Using Network Analysis To Predict Learned Strategies For Web Development Tasks

Adam Lupu

Northwestern University

Using Network Analysis To Predict Learned Strategies For Web Development Tasks

Abstract

Analyzing learning outcomes using a network perspective is a relatively new field of study. An intuitive approach using existing social network theories (i.e. self-interest, contagion, cognitive knowledge networks, etc.) suggests the potential for many relevant social learning network structures. However, few have been successfully modeled and tested (exceptions include Rob, Stevce, and Andrew, 2002 and Aviv, Erlich, Ravid, and Geva, 2003). In this paper, I propose a method for predicting the strategies novice web developers learn and adopt based on their network connections. I begin by describing the specific bounded network I propose to study: a learning organization that trains prospective web developers using instruction, mentorship, and community engagement. I suggest a method of data collection that will allow researchers to observe this network over the course of three months. The data we aim to collect includes the actors, their position in the network, their relationships with each other, and their learned strategies. I then offer a formula for predicting learner strategy adoption based on their connections. This formula is constructed from hypotheses based on Situated Learning (Lave and Wenger, 1991) and Contagion theories in social learning (Bandura, 1986). I propose using this formula as a starting point to simulate fictitious networks and compare them to the observed network using the analysis software SIENA. Once completed, this research could stimulate theoretical advancement in the conjunctive application of Social Network Analysis and the Learning Sciences as well as offering testable hypotheses for the prediction of skill acquisition and strategic choice based on one's positioning in a learning network. Such advances could be used to develop training and professional development programs that target the specific strategies one wishes to see multiple learners acquire.

Motivation

Computer programming is a highly sought after skill, and industry demand for software developers with extensive programming expertise continues to grow (Prabhakar, Litecky, and Arnett, 2005). Until recently, aspiring programmers looking to fill this demand had two options: learn on their own or study at a college or university. Learning on one's own can be quite challenging, but learning in a university can be worse. Many students in undergraduate computer programming classes suffer from poor grades, frustration, and mental fatigue (Braught, Wahls, and Eby, 2011; Murphy, Fitzgerald, Hanks and McCauley, 2010). Additionally, at least one third of students enrolled in a college-level introduction to programming course will either fail or drop out (Bennedsen and Caspersen, 2007). Unfortunately, students who continue and graduate with degrees in computer science often lack necessary skills for employment, including communication and collaboration (Begel and Simon, 2008).

Perhaps in response to these findings, a third educational option for novice programmers has emerged. Commercial organizations like Code Academy and General Assembly offer training programs that teach a modern programming language and professional software development skills to beginners. Organizations like RailsBridge seek to entice women, who in 2009 received only 18% of undergraduate computer science degrees (NCWIT, 2011), into the software development industry. Yet, to date, there are no existing studies analyzing the learning that takes place in these organizations.

With computer programming education expanding beyond universities and do-it-yourself manuals, there is a growing opportunity for the development of more effective pedagogical practices. In this research proposal, I argue that a closer examination of the influence network effects have on the skills students learn and the strategies they employ to conduct tasks, can help improve the educational opportunities that universities and commercial training programs provide. To make this argument, I will begin with a description of the benefits network analysis can offer to this particular domain of learning.

Professional web development requires practitioners to perform computer-programming tasks, apply software engineering practices, and employ domain specific collaboration techniques. To accomplish these three objectives novice web developers must learn the knowledge and skills already available to experienced professionals. One form in which this learning occurs is through the interactions between novices and experts, or “new-comers” and “old-timers” (Lave and Wenger, 1991). Another feature of this particular domain is the plethora of different strategies practitioners can employ to accomplish similar goals. For instance, one expert web developer might use a certain set of computer programming commands to obtain the same results that another expert might use a different set of commands to obtain. As another example, two teams in the same office may use slightly different collaboration procedures, which eventually evolve into divergent cultural work norms.

