HYBRID COMPARATIVE PREDICTIVE MODELING

by

Mohammad Asif Nawaz

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2017

Approved by:

____
Dr. Mirsad Hadzikadic

____
Dr. Zbyszek Ras

____
Dr. WeichaoWang

____
Dr. M. Yasin Akhtar Raja

____
Dr. Joseph M. Whitmeyer

ABSTRACT

MOHAMMAD ASIF NAWAZ. Hybrid Comparative Predictive Modeling. (Under the direction of Dr. MIRSAD HADZIKADIC)

In this research, a hybrid predictive model was proposed for the decision-making process. Predictive models can be built through the use of Machine Learning using different classifiers/algorithms to predict results as well as provide recommendations to the management for the student placement in appropriate programs of study and to the students for the adoption of appropriate study strategies and habits. A predictive model through Machine Learning was used in conjunction with probabilistic classification and clustering of specific segments within the data to increase the rate of success for an improved decision-making process. Variance in the actual and predicted results with respect to the difference in success rates can assist the decision makers in student placement. An aggregate of all the processes with the help of Cobb-Douglas utility function leads to a Hybrid Predictive Model, which combined two different phases for better placement, an increased rate of success, and an overall improved decision-making process. The introduction of Cobb-Douglas utility function can further streamline the process to check any external factors that may have influenced the predicted results. This model can also be applied in the corporate sector to maximize any program or individual efficiency by placing and training individuals with respect to individuals/employees aptitude, background, and personality type and learning styles. This type of model can also be applied in the same capacity in different organizations to maximize the program efficiency, placement, and employees' capabilities.

ACKNOWLEDGEMENTS

Special thanks to my excellent mentor, adviser, and chair Dr. Mirsad Hadzikadic for his patience, continuous guidance, and support throughout my Ph.D. studies. His encouragement and direction made me think out of the box. Special thanks to all my committee members, family members, relatives, dear friends, professors, and staff at UNC Charlotte who supported and encouraged me throughout my Ph.D. studies. I am also grateful to the following mentors and academics for their time and guidance in reviewing, correcting, and acknowledging my research work: Dr. Joseph Whitmeyer; Mr. Curtis Price; Lieutenant Colonel Phillip W. Thomas; Dr. J.T. Folsom-Kovarik; Dr. Rick Dietrich; and Dr. Umar Farooq. Special thanks to Inna Ruble, Major Timothy J. Ghormley, Mr. Michael P. Judge, Dr. David B. Carmichael, and Dr. Amir Nabipour for encouragement, guidance and moral support. I cannot appreciate enough, my parents Rab Nawaz Choudhry and Bazgha Birjees Nawaz for their prayers and pushing me to start and finally complete this study. I am also truly blessed with sincere friends and their spouses who not only morally supported me throughout my Ph.D. but also tolerated my unannounced visits, provided delicious food, and quality time to uplift my spirits.

<u>It takes a village to raise a child, and it takes the full support, and wisdom of all the professors and staff of the University, one's family members, friends, and coworkers to complete a Ph.D.; not to forget the commitment and hard-work from the person doing the Ph.D. is a must too.</u>

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	ix
LIST OF ABBREVIATIONS	x
CHAPTER 1: INTRODUCTION	1
Overview	1
Background	2
Motivation and rationale for this research	5
Purpose of the study	8
Statement of the problem	9
Research objective	10
Evaluation goal	11
My contribution	13
CHAPTER 2: LITERATURE REVIEW	15
CHAPTER 3: METHODOLOGY	31
Overview	31
Detailed overview	34
Algorithm steps	41
PTLS BRKEVEN Snapshot program details	51
Drag or boost effect	58
Teacher effect	64
Project illustration	75
CHAPTER 4: CALCULATIONS & RESULTS	78

Project background and assumptions	78
Tools and limitations	81
Hybrid Model	106
Corrective action	108
CHAPTER 5: CONCLUSIONS & RECOMMENDATIONS	112
Summary of results	113
My contribution	118
Significant findings and future applications	120
Limitations	124
Future recommendations	124
REFERENCES	128
APPENDIX A: PRE-HYBRID MODEL TRIALS & ERRORS	130
APPENDIX B: DATA NOTES	132
APPENDIX C: PT SNAPSHOT PROGRAM NOTES	133
APPENDIX D: DETAILED SVM & DT RESULTS	134

LIST OF TABLES

TABLE 1: Personality Type & Learning Preference	16
TABLE 2: Demo – Sample Data Format 1	24
TABLE 3: TE Experience Chart	70, 109, 126
TABLE 4: TE Qualifications Chart	70, 109, 126
TABLE 5: Personality Type Acronyms	79
TABLE 6: Iterations Abbreviations	83
TABLE 7: Arabic Scenarios	86
TABLE 8: Russian Scenarios	90
TABLE 9: French Scenarios	92
TABLE 10: Combined Language Scenarios	94
TABLE 11: Arabic PTLS Probability Distribution Chart (LN1-Arabic)	98
TABLE 12: French PTLS Probability Distribution Chart (LN3-French)	99
TABLE 13: Russian PTLS Probability Distribution Chart (LN5-Russian)	100
TABLE 14: Personality Type Preference Chart	101, 116
TABLE 15: Combined Language PTLS Probability Distribution Chart	102
TABLE 16: Personality Type Preference Chart 2	103, 117
TABLE 17: TE Effort Chart	110
TABLE 18: LN-PT Rate of Success Chart	115
TABLE 19a: Hybrid Model Key Results	119
TABLE 19b: Hybrid Model Results Overview	119
TABLE 20: Teacher Effort Overview	120

TABLE 21: Drag and Boost Findings 124

LIST OF FIGURES

FIGURE 1: Phase 1	32, 75
FIGURE 2: Phase 2	33, 76
FIGURE 3: Phase3	33, 77
FIGURE 4: External Factors – Extended Research	34, 77
FIGURE 5: PTLS BRKEVEN Snapshot Program Screen 1	52
FIGURE 6: PTLS BRKEVEN Snapshot Program Screen 2	53, 104
FIGURE 7: PTLS BRKEVEN Snapshot Program Screen 3	53, 104
FIGURE 8a: PTLS BRKEVEN Snapshot Program Screen 4	54, 105
FIGURE 8b: PTLS BRKEVEN Snapshot Program Screen 4	54, 105
FIGURE 9: PTLS BRKEVEN Snapshot Program Screen 6	56
FIGURE 10a: PTLS BRKEVEN Snapshot Program Screen 5	60, 108
FIGURE 10b: PTLS BRKEVEN Snapshot Program Screen 5	60
FIGURE 10c: PTLS BRKEVEN Snapshot Program Screen 5	61
FIGURE 11: PTLS BRKEVEN Snapshot Program Screen 7	74, 110

LIST OF ABBREVIATIONS

AD	Arabic Language
CLN	Combined (Arabic + French + Russian)
DA	Diagnostic Assessment
DLAB Test/Score	Defense Language Aptitude Battery Test/Score
DLI	Defense Language Institute
DLPT	Defense Language Proficiency Test
DT	Decision Tree
E&L (Learning Style Questionnaire)	Ehrman-Leaver
EBM	Equation-Based Modeling
ENFJ	Extravert, Intuitor, Feeler, Judger
ENFP	Extravert, Intuitor, Feeler, Perceiver
ENTJ	Extravert, Intuitor, Thinker, Judger
ENTP	Extravert, Intuitor, Thinker, Perceiver
EP	Expected/predicted to succeed or pass
ESFJ	Extrovert, Sensor, Feeler, Judger
ESFP	Extravert, Sensor, Feeler, Perceiver
ESTJ	Extrovert, Sensor, Thinker, Judger
ESTP	Extrovert, Sensor, Thinker, Perceiver
Exp	Experience
FaAD	Did not succeed in Arabic in getting ILR level 2 or above.
FaCLN	Did not succeed in CLN in getting ILR level 2 or above
FaFR	Did not succeed in French in getting ILR level 2 or above.
FaRU	Did not succeed in Russian in getting ILR level 2 or above.
FR	French Language
G-D or GD	Global Deductive
G-I or GI	Global Inductive
ILR	Interagency Language RoundTable
INFJ	Introvert, Intuitor, Feeler, Judger
INFP	Introvert, Intuitor, Feeler, Perceiver
INFP	Introvert, Intuitor, Feeler, Perceiver
INTJ	Introvert, Intuitor, Thinker, Judger
INTP	Introvert, Intuitor, Thinker, Perceiver
ISFJ	Introvert, Sensor, Feeler, Judger
ISFP	Introvert, Sensor, Feeler, Perceiver
ISTJ	Introvert, Sensor, Thinker, Judger
KSA	Knowledge – Skills – Abilities
LANG	Language
LASC	Language School

LN	Language
LN1	Arabic Language
LN3	French Language
LN5	Russian Language
LS	Learning Style
MBTI (Personality Type Questionnaire)	Myers Bridge Type Indicator
ML	Machine Learning
ML-OPI	OPI value derived for Machine Learning Calculations
N-Age	Normalized Age scores
N-DLAB	Normalized DLAB score
OP-FV	OPI-Face Value
OPI	Oral Proficiency Interview
PaAD	Would succeed in Arabic in getting ILR level 2 or above.
PaCLN	Would succeed in CLN in getting ILR level 2 or above
PaFR	Would succeed in French in getting ILR level 2 or above.
PaRU	Would succeed in Russian in getting ILR level 2 or above.
P-D or PD	Particular Deductive
P-I or PI	Particular Inductive
P-M-I	Passion-Motivation-Innovation
PT	Personality Type
PT Questionnaire	Personality Type Questionnaire
PTLS	Personality Type – Learning Style
Q	Qualifications
RU	Russian Language
SR%	Success Rate %
ST	Student/s
SVM	Support Vector Machine
TE	Teacher

CHAPTER 1: INTRODUCTION

Overview

The selection criterion for student placement in specific programs in many institutes is mostly based on the student's past performance and some kind of entrance exam. Most of these entrance exams can be mastered with practice over time. There are various entrance exams and aptitude assessments such as SAT, DLAB, GRE, GMAT, and GT to arrange students for admissions in different programs and fields of study. Many teachers acknowledge the importance of learning styles (LS) and personality types (PT) in learning. However, it can be one step further, to devise a model that utilizes students' personality type and learning style preference (PTLS), which can both assist in predicting student success and aide decision makers in placing a student in a program of study where they are more likely to succeed. Further, by analyzing this success rate, trained teachers can assist the students in adapting the attributes of the most successful PTLS with the highest success rate within a specific program. This predictive analysis can be done by making specific predictive models for classification, clustering, and association by Machine Learning techniques through Support Vector Machine (SVM) and Decision Tree (DT) algorithms and classifiers. Predictions through SVM and DT can be verified by cross-checking the final/actual results.

By conducting a comparative analysis of various predictions, one may even get better predictions. However, in EBM, predictions are not 100% accurate. Is there a way to improve the prediction accuracy? Sometimes there is a big difference between the actual and predicted results. Machine Learning results can further be combined with probabilistic classification to get the success rates within each personality type segment.

In this research, the focus is only on the speaking skills because Oral Proficiency Interview (OPI) is the predictor, dependent variable, and the benchmark to measure the success outcome. In this study, the success outcome is considered ILR level 2 or above.

Sometimes, it is also assumed that some related external factor might have influenced the results. Figuring out all the external factors is tough. Comparative analysis of the predicted and actual result may be computed to an extent to look into the PTLS factors along with the known external factors that made that student successful in the subject contrary to the predictions and vice-versa. Different methods and models will be looked at to check the accuracy of the predictive results, and a better method and model is proposed for predictions and placement to improve the existing decision-making process. Comparative analysis will also look into students with specific PTLS to be more successful in the area of study different than their current or initial choice of study. In this research, the relationship of teacher's influence as an external factor will also be looked at through a utility function.

Background

In one of the Army Departments, students are placed in different language programs based on the needs of their chain of command and students' background, which sometimes includes students' Defense Language Aptitude Battery Test (DLAB) scores (Petersen & Al-Haik, 1976). However, if students are scientifically placed in specific programs by considering their personality types and learning styles preferences, in conjunction with their background, then it may lead to even better results. This kind of placement can reduce the failure rate, improve results, and may lead to enhanced retention and life-long/continuous learning, which can transform students into proactive learners. Here the specific Army department will be referred to as the language school (LASC). To improve the results of the language program, a research project for predictive modeling was initiated. The goal of this project was to explore any scientific ways to predict student placement that is more compatible with students personality types and learning styles and background.

Personality is a complex domain and most of the time, psychologists, and teachers/trainers are trying to figure out the magic formula where teachers can tailor their instruction/teaching according to students' specific personality traits and learning styles, so students can get the most out of their learning. In other words, it is an effort to make learning more compatible to students' capabilities with respect to students' personality types and learning styles preferences. In this way, students may be able to learn more efficiently.

Much research is done on personality types and learning styles (some of this research will be discussed in the literature review section of this proposal); however, it would be one step further to make this research more beneficial in developing better methodology and models with respect to PTLS preferences that can look into students' success rate in particular programs which can also assist the decision makers in recommending the students in different programs.   Moreover, with this kind of a model, one can train the students to adapt the PTLS preferences of the students with the highest success rate in a specific program.

According to the Myers-Brigg type indicator (Boeree, 1997) (Clawson, 1997), there are 16 different personality types. Furthermore, according to the Ehrman-Leaver

(E&L) cognitive styles construct (Leaver, 1997) (ME Ehrman, Leaver, & Skekhtman, 2002) there are ten different combinations of Synoptic vs. Ectenic combinations of learning styles. According to the E&L Questionnaire, "Synoptic" side has Learner Style attributes such as Global and Inductive whereas "Ectenic" side has Learner Style attributes such as Particular and Deductive. One can infer from the E&L Questionnaire that Synoptic and Ectenic attributes are the opposite ends of the spectrum.

How students prefer to learn can be linked to their personality preferences, and this can also be related to what motivates these students; according to Leaver, B.L., M. Ehrman, and B. Shekhtman in their book "Achieving success in second language acquisition", on page 113 in the personality section, "What we pay attention to and remember, how we prefer to learn, what makes us anxious, what motivates us, and what we are confident at, all are linked to our personality preferences." In this research unidentifiable data was given for modeling. The transmuted data was provided from approved Personality Type and Learning Style questionnaires to get the personality types and learning styles preferences of students (these forms are industry standards in the Defense Language Institute and also widely used in the Defense Language Institute and various Language Training Detachments in the Department of Defense (DoD) facilities). In this research, results are measured by Oral Proficiency Interview (OPI) results (Liskin-Gasparro, 2003), which is the benchmark and dependent variable in this predictive modeling. Interagency Language Roundtable (ILR) (Herzog, 2007) level is a language proficiency scale, a particular kind of a grading system that provides the description of various proficiency levels in specific languages. ILR level is measured by conducting an assessment of any specific language proficiency by an OPI or by Defense Language

Proficiency Test (DLPT). ILR levels range from 0, 0+, 1, 1+, 2, and 2+ to ILR level 5. ILR levels can be in any of the four skills i.e., reading, writing, speaking, and listening. In this research, we are only focused on the speaking skills because OPI is mainly Oral Proficiency Interview.

This is a classification problem. Given data is normalized for Machine Learning (Salzberg, 1994), and then by using Support Vector Machine (S. B. Kotsiantis, 2007) and Decision Tree Algorithm Classifiers (Neville, 1999) (Safavian & Landgrebe, 1990), a model is devised to test the data for predictive results. Actual results are compared with the predicted results, and the difference of results is analyzed for the recommendations in light of the PTLS. Different language results are also compared to see if the student with a specific PTLS would perform the same way in one language as compared to the other. To validate the model one has to compare the predicted results with actual results and further demonstrate in light of the data how predicted results changed with respect to student's PTLS. By combining the two methods through Cobb-Douglas utility function in a program like C-sharp leads to a hybrid comparative predictive model for better predictions and to an improved decision-making process. In this research, other than PTLS, one also looks further into factors such as teacher impact that may have influenced the results.

Motivation and rationale for this research

The motivation for this study was to explore the results with reference to PTLS preferences and find any new or non-traditional ways to achieve a better success rate in particular subjects with respect to those preferences. The motivation was also to increase the success rate of the students in their field of study. Here the success is determined by the OPI score. In one of the LASC briefs, the Director of the LASC floated the idea to find new methods and techniques based on students KSA (Knowledge–Skills–Abilities) to enhance student performance to become highly successful in their language studies which may also lead to a life-long learning (when it comes to the retention and enhancement of the acquired language skills). In LASC, the specific language experts diagnose any learning issues of the students and then recommend these students specific learning strategies to overcome their learning weaknesses in light of their PTLS preferences. In this specific scenario, the language experts are trained by their organization and these language experts are considered the subject matter experts in their field of teaching. Any of these language experts has to get a specific certification before the language expert can diagnose and provide recommendations for any learning issues of the students. This specific certification training provides a two prong tool, based mostly on PTLS preferences that helps students enhance their learning strengths and also to overcome their learning issues/weaknesses with specific recommendations given by a certified language expert within the organization. PTLS is used to diagnose the learning issues or to enhance the learning strengths of the students with specific strategies during the course of study. However, PTLS is not primarily used to predict the results or to place students in different programs for predictive modeling. Motivation to do this study is to devise a new methodology for predictive modeling through a comparative Hybrid Model in light of the literature review for better results. This research will help students and teachers understand their learning process with respect to their PTLS preferences by devising a model that increases the rate of success and makes the learning process easier and proactive. This type of model will also help the decision makers to see which PTLS preference has the higher rate of success when it comes to student placement. Learning these specific subjects/languages is part of the training for these students. These courses and training sessions are sometimes critical to these students' success in their career advancement and job requirements. If a better method and system can be devised which is more effective than the existing method of predicting student successes and better placement than it would improve the overall results of the LASC, which can be verified by comparing the results before and after the implementation of such method and system. Currently students are sorted out for different programs of study (in this case different languages) by their chain of command in view of their DLAB scores and their command's preference.

According to the ("DLAB Test Score Range," 2014) language categories and their DLAB scores are given below:

"Because not all Languages are created equal, the US military has devised a means in which to grade language levels based on their difficulty to learn. Their breakdown is as follows:

95 for a ***Category I language*** *(Dutch,* ***French****, Italian, Portuguese, and Spanish)*

100 for a Category II language (German)

105 for a ***Category III language*** *(Belorussian, Czech, Greek, Hebrew, Persian, Polish,* ***Russian****, Serbian/Croatian, Slovak, Tagalog [Filipino], Thai, Turkish, Ukrainian, and Vietnamese)*

110 for a ***Category IV language*** *(****Arabic****, Chinese, Japanese, and Korean)*

According to the military, languages like Chinese, Korean, and Arabic are much harder than French, Italian, and Portuguese. This author would not disagree with *them. Languages like Chinese, Korean, Japanese, and Arabic do not use the Greek alphabet. Therefore language learners of these types have to learn a whole new alphabet and in most cases have to learn new sounds."*

Students' personality type and learning styles were not taken into account in placing these students in specific programs.

Purpose of the study

The purpose of the study was to find an improved method of predicting results, which gives better results as compared to previous methods and also to assist the decision makers in recommending students for placement in different programs to achieve the best results. Another objective was to share the findings and the recommendations with the trainers in order to help the students improve their learning by comparing and adapting their PTLS preferences to the PTLS that performed the best in a specific program. Students can adopt behaviors similar to those exhibited by students who have been successful. Additionally, a student who knows the strengths and weaknesses of their particular learning preferences can also modify their behavior in order to learn more efficiently.

Moreover, in this study, the difference in actual and predicted results is also looked at, with the intention to calculate the external factor that contributed to that difference in results. Every student has their own personality and learning style so it would be beneficial to look at those personality types and learning styles traits with reference to different programs for predictive modeling. Various permutations and factors in the model are based on specific personality type and learning style combinations. Literature references for PTLS, Machine Learning, and various tools for

Predictive Modeling will be discussed in detail in the Literature Review Section. The literature review will provide the basis for the methodology and model in devising this system.

Statement of the problem

How to maximize the learning potential of the individuals (students) by placement based on their personality types and learning styles preferences through predictive modeling?

The research conducted in this project can provide an improved insight in predicting which students may have more potential to pass the course with a better rate of success with respect to their PTLS preferences in specific programs. LASC can also help their students by focusing on the students that the predictive model indicated as not successful but the rate of success of these students can increase by intervention through academic strategies with respect to their PTLS preferences. Here assumptions are based on the calculations from the unidentifiable data[1] used in this project which is transmuted from specific Personality Type (PT Questionnaires) and Learning Style (E&L forms/questionnaires) for PTLS preferences, and a predetermined course timeline. Limitations of this research are based on teacher intervention and students interacting with each other and the teacher, the impact of that interaction may affect in changing the PTLS preferences of the students which may influence the predicted results.

