# Using Automatic Detection and Characterization to Measure Educational Impact of nanoHUB


Michael Zentner
HUBzero
Purdue University
West Lafayette, IN, USA
mzentner@purdue.edu

Nathan Denny
HUBzero
Purdue University
West Lafayette, IN, USA
ndenny@purdue.edu

Krishna Madhavan
Dept. of Engineering Education
Purdue University
West Lafayette, IN, USA
cm@purdue.edu

Swaroop Samek
Network for Computational Nanotechnology
Purdue University
West Lafayette, IN, USA
ssamek@usc.edu

George Bunch Adams III
Dept. of Computer Science
Purdue Universtiy
West Lafayette, IN, USA
gba@purdue.edu

Gerhard Klimeck
Network for Computational Nanotechology
Purdue University
West Lafayette, IN, USA
gekco@purdue.edu



***Abstract*— The science gateway and online community nanoHUB hosts over 4000 technical resources related to nanoscience and nanotechnology and online capabilities for nano community engagement. nanoHUB also hosts over 500 online simulation tools. nanoHUB serves the nano community spectrum ranging from undergraduate students to high profile researchers. In this paper, the evolution of nanoHUB online simulation is discussed along with the impact of that simulation on student behavior. With over 52,000 simulation users, the nanoHUB team is not personally aware of most new classrooms that adopt simulation in their syllabi. Yet, these classroom users feed the next generation of nano community contributors. A method is presented to detect classroom by clustering coordinated behavior among simulation users, thereby automatically detecting adoption of simulation tools in a classroom environment. Several prototypical patterns of clustered behavior are analyzed, ranging from peripheral to systemic classroom integration of simulation. Visualizations of detailed user behavior illustrate the varying behavior structures. Between the fall of 2000 and the fall of 2011, in 846 clustered behaviors have been detected. This number of classroom settings is on a continuous growth trend as nanoHUB becomes more widely adopted. A discussion on the rate of adoption of published simulation tools in clustered behaviors is presented.**



***Keywords—science gateway, nanoHUB, data analytics, engineering education, user behavior analytics***


## I. INTRODUCTION

Personalization and demonstration of educational impact in scientific communities or scientific cyber-environments requires an understanding of how people learn and conduct science. In this paper, we define "cyber-environments as a collection of computational, visualization, and data management resources presented to an engineering community through an easy-to-access and easy-to-use online portal" [1]. Another definition from the online science gateway perspective which is highly consistent with the definition we adopt can be found in [2]. They define science gateways as "a framework of tools that allows scientists to run applications with little concern for where the computation actually takes place."

Regardless of which definitions of online scientific communities we consider, it is clear that community formation around scientific grade simulation tools and the ability to perform complex computations are important ingredients. It is in this context that personalization plays an extremely critical role. A clear understanding of user behavior within the online scientific community will allow better feature and content design as well as provisioning of computational and data resources. This paper is positioned in the context of real data collected based on observation of a large set of users in one of the most successful engineering cyber-environments, nanoHUB.org [3,4].

## II. NANOHUB.ORG: OUR EXPERIMENTAL APPARATUS

nanoHUB is an online scientific virtual organization serving the nanoscale engineering and science communities, and is operated by the National Science Foundation funded Network for Computational Nanotechnology (NCN). NCN has been funded by the US National Science foundation since 2002 as a national resource, while nanoHUB had been in existence since 1998. At the time of this writing, nanoHUB serves an extremely large community of over 1,400,000 unique users (a user is someone who either downloads a resource, or spends in excess of 15 minutes on the site) worldwide on an annual basis. The number of users participating in the nanoHUB community has been growing at a rapid pace over the past decade. nanoHUB hosts over 4,000 technical resources that include animations, courses, learning modules, notes, presentations, publications, series, teaching materials, and workshops. More importantly, nanoHUB also hosts 400+ online simulation tools. On an annual basis, these simulation tools serve more than 12,000 users running over 500,000 simulations.

The nanoHUB delivery mechanism is unique. Simulation users do not need to download or install code locally. Instead, they can run simulations directly in the browser, accessing a variety of back-end compute clusters provided through nanoHUB. Such simulations are often integrated components of student coursework and provide a modality of interactive exploration not available with presentation or publication only resources. Because these tools are delivered in a browser, the barrier to use is eliminated. These interactive tools invite users to ask "What if?" questions where multiple runs can be compared without data downloads. In short, students interact with the site, rather than simply downloading resources for offline use.