If learners of a craft, such as web development, seek access to a community of practice, such as the professional software development industry, they must construct an identity consistent with that community (Lave and Wenger, 1991). In constructing this identity, they learn new behaviors, practice new habits, and adopt strategies that allow them to perform the tasks required of community participants. In this proposal, I focus on identifying which strategies learners adopt as a means of observing the learning process they experience during the transition from new-comer to community member.

If novice web developers are to become professional practitioners, they must navigate the pre-existing network of professionals and work strategies in order to find their own professional identity and make their own strategic choices. In order to understand how learners do this, we must identify how strategic choices are practiced and adopted. Social Network Analysis offers a way to observe learners as they enter a network of more experienced practitioners and adopt working strategies. Through network analysis we can examine and quantify how a learner's connections to experts, advisors, and peers changes over time and how their strategic choices shift and adapt along with their network relationships. Through Social Network Analysis we can attempt to identify what connection (if any) a learner's network relationships have with the strategies they learn to use. This helps us avoid attempts to interpret what is happening in the "black box" (the mind of the learner) (Hamlyn, 1990), enabling us instead to focus on the observable changes in their behavior, practices, and problem solving techniques.

Research Questions

In order to analyze the learning that occurs as a result of network ties, we need to define the network structure we wish to examine and the learning outcomes we wish to observe. I aim to study an existing network in a learning organization similar to those mentioned above. The proposed network of study is a bounded network (n=50) of web development students, their instructors, and mentors. During a three-month period, students form relationships with each other in the form of friendships and/or collaborative working relationships. They also have one formally assigned mentor and may receive informal advice from other mentors. The instructor provides directed advice, assistance, and training.

From a network perspective, the learners, mentors, and the instructor are the actors and the relationships they form comprise the ties. Therefore, any particular learner has an in-degree tie with the instructor, an in-degree tie with their formal mentor, the potential for in-degree or reciprocal ties with other learners, and the potential for in-degree ties with other mentors. Each mentor has an out-degree tie with at least one student, and the instructor has an out-degree tie with every student. The direction of the tie represents the flow of training and, as I hypothesize below, the flow of strategic choice. Below is an example graph of how this network might look if reduced to only a few nodes.