This research can also lead to any future research to devise models tailored to specific courses that can be utilized with particular PTLS permutations as per student

---

[1] In this study, unidentifiable student data was given for the modeling purposes in the form of variables and numbers only, such as X1, X2, X3, which does not reveal student identity due to confidentiality and organizational protocol. This unidentifiable data (student names as X1, X2, X3…) is only used for the modeling purposes. Permission letter (Reference #: AOJK-EDG-LA) to use the unidentifiable data was given on 4th November 2015 is on file, which does not require an IRB approval for this specific modeling.

needs. In such cases, a tailored model may suggest learning strategies with respect to student's particular PTLS preferences to achieve better results; especially, if student's PTLS preference differs with the PTLS preferences of the students with the highest success rate i.e., student's particular PTLS preference is not compatible with the specific program so student is taught to adapt the PTLS preference of those students whose PTLS preference has the highest success rate. This type of corrective action may decrease the student failure rate, enhance learning retention, and lead to life-long/continuous learning. This is due to the fact that various personality types and learning styles have different learning strategies. By applying the strategies from the reviewed articles in this proposal's literature review, one can lead these students (whose PTLS permutation is not the best option for a specific program) to a better learning environment that can save time and cost by producing highly productive learners. One may also calculate different scenarios where the cost savings, timelines of courses, and learning efficiency of students can be connected. Learning efficiently by integrating PTLS model can have different connotations such as by incorporating the strengths of students' personality types and learning styles, one may look into, how to calculate the optimum level of learning with respect to materials learned and timelines i.e., if students can learn more materials in the same time period or if the students can learn the same materials in reduced time period.

Research objective

Under the guidance of Dr. Mirsad Hadzikadic – Executive Director, Data Science Initiative-Director, Complex Systems Institute, a hybrid comparative predictive student placement model was proposed to increase language program efficiency by developing a student placement model that would help predict language results with respect to the ILR levels and further assist decision makers in language placement recommendations as well as help students selected languages adopt strategies and techniques based on their PTLS preferences. Moreover, the external factors such as the effect of teacher influence on the results can also be calculated.

Evaluation goal

The goal of this research is to achieve 70% success rate. If students are placed in specific language programs based on results and recommendations from the hybrid predictive model, then the probability of students to be successful can be 70% i.e., seven out of ten students will be successful (where the success factor is ILR level 2 or above on the OPI)[2].

Student success rate in learning the specific language can be increased by assisting and placing the students in specific language programs that show higher success rate with respect to particular learning style and personality type preferences as determined by the data analysis. Specific PTLS preference attributes may be compatible with students' aptitude towards a specific language. Further research can be conducted beyond this research project where this type of predictive placement can be beneficial to the organizations for training the students or employees in a proactive way where failure or turnover rate is very high. This kind of predictive modeling can lead the research further to optimize training timelines, which may also lead to the cost savings for the organization. Results from this research can be further studied and validated by experiments in making learning or training more systematic in such a way that

[2] 70% success rate is proposed as a measure of success in this case. Current results show that around 45% of the students secure ILR level 2 in the OPI; by following the recommendations of the research and by placing students through this Hybrid Model 70% of the students are expected to get ILR level 2 in their OPI.

students/employees with various PTLS in a specific program can be taught about their PTLS strengths and weaknesses along with the strategies in light of any program objectives i.e., in this case, sometimes the focus is on speaking skills and sometimes the focus may be on reading or listening skills. In this particular research, the program focus is on speaking. PTLS preference compatibility with the course of study may help in retention, which may lead to life-long learning. Overall, this can make the learning process more productive. Findings of this research can be applied to programs and academic areas other than the language programs based on the results validated by the predictive models that can be developed and tailored in light of the specific needs of an organization and course/training objectives.

When it comes to better student results in LASC, it is a team effort, which includes many factors that contribute to the success of an engaging and intense language program; these factors range from student motivation, students' efforts and background; experienced and qualified teachers; proactive management; dynamic leadership; recommendations from the language experts, guidance from cognitive enhancement and performance coaches; and a practical & flexible curriculum. With all the factors mentioned above, if students are also placed as per their PTLS preferences and background (here background can be DLAB scores, knowledge of other languages, age, etc.) then this persistent success rate of getting higher results can be maintained and even enhanced. Moreover, to maintain and even improve the high success rates of an intense language program, factors like flexible syllabus (flexibility); teachers that adapt to the flexible syllabus (adaptability), organizational objectives, and student needs; sharing/collaborative system where colleagues share their successful techniques with their peers; students that share their ineffective and effective learning techniques, can maintain or enhance those consistent success standards. But scientifically placing the students as per PTLS model can make the higher success factor for the organization even more certain. However, this research focus will be limited to a hybrid comparative predictive modeling. How PTLS model can be devised for student placement and how to smoothly integrate this type of design in the already existing and established system for better predictions and decision making? Hidden facts can be discovered from educational data through data mining techniques (Okeke Ogochukwu & Ezenwegbu Nnamdi). With Machine Learning, one can explore any significance in the data to develop predictive student models in projecting students' performance in LASC. The focus of this project is to develop a viable classification system by comparative predictive modeling to identify students that are more compatible to produce better results in specific languages in various language programs at LASC.

My contribution

An aggregate of all the processes lead to the formation of a Hybrid Predictive Model that contributes to a better student placement and training, an increased success rate, and an overall improved decision-making process for the management and the trainers. In this novel contribution, a Hybrid Model was created by combining different methods for improved predictions for better decision making. These methods (Phase 1 & Phase 2) were combined through Cobb-Douglas utility function; the aggregate result gave an outcome which was better than before, this optimal outcome was not possible without combining these methods. This method was automated through C-sharp program and could be applied to similar scenarios for improved results. Moreover, through comparative analysis drag and boost effect was introduced which could further be computed through Cobb-Douglas utility function to look into the external factors that influenced the predictive outcome. One could also take measures to adjust those external factors for better results.

CHAPTER 2: LITERATURE REVIEW

Literature review in this section provides a base for further steps in this research process. This literature review has two parts. The first part focuses on the research already done on personality types, learning styles, and their importance in student learning. This section will provide the understanding of Personality Types and Learning Styles. The second part focuses more on tools needed to utilize data with Machine Learning, and algorithms that would help in formulating a methodology to devise a predictive model to check results, which can assist in the decision making process of student placement.

According to Carl Jung (Boeree, 1997; Clawson, 1997), "there are four functions in personality theories, first is sensing, second is thinking, third is intuiting, and fourth is feeling. All of us have these four functions with different proportions. Most of us develop only one or two of the functions, but our goal should be to develop all four."

Carl Jung talks about the personality types in detail which would help in this research to sort out specific PTLS combinations in predictive modeling and student placement in particular programs.

(Boeree, 1997) Personality types attribute, and descriptions are given in Carl Jung personality theories section. (Leaver, 1997; Leaver, Ehrman, & Shekhtman, 2005) mentions, "MBTI combines four domains of personality attributes into sixteen different personality types as given below. The significance of these types for the classroom teaching is summarized very briefly in the chart below and in the explanation that follows (Keirsey & Bates, 1988)."

Table 1: Personality Type & Learning Preference

ESFJ	cooperative groups
ESTJ	organization, clear instructions, deadlines
ENFJ	one-on-one or with peer groups
ENTJ	leading a group of peers in a project
ESFP	actively with a group and with choice
ESTP	games, negotiations, simulations
ENFP	real-life applications, projects
ENTP	analysis, invention, develop new procedure
ISFJ	manuals, assisting others
ISTJ	details, calculations
INFJ	plays, poetry, visual images, archetypes
INTJ	manipulation of theory, logical problems
INFP	creative writing, metaphor, impressionism
ISFP	practice, play, action, concretization
INFP	relating, writing, acting
INTP	research, systematize, theorize

According to Betty Leaver (ME Ehrman et al., 2002; Madeline Ehrman & Leaver, 2003; Leaver, 1997) "Cognitive styles in the service of language learning," E&L questionnaire can be used as an instrument to analyze student's learning styles within Synoptic and Ectenic boundaries. For example, Global and Inductive are on the Synoptic side, whereas particular and deductive are on the Ectenic side. According to the article (Madeline Ehrman & Leaver, 2003), Synoptic apparently trusts their gut, but Ectenic tends not to trust their guts as much as compared to the Synoptic learners. In this article, a case study of two language students is discussed. E&L questionnaire findings are discussed, and then some strategies are given to enhance student learning. Here to be noted that in this ongoing research, initially, two sets of the learning style attributes are considered for the student placement and predictive modeling, which are "Global vs. Particular" and "Inductive vs. Deductive." In this article, the emphasis is on student learning style not on student placement as per their strengths and hidden potential. E&L questionnaire is one of the tools that show some of the learning style attributes.

Rebecca L. Oxford (Oxford, 1990) has given a list of language activities to enhance the language skills as per learning styles and further provided language tasks or situations for strategy search game too. Language activities such as Dating Game, Guess What, Short Haired/Long Haired Dictionary, Espionage, and Toothache, etc. These activities and strategies are helpful and can be used in line with the personality type. However, these strategies and activities are not used in developing a placement model but utilized to enhance student learning after students are placed in specific programs as per their PTLS.

Here to be noted that the past research has been conducted where students personality types were looked at to check any relationship between personality type and performance in specific programs such as undergraduate pilot training (Davis, 1989), "however the research has failed to establish a consistent link between personality and pilot performance" page 11 (Davis, 1989); the correlation of success (performance) with respect to PT was extremely small. “Also, studies conducted by Robert L. Helmreich, Department of Psychology, the University of Texas at Austin, agrees that research done in the recent past has failed to establish a consistent link between personality and pilot performance.(Davis, 1989; Helmreich & Wilhelm, 1987).” There may be other reasons that did not show any significant relationship between PT and performance such as lack of proper methodology (Davis, 1989; DOLGIN & GIBB, 1988).

When any research is conducted to understand the dynamics of personality types and learning styles and different attributes within the PTLS, it may be beneficial to also look into various learning strategies too, learning strategies that are compatible with specific personality type and learning style preferences may show some connection of PTLS attributes with performance when it is linked with specific strategies that are in line with the PTLS preferences. In Rebecca L. Oxford’s book “Teaching and Researching Language Learning Strategies” (Oxford, 2013); S2R model (strategic self-regulation model) is discussed; concept 1.10 specifically goes over the Extroverted and Introverted style. Here to be noted that some of the language instructors limit their perception of extroverts and introverts as extroverts are more outspoken, and introverts are the quiet ones. However, in this section, Dr. Rebecca L. Oxford provided the definition of extrovert be the people who prefer to gain energy from (other) people and activities; these learning attribute students like to work with others. She further goes in the tactic related to the style aspect with examples. Further in strategy that corresponds to the tactic is “Interacting to learn and communicate.” However, in introverted style domain, learners get energy from inner thoughts and feelings and like to work alone. A strategy that corresponds to the tactic would be the focus on reasoning. In this research of developing student placement model, learner’s personality type has “Extrovert vs. Introvert” attributes. Concept 1.10 from “Teaching and Researching Language Learning Strategies” goes one step further in understanding extrovert and introvert learners, which will be very useful in understanding the personality type combination with specific learning styles. This leads to another aspect of this research, which is; the person conducting this kind of a research should be knowledgeable about PTLS theories and trained in interpreting the PTLS preferences in the right context; also keep in mind the ethical predicaments of labeling others with a specific PTLS because these PTLS preferences may vary depending on the environment and person’s adaptability to the circumstances with specific training sessions (Wurster, 1993).

Patsy M. Lightbown and Nina Spada (Lightbown & Spada, 2006) comments that some learners learn languages more quickly than others. Even in the first language acquisition, the rate of development varies widely. In Table 3.1 on page 55, several characteristics of a language learner are given, such as learner enjoys grammar exercises, analyzes his or her own speech and the speech of others, is willing to make mistakes, is a willing and accurate guesser, and has a good self-image and lots of confidence. These learning characteristics can be tied to certain personality types. Moreover, these characteristics can also be used by letting the learners become aware of their specific personality types and how these learners can use their strengths and weaknesses in their favor to learn at a faster rate efficiently. The Appendix of Strengths and Weaknesses of Each Type summarized in the research article in “A Comparison of Personality Type and Learning Style of Elementary Education Majors, Math Majors, and Math Professors: Cultures in Conflict is an excellent overview of the strengths and weaknesses of different personality type preferences and the first recommendation in the conclusion states that "Faculty and students, in general, could benefit from a better understanding of individual personality types and how these influence their approach to learning, to teaching, and to interpersonal relationships" (Martin, 1992), which reaffirms the importance of understanding the weaknesses and strengths in different personality types in learning and teaching.

Christine Campbell and Deanna Tovar discusses Learner-Centered Curriculum and Instruction in their "Communicative Language Teaching" article (Tovar, 2007) "Perhaps Nunan's (2006) most succinct definition of learner-centered instructions follows: [It has] two dimensions: (1) Learner involvement in making choices about what to learn, how to learn, and how to be assessed. (2) Learners are actively involved in learning through doing." Here to be noted that in the first dimension, the segment "how to learn" is in line with determining learners' particular learning style and personality type i.e., when learner knows his/her specific personality type, and learning style than the learner would know how to learn effectively by using specific techniques that are in-line with that specific learning style and personality type. Moreover, if learner lacks specific learning attributes or if learner's learning style and personality type is in contrast with the specific subject or teacher's teaching style; then the learner can adapt specific strategies to overcome his/her weakness to become better in that specific field and learn more proactively in a specific class where teacher's teaching style is very different from student's learning style and personality type. For example, if a specific learner is a visual learner and teacher's preferred instruction is highly aural/auditory i.e., teacher is only giving lectures, and then the learner will have difficulties to progress in that class.

However, some strategies can enhance learner's aural skills by various techniques. Similarly, if teacher is teaching with inductive and global preferences and the student has more of the deductive and particular learning attributes, it may lead to a frustrated learner but if student/learner is made aware of the fact, how to extract information and learn from an inductive and global teacher, and how to transform the deductive and particular skills in-line with inductive and global domain then the learner can have a much better learning experience.

(Leaver, 1997) All students can learn or acquire knowledge in one way or another, what the learners may not know is how to be able to learn in the way prescribed by a specific program or teacher. According to Betty Leaver, "The teacher's role is to orchestrate the miracle by focusing on the student who is not learning and rearranging the environment, task, or subject matter so that the student can learn. By accommodating learning style profiles and empowering students, teachers can 'deconflict' in teaching and learning styles that interferes with students' ability to learn. Teachers who have used a specific learning styles system may need to reorient themselves to a different concept of learning styles if they are to teach the individual students in their classes (Hatch, 1997). This concept looks at learner profiles as complex descriptions of how each student learns. The ways in which learners relate to other people and to the physical and intellectual world around them influence their learning. Students reveal their learning style preferences by everything they do and not do and by everything they say and not say."

From the literature review, one may infer that when we analyze learning styles and personality type of a learner; sometimes it reveals a lot about person's lifestyle, habits, their likes and dislikes, their comfort zone, factors that influence their motivation, and even the environment they were raised in. Personality types and learning styles are developed over time. One can devise strategies and tactics to enhance their learning by understanding how to acquire, process, and comprehend the knowledge by different ways. Some may process the information quickly by visual charts and graphs, and some may process the information faster by reading and discussion. Everyone has their own learning limitations. Once a learner is aware of his/her learning preferences, his/her personality style and attributes of that personality style then the learner can use those attributes in processing and acquire the information in a much efficient and faster way.

(Leaver, 1997) Betty Leaver also talks about Cognitive Styles in “Styles and Profiles” in her book “Teaching the Whole Class.” Cognitive styles refer to thinking processes, a complex set of actions that takes place in the mind. To think, intake of information must occur, followed by processing, storage, and reconstruction of that information, as well as the generation of unique thought. The ways in which people perceive and process information affect how they learn. Here the processing of information depends on the intake of information first, and intake of information is directly related to the learning style i.e., how one is acquiring the information. If the learner is used to acquire information by visual aids then even if learner spends hundreds of hours to acquire the information by aural or auditory aids; this would not be highly productive for the learner, it might even become frustrating and counterproductive to the learner, which in turn would lead to demotivation (because learner is forced to learn contrary to his/her learning style and personality type).

Letting the learner know their learning styles and personality types and further guiding the learner with appropriate strategies to use those learning styles and personality types in an efficient manner will make the learning process easier. Moreover, the learner will get more out of their learning by getting engaged in the learning process, and this may not let the learners become demotivated.

In this project, Machine Learning is used to streamline the given data for various calculations, analysis, and predictive modeling. According to Tom Michael Mitchell in his article, The Discipline of Machine Learning, (Mitchell, 2006)“Machine Learning is a natural outgrowth of the intersection of Computer Science and Statistics. “The question that largely defines Statistics is “What can be inferred from data plus a set of modeling assumptions, with what reliability?” “Machine Learning incorporates additional questions about what computational architectures and algorithms can be used to most effectively capture, store, index, retrieve and merge these data, how multiple learning sub-stacks can be orchestrated in a larger system, and questions of computational tractability. Machine Learning methods are already considered the best methods available for developing particular types of software, in applications where the application is too complex for people to design the algorithms manually”.

After establishing the specific terminologies and variables such as OPI (Oral Proficiency Interview) (Liskin-Gasparro, 2003), personality types and learning styles, unidentifiable student data is analyzed that reveals specific personality type and learning styles along with other variables in Table 2. It is a classification problem, and Machine Learning will be applied to develop a methodology for predictions and student placement, primarily based on the PTLS.

Table 2: Demo – Sample Data Format 1

LANG	Student	OPI ILR Level	Person. Type	Learn. Style	DLAB	Age
AD	AD-X1	1+	INTP	G-I	119	22
AD	AD-X3	2	ESTJ	G-D		27
AD	AD-X4	1+	ESTJ	P-D		32
AD	AD-X6	1+	ISTJ	G-D		28
AD	AD-11	1+	ESTP	G-I	93	24
AD	AD-19	1+	ISTJ	G-D		27
AD	AD-23	2	ISFJ	P-D	132	29
AD	AD-24	2	ISTJ	P-D		27
AD	AD-25	2	ESTJ	G-I		27
AD	AD-26	2	ESTP	G-D		29

In the article “The Discipline of Machine Learning,” Tom Mitchell mentions that many different learning algorithms have been proposed and evaluated experimentally in different application domains. One theme of research is to develop a theoretical understanding of the relationships among these algorithms, and of when it is appropriate to use each” (Mitchell, 2006). In the case of developing a placement model for students in light of PTLS and OPI scores, Machine Learning techniques will be employed extensively to predict the results with respect to PTLS.

Different algorithms can be used in Machine Learning to devise predictive models. In this specific case, Decision Trees (Neville, 1999) and Support Vector

Machines (SVMs) (Sotiris B Kotsiantis, Zaharakis, & Pintelas, 2007; Meyer & Wien, 2015) are used to achieve better accuracy for classification purposes. "SVM are supervised learning algorithm models within Machine Learning to analyze data used for classification and regression analysis especially when it comes to non-linear classification, regression and outlier detection with an intuitive model representation" (Meyer & Wien, 2015). "In today's Machine Learning applications, support vector machines (SVM) are considered a must try—it offers one of the most robust and accurate methods among all well-known algorithms. It has a sound theoretical foundation, requires only a dozen examples for training data, and is insensitive to the number of dimensions. In addition, efficient methods for training SVM are also being developed at a fast pace" (Meyer & Wien, 2015). According to the top 10 algorithms (Wu et al., 2008), "SVM is one of the top 10 algorithms in data mining for classification purposes." When it comes to the Decision Trees, in the article "A Survey of Decision Tree Classifier Methodology" by S. Rasoul Safavian & David Landgrebe, (Safavian & Landgrebe, 1990)"The main objectives of decision tree classifiers are: 1) to classify correctly as much of the training sample as possible; 2) generalize beyond the training sample so that unseen samples could be classified with as high of an accuracy as possible; 3) be easy to update as more training sample becomes available; 4) and have as simple a structure as possible".

In the article Comparative Analysis of Decision Tree Algorithms for predicting undergraduate students' performance in computer programming, "student data from various Math and Physics classes was used to check the accuracy of three decision tree algorithms to predict the performance of students in computer programming. Results from this research can be used by non-expert user of data mining such as a teacher that can directly use the results obtained by these algorithms to identify students deficiencies and then to assist these students accordingly to prevent any failures" (Hambali Moshood). When it comes to predicting performance in computer programming, past results from Math and Physics can provide an easy and good foresight to the teachers to help students come back on track; this is much simpler as compared to looking into PTLS of students and then assisting them in improving their chances of success in the language or any other field of study. In PTLS research, focus would be in improving the rate of success by looking into different classification techniques and to see which PTLS has a higher success rate in which specific language. In PTLS research project, teacher can look into a specific PTLS preference that has a higher rate of success and then teaches students strategies which are in line with that specific PTLS that has a higher rate of success with respect to the specific language. Further, in the PTLS project, the decision makers can also look into the difference in the predicted and actual results, and analyze that difference with the intention of checking any external factors and give error correction such as tailored teacher training to improve the rate of success.

The PTLS placement model research is based primarily on personality types and learning styles and the predictors such as DLAB scores can also be used in conjunction with PTLS to refine the model further in this classification process. However, instead of looking further to add more attributes for better predictions, such as including DLAB scores, one can look first into Probabilistic Classification and Clustering in Relational data for the PTLS scenarios.