While the value proposition of interactivity and deeper learning for students is clear, nanoHUB relies heavily on


This work was supported with funding from NSF grants EEC-0228390, EEC-0634750, OCI-0438246, OCI-0721680, EEC-0956819, and EEC-1227110.

Presented at Gateways 2018, University of Texas, Austin, TX, September 25–27, 2018.
https://gateways2018.figshare.com/

contributions from other expert members within the community to sustain a steady pace of new materials. Anyone can become a contributor of simulation tools or other resources. As such, there must also be a value proposition for the creators of these new materials. This value is delivered through impact measurement. Content creators and contributors are able to see how their materials diffuse through the research and education communities and utilize these measurements to promote their impact in their field. These two elements are what makes nanoHUB unique in the field of online scientific research and education facilities and are a focus of this paper.

## III. Patterned Simulation: A Fingerprint of Students

Based on anecdotal evidence, the nanoHUB team was aware that a non-trivial component of the simulation use was due to students using simulation tools for the first time in their coursework. We hypothesized that in addition to our anecdotal evidence, there were many types of classroom use of simulation tools of which we were not aware that occurred spontaneously in our online community. We undertook the effort to systematically and automatically recognize these student characteristics as part of our impact measurement system.

Our first step in this process was to manually substantiate this hypothesis. One method for this was to examine self-declared registration information. When nanoHUB users create an account, they are asked a series of questions regarding their affiliation and demographics. A quick perusal of information provided voluntarily by users indicated that many people stated they were uncomfortable supplying personal data, others misstated their institutional affiliation as evidenced by comparing the location of the institution to the geographic location of the IP address originating the registration, and yet others took the opportunity to provide obviously false and sometimes humorous responses.

Deciding not to rely on self-declared information, we therefore developed a visualization we call raindrop plot. The raindrop plot aligns individual users along the vertical axis and time discretized in daily buckets along the horizontal axis. When a user activates a tool on a given day, a dot is recorded in the raindrop plot. Dots are colored to indicate which tool has been accessed. Dots may also assume multiple colors in the event more than one tool was activated by any given user on a given day. The result is that each row corresponds to a picture of a user's longitudinal usage record over time, as shown in Figure 1. Stacking users on top of each other then allows one to visually scan for patterns. The example in Figure 1 shows what appear to be similar patterns between User 1 and User N, but uncorrelated patterns with respect to User 2.

A raindrop plot based on a semester's worth of activity, with users sorted in such a manner that users who make the earliest to latest appearance in the semester are arranged from top to bottom, is shown in Figure 2. By visual inspection, there appear to be two vertical sections of the plot that exhibit coordinated simulation tool usage behaviors. Mixed in these vertical sections are users who do not appear to belong to the pattern as well. Examining the geographic location of the IP numbers of users in some of the more prominent visually identified patterns allowed us to contact instructors at institutions in those geographic locations to confirm our classification and identification. Analyses of our data from the confirmation studies showed that nanoHUB tools were being used in classroom settings as hypothesized.

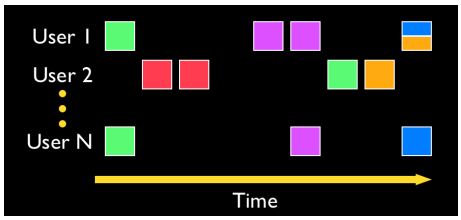


*Figure 1 – Example Raindrop Plot that visualizes the temporal usage of different nanoHUB tools (indicated by a different color) as a function of time, where each dot corresponds to activity on a specific day. Different users are indicated through different rows. In this specific example User 1 looks somewhat similar to User N, while User 2 does not bear much similarity to either User 1 or User N.*

## IV. Methods, Theory, and Calculations

To automatically detect and identify clustered users, several existing algorithms were considered and new algorithms developed. Existing algorithms fall into two broader categories based on k-means and hierarchical clustering methods. A key aspect of such algorithms is the focus on partitioning a set of N members into M clusters based on some differentiating feature of the members. However, in the case of nanoHUB, patterned classroom use is exhibited in only a fraction of the N members. The remaining members of N are not differentiable by the same criterion. The focus then needs to be selecting an appropriate unknown number of clusters out of the N members, and leaving the members of N that do not fit in any cluster as unclustered elements. Therefore, customized algorithms were created. The first new algorithm is the calculation of a similarity measure between individual users. With pairwise similarities calculated, a novel clustering method then was created to group like users together. Each of these methods is described in more detail in the sections below.