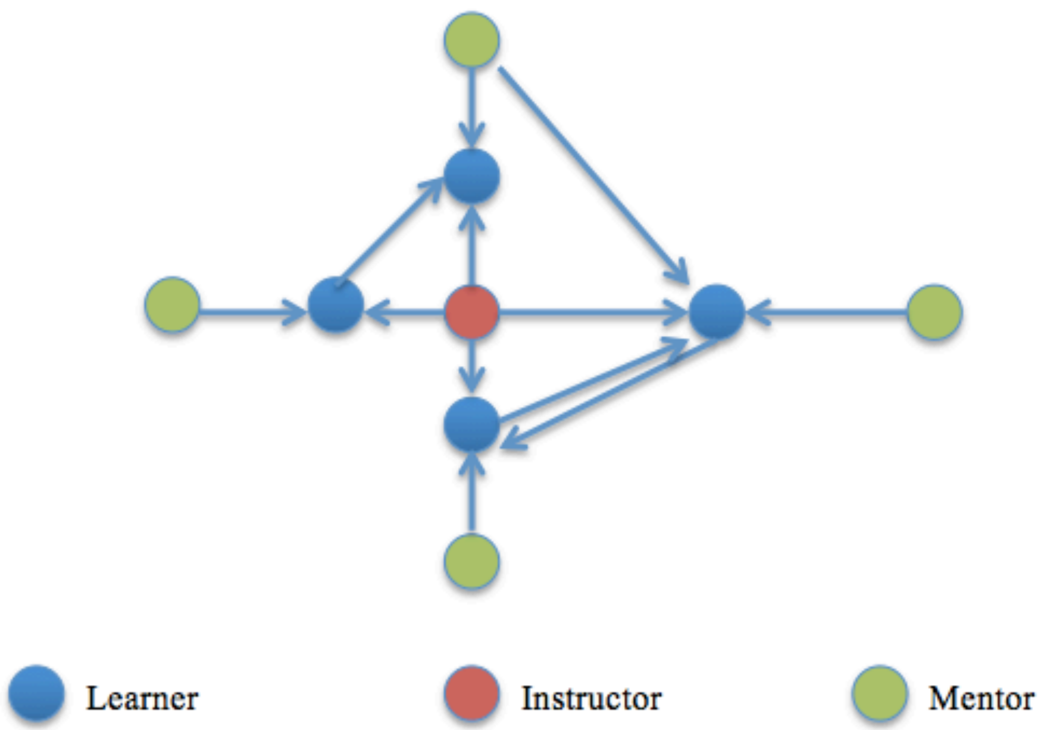

Figure 1 - An example of the types of actors and ties in the proposed network of study. *Note: this graph shows only nine actors and is not meant to represent the complete network of 50 actors proposed for this study. The arrows represent an influence relationship between the two nodes and the direction of influence. Notice that learners can influence each other but not their mentors, the instructor influences each student, and mentors may influence more than one student.

Since the same web development tasks can be completed using several different strategies, and since strategy awareness and strategic preferences vary throughout the industry, the instructor and each of the mentors may prefer differing or similar strategies. Therefore each node has an organizational positioning attribute (i.e. learner, mentor, instructor) and a strategy attribute vector (indicating the strategies adopted by that individual).

The goal of this research is to predict how the strategic preferences of a learner's network connections influence the strategies that learner adopts over time. In other words, *can a learner's network ties predict his or her chosen strategies when asked to complete a recently learned task?* While examining this network to answer this question, I propose the following hypotheses. 1) Learners have a higher likelihood of adopting a strategy if their formal mentor employs that strategy. In other words, strategies are contagious from mentor to learner. 2) Learners are more likely to adopt a strategy if they have ties to other students who adopt that strategy. In other words, strategies are contagious through physical and digital proximity with fellow students. 3) Learners are more likely to adopt a strategy if the instructor employs that strategy. In other words, strategies are contagious from instructor to student. 4) The more popular a strategy is within the network, the more likely each student is to adopt that strategy. In other words, strategic homophily increases throughout the network over time.

In order to test these hypotheses, we must identify existing and emergent network ties, we must identify the strategies each actor adopts (or has already adopted), and we must construct a model to test the goodness of fit against the observed network. I will now propose a method of data collection and analysis for achieving these three objectives.

Methods

As noted above, the network under consideration is comprised of several different ties amongst different types of actors. Each of these ties would need to be tracked in order to obtain a complete picture of an individual learner's network at any given time. Since interactions can occur both online (chat rooms, instant messaging, video conferencing, etc.) and offline (face to face meetings, classroom instruction, office hours, etc.) we must combine both observational and digital trace data. We will begin by observing the ties between learners. This data can be conveniently collected due to the nature of student classroom interactions. During class, students “pair” with each other to take notes on the lecture and complete lab activities. These pairing will be recorded, and the number of times two students pair with each other will represent the strength of their tie. To supplement this data, we will have students complete a survey asking them to identify from a list of students in the class, whom they shared or received information from during the class. If we also ask students to identify who they think shares or receives information from whom, as in a Cognitive Social Structure (Krackhardt, 1987), this data could be mined in the form of a row-dominated LAS (to determine who learners believe they influence), a column-dominated LAS (to determine who learners identify as their own influencers) or a consensus structure (to identify who a majority of the network believes influences whom). With the survey data and the observational data, a threshold of evidence could be established to determine the influence relationships that exist amongst the learners. Any relationship for which that threshold is reached can be said to exist. Any relationship that fails to reach this threshold can be ignored for the purposes of this study. This threshold would have to be determined after making all observations and collecting all surveys as it would be relative to the signal to noise ratio in the data collected.