“Supervised and unsupervised learning methods have traditionally focused on data consisting of independent instances of a single type. However, many real-world domains are best described by relational models in which instances of multiple types are related to each other in complex ways” (Taskar, Segal, & Koller, 2001). When it comes to different personality types and learning styles, different personality types and learning styles are instances of multiple types that are related to each other in complex ways. Various personality types in different languages can be treated as a probabilistic relational model over a set of instantiations within the specific language. According to the article Probabilistic Classification and Clustering in Relational Data, “many real-world domains have rich relational structure, and traditional Machine Learning algorithms, ignore this rich relational structure” (Taskar et al., 2001). However, by using Probabilistic Classification and Clustering techniques in conjunction with SVM and Decision Tree classifiers in a Hybrid Model may help understand PTLS predictive modeling in a more efficient way. Results from the probabilistic classification and from the SVM & DT classifiers can be combined through a utility function such as Cobb-Douglas utility function. “Cobb-Douglas is a concept from economics that shows the relationship between two or more inputs and the amount of the output that can be produced by those inputs such as explained in the article Human-Capital Investments and Productivity” (Black & Lynch, 1996) . In the book Managing Complexity: Practical Considerations in the Development and Application of ABMs to Contemporary Policy Challenges on page 51, “The Utility Function” is mentioned; “utility function can incorporate relevant theories and factors which show some kind of a relationship between variables, and it can instantiate different theories by adjusting parameters” (Hadzikadic, O'Brien, & Khouja, 2013). "ACSES model has been devised by using Cobb-Douglas utility function. The Cobb-Douglas production function can be used to show the relationship between two or more inputs. In the case of developing an Agent-Based Model (ABM) ACSES (Actionable Capability for Social and Economic Systems) model, Cobb-Douglas function model was used because it is easily expandable to include additional preferences or values or motivations if they are important for a theory" (page 52)(Hadzikadic et al., 2013). Specific version of the Cobb-Douglas utility function is given in the form of equation (2) on page 55 in the Managing Complexity: Practical Considerations in the Development and Applications of ABMs to Contemporary Policy Challenges, "$U = (1-L)^{WL} (1-C)^{WC}(1-I)^{WI}(1-E)^{WE}(1-V)^{WV}(1-F)^{WF}(1-R)^{WR}$; where L is loyalty to leader, C is coercion, I is ideology, E is economic welfare, and R is the repression and social influence for defying repression and the weights W_x for motivation x, give the relative importance of the different motivations to the agent and the relative effect they have on U" (Hadzikadic et al., 2013). On the same pattern, with the help of Cobb-Douglas utility function, input from the SVM and DT classifiers and the input from the probabilistic classification that shows the rate of success can be combined to calculate an improved rate of success and a better insight to predict student success in light of students' PTLS preferences. This also gives a better picture to the management when one has to sort out different students to place these students in various programs based on their PTLS preferences and the success rate of these PTLS preferences in a specific program.

There may still be differences in actual and predictive results. It can be due to the nature of data and how the classifiers and algorithms work, or it can be caused by external factors. When it comes to students' success one of the factors other than students' personality, aptitude, and hard work can be teacher experience, qualifications, and how the teacher taught the concepts to the students. External factors such as teacher's influence are hard to calculate precisely in students' success; however, it exists. It is usually assumed that other than students' own efforts, teacher's role also plays a factor in students' success. If a teacher is experienced, qualified, and engaged with students with innovative teaching strategies, passion, and motivation then the students are more motivated to learn and excel in that subject. In this research, the focus is on comparative predictive modeling through a Hybrid Model; however, the difference in actual and predicted results is also looked at as a consequence of a related external factor such as the teacher effect. This teacher effect can also be calculated by introducing a utility function such as a Cobb-Douglas function on the same pattern as in the article "Some Characterizations of the Cobb-Douglas and CES Production Functions in Microeconomics" (Wang & Fu, 2013). On the same pattern, with the help of Cobb-Douglas utility function, teacher effect can also be integrated in the predictive modeling where teacher effect can be calculated by considering teacher experience, qualifications, and passion/motivation/innovation; where teacher passion, motivation, and innovation & creativity to teach can be referred to as the teacher effort. Here teacher experience, qualifications, and effort can be factored in the Cobb-Douglas function by labeling these factors such as in Argumentation Theory (Grossi, 2010) and then check through experimentation and by validating the results. Moreover in studying external factors like teachers effort and ability can be looked at by keeping in view Attribution Theory, how teachers and students effort and motivation can contribute to increased probability of achievement behavior (Weiner, 1972) (Kelley & Michela, 1980).

Some research has been done in the past that is used in admitting students in different programs which is based on their past performance i.e.,, grades and entrance test results (Adhatrao, Gaykar, Dhawan, Jha, & Honrao, 2013) and that research was based on the past performance of the students and the primary independent variables used in this research were merit scores or marks scored in the entrance examination, gender, the percentage of marks scored in Physics, Chemistry, and Mathematics in the board examination of class XII and admission type. "Finally, the class attribute was added, and it held the predicted result, which can be either "Pass" or "Fail"" (Adhatrao et al., 2013). However, PTLS placement model research is based on primarily personality types, and learning styles and chances of success are calculated for improved decision making by combining different models and methodologies; moreover, external factors related to the student success are also looked at for better decision making.

CHAPTER 3: METHODOLOGY

The purpose of this research project was to devise a methodology and a model for better predictions and improved recommendations for placement. This model can contribute to an increased rate of success and also assist the decision makers to improve their program effectiveness when it comes to student teaching and learning with respect to different subjects (course of studies such as the Arabic, French, and Russian languages). In short, the aim was, “how to devise a better predictive model based on students’ personality types and learning style preferences, which leads in determining an increased rate of success?”

Overview

In this research, a hybrid approach was proposed, where two different processes were combined to devise a Hybrid Predictive Model for improved predictions and student placement. The results from the Hybrid Model would be more accurate and closer to the actual results as compared to the predicted results from the two different processes separately. This problem was taken as a classification problem. In Phase 1, given data was normalized and converted for Machine Learning (Salzberg, 1994). Then by using a Support Vector Machine (S. B. Kotsiantis, 2007) and Decision Tree Algorithm Classifiers (Neville, 1999) (Safavian & Landgrebe, 1990), a classifier/model was developed to test the data for predictive results. Actual results were then compared with the predicted results and the difference between actual and predicted results was analyzed to make Phase 1 recommendations. Different language results were also compared and an additional run with various classifiers were performed to see if the student with a particular PTLS would perform the same way in other languages. From the classifiers, a combined success probability of the results with reference to each PTLS was calculated. In Phase 2, PTLS preferences were categorized through probabilistic classification. The success rate of each personality type and learning style was calculated, and updated whenever any new data was added. The success rate of each PTLS was automatically updated whenever any new data was added. This updating reveals any change in success rate and also shows the consistency in the success rate based on each PTLS preference. Results from Phase 1 and Phase 2 were combined into a Hybrid Model by using the Cobb-Douglas function (Phase 3), which gives a more accurate prediction of the success rate of each PTLS with respect to the actual results. In addition to PTLS, this research explored another factor, teacher impact that may have influenced the results by using the Cobb-Douglas utility function.

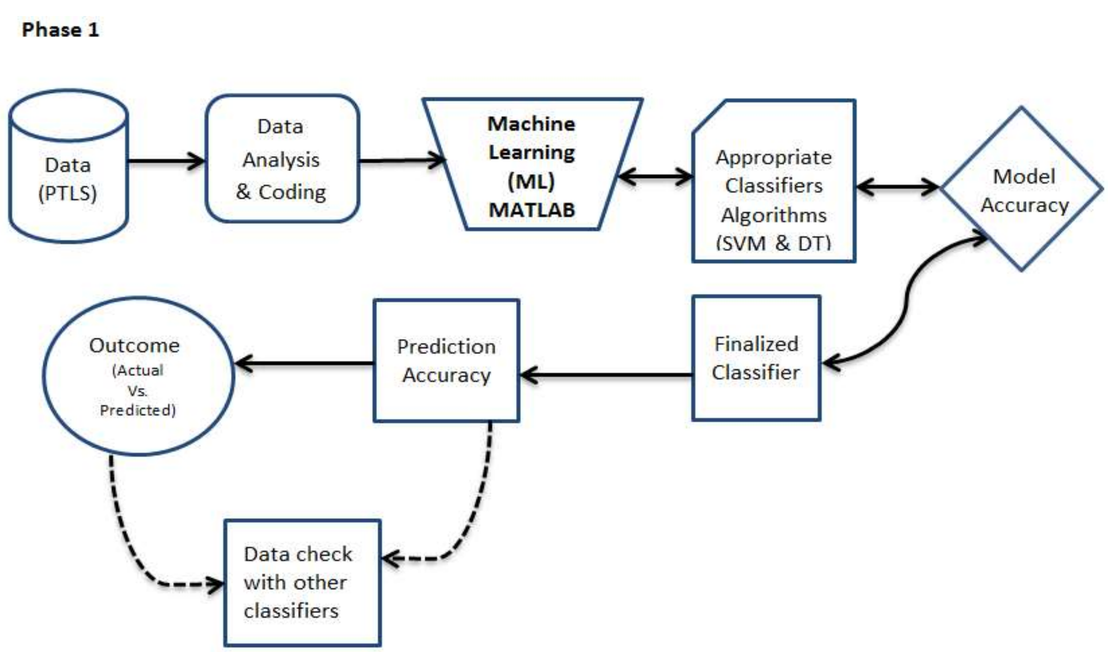


Figure 1. Phase 1

Phase 2

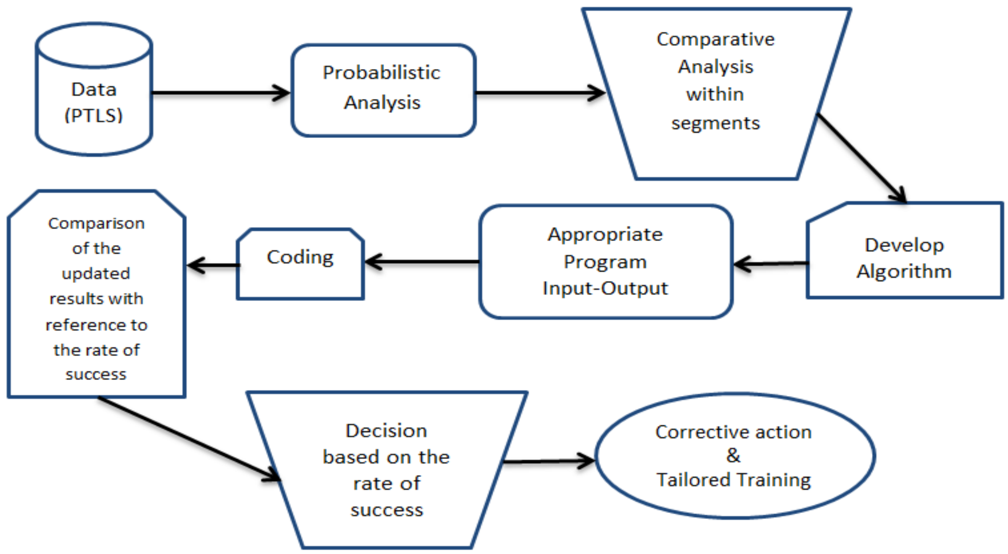


Figure 2. Phase 2

Phase 3

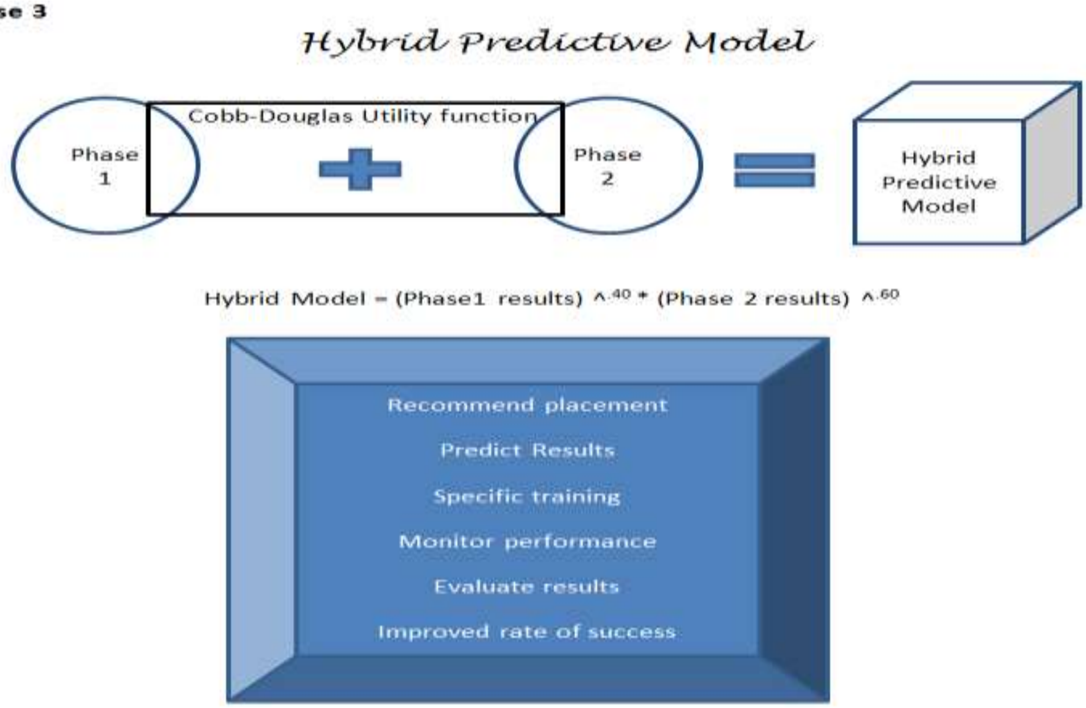
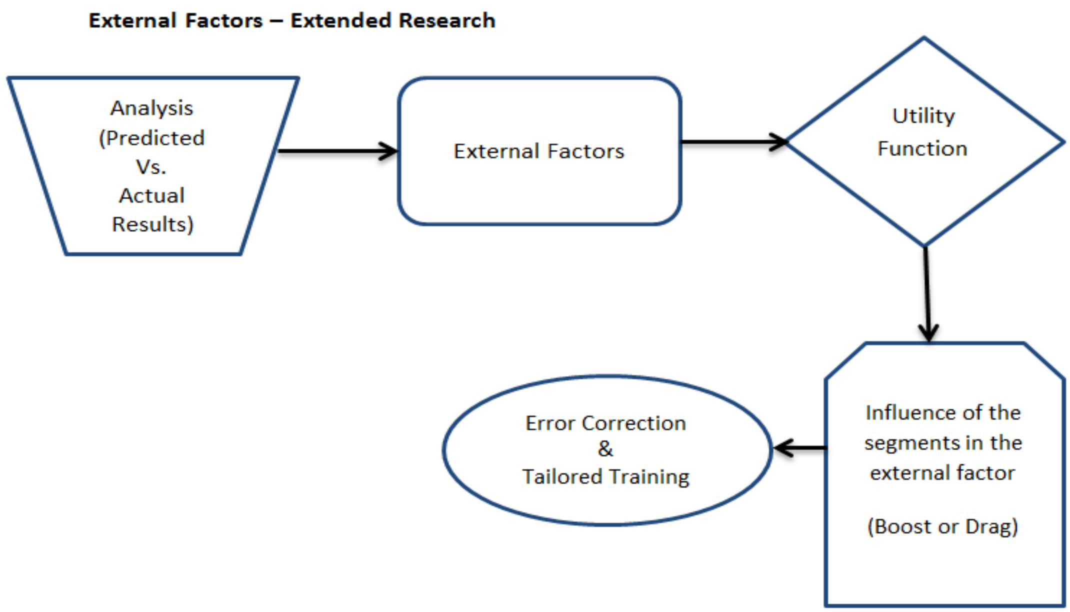


Figure 3: Phase 3



$$Y = (\text{TE-Experience})^{.25}(\text{TE-Qualifications})^{.35}(\text{TE-Effort})^{.40}$$

Figure 4: External Factors – Extended Research

Detailed overview

Phase 1

In Phase 1, given data was cleaned and coded for the Machine Learning process. In Machine Learning, different classifiers and algorithms were analyzed, and then the data was run by selecting specific classifiers. In the literature review section of this paper, the explanation for selecting and using different classifiers such as decision trees and SVM classifiers is already discussed (Meyer & Wien, 2015; Neville, 1999; Safavian & Landgrebe, 1990; Wu et al., 2008).

In the first phase, Machine Learning process was done using MATLAB. Result data from Russian, French, and Arabic languages was used separately to develop Trained Classifier Models in each language. This research was primarily based on PTLS preferences (personality types and learning style preferences), and one of the primary objectives was to check the influence of PTLS with respect to student performance. Here student performance was measured by the Oral Proficiency Interview (speaking proficiency test) (Liskin-Gasparro, 2003).

Three different languages of three separate category levels were taken, i.e., three different languages of three different difficulty levels ("DLAB Test Score Range," 2014). Russian is considered CAT 3 language (Category 3), Arabic is considered CAT 4 (Category 4) and French considered as CAT 1 (Category 1). In this case, students' performance can also be analyzed with respect to PTLS preferences when it comes to different subjects with various difficulty levels. In learning style, the primary attributes taken for predictive modeling were Global vs. Particular and Deductive vs. Inductive. Age and overlapping PTLS attributes data was also used with the original PTLS data for the main model. Some additional features in conjunction with PTLS preference attributes were also used in separate iterations to check any significant impact in improving the model or prediction accuracy. Additional features that were looked at to check for any significant impact included DLAB scores, other languages spoken by the students, education level, status (rank or job title if available), additional learning style attributes such as field independent vs. dependent, field sensitive vs. insensitive, and analog vs. digital.

It is to be noted that DLAB score was not available for all of the student data; this significantly reduces the data size, and the focus in this specific research was mainly PTLS preferences with respect to the OPI performance. Here performance was measured by OPI scores (speaking skill proficiency). The current standard in LASC is ILR level 1+ in OPI speaking. Though the standard is ILR level 1+ but the management desires the students to get ILR level 2 or above in speaking in the same time period. In this project

ILR level 1+ or below in speaking was considered a failure or “not successful” and ILR level 2 or above in speaking was considered as pass or “successful.” For the record, management is striving for the students to exceed the standard i.e., get ILR level 2 or above in speaking. This was one of the reasons that management was also looking into the PTLS preferences i.e., if there can be a significant effect on student performance by tapping into PTLS preferences. In the last two years, overall results indicated, 35-47% of the students exceeding standards i.e., getting ILR level 2 or above. Depending on the specific language, exceeding standards was anywhere from 5% to %100; in this project, Russian, Arabic, and French are considered only (due to data and other limitations i.e., these three languages had more data than other languages such as Chinese, Korean, Thai, etc.). Based on the given data for this project, French results exceeded standards 17%, Russian exceeded standards 65%, and Arabic exceeded standards 19%; here the base standard was ILR level 1+. Management was always looking for the ways and methods that would contribute to exceeding standards, which was obvious from the past 24 months’ results. If the PTLS preferences with respect to this OPI performance outcome is also investigated, and the findings from the PTLS predictive modeling can be integrated in the ongoing training then this may further improve results predictability, which would help evaluate and improve further the program evaluation process. For comparison purposes and to look at any significance, combined results of Arabic-French-Russian were also looked at and were referred to as CLN (languages combined).

When the data was run using Machine Learning processes, model accuracy was checked with respect to Decision Trees and SVM classifiers. Trained Classifiers with the highest accuracy was selected for the test data and predictions. Trained Classifier

(Model) with the highest accuracy for each language was formulated. Test data for each language was run with the respective trained classifier to get the predicted results i.e., test data of RU (Russian language) was run with the RU trained classifier. After running the test data of the specific language with its' respective classifier model the test data of that language was also run with the classifiers of the other languages' trained classifiers i.e., test data of RU was run with the RU Trained Classifier and also run with the Arabic (AD) Trained Classifier and the French (FR) Trained Classifier. This was done to check two things i.e., if a specific student may be more successful in the other language i.e., student placed in the Russian Language had the potential and predictability to pass in the Russian language, but could the student pass in the Arabic and French languages too. Further by using the classifiers with this way, one could also predict if a student failed in the Russian language, would the student pass in the Arabic or French language. Moreover, if at least two of the classifiers predict that the student would succeed then the success rate for that student should be high to pass. This was another reason three languages of different category levels were taken i.e., Category1 (CAT1) is considered easier than CAT3 or CAT4, CAT3 is considered easier than CAT4. French is CAT1, Russian is CAT3, and Arabic is CAT4. Once could also look at the results, if the difficulty level had to do anything with the results. If the classifier predicts that student passes in one language than the other two then the probability of that student passing in that particular language should be higher as compared to the other two languages; so the student could be recommended to be placed in the language where that specific PTLS preference student had a higher probability or higher rate of success as compared to the other languages. Also, if all the classifiers were predicting that the student would succeed in all the three languages then the specific student had much higher probability to pass in any of the languages as compared to other students that were not predicted to pass in all three.

The predicted results from the trained classifiers could also be used in other ways to check the pass or fail predictability and to assist the decision makers when the decision makers were asked to place the students in specific programs (scientifically); these predictions could be used in conjunction with another selection criterion (if there is any) for a better chance of success rate.

