Preparation Assignment

Is Instructionism always bad ?

Before I state my stance on Sweller's claims about Minimal Guidance, I believe I should first answer this question since a lot of what Sweller says seems to support instructionism and the theories we have read so far are all anti-instruction.

I believe instructionism is often necessary for high risk feilds such as farming, fishing, aircraft pilots, drivers, etc. A lot of communities of practice are also based on instructionism. For example, traditional dance communities are almost entirely instructive for the newcomers. Similarly, any martial arts community is instructive for the most part. These communities work well and the newcomers don't dis-identify with the practice.

While Dewey says that perplexity is a necessary component for experience, I believe it is "desire" or "incentive" that is the root cause of any experience. Most kids who learn how to play a piano piece really well or how to speed cube or even how to solve arithmetic really fast are not really perplexed as to how to do it. Instead they do it because "it's cool". While Dewey will not call it an experience since a speed cuber is not really going through any inquiry while his simple practice, it is almost certainly a *happy* and a *cognitive* change.

I am currently enrolled in a course for learning Blender. The concepts are taught to us in a bottom-up approach. We are first told how to do something, are made to walk through an example and then given assignments which we can do using the tools learnt in various ways. We are not given the assignment first and then told to figure out how to do it, thereby learning concepts. The reason for this is that everyone already has an incentive to study the procedures and tricks and don't need more incentive to do it. Unguided assignments would force to rely on what we know already, and while it may strengthen those concepts, they won't make us seek the optimal strategy. The problem that worked examples and instructions solve here is the expansive search space and the extra cognitive load associated with searching things by ourselves, choosing the best one, picking sources to read from, figuring out exceptions and easter eggs through unnecessary trial and error and overall struggling and feeling frustruated much more than we need to. From this list of bad effects that I gave, most happened with me when I tried learning it on my own some two years ago.

The message is clear; unguided learning causes extra (cognitive) load and often frustration, potentially causing dis-identification and believing the task is "too hard".

Coming back to my point about incentives; although when learning in a bottom-up approach (schema first, application later) the acquisition of schema is not as rewarding as compared to the Deweyan top-down approach and might even remove some sense of agency from the learner since he is now forced to forced to do things that are meaningless (abstract) in the short term (and will become very useful later), it is undoubtedly more efficient and provides a sense of security (structure) to the learner.

Guided discovery

While *unguided* discovery is obviously harder and tends to fail a lot, guided discovery does not necessarily lead to failure. I experienced guided discovery routinely when I was practicing for JEE. The worksheets I solved very well sequenced and I could (re)discover most of the 12th grade mathematics and physics all on my own simply by working through the questions. JEE problems are also known for employing problematisation a lot to "trip off" students with un-refined concepts. This unintentionally also serves the students since the practice questions are also similarly problematised and lead to perturbations, and perplexity and usually ends with either more steps added to an existing method (constructed by the students) similar to adding more code to an algorithm for edge cases (bricolage), refinement of concepts (similar to accomodation), or even picking up or rediscovering a new method or generalisation (in the ZPD).

This is a good point where I should point out that the studies cited by Sweller were all comparing unguided discovery of a concept much outside the ZPD with either worked examples or instruction.

One study in particular gave a physical setup to the two (guided and unguided) group and asked each to optimise for a particular effect based on some variables in the setup. The unguided group was unable to figure out how to model completely, but made some slight progress. The guided group on the other hand was given the model (the result of reflective-abstraction done by the *scientists* who created the model) and were successful in the task. The same two groups when asked to do the same thing again for a slightly different setup which *also* fit the model showed the same result again. The researchers concluded that "unguided dicovery doesn't work" while the real conclusion should've been "unguided discovery doesn't work when the sequencing is trash and the task is miles away from the ZPD and the guided group is spoon-fed the surface level practical schema for 100% efficiency in a narrow set of tasks without any concern for integration of this schema into their existing schemes" .