To determine the ties between students and mentors, we will first construct a matrix of formal student mentor relationships. This will contain mostly null ties since each student is formally assigned only one mentor. A second matrix of informal offline interactions between students and mentors is also required. In order to create this matrix, both students and mentors will need to be surveyed to assess the offline, informal ties students create with mentors as well as the mentor/mentor ties that, although rare, may also exist. Finally, a matrix of online ties is required to document the connections students form with each other and with mentors online. This will require taking digital trace data of online interactions that occur in established classroom chat rooms and online forums, as well as blog commenting and Twitter "following." We will supplement the digital trace data by asking students (on the aforementioned surveys) whether they connected online, offline, or both with each member of the network.

In addition to defining the network ties, we must also identify the programming strategies actors in the network use to perform web development tasks. These strategies can consist of using pre-written code borrowed from other developers (referred to in this network as "gems"), using specific commands or lines of code they themselves write, or overall approaches to designing the models and associations they construct in order to complete the task. Since each strategy is completely new to the novice student, and each student learns more strategies than can be tracked, I propose identifying five core strategies that we can easily test for. The strategies I propose testing for include debugging procedures, user account management methods, database management methods, programming environment preference, and user-story-to-coded-feature workflow. These specific procedures, methods, preferences, and workflows are likely to vary amongst members of any web development network. Since they are also strategies required by professionals and taught to learners as part of the curriculum of this course, they can be easily observed through administered testing.

In order to test for a network actor's debugging procedures, we will design a simple debugging task and have them articulate their process while solving it. To observe user account and database management methods, we will administer a task involving the creation and setup of each. To observe the user-story-to-coded-feature workflow we will have each actor code the same user story and identify any variations. To identify an actor's programming environment preference we will ask them to show and describe their preferred software choices and any modifications they have made to each software tool's default setup. After network members complete these tasks, our team of researchers will code the results and identify the strategic choices each actor made. To ensure that we receive comparable results from each member, we will ask every member of the network, including learners, mentors, and the instructor to perform the same tasks, as well as asking the research team to perform inter-rater reliability checks when coding the results. We will then construct an attribute vector of strategy choices for each actor based on their resulting task output. Therefore, in addition to a scalar positional attribute (i.e. mentor, learner, instructor), each actor in the network will be assigned a vector of strategic attributes representing their five observed strategy choices.

Since the network changes over the course of three months, and since the number and quality of ties increases over time, network data must be gathered longitudinally. In considering when to perform the testing mentioned above, I propose focusing on three distinct times, t_0 (representing the time right before class begins, t_1 (representing the class midpoint, approximately 6 weeks after t_0), and t_2 (representing the conclusion of the class, 12 weeks after t_0). To determine the structure of the network at t_0, we must assess which learners may be already connected to each other, which mentors may already be connected to each other, and the strategy vectors for the mentors and the instructor. The strategy vector for each learner is not necessary to obtain at this point since it is almost certain to be null do to their lack of experience in the discipline. Therefore, at time t_0, we will conduct an initial survey and administer the task test to the mentors and the instructor. Between times t_0 and t_1 we will observe student pairing, survey and digitally trace informal ties among students and between students and mentors, and record formally assigned student/mentor ties. At time t_1 we will ask students to perform the task and assign strategy vectors to each. Between times t_1 and t_2 we will continue collecting data on network tie formation and dissolution as we did between times t_0 and t_1. Finally, at time t_2, we will ask everyone in the network to once again perform the task and update each actor's vector of strategic choices.