In some cases, there might be no selection criterion, and it was just the organizational requirements, or due to the short-staffed issue, there was a need for the individuals in that organization. If the classifier was predicting that the student might fail in the Russian language but might pass in French or Arabic, then it was recommended placing the student in the French or Arabic (which language the classifier was predicting to pass). However, if a student could not be placed in the other language and the only option was to be placed in the Russian language, then one could look into the actual and predicted results from the classifiers that showed which PTLS preference was successful in Russian and train the student to adapt those successful PTLS preference attributes.

In the end, the predicted and actual results were compared, and the difference between predicted and actual results was noted. Also, the model accuracy vs. the prediction accuracy was compared for further analysis and model improvement.

Phase 2

Phase 1 successfully transitioned into the Phase 2 of this project. Phase 2 looked into pure probability and probabilistic classification in each PTLS preference separately. In Phase 1, all the PTLS data of a specific language was processed as one segment or in a vertical sense to develop a predictive classifier. There were some shortfalls to this method in this specific environment; Phase 1 of the project could have projected more accurate results if data was uniformly distributed when it comes to the overall successes and failures and also if the data was uniformly distributed within each of the 16 personality types and learning styles.

In the second phase, data was looked at horizontally i.e., each PTLS preference was looked at individually or as a separate segment. The original data or the base data was in the LN-Comparison1-Base (in Excel) in BRKEVEN charts. BRKEVEN charts were calculated from the results (data) that showed the probability and success rate of a particular Personality Type within each language. BRKEVEN chart of the base data of each language was shown in the file LN1-BRKEVEN1 for Arabic; LN3-BRKEVEN1 for French and LN5-BRKEVEN1 for the Russian language. A BRKEVEN chart with all the three languages combined was also made for analysis, namely CLN-BRKEVEN1. Probability and the (increased) chance of success i.e., rate of success of each PTLS preference was calculated. This probabilistic model was working on the similar concept as of ensemble classifier where sub-segments of data are considered separately for probabilities. This probabilistic model gave % of increased chance of success (SR% - Success Rate %) with respect to different personality types and learning styles, which gave a different snapshot of success and failure as compared to the classifiers' results. This probabilistic model looked into specific PTLS rate of success within specific subjects i.e., in Russian it would be different than in Arabic or French.

This type of probabilistic model might become handy for an organization to get a snapshot that gave the rate of success of overall performance in different segments especially when one dealt with data where all the segments were not equally distributed, and also the data was not evenly distributed. One of the limitations of this project was that data was not big enough at this time. As data increases, the model can be improved for accuracy. In this phase, initially, a probability chart was formulated where the probability chart showed the rate of success of different PT and also further breakdown into specific LS permutations. In this project there were four LS permutations; if a student had a specific PT, then one could further refine the success factor by going into LS sub-segments. However, in this phase, the focus was more on the PT side then to go further into sub-segments of LS; much more data was needed for the LS that was not readily available at this time. However, a system was built which would accommodate the new data in improving prediction accuracy (which is elaborated in the later section – in PTLS Snapshot Program). Based on the data accumulation of around 24 months' period, a probability chart for each language was formulated; this probability chart was the base chart or the launching pad for the next step. Probability chart was called the BRKEVEN, and if it is for Russian language, it is called BRKEVEN LN5. For reference purposes, the initial or base chart with around 24 month's data was also called the 1-BRKEVEN-RU for Russian, and the other languages followed the same pattern respectively.

From these charts, Algorithm steps were devised to extract the data from the chart for further steps. Algorithm steps were devised for each Arabic, Russian, and French languages. Also, algorithm steps were devised for all the three languages combined as CLN[3] for an overall generic snapshot to compare it with each language success rate (to check the overall comparison with respect to the individual languages).

[3] All the three languages AD-FR-RU combined are referred to as CLN

Based on the available data, steps sequence of an algorithm for all the languages including the CLN are given below for reference:

Algorithm steps

For LN = 5 (Russian Language)

1. if personality = ENFJ then the increased chance of success rate = ENFJ 100%
   LS success rate = G-I = 2/3; P-D= 1/3
   Note: Preferred - Judger = Perceiver (Overlapping PT preferences)
   Note: Student numbers ** (less than 15)

2. if personality is not ENFJ then check ENFP

3. if personality = ENFP then the increased chance of success rate = ENFP 75%
   LS success rate = G-I = 2/3; G-D= 1/3
   Note: Preferred - Extreme Extrovert, Intuitor and if Inductive = Deductive or if Global = Particular (Overlapping PTLS preferences)
   Note: Student numbers ** (less than 15)

4. if personality is not ENFP then check ENTJ

5. if personality = ENTJ then the increase chance of success rate = ENTJ 88%
   LS success rate = G-I = 4/7; P-I= 2/7; G-D=1/7
   Note: Preferred- Inductive = Deductive Thinker = Feeler (Overlapping PTLS preferences)
   Note: Student numbers ** (less than 15)

6. if personality is not ENTJ then check ENTP

7. if personality = ENTP then the increased chance of success rate = ENTP 67%
   LS success rate = G-I = 4/4
   Note: Preferred - Inductive = Deductive (Overlapping LS preferences)
   Note: Student numbers ** (less than 15)

8. if personality is not ENTP then check ESFJ

9. if personality = ESFJ then the increased chance of success rate = ESFJ 50%
   LS success rate = G-D = 3/4; P-D = 1/4
   Note: Student numbers ** (less than 15)

10. if personality is not ESFJ then check ESFP

11. if personality = ESFP then the increased chance of success rate = ESFP 100%

LS success rate = G-D = 2/3; G-I = 1/3
Note: Student numbers ** (less than 15)

12. if personality is not ESFP then check ESTJ

13. if personality is ESTJ then the increased chance of success rate = ESTJ 65%
LS success rate = G-D = 20/47; G-I = 12/47; P-D = 12/47; P-I = 3/47
Note: Preferred - Overlapping with other LS

14. if personality is not ESTJ then check ESTP

15. if personality is ESTP then the increased chance of success rate = ESTP 71%
LS success rate = G-D = 2/5; G-I = 2/5; P-I = 1/5
Note: Preferred – Extreme Extrovert
Note: Student numbers ** (less than 15)

16. if personality is not ESTP then check INFJ

17. if personality is INFJ then the increased chance of success rate = INFJ 0%
Note: At this time enough data not available for validation.
Note: Student numbers ** (less than 15)

18. if personality is not INFJ then check INFP

19. if personality = INFP then the increased chance of success rate = INFP 100%
LS success rate = G-D = 1/3; P-D = 1/3; G-I = 1/3
Note: Preferred - Global = Particular
Note: Student numbers ** (less than 15)

20. if personality is not INFP then check INTJ

21. if personality = INTJ then the increased chance of success rate = INTJ 33%
LS success rate = G-I = 1/2; P-I = 1/2
Note: Student numbers ** (less than 15)

22. if personality is not INTJ then check INTP

23. if personality = INTP then the increased chance of success rate = INTP 100%
LS success rate = P-D = 1/1
Note: Student numbers ** (less than 15)

24. if personality is not INTP then check ISFJ

25. if personality = ISFJ then the increased chance of success rate = ISFJ 57%

LS success rate = G-I = 2/4; G-D=1/4; P-I=1/4
Note: Student numbers ** (less than 15)

26. if personality is not ISFJ then check ISFP

27. if personality is ISFP then the increased chance of success rate = ISFP 100%
LS success rate = G-D=1/1
Note: Student numbers ** (less than 15)

28. if personality is not ISFP then check ISTJ

29. if personality = ISTJ then the increased chance of success rate = ISTJ 61%
LS success rate = G-D=16/27; G-I=6/27; P-D=5/27
Note: Preferred - Extreme introvert; Inductive = Deductive (Overlapping PTLS preferences)

30. if personality is not ISTJ then check ISTP

31. if personality type is ISTP then the increased chance of success rate = ISTP 80%
LS success rate = G-I=3/4; G-D=1/4
Note: Preferred – Overlapping PTLS
Note: Student numbers ** (less than 15)

For LN = 3 (French Language)

1. if personality = ENFJ then the increased chance of success rate = ENFJ 33%
LS success rate = G-I = 1/1
Note: Student numbers ** (less than 15)

2. if personality is not ENFJ then check ENFP

3. if personality = ENFP then the increased chance of success rate = ENFP 100%
LS success rate = G-D = 1/1
Note: Student numbers ** (less than 15)

4. if personality is not ENFP then check ENTJ

5. if personality = ENTJ increased chance of success rate = 0%
LS success rate = N/A
Note: Student numbers ** (less than 15)

6. if personality is not ENTJ then check ENTP

7. if personality = ENTP then the increased chance of success rate = ENTP 0%
LS success rate = N/A

Note: Student numbers ** (less than 15)

8. if personality is not ENTP then check ESFJ

9. if personality = ESFJ then the increased chance of success rate = ESFJ 15%
   LS success rate = G-D = 1/2; P-D = 1/2
   Note: Student numbers ** (less than 15)

10. if personality is not ESFJ then check ESFP

11. if personality = ESFP then the increased chance of success rate = ESFP 50%
    LS success rate = G-I = 2/2
    Note: Student numbers ** (less than 15)

12. if personality is not ESFP then check ESTJ

13. if personality is ESTJ then the increased chance of success rate = ESTJ 21%
    LS success rate = G-I = 14/27; G-D = 7/27; P-D = 4/27; P-I = 2/27
    Note: Preferred - Overlapping with other LS

14. if personality is not ESTJ then check ESTP

15. if personality is ESTP then the increased chance of success rate = ESTP 30%
    LS success rate = G-D = 1/3; P-I = 1/3; P-D = 1/3
    Note: Student numbers ** (less than 15)

16. if personality is not ESTP then check INFJ

17. if personality is INFJ then the increased chance of success rate = INFJ 0%
    LS preference in INFJ = N/A
    Note: Enough data not available for any validation
    Note: Student numbers ** (less than 15)

18. if personality is not INFJ then check INFP

19. if personality = INFP then the increased chance of success rate = INFP 0%
    LS success rate = N/A
    Note: Note: Enough data not available for any validation
    Note: Student numbers ** (less than 15)

20. if personality is not INFP then check INTJ

21. if personality = INTJ then the increased chance of success rate = INTJ 0%
    LS success rate = N/A

Note: Note: Enough data not available for any validation
Note: Student numbers ** (less than 15)

22. if personality is not INTJ then check INTP

23. if personality = INTP then the increased chance of success rate = INTP 0%
LS success rate = N/A
Note: Note: Enough data not available for any validation
Note: Student numbers ** (less than 15)

24. if personality is not INTP then check ISFJ

25. if personality = ISFJ then the increased chance of success rate = ISFJ 20%
LS success rate = G-D=1/1
Note: Student numbers ** (less than 15)

26. if personality is not ISFJ then check ISFP

27. if personality is ISFP then increased chance of success rate = ISFP 0%
Note: Note: Enough data not available for any validation
Note: Student numbers ** (less than 15)

28. if personality is not ISFP then check ISTJ

29. if personality = ISTJ then the increased chance of success rate = ISTJ 14 %
LS success rate = G-D=4/9; G-I=3/9; P-D=2/9

30. if personality is not ISTJ then check ISTP

31. if personality type is ISTP then the increased chance of success rate = ISTP 10%
LS success rate = G-I= 1/1
Note: Student numbers ** (less than 15)

For LN = 1 (Arabic Language)

1. if personality = ENFJ then the increased chance of success rate = ENFJ 0%
LS success rate = N/A
Note: Note: Enough data not available for any validation
Note: Student numbers ** (less than 15)

2. if personality is not ENFJ then check ENFP

3. if personality = ENFP then the increased chance of success rate = ENFP 0%
LS success rate = N/A
Note: Note: Enough data not available for any validation

Note: Student numbers ** (less than 15)

4. if personality is not ENFP then check ENTJ

5. if personality = ENTJ then the increased chance of success rate = ENTJ = 57%
   LS success rate from high to low = G-I = 2/4; G-D = 1/4; P-D = 1/4
   Note: Preferred – Other language background and overlapping PTLS
   Note: Student numbers ** (less than 15)

6. if personality is not ENTJ then check ENTP

7. if personality = ENTP then the increased chance of success rate = ENTP 0%
   LS success rate = N/A
   Note: Note: Enough data not available for any validation
   Note: Student numbers ** (less than 15)

8. if personality is not ENTP then check ESFJ

9. if personality = ESFJ and LS = GD, GI then the increased chance of success rate = ESFJ 13%
   LS success rate = G-D = 1/2; G-I = 1/2

10. if personality is not ESFJ then check ESFP

11. if personality = ESFP then the increased chance of success rate = ESFP 33%
    LS success rate = P-D = 1/1
    Note: Student numbers ** (less than 15)

12. if personality is not ESFP then check ESTJ

13. if personality is ESTJ then the increased chance of success rate = ESTJ 20%
    LS success rate = G-D = 11/21; G-I = 7/21; P-D = 3/21
    Note: Preferred – Extreme Extrovert and overlapping with other LS

14. if personality is not ESTJ then check ESTP

15. if personality is ESTP then the increased chance of success rate = ESTP 18%
    LS success rate = G-I = 2/2
    Note: Student numbers ** (less than 15)

16. if personality is not ESTP then check INFJ

17. if personality is INFJ then the increased chance of success rate = INFJ 0%
    LS preference in INFJ = N/A

Note: Note: Enough data not available for any validation
Note: Student numbers ** (less than 15)

18. if personality is not INFJ then check INFP

19. if personality = INFP then the increased chance of success rate = INFP 0%
LS success rate = N/A
Note: Note: Enough data not available for any validation
Note: Student numbers ** (less than 15)

20. if personality is not INFP then check INTJ

21. if personality = INTJ then the increased chance of success rate = INTJ 20%
LS success rate = G-I = 1/1
Note: Student numbers ** (less than 15)

22. if personality is not INTJ then check INTP

23. if personality = INTP then the increased chance of success rate = INTP 50%
LS success rate = G-D = 2/3; G-I = 1/3
Note: Preferred – Prior background in other languages
Note: Student numbers ** (less than 15)

24. if personality is not INTP then check ISFJ

25. if personality = ISFJ then the increased chance of success rate = ISFJ 33%
LS success rate = G-D=1/2; P-D = 1/2
Note: Student numbers ** (less than 15)

26. if personality is not ISFJ then check ISFP

27. if personality is ISFP and LS = G-D then increased chance of success rate = ISFP 100%
LS success rate = G-D= 1/1
Note: Note: Enough data not available for validation
Note: Student numbers ** (less than 15)

28. if personality is not ISFP then check ISTJ

29. if personality = ISTJ then the increased chance of success rate = ISTJ 8 %
LS success rate = P-D=2/4; G-D=1/4; G-1=1/4

30. if personality is not ISTJ then check ISTP

31. if personality type is ISTP then the increased chance of success rate = ISTP 20%
    LS success rate = G-D= 1/2; G-I = 1/2
    Note: Student numbers ** (less than 15)

For CLN = (Generic for the combined Arabic, French, and Russian Languages)

1. if personality = ENFJ then the increased chance of success rate = ENFJ 57%
   LS success rate = G-I = 3/4; P-D = 1/4
   Note: Student numbers ** (less than 15)

2. if personality is not ENFJ then check ENFP

3. if personality = ENFP then the increased chance of success rate = ENFP 67%
   LS success rate = G-I = 2/4; G-D = 2/4
   Note: Student numbers ** (less than 15)

4. if personality is not ENFP then check ENTJ

5. if personality = ENTJ then the increased chance of success rate = ENTJ = 61%
   LS success rate = G-I = 6/11; G-D = 2/11; P-I = 2/11; P-D = 1/11

6. if personality is not ENTJ then check ENTP

7. if personality = ENTP then the increased chance of success rate = ENTP 36%
   LS success rate = G-I = 4/4
   Note: Student numbers ** (less than 15)

8. if personality is not ENTP then check ESFJ

9. if personality = ESFJ then the increased chance of success rate = ESFJ 22%
   LS success rate = G-D = 5/8; P-D = 2/8; G-I = 1/8

10. if personality is not ESFJ then check ESFP

11. if personality = ESFP then the increased chance of success rate = ESFP 60%
    LS success rate = G-I = 3/6; G-D = 2/6; P-D = 1/6
    Note: Student numbers ** (less than 15)

12. if personality is not ESFP then check ESTJ

13. if personality is ESTJ then the increased chance of success rate = ESTJ 31%
    LS success rate = G-D = 38/95; G-I = 33/95; P-D = 19/95; P-I = 5/95

14. if personality is not ESTJ then check ESTP

15. if personality is ESTP then the increased chance of success rate = ESTP 36%
    LS success rate = G-I = 4/10; G-D = 3/10; P-I = 2/10; P-D = 1/10

16. if personality is not ESTP then check INFJ

17. if personality is INFJ then the increased chance of success rate = INFJ 0%
    LS preference in INFJ = N/A
    Note: Enough data not available for validation (Student numbers**)

18. if personality is not INFJ then check INFP

19. if personality = INFP then the increased chance of success rate = INFP 50%
    LS success rate = G-D = 1/3; G-I = 1/3; P-D = 1/3
    Note: Student numbers ** (less than 15)

20. if personality is not INFP then check INTJ

21. if personality = INTJ then the increased chance of success rate = INTJ 27%
    LS success rate = G-I = 2/3; P-I = 1/3
    Note: Student numbers ** (less than 15)

22. if personality is not INTJ then check INTP

23. if personality = INTP then the increased chance of success rate = INTP 57%
    LS success rate = G-D = 2/4; G-I = 1/4; P-D = 1/4
    Note: Student numbers ** (less than 15)

24. if personality is not INTP then check ISFJ

25. if personality = ISFJ then the increased chance of success rate = ISFJ 39%
    LS success rate = G-D = 3/7; G-I = 2/7; P-D = 1/7; P-I = 1/7

26. if personality is not ISFJ then check ISFP

27. if personality is ISFP then increased chance of success rate = ISFP 67%
    LS success rate = G-D = 2/3
    Note: Student numbers ** (less than 15)

28. if personality is not ISFP then check ISTJ

29. if personality = ISTJ then the increased chance of success rate = ISTJ 25%
    LS success rate = G-D = 21/40; G-I = 10/40; P-D = 9/40

30. if personality is not ISTJ then check ISTP

31. if personality type is ISTP then the increased chance of success rate = ISTP 28%
    LS success rate = G-I = 5/7; G-D = 2/7

Based on the above mentioned 31 algorithm steps for each of the given languages, a program was devised in C#. Algorithm steps were coded with respect to C# program script. Program was named as PTLS BRKEVEN Snapshot. Two BRKEVEN charts for each language in MS Excel were uploaded and also any notes which elaborated the PTLS success rate in the text format. First BRKEVEN chart was taken from the LN-Comparison1-Base (Excel file) which had the base data[4]. LN1-BRKEVEN1 was for AD data, LN3-BRKEVEN3 was for French data, and LN5-BRKEVEN5 was for the Russian data. Then the second BRKEVEN chart was taken from the LN-Comparison2-Update (Excel file), which had the updated data. A feature to regularly update the data whenever the new data would be available was integrated into the system to stabilize the system; this feature would also help in analyzing any drastic changes in the success rate of a particular PTLS to further look into the causes of any of these drastic changes. This feature was made by keeping in view the current data; data in some PTLS was limited, but with the passage of time whenever more data would be available, the updated data would give a better picture to check the consistency in the success rate of each PTLS for any particular language. LN1-BRKEVEN2 for the updated Arabic data, LN3-BRKEVEN3 for the updated French data, and LN5-BRKEVEN5 was for the updated Russian data. Whenever the data was updated then, the previous LN-Comparison2-

[4] Base data is the data, which is initially gathered to check the success rate probability. In some places it is also referred to as the reference data or the original given data. Whenever the new data comes, the base data is updated and called the updated data. Whenever the updated data is again updated then the new updated data is called the updated data and the previously updated data becomes the base data. Base data and updated data are regularly updated whenever the new data is available and these data sets are regularly compared to see the rate of change, which shows any change in the rate of success in specific PTLS.

Updated file became the LN-Comparison1-Base, and the newly updated data file would become the Comparison2-Updated file. In this way the base data file can also be called the previous data and the updated data file can be called as the new current or updated data file.

For now, further findings of sub-segments were in the notes section, which gave further information in improving the decision-making process (due to the limited data, resource, and time limitations it was kept in this format; it could be updated and used in other formats later). If the data becomes huge then a database can also be attached in the same way.

PTLS BRKEVEN Snapshot program details

There are seven screens in the PTLS BRKEVEN Snapshot program. In Screen1, data and text files are uploaded. The first file is LN-Comparison1-Base, which has the initial 24 months data (as a base data file or point of reference). This LN-Comparison1-Base file has data of Arabic, French, Russian, and CLN data. This was shown as Excel on the screen 1. Then the LN-Comparison2-Update file was uploaded in the Final Excel box, which was the updated data of AD, RU, FR, and CLN respectively i.e., whenever new data was available, this file had the updated data with reference to the Base data, (Base data file was the original point of reference). Then the text file was loaded in the Text box; this text file had further information and success rate of different learning styles preferences derived from the data.