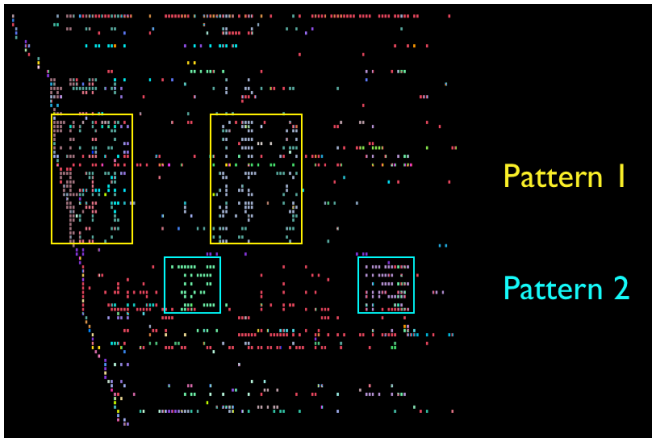


*Figure 2 – Section of a Semester-Long Raindrop Plot for 98 users. Each row indicates shows the longitudinal tool invocation pattern for an individual user. The boxed areas show groupings of users that appear to have similar, but not identical patterns. Such visual evidence is the basis for development of user-user similarity metrics.*

### A. Similarity Measure

One of the first stages of user identification is to classify users who exhibit similar usage characteristics into groups. The key challenge in computing similarity is that any two longitudinal user patterns typically differ both in temporal and in sequential features. We compared similarity measures based on variants of cosine similarity and common tool

sequence determination with little success. An alternative way to determine the similarity between two users is to determine a set of transformations that once undertaken will result in two users whose longitudinal records appear identical. Penalties incurred for each transformation sum to an overall edit distance. This is similar in concept to what Levenshtein edit distance [5] does for pairs of strings. However, our method is not based on the same type of data structure and is not guaranteed to provide an optimal edit distance.

The algorithm begins by representing the activity of user $x$ as a vector $U_x$ of vectors $U_{xd}$ that each contain a list of the tools invoked by user $x$ on day $d$ within an interval of time the first and last days of which are $D_{min}$ and $D_{max}$. Three types of edit transformations are then evaluated.

The first edit transformation involves shifting a tool execution $t$ from a vector $U_{xn}$ to a vector $U_{xd}$ where $d < n$. This transformation carries a penalty of $S_{xdt}$:

$$S_{xdt} = p_s(n-d) \tag{1}$$

$$n > d \wedge t \in U_{xn} \wedge \neg\exists m: (m > d \wedge t \in U_{xm} \wedge m < n) \tag{2}$$

The qualification (2) ensures that there is no vector $U_{xm}$ that contains an instance of tool execution $t$ where $m$ is closer in time to $d$ than $n$. The term $p_S$ is an empirically determined constant.

The second edit transformation is a neighbor insertion. It allows the insertion of a tool execution $t$ into a vector $U_{xd}$ provided that there is an execution of tool $t$ in at least one other vector $U_{xn}$. In other words, if the user already has an execution of tool $t$ in their longitudinal record, another one can be inserted elsewhere in the record with a given penalty $N_{xdt}$:

$$N_{xdt} = p_N|n-d| \tag{3}$$

$$\mathrm{n}: \mathrm{t} \in \mathrm{U_{xn}} \neg\exists \mathrm{m}: (\mathrm{m} \neq \mathrm{d} \wedge \mathrm{t} \in \mathrm{U_{xm}} \wedge |\mathrm{m}-\mathrm{d}| < |\mathrm{n}-\mathrm{d}|) \tag{4}$$

The qualification (4) ensures that the nearest neighboring set $U_{xn}$ in time to $U_{xd}$ that contains an execution of tool $t$ is chosen such that the minimum penalty would be assessed for the insertion. The term $p_N$ is an empirically determined constant.