---Table 1: Longitudinal Data Collection Strategy---

	t_0	Period 1 ($t_0 \rightarrow t_1$)	t_1	Period 2 ($t_1 \rightarrow t_2$)	t_2
Learners	• Survey existing connections • Identify formal mentor	• Observe pairing and collaboration • Collect digital trace data	• Survey changing connections • Test for strategies	• Observe pairing and collaboration • Collect digital trace data	• Survey changing connections • Test for strategies
Mentors	• Test for strategies				• Test for strategies
Instructor	• Test for strategies				• Test for strategies

Having identified the network ties and actor attributes at these three times, we can operationalize the hypotheses present above and construct a network model for testing. The formula for the above hypotheses suggests that learners adopt strategies based on their ties with fellow learners, their ties with mentors, the instructor's strategic choices, and each strategy's popularity within the network. Let $\boldsymbol{L}_i$ be the vector containing the strategic choices learner *i* adopts, let $\boldsymbol{I}$ be the vector containing the instructor's strategic choices, and let $\boldsymbol{M}_j$ be the vector containing the strategic choices of mentor *j*. Finally, let $\boldsymbol{C}$ be the matrix containing the commonality ratio (or popularity quotient) of each strategy. In this case, in order to test the hypotheses above we set:

$$L_i = function[(\sum_j L_{ji})(\sum_k M_{ki})(M_i)(I_i)(C)]$$

Where L_{ji} is the strategic influence learner j has on learner i (zero if none exists), and M_{ki} is the strategic influence mentor k has on learner i (again, zero if none exists). M_i is the strategic influence of the learner's formal mentor, and I_i is the influence relationship the instructor has with learner i.

We must also parameterize this equation so that we can account for the differing amount of influence a mentor might have compared to the instructor or how total exposure time to any influencer in learner i's network might change the degree of influence that person has on learner i. To perform this parameterization we introduce an element multiplier to each vector, a coefficient that is itself a function of the classification of the influencer in relationship to the learner (i.e. formal mentor, instructor, informal mentor, peer) and the time spent learning from that influential individual. Let P_L be the influence coefficient learners have on each other per unit time spent interacting, let P_M be the influence coefficient formal mentors have on learners per unit time, let P_m be the influence coefficient informal mentors have on learners per unit time, and let P_{Ins} be the influence coefficient of the instructor on learners. Extracting these coefficients from the above equation and rendering them visible yields:

$$L_i = function[(\sum_j P_L L_{ji}\, t_{ji})(\sum_k P_{iM} M_{ki}\, t_{ki})(P_M M_i)(P_{Ins} I_i)(C)]$$

In order to test this model against the observed network, we will also have to account for structural zeros, i.e. the types of connections that simply would not occur in this network. These relationship limitations include: learners cannot influence themselves, a learner can only have one formal mentor, and learners cannot influence mentors or the instructor. The network analysis software SIENA allows for the creation of structural zeros such that all simulated networks can be made to obey the same relationship rules as the existing network.

Obviously, this formula is just a first attempt at modeling the observed network. It is likely that several iterations will be required to define the function more fully and identify the coefficients. Indeed, the formula itself may have to be altered at the theoretical level to better model the observed network. There are several theoretical concerns to look for once the data is acquired. First, we must attempt to observe any indication of the "principle of reflected exclusivity" (Krackhardt and Brass, 1994). Is this network tending toward strategic homogeneity or do strategies coalesce amongst those who interact with each other often? Does less time spent with others in the network lead to observably less influence? Second, we must seek to identify learners who may require a threshold number of influencers before they adopt a strategy (Granovetter, 1978). If such effects occur in this network, our model might over predict certain strategy choices. Some learners may not interact with enough members of the network to achieve any specific working strategy and therefore prove unable to perform the tasks we assign them. Third, while learners in this specific network are all novices to web development strategies, some of them may have experience in related fields. Will this create an unequal distribution in the degree of novelty learners experience during the training? Rice (1993) argues that such an inequality might lead to varying levels of receptivity to contagion effects. We may have to account for learners' previous experience or lack thereof. Finally, we may also want to examine the possibility that learners who attempt a strategy and have difficulty adopting it might turn away from that strategy altogether. This would be a manifestation of what McGuire (1966) refers to as "inoculation theory." Might this create an anti-contagion effect? For instance, a mentor that struggles to communicate effectively with their assigned student might have a higher probability of influencing that learner away from their chosen strategy as opposed to helping them adopt it. I propose examining whether any of these effects are observable in the data and adjusting the model accordingly.