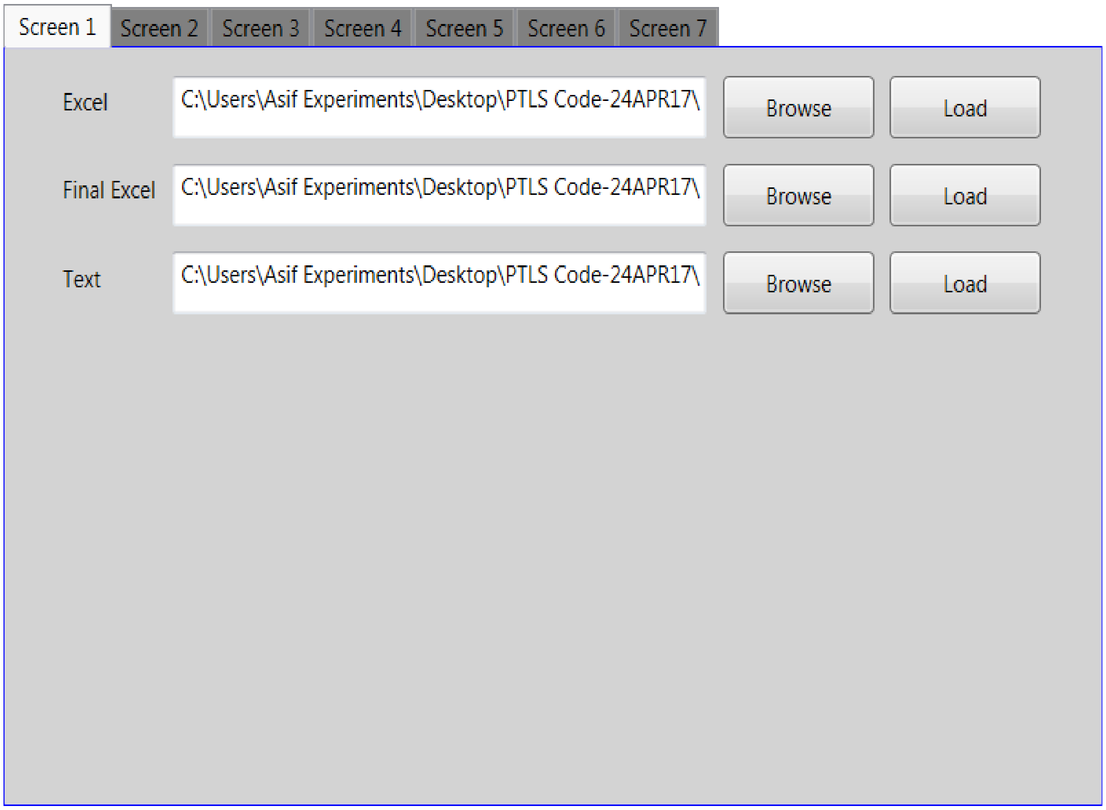
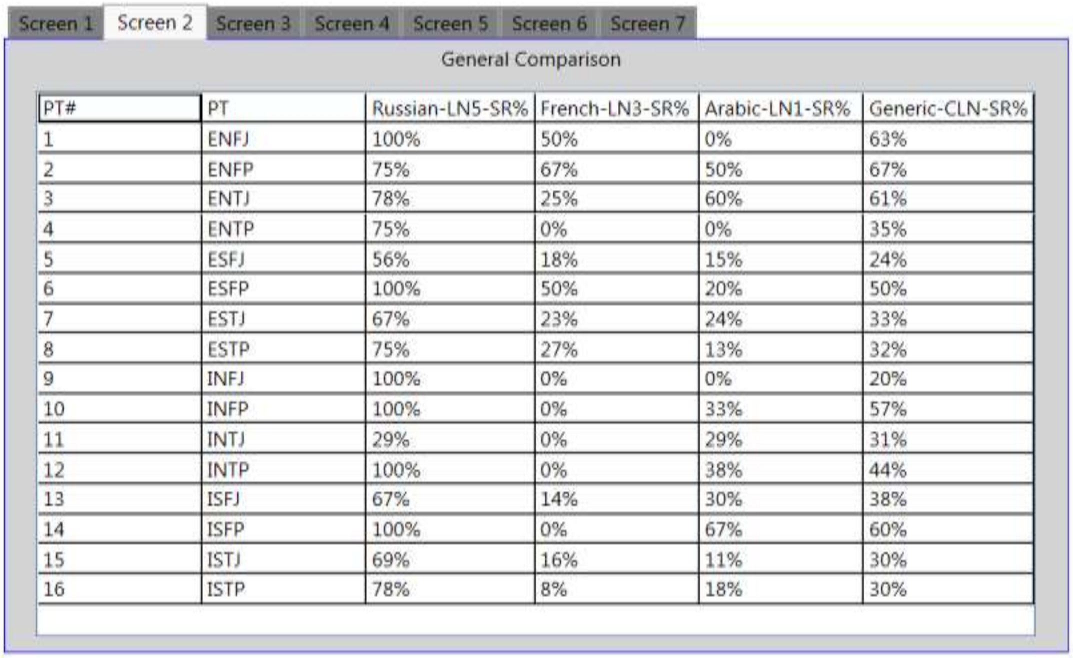


Figure 5: PTLS BRKEVEN Snapshot Program Screen 1

The second screen gave the overall output of each personality type success rate as compared to the other personality type preferences with respect to each language and CLN.

The third screen gave the output with respect to each PT and language plus the probability and success rate of each LS and any additional notes derived from the data.



PT#	PT	Russian-LN5-SR%	French-LN3-SR%	Arabic-LN1-SR%	Generic-CLN-SR%
1	ENFJ	100%	50%	0%	63%
2	ENFP	75%	67%	50%	67%
3	ENTJ	78%	25%	60%	61%
4	ENTP	75%	0%	0%	35%
5	ESFJ	56%	18%	15%	24%
6	ESFP	100%	50%	20%	50%
7	ESTJ	67%	23%	24%	33%
8	ESTP	75%	27%	13%	32%
9	INFJ	100%	0%	0%	20%
10	INFP	100%	0%	33%	57%
11	INTJ	29%	0%	29%	31%
12	INTP	100%	0%	38%	44%
13	ISFJ	67%	14%	30%	38%
14	ISFP	100%	0%	67%	60%
15	ISTJ	69%	16%	11%	30%
16	ISTP	78%	8%	18%	30%

Figure 6: PTLS BRKEVEN Snapshot Program Screen 2

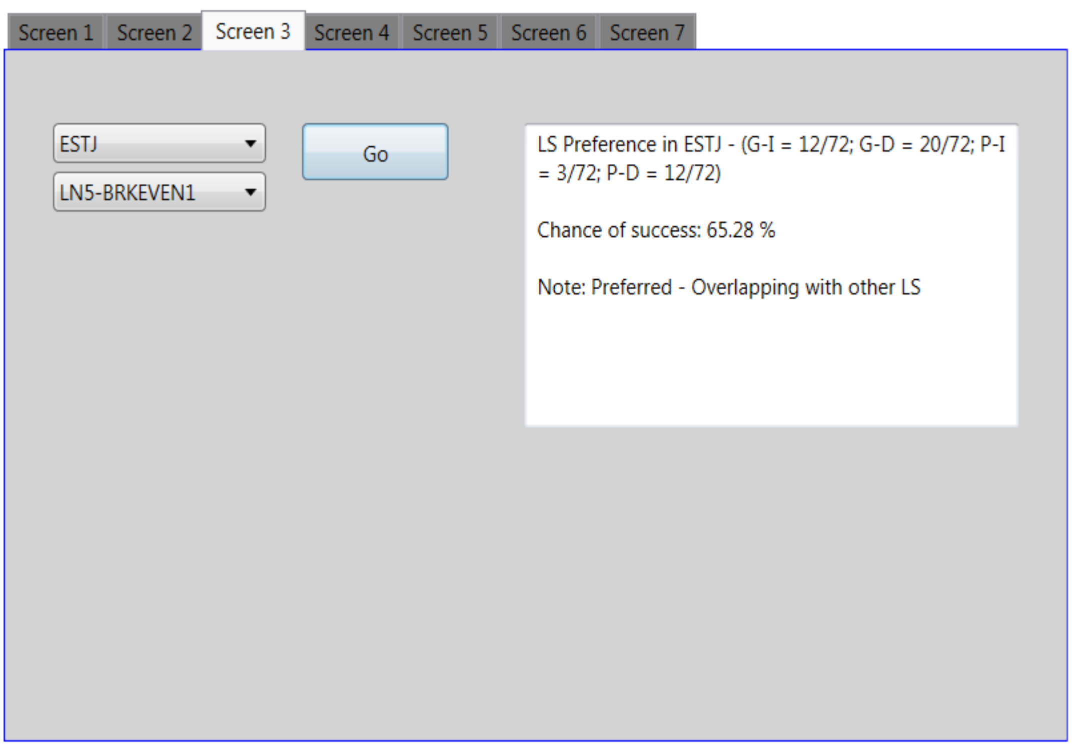


Figure 7: PTLS BRKEVEN Snapshot Program Screen 3

With Screen 4, one could compare the success rate of a specific PTLS with respect to other languages or the success rate of different PTLS with respect to each language.

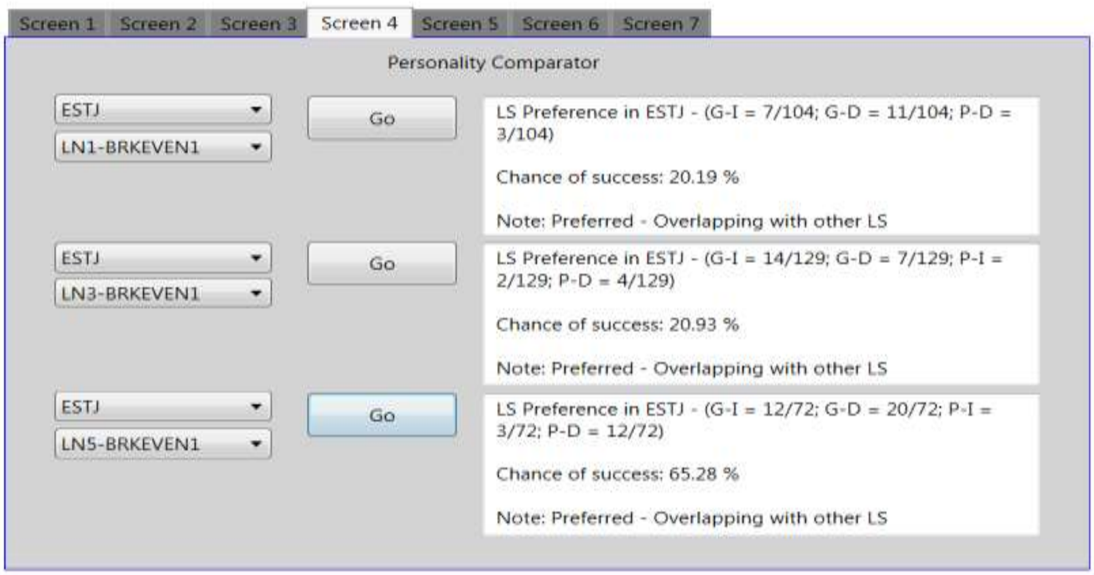


Figure 8a: PTLS BRKEVEN Snapshot Program Screen 4

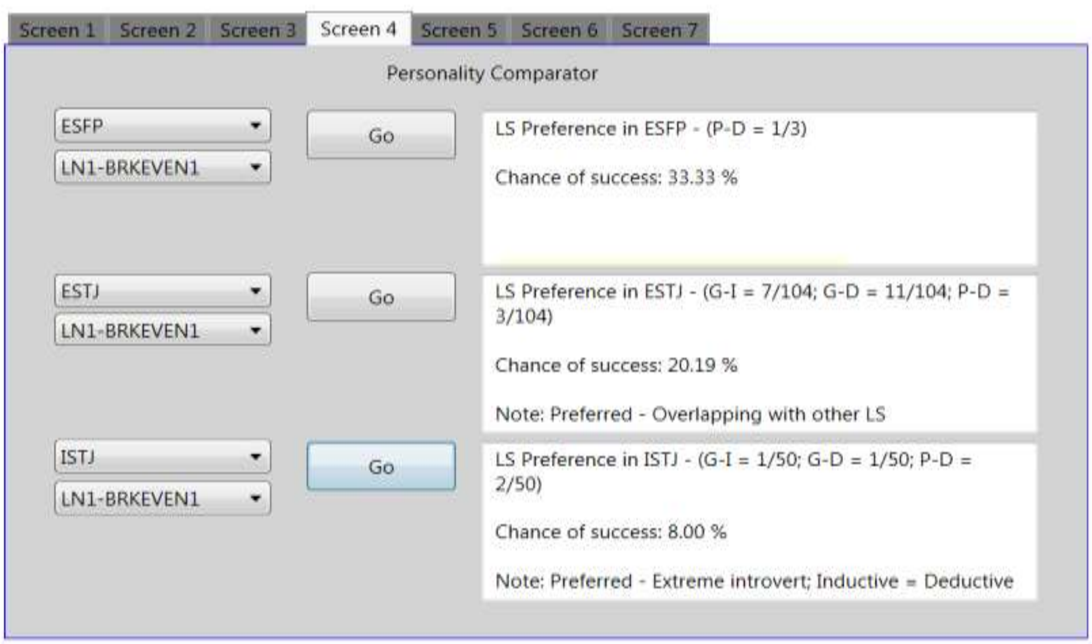


Figure 8b: PTLS BRKEVEN Snapshot Program Screen 4

In screen 4, the success rate of a particular PT could be checked with respect to different languages. In screen 4, the success rate of different personality types could also be checked in the same language. This enabled decision makers or the trainers to check, which PT preference had higher success rate in a particular language; this could also be interpreted as which PT preference was performing better as compared to the other PT preferences in different languages. If a specific PT, results showed that the typical PT was not performing better as compared to another PT in a specific language, then that student could be taught or trained to adapt the learning ways of the PTLS that were performing better in that language. As more data would be added over time, the model would become more stable with better and more consistent results. AD, FR, and RU language programs are ongoing programs, and if needed and authorized, then more data can be accrued to improve the model.

In Hybrid Model, the outcome of Phase 1 was combined with the success rate of different PTLS preferences in the probabilistic model through Cobb-Douglas utility function for better predictions and an improved decision making process. In Phase 1, classifier results within the same language were looked at first. Test data of a specific language was run with the model (trained classifier) of that language. In light of Argumentation theory and Attribution theory one could infer that if every student of the same PT type was predicted successful in that language through the trained classifier of that language then the success rate of that PT should be 100%. If the success rate was 100% with the trained classifier of that language then one could recommend that student for placement in that language; for example in RU, 12 ESTJ students from the test data showed that the P1 phase success rate was 100%. However, the actual results showed that the success rate was 75% i.e., nine out of twelve students were successful. P2 phase showed that the success rate of ESTJ was 67%. However, actual success rate was 75%. But when P1 and P2 phases are combined through Cobb-Douglas utility function then the result was 79%, which is more accurate and closer to the actual results (75%) as compared to P1(100%) or P2(67%) phase alone.

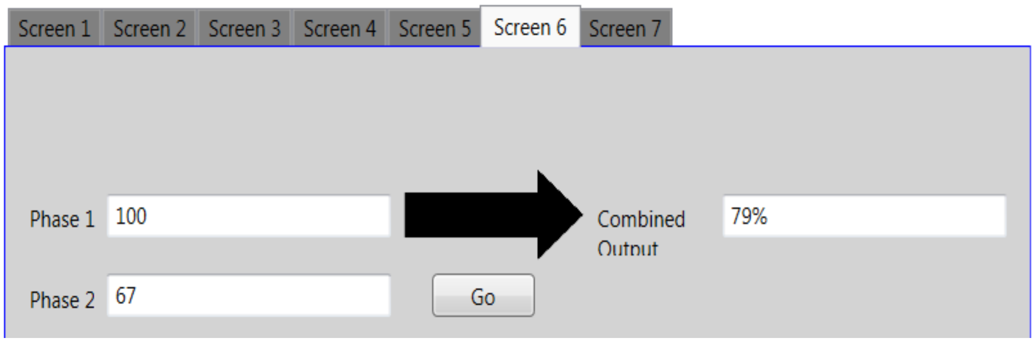


Figure 9: PTLS BRKEVEN Snapshot Program Screen 6

From the literature review, "Cobb-Douglas function showed the relationship between two or more inputs" (Black & Lynch, 1996) and according to (Hadzikadic et al., 2013) "Cobb-Douglas utility function could incorporate relevant theories and factors which showed some kind of a relationship between variables, and it could instantiate different theories by adjusting parameters"; in this case Cobb-Douglas function could be used to show relationship between two or more inputs by combining those inputs to show the aggregate results from Phase 1 and Phase 2 for the Hybrid Model.

Hybrid Model ➔ Combined output = (P1 results) $^{\wedge .40}$ * (P2 results) $^{\wedge .60}$

Hybrid Model was the combined output of Phase 1 and Phase 2. This combined output was more accurate with reference to the actual results. By combining Phase 1 and Phase 2 results, management could foresee more accurate results. Management could also recommend students for placement in a specific langauge where student PTLS had significantly higher success rate. Here P1 and P2 has .40 and .60 exponents. By using Argumentation and Attribution theory (Grossi, 2010; Kelley & Michela, 1980) we could infer that in P1 results were not updated on a regular basis as P2. Initially the test data was checked with the same language classifer and if the success rate was not 100% then the test data was run with the classifiers of other languages to check if the student would pass in any of the other languages. If the success rate was not 100% with the same language classifer but the other language classifiers showed success then half % success was assigned as a weight to the success of the primary language. One does not need to go to the results of the other languages if the primary language has 100% success rate; one only looks at the other languages success rate if the primary language success rate is not 100%. If the other two languages show success then that success rate is added in the primary language success rate for better accuracy. Reason behind this logic is that if one student is predicted to fail with all three classifiers as compared to a student who is predicted to fail in the primary language but showed success with another langauge has higher probability of success then the student who is not successful in any of the languages. To capture that success factor the success rate showing success in other languages was taken into account as 50% of that success if the primary language success was not 100% i.e. highest success rate divided by two from one of the other classifiers.

With predictive modeling one tries to predict the results but sometimes the predicted results are not 100% accurate with reference to the actual results and the difference in actual and predicted results seem completely illogical. Hypothetical Scenarios are discussed for clarification and explanation – Scenario A - For example, the trained classifier of a specific language is predicting that a particular student (student

“X”) would pass in RU and the probabilistic snapshot also reveals that the rate of success is 70% for that particular PTLS. There is another student (student “Y”) predicted to fail by the trained classifiers in the Phase 1 and the rate of success in the second phase is also showing 30%, which is very low as compared to the student “X”. In reality, this particular student “X” should have a much higher chance of success as compared to the student “Y”. However, the actual results showed that the student who was shown successful by the trained classifier and whose success rate in the probabilistic classification was much higher did not succeed and the student who was shown not successful by the classifiers and whose probability was much lower i.e., 30% success rate succeeded in the actual results.

Scenario B - In this scenario, if probablistic analysis shows, three out of four of the same PT were successful in a particular language consistently then success rate of that PT would be 75% but in one instance only one out of four PT were successful (25% success) in that particular language which was consistently showing 75% success rate.

Scenario C - In this scenario, if all the three classifiers are predicting success of a specific student PTLS in Arabic but the student actually fails in Arabic. What can be the reasons for these drastic discrepancies in actual and predicted results? This leads to the introduction and concept of drag or boost effect in this Hybrid Model.

Drag or boost effect

Prediction accuracy may be much higher when Phase 1 and Phase 2 results are used in conjunction with each other; both phases are giving a different perspective of success. Aggregate analysis was used to enhance student placement or student training; however, here the focus was limited to the predictive analysis. Even though the accuracy rate of this type of Hybrid Model could be high but there might still be differences in actual and predicted results as mentioned above in the Hypothetical Scenarios. That difference in the actual vs predicted results could be due to some external factor, which was other than the PTLS. One could refer to the Argumentation Theory (Grossi, 2010) to infer that due to the influence of some external factor there was a big difference between actual and predicted results and in this environment, most probably the external factor could be the teacher effect. Why teacher effect? Because teachers were teaching the students and teachers spent most of the time with the students during their language training. If teachers were well trained and well qualified then teachers could tailor their instruction as per student needs and students PTLS preferences. Teachers could also train the students in teaching/facilitating their students to use their (students) specific PTLS preferences to overcome their (students) learning weaknesses with respect to their PTLS preferences. Actual results were checked with the predicted results. If the actual results were better than the predicted results than it would be called a boost. If the actual results were worse than the predicted results than it would be called a drag. This difference in predicted to actual result i.e., the boost or drag effect was contributed to the external factor.

Screen 5 calculated the difference in success rate of a specific PT within a specific language. If the new success rate (updated/current data) was better than the previous success rate (base data) then that would be a boost.