While Sweller uses the CLT argument for justifying this, saying that the unguided learners are overwhelmed due to the complexity of task, he is actually making a conclusion in similar fashion to a behaviourist. For him, increase in efficiency seems to imply increase in learning. He believes as long as the schema gets printed on the LTM somewhere, it counts as knowledge. But from what we know from the experiment done on the street-dweller kids who were asked questions involving mathematical concepts but framed as money management problems, this isn't the case. The context that this schema was learnt in matters. We know that most people maintain micro-identities which can be thought of as partitions in the full schema. For the kids who are "guided", they form a big split in their schemes approximately dividing it as "real life" and "schools and textbooks". We don't want this to happen. If the students are unable to incorporate the schemas achquired from instruction into their *working* , *real life* parts of schemes, then they will never believe in what they learn and will never have enough confidence to apply any of it to a new *kind* of problem.

Another issue is that just giving the end-to-end procedures as black boxes doesn't allow the student to learn the subroutines which are often more important and lead to easier transfer. Guided discovery on the other hand, forces the kids to discover them.

Worked example effect

Sweller mainly prescribes the modelling part of the cognitive apprenticeship and calls its effect as the "worked example effect". He warns us of the "Expert Reversal Effect" as well, which is when guidance to experts becomes hindrance rather than help; but that is very different from the critiques of instruction that constructivists give, which are mainly the issues of agency and improper integration of schema with the working schema (I will be using this self made term a lot since I have not other way to describe my thoughts).

As is evident from my account (Blender class), the worked example effect does decrease uncertainity, frustration and cognitive load on the WM.

Unfortunately, Sweller and company seem to be ignoring factors such as motivation arising *from* the education and not the other way around, and the fact that only worked examples create robots of students who are unable of critical thinking and praxis, and also things such as confidence in solving new problems.

While the worked example method works well for simple problems that require no decision making or real struggle, it won't ever work in the messy situations that experts actually go through.

Their paper states that the enterprise and the pedagogy for that enterprise shouldn't be expected to be identical and thus refute claims such as lack of validity of schooling knowledge in real world while not giving any account of *only* worked examples succeeding in closing the gap between theory and practice in the intellectually demanding fields such as STEM.

Structuring

Reiser describes 2 modes of scaffolding both of which are used a lot in university level textbook problems. Reiser is mainly focused on how to do these both using software (similar to Papert and Logo). Structuring is when the an outline of the process is laid out by the software itself which allows lower cognitive load at any given time on the student. Helper tools such as calculators and spell checkers also fall into the structuring category since they abstract away the tiny details that would've otherwise caused unnecessary cognitive load. Structuring reduces the solution search space by limiting attention only to the high level objects, and planning on how to use these high level objects/algorithms to do the task. He also puts "focus improving" scaffolds, which dis-allow wandering off into tangents and trying too many things and "monitoring" scaffolds such as timers, test score charts, per-problem timer, accuracy charts, etc. into the structuring category.

He also mentions structuring social interaction through the tool so that the students can understand the proper language for the practice. This is supported by Vigotsky's theory.

The best example of such structuring is StackExchange. The activity system has very explicit rules which are ruthlessly implemented. While this does deter many away from participating, it leads to incorporation of the rules as signs eventually.

By structuring, we not only limit the solution search space but also provide new schemas to the learner that they are to incorporate and use as the sequencing of the tasks proceeds.

As opposed to work examples, structuring is more general and doesn't necessarily spoon feed the learner allowing for some style and critical thinking to still take place.

Problematisation

While a common interpretation of it is "tasks that force you to reflect on what you did" it should not be taken too far since applying actual force; say grading the reflection for example just takes away the point of reflection. In that case, the learner will only use textbook (or worse, AI) reflections and not really learn. Instead, I would interpret problematisation as a "schema refining pertrubation". It's aim is to force critical thinking and refinement of the schema acquired and assimilated through structured problems or worked examples. It does match with the reflection/update part of the Deweyan cycle, but need not be an externalised, systematically written reflection. Instead it is more of an event or problem that leads to comparison and conflict between two (or more) methods or concepts learnt so far, leading to refinement of both. This refinement and accomodation is conscious and needs critical thinking. With enough of a perplexity, this may even lead to inquiry (asking peers, online community, etc.). I saw a lot of this happen during my time at the PACE coaching institute. As I already mentioned, JEE problems are known to be "tricky" (problematised) and this would often cause doubts that would be resolved with the help of the community and teachers.

My stance

My views align the best with Reiser's views. Moreover I also beleive that we already possess enough technology to build self-sustained learning environments as the ones he cited.