The third edit transformation involves a spontaneous insertion of an execution of tool $t$ in a vector $U_{xd}$ without respect to any neighboring sets that also contain executions of $t$ at a penalty $I_{xdt}$. A simple expression of this penalty might be as follows:

$$I_{xdt} = \frac{p_I\left(D_{\max} - D_{\min}\right)}{\left|\bigcup_d U_{xd}\right|} \tag{5}$$

The term $p_I$ is an empirically determined constant. The rationale behind such a penalty is that if the period of time over which users are being compared is large, and the number of unique tools they use during that period is small, then there should be a large penalty for inserting the execution of a different tool. The reason is that a new type of tool is a somewhat fundamental change in behavior. Conversely, if two users are each using ten unique tools, there should be a much smaller penalty for inserting an eleventh tool since the significance of the eleventh tool is diluted by ten other tools. By itself, such a ratio would define a penalty response defined by a single curve. Because such a ratio by itself is too restrictive for assessment of penalties by allowing only one curve shape, discontinuities are introduced into the penalty function as follows:

$$I_{xdt} = \frac{p_I\left(D_{\max} - D_{\min}\right)}{b\min\left(T, \left|\bigcup_d U_{xd}\right|\right) + c\max\left(0, \left|\bigcup_d U_{xd}\right| - T\right)} \tag{6}$$

The terms $T$, $b$, and $c$ are empirically determined constants. The first term in the denominator of (6) adds a dampening effect to the penalty, enforcing a minimum value of the penalty once a given number of tools used is surpassed. The second term in the denominator of equation (6) allows us to shape to the dampening effect, providing a tunable penalty curve. The overall cost to transform two users into identical users can then be expressed as $E_{ij}$:

$$E_{ij} = \sum_{d=D_{\min}}^{D_{\max}} \left( \sum_{t \in U_{jd} - U_{id}} M_{idt} + \sum_{t \in U_{id} - U_{jd}} M_{jdt} \right) \tag{7}$$

$$M_{xdt} = \min(S_{xdt}, N_{xdt}, I_{xdt}) \tag{8}$$

This sum is determined by moving forward one day at a time. On each day, the tools executed by each user are examined. For each tool executed by one user and not the other, each of the three possible move types is examined and the least expensive applied. Because it marches forward linearly in time and greedily chooses the cheapest localized option, the algorithm does not guarantee that the set of transformations chosen is the optimal cost for transforming the two users.

### *B. "Nucleate and Merge" Cluster Assembly*

With user-user similarity determined, clustering can be performed. The nucleation phase of clustering is to assemble a set of highly similar users around each user, in effect creating a cluster of similarity for each user. Cluster $C_i$ is assembled around each user $P_i$ using an edit distance threshold of $H$ by successively taking the union of sets that are initially of singleton membership.

$$C_i = \{P_i\} \bigcup_{x:E_{ix}<H} \{P_x\} \tag{9}$$

Next, clusters $C_j$ are subsumed when they are completely contained within other clusters $C_j \subseteq C_i$. Finally, the merge phase of the algorithm occurs when cluster $C_j$ is merged with $C_i$, and $C_i$ is removed from the set of clusters $C$ under the following conditions:

$$C_i = C_i \cup C_j \tag{10}$$

$$C = C - C_j \tag{11}$$

$$i,j: |C_i \cap C_j| \geq q|C_j| \tag{12a}$$

$$\wedge(\neg\exists m: m \neq i \wedge |C_m| > |C_j| \wedge |C_i \cap C_m| \geq q|C_m|) \tag{12b}$$

$$\wedge(\neg\exists k,l: k \neq l \wedge k \neq i \wedge |C_k| > |C_i| \wedge |C_k \cap C_l| \geq q|C_l|) \tag{12c}$$

The first condition (12a) ensures that the degree of overlap between clusters $C_i$ and $C_i$ is at least some fraction of the membership in $C_i$ where such fraction is determined by an empirical constant $q$ over the interval *(0,1]*. The second condition (12b) ensures that there is not another cluster $C_m$ as an alternative to $C_j$ that is larger than $C_j$ and also meets the threshold condition. The third condition (12c) ensures that