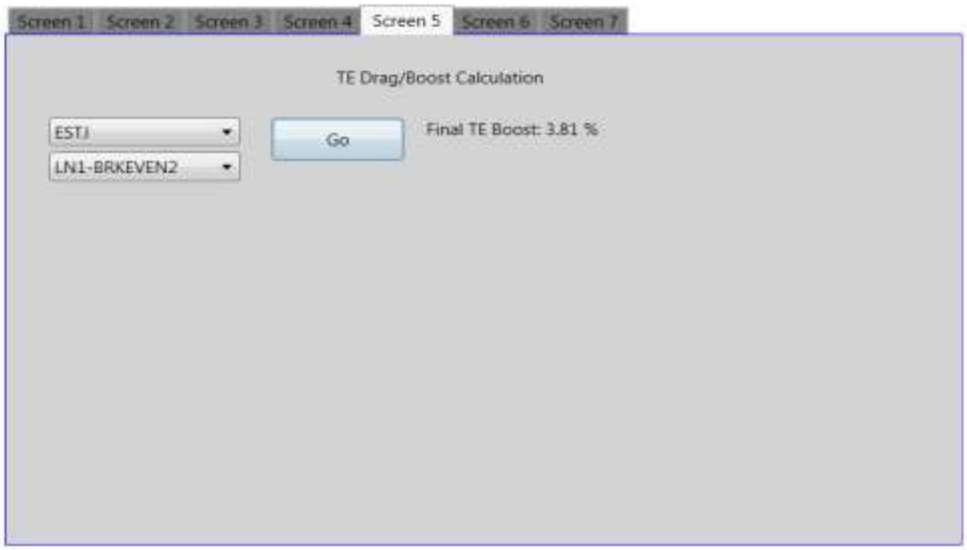


Figure 10a: PTLS BRKEVEN Snapshot Program Screen 5

If the new success rate (updated/current) was worse than the previous success rate (base/point of reference), then that would be a drag.

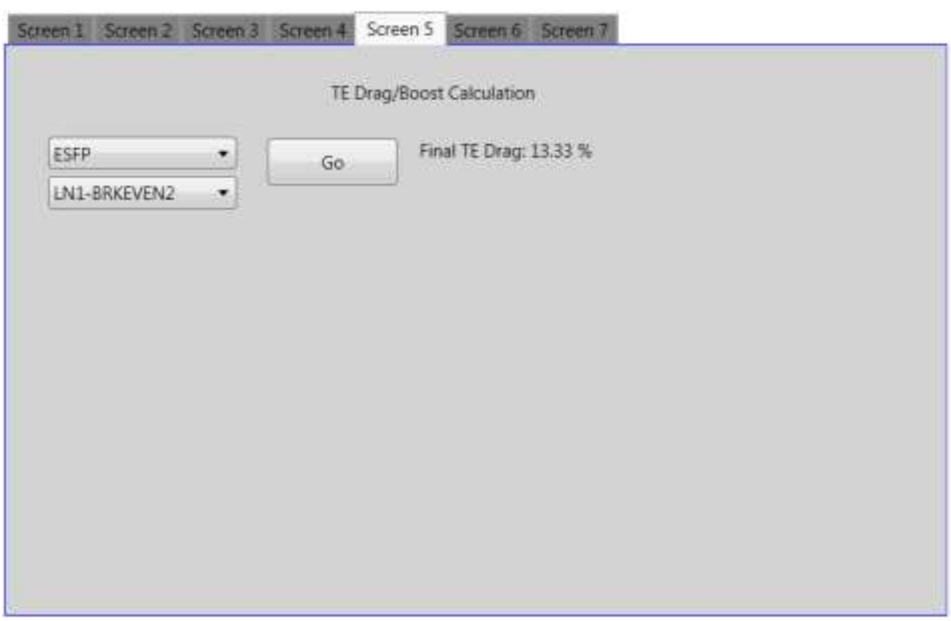


Figure 10b: PTLS BRKEVEN Snapshot Program Screen 5

If there was no difference between the BRKEVEN1 SR% and BRKEVEN2 SR% i.e., if the success rate was same then there would be no further processing.

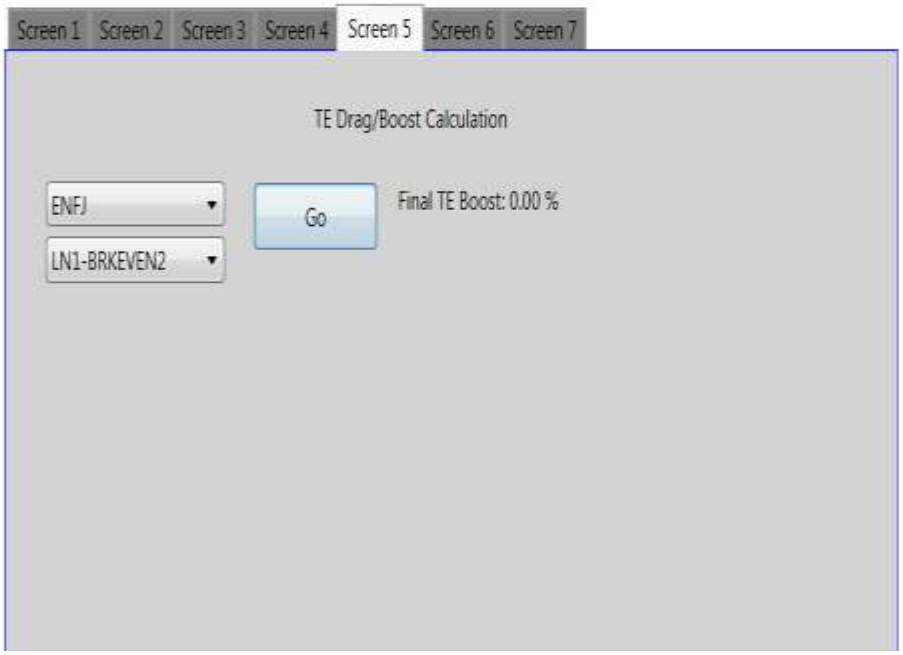


Figure 10c: PTLS BRKEVEN Snapshot Program Screen 5

If the previous success rate was lower than current success rate, i.e., for demonstration purpose: if the previous success rate of ESTJ in LN1 was 20% but the current success rate was around 24%. Then the boost was around 4%. The 4% difference (improvement) is called a boost. If the previous success rate was higher than the current success rate, i.e., previous success rate of ISTJ in LN3 was 14% and the current rate was 7.41% then the current success rate showed a decline of 6.59% or -6.59%; this decline in the success rate was called a drag. That drag and the boost could be discussed further with respect to future research, which would make the Hybrid Predictive Model more detailed oriented with respect to the difference in actual and predicted results. It was also proposed that the teacher effect could further be calculated by a utility function such as Cobb-Douglas utility function (Hadzikadic et al., 2013).

There might be various external factors but it was observed that the student learning and progress was usually more engaged, smoother, or better when the teacher was qualified, trained, and experienced as compared to the class where teacher was not trained, qualified, or experienced. Sometimes there were two equally qualified and trained teachers teaching the same subject but in one class, students were more engaged then the other class due to the teacher's creativity and passion to engage the students; sometimes one teacher was more creative, passionate, and innovative in teaching the same thing as compared to the other teacher; sometimes a teacher was more motivated as compared to the other teacher. Sometimes the same teacher was more enthusiastic and involved in a class at one point in time, but might not be that involved in the same type of class in another point in time; it could be due to the things going on in that teacher's personal life such as divorce or family crisis, etc. Factors such as a teacher going through divorce or a family issue could affect teacher passion, motivation, and innovation-creativity to teach a specific class. In PTLS BRKEVEN Snapshot program, screen 5 calculated the drag or boost factor i.e., the increase or decrease of success rate over time. This drag or boost factor was also the difference in predicted and actual success rate OR between previous and latest/current results that showed success rate. This difference in actual and predicted results OR previous and latest/current/updated results could be due to the external factors and in this case the external factor was assumed as the teacher's effect because it was observed that teacher's qualifications, experience, and the way of teaching (passion-motivation-innovation-creativity to teach) the student would engage and influence students' performance too, which in turn can sometimes affect the predicted results significantly. In this phase drag and boost was contributed to the teacher effect. In teacher effect drag or boost was integrated into three factors: Teacher Experience, Teacher Qualifications, and Teacher PMI. Teacher PMI was teacher's passion, motivation, and innovation & creativity to teach. This teacher passion, motivation, and innovation-creativity to teach were referred to as the teacher effort. Due to limited data (lack of availability of a large data at this time), 5% change or 5% drag or boost effect was ignored i.e., as if the previous and current/latest rate of success was the same. On the other hand, if data of a specific PTLS was too small i.e., under 15 then even a small change could show big differences. When the data in the BRKEVEN chart was large enough then the small changes with the new BRKEVEN would not lead to big fluctuations.

For clarification and understanding of the Hybrid Model, consider the following demo example: In Phase 1, classifiers predict that the student John Doe would succeed, in Phase 2 the success rate (SR) of student John Doe with the specific PTLS preference showed that the rate of success was very high (70% or above). One could infer from the combined analysis of Phase 1 and Phase 2 that there was a high probability that John Doe would pass. However, the actual results showed that the student did not succeed. In fact the actual result showed that five out of 10 students of the similar PTLS did not succeed. The predicted results were contrary to the actual outcome. The rate of success in this specific PTLS over time showed 70%. In reality, at least seven out of 10 students should have passed but only five students passed and the Hybrid Model output showed 50% success rate. In this case, there was a drag of 20%. According to this research, this difference of 20% in actual and predicted results in the rate of success was due to the external factor, which was mostly due to the teacher effect. May be teacher was not trained or not qualified or not passionate or motivated or did not motivate the students enough to succeed. Teacher effect would be calculated by factoring in teacher qualifications, experience, and teacher effort. Teacher effect was the external factor that contributed to that drag when it came to the rate of success.

Teacher effect

To calculate teacher effect, Cobb-Douglas utility function was used (Hadzikadic et al., 2013; Wang & Fu, 2013). Similarly, if the student passed but student was predicted to fail through the classifiers and the SR% for that PT was very low (30% or less) then the difference in the success rate of that specific PT was a boost which was due to the teacher effect. Program was made in such a way that data used in Probabilistic classification could be updated regularly till a large enough data was there for comparisons for the further analysis of the ongoing classes. In C-sharp one could calculate the % of increased chance of success (SR %) with the actual results and boost and drag function was already factored in the program. As explained earlier, boost or drag could be further fragmented into three sections namely teacher qualifications, experience, and effort by using Cobb Douglass function to calculate the impact of different segments within the chosen external factor.

In assigning values for attributes and selecting attributes and scale for the Cobb-Douglas utility function, logic of argumentation theory (Grossi, 2010) and attribution theory (Kelley & Michela, 1980; Weiner, 1972) was used.

For demonstration purposes, let’s look at the following example (Example A) to understand the boost or drag with respect to Cobb-Douglas utility function:

Example A: If the rate of success in RU for personality type (ENTJ) = 70% and the rate of success increased to 90% then the difference would be considered as a boost of 20% because the rate of success was better than before. According to the proposed Hybrid Comparative Model this 20% difference was (mostly) due to the teacher experience, qualifications, and effort (where effort was combination of teacher passion and motivation to teach and how the teacher taught in an innovative and creative way).

The original Cobb-Douglas function, equation is:

$Y = AL^{\beta}K^{\alpha}$

Y = total production (the real value of all goods produced in a year)

L = labor input (the total number of person-hours worked in a year)

K = capital input (the real value of all machinery, equipment, and buildings)

A = total factor productivity

α and β are the output elasticities of capital and labor, respectively. These values are constants determined by available technology.

According to Managing Complexity: Practical Considerations in the development and Application of ABMs to Contemporary Policy Challenges, (from pages 51-52)(Hadzikadic et al., 2013) “The ACSES model uses the Cobb-Douglas utility function, which specifies utility as the product of preferences (i.e., values, interests, motivations) raised to a fractional power. Equation (1) shows the general form of a Cobb-Douglas function P_i denoting the i-th out of n preferences, and wi including the relative weight importance of that preference:

Equation 1 ➔ $U = P_1^{w1}\ P_2^{w2}\ P_3^{w3} \ldots . P_n^{wn}$

The exponents are required to sum 1.

Cobb-Douglas utility function has several desirable features, including simplicity and modularity, which means that it is easily expandable to include additional preferences or values or motivations if they are important for a theory….." "A utility function can incorporate relevant theories and factors which shows some kind of a relationship between variables, and it can instantiate different theories by adjusting parameters" (Hadzikadic et al., 2013). ACSES model has been devised by using Cobb Douglas utility function. Cobb-Douglas production function can be used to show relationship between two or more inputs. "In case of developing an Agent Based Model (ABM) ACSES (Actionable Capability for Social and Economic Systems) model, Cobb-Douglas function model was used because it is easily expandable to include additional preferences or values or motivations if they are important for a theory" (page 52)(Hadzikadic et al., 2013). Specific version of the Cobb-Douglas utility function is given in the form of equation (2) on page 55 in the Managing Complexity: Practical Considerations in the Development and Applications of ABMs to Contemporary Policy Challenges, "$U = (1\text{-}L)^{WL}\ (1\text{-}C)^{WC}(1\text{-}I)^{WI}(1\text{-}E)^{WE}(1\text{-}V)^{WV}(1\text{-}F)^{WF}(1\text{-}R)^{WR}$; where L is loyalty to leader, C is coercion, I is ideology, E is economic welfare, and R is the repression and social influence for defying repression and the weights Wx for motivation x, give the relative importance of the different motivations to the agent and the relative effect they have on U" (Hadzikadic et al., 2013).

On the same pattern, with the help of Cobb-Douglas utility function, teacher effect could also be integrated in the predictive modeling where parts of teacher effect could be calculated by considering teacher experience, qualifications, and effort.

$U = P_1^{w1}\, P_2^{w2}\, P_3^{w3} \ldots . P_n^{wn}$

Y = change of success rate or the difference of the percentage of success

A = Teacher Experience = TE-Exp

B = Teacher Qualifications = TE-Q

C = Teacher Effort (Passion, Motivation, and Innovation & Creativity in teaching) = TE-Eff

A = Teacher experience was important and made a difference but if teacher did not have the knowledge and qualifications then it had limitations. Facility where the research was conducted, some language teachers were teaching there because they were native speakers but did not have proper training or qualifications like pedagogy or not even familiar with some subject matter details like grammar rules. Experience was important and definitely made some positive impact on student learning; however a teacher with a Bachelors or Masters' degree with five years of experience would have a different outlook and approach to teach his/her students as compared to a Bachelors or Masters' degree without any experience. The teacher eventually had to learn the basics of teaching and how to deal with the students in different situations and how to teach students of different personalities and needs differently. Experience helped and was very important but solely experience wouldn't suffice the qualification gap. Based on observation and data, the exponent weight was assumed to be less than the qualifications in this project.

B = Teacher qualifications. If teacher was qualified, knowledgeable, and subject matter expert in his/her subject then the teacher could teach students in a better way as compared to a person who was less qualified. With experience that teacher could become more effective because that teacher had the knowledge and background and a better perspective. For example in language teaching a person with language rules, grammar knowledge, expertise in teaching with multiple intelligence skills, and with personality styles and learning styles attributes knowhow; a qualified teacher could prepare students much better as compared to the person who was not aware of those concepts. Moreover, a teacher with the degree in the same subject was more effective than the teacher whose degree was not directly related to the subject which was being taught.

C = Teacher Effort = Combination of teacher's passion, motivation, and creativity-innovation to teach the students. Analysis of different language teachers indicated that the teachers who came-up with different engaging task-based activities, motivation, and engaged their students had better results but without qualifications and experience, only effort had its' limitations; however in language teaching it was observed that when experienced and qualified teachers, who were passionate, motivated, and creative, engaged and trained their students, got better results as compared to the ones who did not have all these factors. When a teacher was passionate and motivated about teaching then the teacher was most of the time thriving to engage the students with creative and innovative ways to learn and teach.

$$Y = (\text{TE-Experience})^{.25}(\text{TE-Qualifications})^{.35}(\text{TE-Effort})^{.40}$$

It was easier to assign a value and scale to teacher qualifications or experience from the data but it was hard to assign a quantitative value to teacher effort (passion-motivation-innovation/creativity in teaching). The objective of this method was to evaluate teacher effort and able to quantify it with other factors, which could help the decision makers in making decisions.

In case of Hybrid Comparative Predictive Model, multiple of all the attributes was equal to the change in the output. Based on the A = TE-Experience, B = TE-Qualifications, and C = TE-Effort. Weights and elasticity outputs were assumed uniform for all the factors to have consistency for uniform calculations.

$$Y = (\text{TE-Experience})^{.25}(\text{TE-Qualifications})^{.35}(\text{TE-Effort})^{.40}$$

A = TE – Experience and the exponent is .25 (as explained earlier, in light of argumentation and attribution theory, weight of experience (exponent) was much lower than the qualifications and effort)

B = TE – Qualifications and the exponent is .35

C = TE – Effort and the exponent is .45

Y = the difference in %chance of success OR the change in the rate of success (change in SR%).

Note: From drag and boost, absolute values were used in the Cobb-Douglas utility function calculations.

Here, the Cobb-Douglas function was applied to analyze the boost or drag, which was the difference in the rate of success. In this project, boost or drag was due to the external factor, which was the teacher effect. Teacher effect was further synthesized into teacher experience, teacher qualifications, and teacher effort. If the boost (output difference) was 20% as mentioned in Example A, then this 20% was Y (the difference in the change in the rate of success).

In light of the above mentioned explanation with respect to argumentation and attribution theory; for the utility function following values were assigned for the calculations.

Table 3: TE Experience Chart

	Experience	**Face value assigned for calculations**
1	Two years or less $X \leq 2$	.19
2	6 years or less but more than 2 years $2 < X \leq 6$	.39
3	10 years or less but more than 6 years $6 < X \leq 10$	.59
4	15 years or less but more than 10 years $10 < X \leq 15$	.79
5	More than 15 years $X > 15$	.99

TE-EXP = Teacher experience and methodology = A

Table 4: TE Qualifications Chart

	Qualifications	**Face values assigned for calculations**
1	HS/HS+	.19
2	BA/BS	.39
3	BA/BA in language/target language/linguistics/teaching	.59
4	MA/MSc.	.79

5	MA/MSc. or above (PhD). OR MA/MSc. in language/target language/linguistics/teaching.	.99

There would be less fluctuations in the calculations, once the PT Student numbers in the calculations were 15 or above (in most of the statistical analysis, 30 is considered a minimal sample size; in light of the lack of data availability of some personality types; 15 data entries were considered a respectable number till the data of each personality type reached 30). In normal circumstances i.e. data is large enough, then ignore +/- 5% change in success rate.

** Currently, if any of the personality type data was below 15 then ignore +/- 10% change in success rate i.e., 10%. When numbers are below 15 i.e., small change would end up in a big difference that would make the calculation biased.

If at least two out of the three classifiers were predicting success and if the PT success rate of that specific PT was 70% or above ($X \geq 70\%$) then there was a very high probability to succeed i.e., student was highly recommended for that program as compared to those students who had less than two of the three classifiers predicting success and whose PT was 70% or less.

Logically, one first looks into the results of the three classifiers and if two of the classifiers were not predicting success then one looks at the BRKEVEN chart; if the probability of success in the BRKEVEN chart is 70% or above then there was a high probability of that student to succeed as compared to the students whose PT probability was also low.

If student fails, even though all the predictions indicated a success; then external factors such as teacher effect was analyzed with respect to boost and drag. Factors (segments) within the teacher effect (external factor) could be analyzed further with respect to boost and drag in Cobb-Douglas function in detail in future research.

The impact of technology on each segment within the external factor could also be calculated (which was not in this research but for future research). If new technology was introduced such as new smartboard technology, which was helping the teacher in teaching and saving time then an increment of .05% was added to the values assigned in percentage with respect to the scale i.e., TE experience. If smartboard technology was not working or teacher did not know the use of technology then .05% is subtracted from the values assigned in percentage with respect to the teacher experience.

Also, the impact of professional development on the external factor (teacher effort and qualification) can be further analyzed in the future research. Any professional development training in between the scale would have a uniform .0416% increase; keeping in view, at the max, normally, there is one professional development training every month then it could lead to 0.5% increment in the values in percentage. With one professional development training 1% of HS+ would become 1.0416% for scale 1.

C = Motivation/Passion/Teaching with innovative and creative techniques

C was not quantifiable like experience and qualifications. With the help of Cobbs-Douglas utility function C could be calculated

If the boost was 20%

Y = 20% = .20

A = teacher experience = if teacher experience was three years of experience = (according to the TE Experience Chart) = A = 0.39

B = teacher qualifications = if teacher qualifications were BA/BS in the same subject = (according to the TE Qualifications Chart) = B = 0.59

EQUATION for calculations ➔ $Y = (A)^{.25}(B)^{.35}(C)^{.40}$

$0.20 = (0.39)^{.25}(0.59)^{.35}(C)^{.40}$

$0.20 = 0.790*0.831*(C)^{.40}$

$0.20 = 0.6569*(C)^{.40}$

$0.20/0.6569 = (C)^{.40}$

$(C)^{.40} = 0.3044$

C = 0.0511

C = 5%

Out of 20%, 5% was the effort of the teacher that contributed to that 20% of the boost.

Figure 11: PTLS BRKEVEN Snapshot Program Screen 7

In this case boost was due to the teacher motivation, passion, and teacher's innovative and creative teaching during that time. Teacher motivation/creativity may vary day to day with the class, students, and external factors but teacher experience, qualifications, and course length did not change that rapidly. One could measure the teacher qualifications, experience, and course length but motivation could not be measured the same way like experience, qualifications, and course length.

