability of the task-environment, available knowledge, and capabilities of both learner and teacher. In both the design of tests and the interpretation of learner behavior or results, it is important to take into account the goals of the learner and how they compare to the used performance measure: if the learner performs poorly, is it because they lack skill/knowledge, did they misunderstand the instruction, or did they simply not care to do well?

3.4 Training

Learning systems adjust their knowledge as a result of interactions with a task-environment. Viewed from a teacher's (and intentional learner's) point of view, we refer to this as "training" as the goal is to become better at some task. Nevertheless, we should not neglect the possibilities that erroneous things can be learned, and desirable things can be unlearned. The goal of the teacher is to influence the learner's task-environments in such a way that progress towards the is facilitated. AP is interested in predicting how a learner's knowledge/skills will change as a result of interacting with a particular class or instance of a task-environment, and to allow us to construct (or alter) task-environments in order to train a particular skill or impart particular knowledge.

Training is roughly analogous to testing, but each has a different goal: The goal of training is to to move the learner from one state to another—to get knowledge into the learner—while testing is about getting an accurate model or measure of the learner's skill at some point(s) in time—getting information out of the learner. Both make heavy use of both task theory and learner theory. Training theory is mainly concerned with how interactions with the environment affect the epistemic and emergent properties of the learner (i.e. knowledge and performance). As with test theory, there will be different kinds of training (e.g. repeated exposure to similar stimuli vs. one-time explanations) which may occur intentionally or not, and success will depend on the goals of the learner.

Many theories of learning/training already exist in e.g. educational science, developmental psychology and animal training. Such theories may usefully be plugged into our Pedagogical Pentagon to facilitate the science of teaching if the learner is indeed human (or an animal). For AI, the assumptions these theories make typically do not hold. Nevertheless, it is worthwhile to figure out which theories do apply to which kinds of AI. For instance, approaches surrounding Vygotsky's zone or proximal development, where most learning occurs in tasks that are only just beyond the learner's current skill level, seem applicable to many different learning systems [14], and it may be possible to adapt or generalize Piaget's stages of cognitive development to the AI domain [3, 10].

Training is also closely related to the established ML subfield of computational learning theory, which concerns itself with the formal analysis of learning in AI systems. So far, it seems this has mostly been concerned with calculating bounds on how many interactions are necessary to achieve a certain level of performance. In addition to this, we are also interested in the content of those interactions, and the specifics of how the learner's knowledge changes.

3.5 Teaching

Teaching is what artificial pedagogy is all about: we want to analyze and design teaching strategies and interactions, using the other concepts and theories we discussed. A teacher should test the learner in order to obtain information that informs the way they proceed to train the learner by altering the task-environment from the learner's point-of-view.[5] This should all be done according to the constraints specified in the teaching task, and with the limitations on knowledge and capabilities of the teacher. It will likely combine knowledge from theories of testing and training to create environments that both allow the teacher to observe progress and encourage it—ideally simultaneously—and avoid adverse interactions between testing and training.

It would be valuable to be able to model and categorize teachers in relation to learners and task-environments. For instance, teachers can be visibly present or not (e.g. they can just change the environment without appearing in it). Or if they teach by demonstration, it may be important to consider how good they are at the task that is demonstrated and how similar their body is to the learner's.

There are many different teaching techniques that can be employed: e.g. heuristic rewarding, decomposition, simplification, situation selection, teleoperation, demonstration, coaching, explanation, and cooperation [2]. Using the other corners of the Pedagogical Pentagon, teaching theories should be able to tell us how to tailor these teaching techniques to different situations (i.e. learner-task combinations + constraints) and what results we can expect. Some more or less full curricula have been developed for teaching AGI, such as the AGI Preschool [3] and GoodAI's School for AI [11]. We believe these constitute important and highly promising pedagogical programs, that could be further improved with an even better understanding of the aspects of AP we have discussed.

[5] Note that if the teacher is in the learner's task-environment, every policy change alters the task-environment in some way.

4 Conclusion

We argue for the importance of artificial pedagogy for artificial intelligence and present a conceptual framework to aid in the structured and systematic study of this field. The Pedagogical Pentagon identifies five core concepts involved in pedagogical interactions: learners, task-environments, testing, training and teaching. The complexity of AP can be somewhat mitigated by studying one corner of the pentagon while keeping the others fixed. Partial theories of tasks and learners could possibly be made without reference to testing, training and teaching, and testing and training could (mostly) be studied in part without referring to teaching, but a complete understanding of all aspects of learning will not emerge unless the constraints that each of these put on the others are included in the picture. By organizing AP in this way we hope to facilitate the tractable study of this challenging domain, and provide a conceptual framework in which acquired knowledge can easily be organized.

References

1. Bieger, J.: Artificial Pedagogy: A Proposal. In: HLAI Doctoral Consortium (2016)
2. Bieger, J., Thórisson, K.R., Garrett, D.: Raising AI: Tutoring Matters. In: Proceedings of AGI-14. pp. 1–10. Springer, Quebec City, Canada (2014)
3. Goertzel, B., Pennachin, C., Geisweiller, N.: AGI Preschool. In: Engineering General Intelligence, Part 1, pp. 337–354. Atlantis Press (2014)
4. Hernández-Orallo, J.: The Measure of All Minds: Evaluating Natural and Artificial Intelligence. Cambridge University Press (2016)
5. Jordi Bieger, Kristinn R. Thórisson, Bas R. Steunebrink: Evaluation of General-Purpose Artificial Intelligence: Why, What & How. In: Evaluating General-Purpose AI 2016. The Hague, Netherlands (2016)
6. Kotseruba, I., Gonzalez, O.J.A., Tsotsos, J.K.: A Review of 40 Years of Cognitive Architecture Research: Focus on Perception, Attention, Learning and Applications. arXiv:1610.08602 [cs] (2016)
7. Krathwohl, D.R.: A revision of Bloom's taxonomy: An overview. Theory into practice 41(4), 212–218 (2016)
8. Legg, S., Hutter, M.: A collection of definitions of intelligence. In: Advances in AGI: Concepts, Architectures and Algorithms. vol. 157, pp. 17–24 (2007)
9. Marcus, G., Rossi, F., Veloso, M. (eds.): Beyond the Turing Test, AI Magazine, vol. 37. AAAI, 1 edn. (2016)
10. Piaget, J.: Piaget's Theory. In: Inhelder, B., Chipman, H.H., Zwingmann, C. (eds.) Piaget and His School. Springer Study Edition, Springer (1976)
11. Rosa, M., Feyereisl, J., Collective, T.G.: A Framework for Searching for General Artificial Intelligence. Tech. rep., GoodAI, Prague, Czech Republic (2016)
12. Steunebrink, B.R., Thórisson, K.R., Schmidhuber, J.: Growing recursive self-improvers. In: Artificial General Intelligence. pp. 129–139. Springer (2016)
13. Thórisson, K.R., Bieger, J., Thorarensen, T., Sigurðardóttir, J.S., Steunebrink, B.R.: Why Artificial Intelligence Needs a Task Theory — And What it Might Look Like. In: Proceedings of AGI-16. Springer-Verlag, New York, NY, USA (2016)
14. Vygotsky, L.S.: Interaction between Learning and Development. In: Mind in Society: The Development of Higher Psychological Processes, pp. 79–91. Harvard University Press, Cambridge, MA (1978)