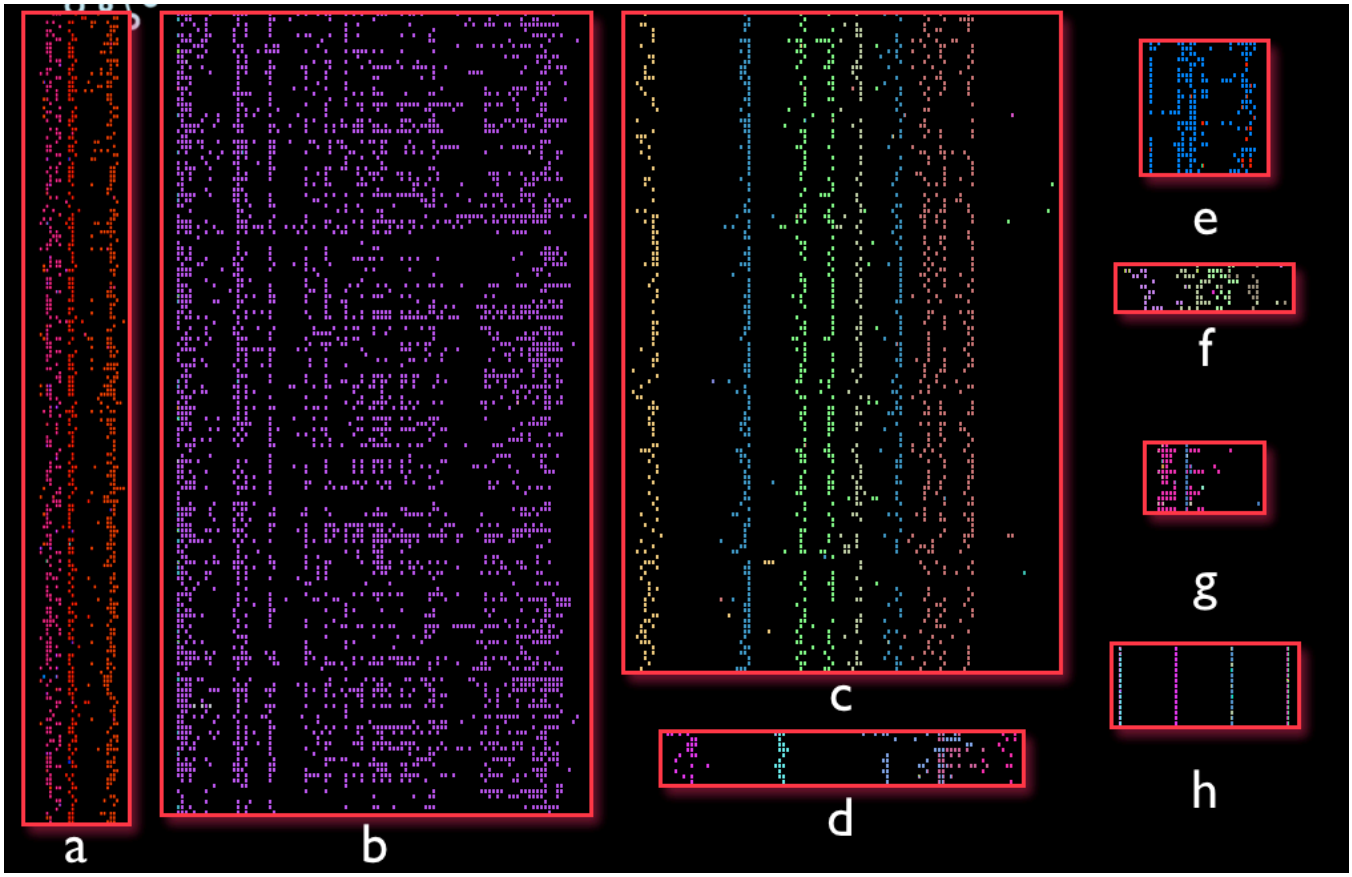


*Figure 3 – Raindrop Plots of Specimen Clusters. This is a sampling of the 846 clusters detected, showing a variety of behavioral patterns ranging from single to many tool use, short to long time duration, sparse to dense activity patterns, and highly synchronized to marginally coordinated use.*

there is not another pair of clusters $C_k$ and $C_l$ that could be merged starting with a larger cluster than $C_i$. After each merge, all remaining clusters are again tested for being subsumed. The effect is to recursively merge the sets that will result in the largest merges first. This allows longitudinal patterns that, as a pair, may not seem similar to be classified as similar by virtue of bridging members in their respective clusters. Users A and B may not by themselves be similar enough to be clustered together, but a user C who has strong similarity to both may define a bridge through which A and B can be clustered together. A conventional clustering algorithm would separate these two sets. However, it is our goal to bring together as many similar behaviors as possible. Here, bridging members are the connection between what might visually be recognizable as similar behaviors but which would otherwise be mathematically ignored. Merging is repeated until no pair of clusters can be found to merge.