Also to be noted that same teacher might have different results for different classes; here teacher qualifications, experience, and course length hardly changed; however the only factor that changed was the motivation. With this method, an attempt was made to calculate the effort (motivation/passion/creativity/innovation) to teach a particular class for a specific duration.

This could be validated by qualitative methods such as surveys or interviews through the students' feedback to check what the students perceived about teacher's effort (motivation/passion/creativity/innovation to teach) in that specific class. At this time those qualitative surveys were not part of this research project.

Project illustration

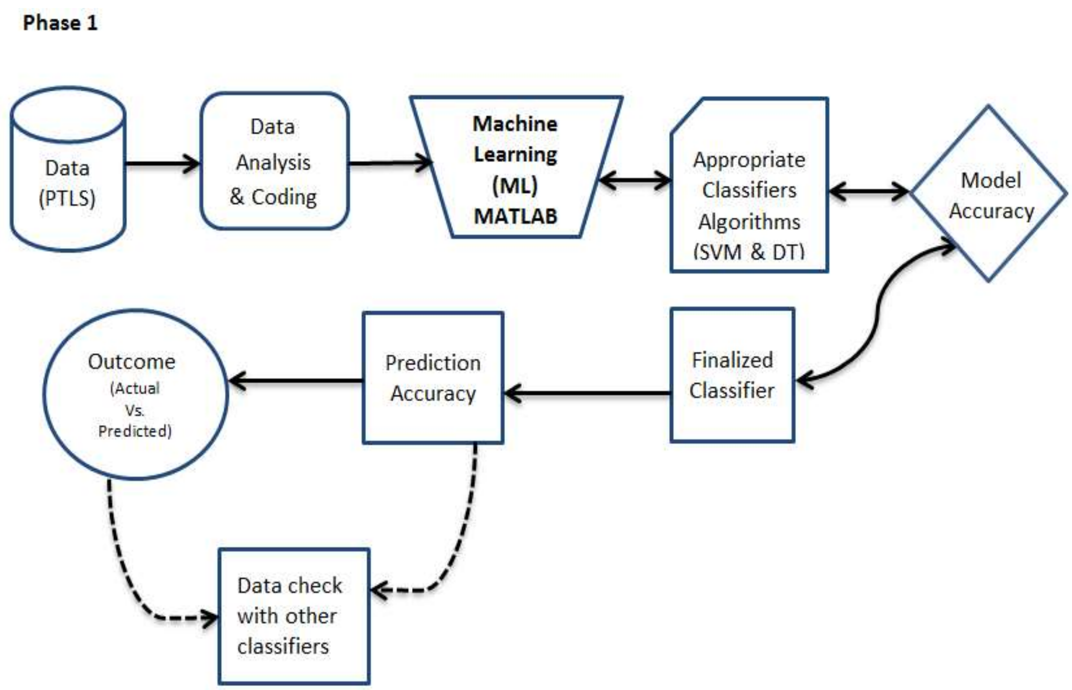


Figure 1: Phase 1

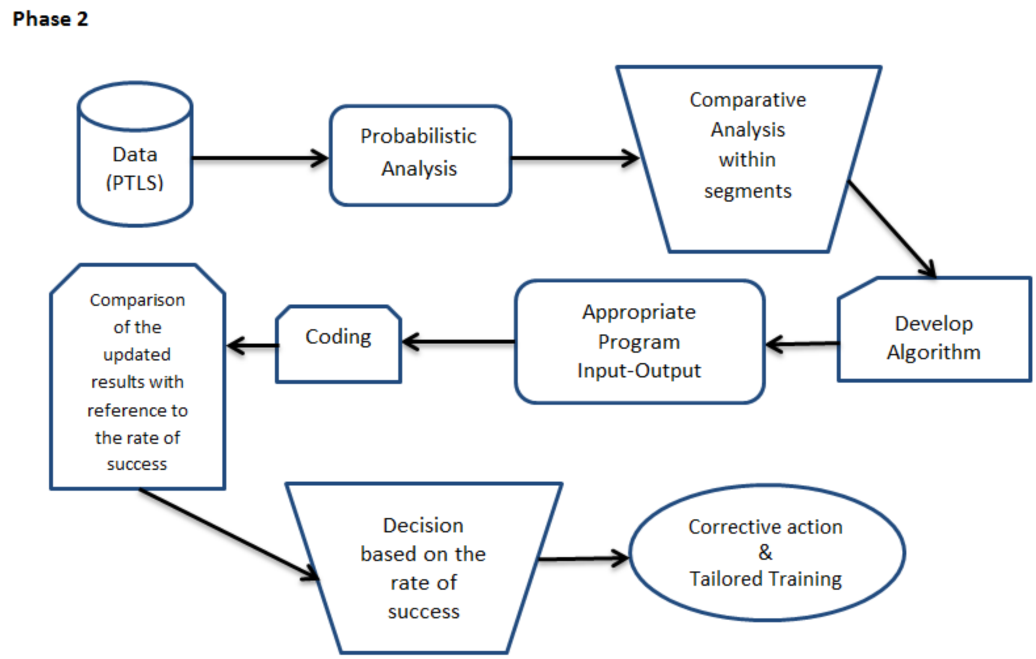


Figure 2: Phase 2

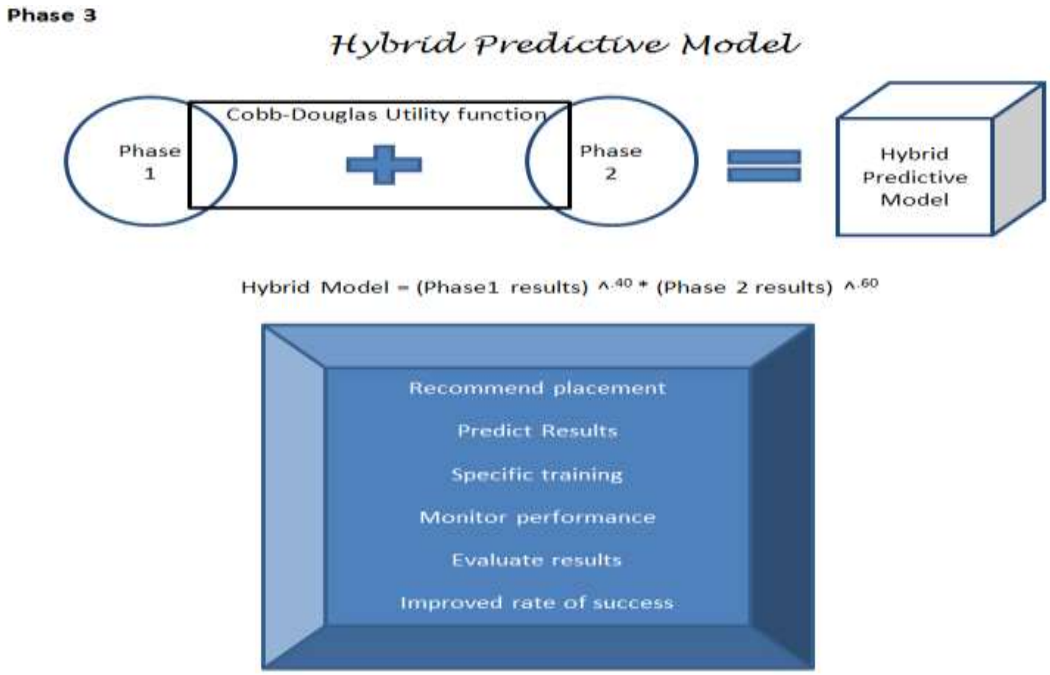
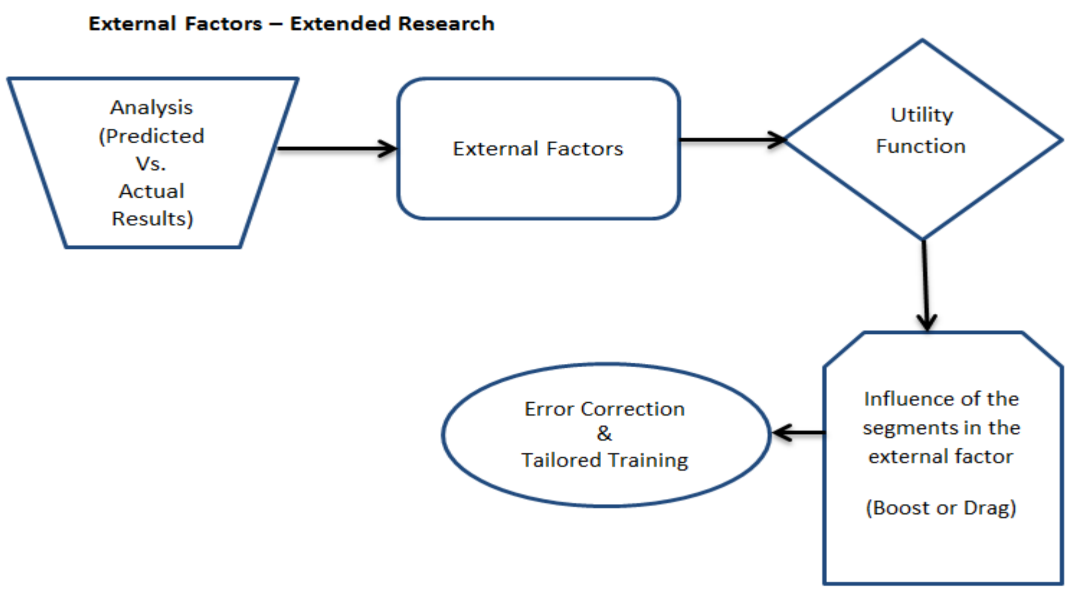


Figure 3: Phase 3



$$Y = (\text{TE-Experience})^{.25}(\text{TE-Qualifications})^{.35}(\text{TE-Effort})^{.40}$$

Figure 4: External Factors – Extended Research

CHAPTER 4: CALCULATIONS & RESULTS

Project background and assumptions

In this project three languages were taken for analysis and predictive modeling. Course length was 24 weeks and classes were conducted for five days a week. Students had six hours of language training every day. Every class varied from four to six students with one instructor assigned to each class. Oral Proficiency Interview (OPI) was the benchmark for the course outcome or final result. Performance of the students was measured in the end by Oral Proficiency Interview (OPI). Here OPI referred to the speaking proficiency. Student proficiency in speaking the specific language was measured through OPI where the scale of measurement was Interagency Language Roundtable (ILR) levels. OPI scores were based on the ILR proficiency. ILR levels varied from ILR level 0 to ILR level 1, ILR level 1+, ILR level 2, ILR level 2+, ILR level 3, ILR level 3+, ILR level 4, ILR level 4+ and ILR level 5. ILR level 5 was the highest level which was equal to the native proficiency level(Herzog, 2007). ILR levels were based on each skill namely speaking, listening, and reading. Here the focus was on speaking skill because data was based on the speaking skill proficiency that was measured by OPI through ILR levels.

Permission to conduct the research and to use the unidentifiable data for this project was taken from the concerned authority on 04 NOV 2015. Data used for this specific project was in the form of numbers and variables only, which constitutes unidentifiable information and according to the official letter an IRB was not required for this kind of a modeling project.

Data used for this project was non-identifiable PTLS data of students in various language programs. Data denoted the student as a generic variable (X1, X2, X3, etc.) if the student was in the Russian language class then the variable would be RU-X1. Teacher information would be denoted by T1, T2, T3, etc. Teacher of the Russian class would be denoted by T1-RU. This problem was taken as a classification problem to predict the results of the students in reference to their PTLS. To identify the languages, face validity in conjunction of scoring function was used for calculations and then face validity was converted further into binary form for some calculations.

There were 16 different personality types namely: ESFJ, ESTJ, ENFJ, ENTJ, ESFP, ESTP, ENFP, ENTP, ISFJ, ISTJ, INFJ, INTJ, INFP, ISFP, INFP, and INTP. Where E = Extrovert/Extravert, I = Introvert, S = Sensor, N = Intuitor (from Intuition), T = Thinker, F = Feeler, J = Judger, P = Perceiver.

Table 5: Personality Type Acronyms

ESFJ	Extravert, Sensor, Feeler, Judger
ESTJ	Extravert, Sensor, Thinker, Judger
ENFJ	Extravert, Intuitor, Feeler, Judger
ENTJ	Extravert, Intuitor, Thinker, Judger
ESFP	Extravert, Sensor, Feeler, Perceiver
ESTP	Extravert, Sensor, Thinker, Perceiver
ENFP	Extravert, Intuitor, Feeler, Perceiver

ENTP	Extravert, Intuitor, Thinker, Perceiver
ISFJ	Introvert, Sensor, Feeler, Judger
ISTJ	Introvert, Sensor, Thinker, Judger
INFJ	Introvert, Intuitor, Feeler, Judger
INTJ	Introvert, Intuitor, Thinker, Judger
INFP	Introvert, Intuitor, Feeler, Perceiver
ISFP	Introvert, Sensor, Feeler, Perceiver
INFP	Introvert, Intuitor, Feeler, Perceiver
INTP	Introvert, Intuitor, Thinker, Perceiver

Personality type data was collected through PT questionnaires by authorized and certified personnel. PT questionnaires were similar to the MBTI type questionnaires. Learning Style data was also provided. Learning style data was collected through (Ehrman and Leaver) E&L Learning Style questionnaires, which is an industry standard in Defense Language Institute (DLI) and various Learning Detachments of the DLI. For this project, initially, only four attributes were considered. Specific learning styles used for this research were Global vs Particular and Inductive vs Deductive. In Global processing a learner preferred to look at the big picture instead of the details of the picture. Learner preferred to focus more on the forest instead of the trees. If a learner's learning style was particular instead of global, learner preferred to focus more on the trees instead of the forest. In language learning an Inductive processor preferred to devise their own rules and formulas from the examples and then applied those rules to learn and enhance their language skills. For example a learner would figure out from examples, the order of verb, subject, and object in making sentences in learning a language. However in Deductive processing the learner preferred to get the formulas and grammar rules first before start practicing and learning the language; though it is a classification problem and specific classifiers have built in feature to check redundancies and correlation but Principal Component Analysis correlation was also considered. Data provided had learning style permutation which looked like G-I or GI, G-D or GD, P-I or PI, and P-D or PD. Where G-I = Global and Inductive, G-D = Global and Deductive, P-I = Particular and Inductive, and P-D = Particular and Deductive.

Tools and limitations

For the first phase, MATLAB was used to generate the language classifiers for each language. Initially, data was cleaned, coded, and then transformed into binary form for convenience and accuracy. When using MATLAB, for calculation purposes, all the personality types and learning styles were given a specific number or a face value. Each specific category had the same number, for example all the ESTJ personality types had number 7. Then the number 7 is transformed into binary form with respect to other personality types; same operation was performed on the learning style, and other variables. Student age and any information about overlapping learning styles or personality types with other learning styles or personality types were also given. DLAB scores were also given but DLAB scores were not available for all the students, which reduced the data to a significant level. However, with the available DLAB data, experiment was run to check if the available DLAB scores can gave any better accuracy in developing a model. Here, the DLAB score data was limited and the primary focus of this research was on PTLS preference data only; so DLAB score data iteration not preferred at this time for the calculations in the Hybrid Model. For any future research, if enough DLAB data was available then this Hybrid Model could be improved and refined to get better model accuracy with the trained data and also better prediction accuracy with the test data. In this research, various iterations were run to see if any specific combination of the variables had better results or more significance. This was shown in Iterations Table 1 in detail. Due to the limited DLAB data, available DLAB scores were only used to check if it had any significant impact in getting higher model accuracy with the trained data or a better prediction accuracy rate with the test data; this might be useful for the future research. Also, within the learning style preferences, some other variables other than "inductive vs deductive and global vs particular" were used to check if adding any other available variables affect the model or prediction accuracy. However, in this project the primary focus was on the 16 personality types and the specific attributes (inductive vs deductive and particular vs global) from the learning styles; other variables primarily used for this research were OPI scores, age, and overlapping PTLS preferences. Additional available variables such as other languages speaking ability at ILR Level 1 or above, education, job title/rank, additional learning style attributes (Field sensitive vs insensitive, Field dependent vs independent, analogue vs digital) were also considered to run different iterations to check any impact of additional variables on model or prediction accuracy. These additional variables could be further researched in future research projects.

Phase 1

In this first phase of the project, six different iterations were run with different combinations of available variables to check any major impact on the model accuracy to develop the appropriate classifier for each language for testing, including CLN, where CLN represented data of all the languages combined (French, Russian, and Arabic). For reference, information of the iterations with various combinations of the variables was as follows:

Note: Naming convention in the iteration was shown for the Arabic data as an example for reference purposes (For example in the first iteration: 1-AD_ML PTLS-DLAB was for Arabic data and abbreviation for the Arabic language was denoted by AD; ML denoted data changed for Machine Learning calculations in MATLAB, PTLS denoted Personality Types & Learning Styles); for French data abbreviation would be FR (1-FR_ML PTLS-DLAB), for Russian data abbreviation would be RU (1-RU_ML PTLS-DLAB), and for CLN data abbreviation would be CLN (1-CLN_ML PTLS-DLAB).

Table 6: Iterations Abbreviations

Iterations	**Name in the Excel Sheet**	**Variables used for calculations**
1	**1-AD_ML PTLS-DLAB** *(AD = Arabic; ML = Machine Learning; PTLS (Personality Types & Learning Styles); -DLAB = without DLAB*	OPI, PT, LS, OI-FV (other information i.e., any overlapping of LS or PT attributes) , and age
2	**2-AD_ML PTLS+DLAB** *(AD = Arabic; ML = Machine Learning; PTLS (Personality Types & Learning Styles); +DLAB = with DLAB*	OPI, PT, LS, OI-FV (Other information i.e., any overlapping of LS or PT attributes), age, and DLAB

3	**3-ADef_ML-DLAB** *(AD = Arabic; ef = extra features; ML = Machine Learning; PTLS (Personality Types & Learning Styles); -DLAB = without DLAB)*	OPI, PT, LS, OI-FV (Other information i.e., any overlapping of LS or PT attributes), OLFV (other languages that student can speak at ILR level 1 or above), ED (education level of the student at the time of entry in the course), R (job title/rank), other LS attributes (field sensitive vs field insensitive, field dependent vs field independent, analog vs digital), and age
4	**4-ADef_ML+DLAB** *(AD = Arabic; ef = extra features; ML = Machine Learning; PTLS (Personality Types & Learning Styles); +DLAB = with DLAB)*	OPI, PT, LS, OI-FV (Other information i.e., any overlapping of LS or PT attributes), OLFV (other languages that student can speak at ILR level 1 or above), ED (education level of the student at the time of entry in the course), R (job title/rank), other LS attributes (field sensitive vs field insensitive, field dependent vs field independent, analog vs digital), age, and DLAB
5	**5-ADef_ML-DLAB-R** *(AD = Arabic; ef = extra features; ML = Machine Learning; PTLS (Personality Types & Learning Styles); -DLAB = without DLAB; -R without job title/rank)*	OPI, PT, LS, OI-FV (Other information i.e., any overlapping of LS or PT attributes), OLFV (other languages that student can speak at ILR level 1 or above), ED (education level of the student at the time of entry in the course), R (job title/rank), other LS attributes (field sensitive vs field insensitive, field dependent vs field independent, analog vs digital), and age
6	**6-ADef_ML+DLAB-R** *(AD = Arabic; ef = extra features; ML = Machine Learning; PTLS (Personality Types & Learning Styles); +DLAB = with DLAB; -R without job title/rank)*	OPI, PT, LS, OI-FV (Other information i.e., any overlapping of LS or PT attributes), OLFV (other languages that student can speak at ILR level 1 or above), ED (education level of the student at the time of entry in the course), R (job title/rank), other LS attributes (field sensitive vs field insensitive, field dependent vs field independent, analog vs digital), age, and DLAB

Cleaned and coded datasheets for Machine Learning process via MATLAB were used, which are available to review in a separate folder for any reference with the following names:

a. Final AD Master Data List (for the Arabic language calculations)

b. Final-FR Master Data List (for the French language calculations)

c. Final-RU Master Data List (for the Russian language calculations)

d. Final-CLN Master Data List (for the combined language data calculations)

Various iterations were run with each of the respective languages data (AD, FR, and RU) to develop an appropriate classifier within each language as mentioned in the Iteration Table 1. Support Vector Machine (SVM) and Decision Tree (DT) classifiers were used to check the model accuracies and to finalize the appropriate classifiers within each language. Test data was run to check the predictions. Predicted results of the test data were then compared with the actual results of the test data to check the prediction accuracy with the specific classifiers. Here to be noted that initially data was trained with the SVM and DT classifiers to finalize a classifier for the specific language, that accuracy was model accuracy i.e., model with better accuracy would be preferred or used for better predictions. Once the model with the highest accuracy was selected, trained classifiers were developed and finalized with respect to each language to check the test data. Then, the test data was run through the classifiers to check the prediction accuracy by checking the actual results.

Each language test data was run through the selected classifiers developed for that language. After running the language data with their respective classifiers, test data for each language was also run with the classifiers of other languages i.e., Arabic test data was run with AD (Arabic) classifier and the Arabic test data was also run with the French and Russian Classifiers. This would predict if a student studying in Arabic language could pass the French or Russian language course too. Here the assumption was that if all or at least two out of three classifiers showed that a student would succeed then the probability for the student to succeed would be very high. A classifier for CLN was also made to check if there was any significant difference in model or prediction accuracy.