Conclusion

I have proposed an amalgam of Social Network Analysis and Learning Science research to study the socially influenced adoption of learned strategies in a professional web development network. Additionally, while not proposed here, this method for studying a learning network could also be reversed, starting with a measurement of existing strategic choices and predicting the influence relationships that exist in the network. Such a study would be a natural continuation of this line of inquiry. While either approach requires the collection of a significant amount of data and allowances for several theoretical considerations, the potential impact and broader implications of the research are well worth the effort. First, this research represents a multi-discipline approach to understanding learning networks, harnessing the quantitative evaluation of Social Network Analysis and the qualitative analysis inherent to the Learning Sciences. Currently, research attempting to link these two fields is rare. Second, a theoretical mechanism for predicting skill acquisition and strategy adoption in professional communities represents a significant advancement to employee development and training programs. Third, such a theory, if verifiable, would be a valuable contribution to the study of organizational management as it offers a more deliberate approach to the distribution of skills and preferred strategies within organizations. If this research is successful, it will lay the foundation for future design-based research to help organizations optimize how they form and manage learning networks, how they pair mentors with learners, and how they plan their trainings and educational efforts.

References

Aviv, R., Erlich, Z., Ravid, G., & Geva, A. (2003). Network Analysis Of Knowledge Construction In Asynchronous Learning Networks. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.2.9044

Bandura, A. (1986). Social foundations of thought and action: A cognitive social theory. *Pretince Hall, Englewood Cliffs, New York*.

Begel, A., & Simon, B. (2008). Struggles of new college graduates in their first software development job. *Proceedings of the 39th SIGCSE technical symposium on Computer science education*, SIGCSE '08 (pp. 226–230). New York, NY, USA: ACM. doi:10.1145/1352135.1352218

Bennedsen, J., & Caspersen, M. E. (2007). Failure rates in introductory programming. *SIGCSE Bull.*, *39*(2), 32–36. doi:10.1145/1272848.1272879

Braught, G., Wahls, T., & Eby, L. M. (2011). The Case for Pair Programming in the Computer Science Classroom. *Trans. Comput. Educ.*, *11*(1), 2:1–2:21. doi:http://doi.acm.org/1921607.1921609

Cross, R., Borgatti, S., & Parker, A. (2002). Making Invisible Work Visible: Using Social network Analysis to support strategic collaboration. *California Management Review*, *44*(2), 25-47. doi:10.1093/0195165128.003.0006

Granovetter, M. (1978). Threshold models of collective behavior. *American journal of sociology*, 1420–1443.

Hamlyn, D. W. (1990). *In and out of the black box: On the philosophy of cognition*. Basil Blackwell.

Krackhardt, D. (1987). Cognitive social structures. *Social Networks*, *9*(2), 109–134.

Krackhardt, D., & Brass, D. J. (1994). Intraorganizational networks. *Advances in social network analysis*, 207–229.

Lave, J., & Wenger, E. (1991). *Situated learning: legitimate peripheral participation*. Cambridge University Press.

McGuire, W. J. (1966). Attitudes and opinions. *Annual review of psychology*, *17*(1), 475–514.

Murphy, L., Fitzgerald, S., Hanks, B., & McCauley, R. (2010). Pair debugging: a transactive discourse analysis. *Proceedings of the Sixth international workshop on Computing education research*, ICER '10 (pp. 51–58). New York, NY, USA: ACM. doi:10.1145/1839594.1839604

Prabhakar, B., Litecky, C. R., & Arnett, K. (2005). IT skills in a tough job market. *Commun. ACM*, *48*(10), 91–94. doi:10.1145/1089107.1089110

Rice, R. E. (1993). Using network concepts to clarify sources and mechanisms of social influence. *Progress in communication sciences*, *12*, 43–62.