## V. Results and Discussion

The above methods were run on data starting from Fall 2000 continuing through Fall 2011. The time intervals chosen were [January 1, June 30] and [July 1, December 31] for each year to roughly correspond to semester by semester schedules. A total number of 846 clusters were detected during that time period.

Bulk measures like those shown above indicate that newcomers are being served in structured classroom settings and that this trend is growing as the number of tools on nanoHUB grows. However, they do little to provide a picture of what is happening within those clusters. Using the raindrop plot to examine some of the generated clusters indicates that there are several important features that characterize clusters. Figures 3a-h show several specimen clusters. Visual examination of these clusters indicates that several attributes help to classify the variety of classroom behaviors:

- Number of participants – Figure 3a shows a cluster with a very large number of obviously coordinated participants. Figure 3f shows a very small number of participants but clearly they also exhibit coordinated behavior.
- Number of tools used – Figure 3c shows the largest variety of tools used. Figure 3b shows only a single tool with a few sporadic instances of other tools, used intensely over a long period of time.
- Time span over which the tools are used (cluster duration) – Figure 3b and Figure 3c show use over a long period of time. Other clustered behaviors last for only one day.
- The intensity of usage (density) – Figure 3b shows very intense usage of a single tool. Figure 3d is much less intense. However, Figure 3d begins with a fairly regular pattern and at the end becomes less coordinated. This may be indicative of a switch from

structured coursework to end of semester self-driven projects.

- Sequence of tool use – Most of the plots show a consistent sequence of tool use as the cluster members switch from one tool to the next. However, Figure 3g shows some users altering their sequence relative to other users.
- Time synchronization – Figure 3h shows a highly time synchronized pattern, as though a group of newcomer students were in a lab together, all executing the same tools during that lab session.

## VI. Conclusions

We continually measure impact in new and innovative ways with the goal of engaging students and incentivizing them to create content in the future. In this paper, we have discussed one of these key measurement methods where students are detected by virtue of their behavior, not their own declarations about themselves. With the presented detection method, we have demonstrated that:

- Students participate in a large number of clustered classroom style settings.
- Student participation is on a continually increasing trend.
- The structure of these clusters is highly varied and may be characterized by a large number of descriptive dimensions.
- The size of clusters varies from quite large to focused small groups.


## Acknowledgment

nanoHUB spans a 15-year effort and involves contributions from over 1,000 people. As such it is impossible to list all contributors. We recognize and acknowledge: Mark S. Lundstrom the creator of nanoHUB, Michael McLennan who created Rappture, Richard Kennell who created the middleware that currently powers all nanoHUB simulations, and finally Supriyo Datta who embraced nanoHUB for the development of all his fascinating quantum transport classes over the past 10 years. The visionary leadership of Lynn Preston and Mikhail Roco is also gratefully acknowledged.


## References


[1] Madhavan, Krishna PC, Beaudoin, Diane, Shivarajapura, Swaroop, Adams, GB, & Klimeck, Gerhard. (2010). *nanoHUB. org serving over 120,000 users worldwide: It's first cyber-environment assessment.* Paper presented at the Nanotechnology (IEEE-NANO), 2010 10th IEEE Conference on.

[2] Wilkins-Diehr, Nancy, Gannon, Dennis, Klimeck, Gerhard, Oster, Scott, & Pamidighantam, Sudhakar. (2008). TeraGrid science gateways and their impact on science. *Computer, 41*(11), 32-41.

[3] Klimeck, G., McLennan, M., Brophy, S. B., Adams III, G. B., & Lundstrom, M. S. nanoHUB.org: Advancing Education and Research in Nanotechnology. Computing in Science & Engineering 10, 17-23 (2008).

[4] Madhavan, K., Zentner, M., & Klimeck, G. (2013). Learning and research in the cloud. *Nature Nanotechnology*, *8*(11), 786.

[5] Levenshtein, Vladimir I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet physics doklady, 10*(8), 707-710.