Detailed results specifying model accuracy are attached in the Index-A; however the summary of the results of model accuracy and prediction accuracy with the same language classifier and with other classifiers are shown in Arabic Scenarios Table, Russian Scenarios Table, French Scenarios Table, and Combined Language Scenarios Table. The Combined Language Scenario was used primarily to check any significant change in model or prediction accuracy as compared to specific languages. When it was said prediction accuracy, it meant the prediction accuracy with respect to the actual results i.e., how many predictions were true with respect to the actual results. In prediction accuracy test data was used. Model accuracy was derived by training the data with reference to SVM and DT classifiers.

Table 7: Arabic Scenarios

				Model Accuracy	
Iterations			**DATA**	SVM	DT
1	1-AD_ML PTLS-DLAB		228 (Trained Data)	81.1%	79.8%
	1-AD_ML TestD		58 (Test Data)		
	1-Results-AD_ML				
	AD Prediction Accuracy		39, 38	67%	66%
	FR Assumption%	2PaFR-FaAD			
	RU Assumption%	35PaRU-FaAD			
2	2-AD_ML		92	83.7%	81.5%

	PTLS+DLAB		(Trained Data)		
	2-AD_ML TestD		28 (Test Data)		
	2-Results-AD_ML				
	AD Prediction Accuracy		19, 17	68%	61%
	FR Assumption%	6PaFR-FaAD			
	RU Assumption%	16PaRU-FaAD			
3	3-ADef_ML-DLAB		228 (Trained Data)	81.1%	76.8%
	3-ADef_ML TestD		58 (Test Data)		
	3-Results-ADef_ML				
	AD Prediction Accuracy		39, 40	67%	69%
	FR Assumption%	2PaFR-FaAD			
	RU Assumption%	29PaRU-FaAD			
4	4-ADef_ML+DLAB		92 (Trained Data)	83.7%	81.5%
	4-ADef_ML TestD		28 (Test Data)		
	4-Results-ADef_ML				
	AD Prediction Accuracy		19, 19	68%	68%
	FR Assumption%	2PaFR-FaAD			
	RU Assumption%	13PaRU-FaAD			
5	5-ADef_ML-DLAB-R		228 (Trained Data)	81.1%	79.8%
	5-ADef_ML TestD		58 (Test Data)		
	5-Results-ADef_ML				
	AD Prediction Accuracy		39, 40	67%	69%
	FR Assumption%	2PaFR-FaAD			
	RU Assumption%	41PaRU-FaAD			
6	6-ADef_ML+DLAB-R		92 (Trained Data)	83.7%	80.4%
	6-ADef_ML TestD		28 (Test Data)		
	6-Results-ADef_ML				
	AD Prediction Accuracy		19, 18	68%	64%
	FR Assumption%	2PaFR-FaAD			
	RU Assumption%	13PaRU-FaAD			

Interpretation of the highlights and results from the Arabic Scenarios Table were elaborated for clarification and review with reference to Iteration 1. Other iterations in the Arabic Scenarios Table, Russian Scenarios Table, French Scenarios Table, and the Combined Language Scenarios Table followed the same pattern.

In iteration 1, there were 228 data entries. As mentioned in Table 6, the variables considered for calculations in this iteration were OPI, PT, LS, OI-FV (other information i.e., any overlapping of LS or PT attributes), and age. Support Vector Machine (SVM) classifier gave 81.1% model accuracy and Decision Tree (DT) classifier gave 79.8% model accuracy. Difference of 1.3% (this was not a very significant difference especially with this limited data). Available test data for this iteration was 58 data entries. AD SVM classifier had 67% accurate prediction rate and AD DT classifier had 66% prediction rate (at this time, with this limited data, 1% difference was not significant). Prediction rate accuracy referred to the prediction accuracy with respect to the actual results.

By looking at all the 6 iterations/scenarios in Arabic, the model accuracy with SVM classifier was slightly higher than DT classifier. However the difference with respect to the data size was negligible i.e., it varied from 1.3% to 4.3%. Enough DLAB scores were not readily available at this time; however, DLAB feature increased the model accuracy to 1.7% in DT classifier and 2.6% in SVM classifier.

Extra features (not including DLAB) did not impact the results in SVM classifier when it came to improved model accuracy. With the primary data, the SVM model accuracy was 81.1% and with extra features (excluding DLAB) the SVM model accuracy was 81.1%, by adding DLAB the accuracy increased to 83.7%.

Extra features (not including DLAB) other than the primary features/variables, the impact of additional features on the model accuracy was negligible or had slightly an adverse effect in DT classifier, especially when it came to enhance model accuracy; without DLAB and extra features, the model accuracy in DT model was 79.8%; by using extra features the model accuracy dropped to 76.8%, which was a 3% drop. However by using DLAB the model accuracy reached 81.5%, which was 1.7% higher accuracy as compared to "in the absence of DLAB scores"; however due to the limited DLAB scores and extreme difficulty in getting DLAB data (limited availability), and not much difference in the model accuracy with or without extra features, iteration 1 classifier was preferred in the AD to be used at this time. Moreover, the primary focus in this research was on PTLS; additional features were available for the given data to check if there was any significant change in the model or prediction accuracy. With respect to this data there was not a significant impact on the model or prediction accuracy. Hence, all the iterations were shown with the result snapshots but only first iteration findings would be discussed and recommendations would be based on the first iteration in concluding this research at this time. Once enough data was available where at least each personality had a minimum of 30 entries then further research could be conducted by making this research as a base model.

AD test data was checked with the French Classifier which showed that two of the students studying Arabic language that did not succeed in the Arabic language might succeed in the French language. With Russian Classifier it showed that 35 of the students that would fail in Arabic language might succeed in the Russian language.

Table 8: Russian Scenarios

				Model Accuracy	
Iterations			**DATA**	SVM	DT
1	1-RU_ML PTLS-DLAB		178 (Trained Data)	66.3%	60.1%
	1-RU_ML TestD		36 (Test Data)		
	1-Results-RU_ML				
	RU Prediction Accuracy		30, 31	83%	86%
	FR Assumption%	0PaFR-FaRU			
	AD Assumption%	1PaAD-FaRU			
2	2-RU_ML PTLS+DLAB		131 (Trained Data)	71.8%	70.2%
	2-RU_ML TestD		3 (Test Data)		
	2-Results-RU_ML				
**	**RU Prediction Accuracy**		2, 2	67%	67%
	FR Assumption%	0PaFR-FaRU			
	AD Assumption%	0PaAD-FaRU			
3	3-RUef_ML-DLAB		178 (Trained Data)	68%	68.5%
	3-RUef_ML TestD		36 (Test Data)		
	3-Results-RUef_ML				
	RU Prediction Accuracy		26, 24	72%	67%
	FR Assumption%	0PaFR-FaRU			
	AD Assumption%	0PaAD-FaRU			
4	4-RUef_ML+DLAB		131 (Trained Data)	65.6%	67.9%
	4-RUef_ML TestD		3 (Test Data)		
	4-Results-RUef_ML				
**	**RU Prediction Accuracy**		2, 2	67%	67%
	FR Assumption%	0PaFR-FaRU			
	AD Assumption%	0PaAD-FaRU			
5	5-RUef_ML-DLAB-R		178 (Trained Data)	66.3%	62.9%
	5-RUef_ML TestD		36 (Test Data)		

	5-Results-RUef_ML				
	RU Prediction Accuracy		30, 25	83%	69%
	FR Assumption%	0PaFR-FaRU			
	AD Assumption%	0PaAD-FaRU			
6	6-RUef_ML+DLAB-R		131 (Trained Data)	68.7%	71%
	6-RUef_ML TestD		3 (Test Data)		
	6-Results-RUef_ML				
**	**RU Prediction Accuracy**		2, 2	67%	67%
	FR Assumption%	0PaFR-FaRU			
	AD Assumption%	0PaAD-FaRU			

In the Russian Scenarios Table, in iteration 1, there were 178 data entries. Support Vector Machine (SVM) classifier gave 66.3% model accuracy and Decision Tree (DT) classifier gave 60.1% model accuracy; difference of 6.2% between SVM and DT model accuracy. Available test data for this iteration was 36 data entries. AD SVM classifier had 83% accurate prediction rate and AD DT classifier had 86% prediction rate (at this time, with this limited data, 3% difference was not significant). However, it is to be noted that the Model accuracy in DT classifier was 60.1% but the prediction accuracy was 86%, and the prediction accuracy was the highest with Iteration 1 with DT as compared to any other iterations. In light of this result with respect to the available data and to keep consistency with other languages, iteration 1 DT classifier was considered in RU for now. Here to be noted that the actual results in RU reveal better prediction accuracy rate with reference to the model accuracy rate. It can be due to factors such as teachers, which can be analyzed further if the updated data reveals this pattern consistently.

RU test data was also checked with the French Classifier and showed that there would be no impact. However, one of the students studying Russian language who would not succeed in the Russian language might succeed in the Arabic language.

Table 9: French Scenarios

				Model Accuracy	
Iterations			**DATA**	**SVM**	**DT**
1	1-FR_ML PTLS-DLAB		246 (Trained Data)	80.9%	78.9%
	1-FR_ML TestD		56 (Test Data)		
	1-Results-FR_ML				
	FR Prediction Accuracy		38, 37	68%	66%
	RU Assumption%	32PaRU-FaFR			
	AD Assumption%	0PaAD-FaFR			
2	2-FR_ML PTLS+DLAB		92 (Trained Data)	84.8%	80.4%
	2-FR_ML TestD		28 (Test Data)		
	2-Results-FR_ML				
	FR Prediction Accuracy		19, 20	68%	71%
	RU Assumption%	13PaRU-FaFR			
	AD Assumption%	0PaAD-FaFR			
3	3-FRef_ML-DLAB		246 (Trained Data)	80.9%	78.5%
	3-FRef_ML TestD		56 (Test Data)		
	3-Results-FRef_ML				
	FR Prediction Accuracy		38, 37	68%	66%
	RU Assumption%	24PaRU-FaFR			
	AD Assumption%	0PaAD-FaFR			
4	4-FRef_ML+DLAB		92 (Trained Data)	84.8%	82.6%
	4-FRef_ML TestD		28 (Test Data)		
	4-Results-FRef_ML				

	FR Prediction Accuracy		19, 17	68%	61%
	RU Assumption%	11PaRU-FaFR			
	AD Assumption%	0PaAD-FaFR			
5	5-FRef_ML-DLAB-R		246 (Trained Data)	80.9%	79.7%
	5-FRef_ML TestD		56 (Test Data)		
	5-Results-FRef_ML				
	FR Prediction Accuracy		38, 37	68%	66%
	RU Assumption%	24PaRU-FaFR			
	AD Assumption%	0PaRU-FaFR			
6	6-FRef_ML+DLAB-R		92 (Trained Data)	84.8%	76.1%
	6-FRef_ML TestD		28 (Test Data)		
	6-Results-FRef_ML				
	FR Prediction Accuracy		19, 17	67%	61%
	RU Assumption%	11PaRU-FaFR			
	AD Assumption%	0PaAD-FaFR			

In the French Scenarios Table, in iteration 1, there were 246 data entries. Support Vector Machine (SVM) classifier gave 80.9% model accuracy and Decision Tree (DT) classifier gave 78.9% model accuracy; difference of 2% between SVM and DT model accuracy, which is not that significant with respect to this data. Available test data for this iteration is 56 data entries. FR SVM classifier had 68% accurate prediction rate and FR DT classifier had 66% prediction rate (at this time, with this limited data, 2% difference is not significant). However, it is to be noted that the prediction accuracy with SVM in all the iterations is consistent i.e., 68%, this infers that extra features did not have any impact on the prediction accuracy. In light of the prediction accuracy rate result with respect to the available data, iteration 1 SVM classifier is considered the best in FR.

FR test data was also checked with the Russian Classifier which shows that 32 of the students that were not successful in FR would be successful in the Russian language and none of the students studying French language would succeed in the Arabic language.

Table 10: Combined Language Scenarios

				Model Accuracy	
Iterations			**DATA**	**SVM**	**DT**
1	1-CLN_ML PTLS-DLAB		652 (Trained Data)	68.7 %	67.5 %
	1-CLN_ML TestD		150 (Test Data)		
	1-Results-CLN_ML				
	CLN Prediction Accuracy		80, 82	53%	55%
	RU Assumption%	74PaRU-FaCLN			
	AD Assumption%	4PaAD-FaCLN			
	FR Assumption %	3PaFR-FaCLN			
2	2-CLN_ML PTLS+DLAB		316 (Trained Data)	66.3 %	61.0 %
	2-CLN_ML TestD		59 (Test Data)		
	2-Results-CLN_ML				
	CLN Prediction Accuracy		41, 39	69.4 %	66%
	RU Assumption%	23PaRU-FaCLN			
	AD Assumption%	3PaAD-FaCLN			
	FR Assumptions%	12PaFR-FaCLN			
3	3-CLNef_ML-DLAB		652 (Trained Data)	68.4 %	69.6 %
	3-CLNef_ML TestD		150 (Test Data)		
	3-Results-CLNef_ML				
	CLN Prediction Accuracy		90, 37	60%	58%
	RU Assumption%	60PaRU-FaCLN			
	AD Assumption%	0PaAD-FaCLN			
	FR Assumptions %	3PaFR-FaCLN			
4	4-CLNef_ML+DLAB		316 (Trained Data)	66%	65.1 %
	4-CLNef_ML TestD		59 (Test Data)		
	4-Results-CLNef_ML				

	CLN Prediction Accuracy		41, 36	69%	61%
	RU Assumption%	19PaRU-FaCLN			
	AD Assumption%	4PaAD-FaCLN			
	FR Assumptions%	7PaFR-FaCLN			
5	5-CLNef_ML-DLAB-R		652 (Trained Data)	68.3 %	67.6 %
	5-CLNef_ML TestD		150 (Test Data)		
	5-Results-CLNef_ML				
	CLN Prediction Accuracy		92, 92	61.3 %	61.3 %
	RU Assumption%	60PaRU-FaCLN			
	AD Assumption%	0PaAD-FaCLN			
	FR Assumptions%	3PaFR-FaCLN			
6	6-CLNef_ML+DLAB-R		316 (Trained Data)	68.6 %	60.6 %
	6-CLNef_ML TestD		59 (Test Data)		
	6-Results-CLNef_ML				
	CLN Prediction Accuracy		42, 39	71.1 %	66.1 %
	RU Assumption%	19PaRU-FaCLN			
	AD Assumption%	4PaAD-FaCLN			
	FR Assumptions%	7PaFR-FaCLN			

In the Combined Language Table, in iteration 1, there were 652 data entries. Support Vector Machine (SVM) classifier gave 68.7% model accuracy and Decision Tree (DT) classifier gave 67.5% model accuracy; difference of 1.2% between SVM and DT model accuracy, which is not that significant with respect to this data. Available test data for this iteration is 150 data entries. CLN SVM classifier had 53% accurate prediction rate and CLN DT classifier had 55% prediction rate (at this time, with this limited data, 2% difference is not significant). However, CLN (combined language scenario of Arabic, French, and Russian) was developed to check, if there would be any major impact with reference to improved model accuracy or an improved prediction rate. CLN results showed that there was no significant impact on the model accuracy but the prediction/actual accuracy rate dropped significantly with respect to all the languages scenarios prediction/actual accuracy. In Arabic, the SVM and DT prediction accuracy was 67% and 66% as compared to CLN prediction accuracy of 53% and 55%; in Russian the SVM and DT prediction accuracy was 83% and 86% as compared to CLN prediction accuracy of 53% and 55%; and in French the SVM and DT prediction accuracy was 68% and 66% as compared to CLN prediction accuracy of 53% and 55%.

In light of the results with respect to the available data, it can safely be concluded that it was better to use the language data separately for predictive analysis as compared to combining different language data together for predictive analysis.

Notes:

FaAD = did not succeed in Arabic (ILR level 2 or above)
FaFR = did not succeed in French (ILR level 2 or above)
FaRU = did not succeed in Russian (ILR level 2 or above)
FaCLN = did not succeed in CLN (ILR level 2 or above)
PaAD = would succeed in Arabic in getting ILR level 2 or above
PaFR = would succeed in French in getting ILR level 2 or above
PaRU = would succeed in Russian in getting ILR level 2 or above
PaCLN = would succeed in CLN in getting ILR level 2 or above

How could this predictive analysis become better i.e., prediction accuracy was rarely 100% accurate when it came to the actual results. Could this predictive analysis be improved by combining something with it? How the decision makers and trainers could get a better insight in helping their students with different personality types preferences for an improved success rate. In the first phase of the project predictive analysis was done by using SVM and DT classifiers and data was taken as one segment i.e., all the personality types and learning style data within a language was combined; it was more of an overall vertical analysis. Predictive analysis by classification through Machine Learning via SVM and DT was one way to look at the data and helped the decision makers made decisions to assist or place the students accordingly.

Phase 2

Could data be looked at a different way where one could look at the success rate of each PTLS to assist the students or to assist the decision makers in placing the students accordingly, so that the success rate of each PTLS could be improved somehow?

In the second phase, focus was on probabilistic classification where each PTLS data was looked at in a different way. Then these different insights could be looked at as a whole to assist the students for better decision making to recommend student placement?

As mentioned earlier, this specific data was not evenly divided among different personality types; ESTJ and ISTJ personality preferences dominated the other PT preferences and with the current data. However, in the next phase each PT type was looked at as a separate entity to calculate the success rate of each PT by calculating their probability with respect to the available data within each segments and then comparing the probability of success rate of different personalities within each language and also with the other languages. It was devised in such a way that each segment was looked at as a separate entity to check the success rate of a particular PTLS. To make the system more efficient, whenever any new data was available or as soon as the new data was added to the original or base data, each PTLS segment was updated that provided the updated success rate of each PT with LS preferences. Then each segment success rate (SR) was analyzed to check which PT had the higher success rate as compared to the other PT. Moreover each PT was further looked at by going further in the LS preferences success rate within each PT. In this way, if there was a situation where there were five students and three out of those five students were recommended to be placed in the

Arabic language but the decision makers could not know which three students should be placed in which order. Firstly, through SVM and DT classifiers, it was checked if these students were predicted to succeed. If the SVM or DT classifier easily sorted out the three students that would succeed then one part of the problem was resolved in the first phase. However, if SVM and DT classifier showed that all five would not succeed but the management had to place the best three in the Arabic language, then in phase two, one could check, which PTLS had the higher rate of success. If one of the students was ESFJ, another one ISFJ, one ENTP and two others as ESTJ, then by looking at the probability chart, one could comfortably infer from the data that their preference of placement could be: ISFJ, ESTJ, ESFJ, and ENTP.

Table 11: Arabic PTLS Probability Distribution Chart

LN1-ARABIC									
					LS				
					G-I	G-D	P-I	P-D	
PT#	**PT**	**ST**	**Success**	**SR%**	**1**	**2**	**3**	**4**	
1	ENFJ	1	0	0%					
2	ENFP	2	1	50%	1				
3	ENTJ	10	6	60%	2	3		1	
4	ENTP	8	0	0%					
5	ESFJ	20	3	15%	2	1			
6	ESFP	5	1	20%				1	
7	ESTJ	125	30	24%	10	16		4	
8	ESTP	15	2	13%	2				
9	INFJ	1	0	0%					
10	INFP	3	1	33%				1	
11	INTJ	7	2	29%	2				
12	INTP	8	3	38%	1	2			
13	ISFJ	10	3	30%		2		1	
14	ISFP	3	2	67%	1	1			
15	ISTJ	57	6	11%	2	1		3	
16	ISTP	11	2	18%	1	1			
	Total	**286**	**62**		**24**	**27**	**0**	**11**	**62**
	Total		**22%**		**39%**	**44%**	**0%**	**18%**	

With this table, one could look at the success rate in a different way. ENTP had 0% success rate. However, ISFP had 67% success rate. If there was no other choice and management had to place an ENTP in the Arabic program, then the trainers could assist the ENTP preference student by adapting the learning habits which made ISFP successful or pair the ENTPs with the PT preference students who were extremely successful in that particular language; in this way ENTP could adapt the ways which might be helpful for the ENTPs to succeed in this program.

LN1-Arabic updated data revealed the success rate of different PTs in the following order:

ISFP, ENTJ, ENFP, INTP, INFP, ISFJ, INTJ, ESTJ, ESFP, ISTP, ESFJ, ESTP, ISTJ, ENFJ, INFJ, ENTP. Here ENTP, ENFJ, and INFJ have zero rate of success however, ENFJ and INFJ had only one student data but ENTP had eight students that did not succeed.

Table 12: French PTLS Probability Distribution Chart (LN3-French)

LN3-FRENCH

					LS				
					G-I	G-D	P-I	P-D	
PT#	**PT**	**ST**	**Success**	**SR%**	**1**	**2**	**3**	**4**	
1	ENFJ	4	2	50%	1	1			
2	ENFP	3	2	67%	1	1			
3	ENTJ	4	1	25%		1			
4	ENTP	1	0	0%					
5	ESFJ	17	3	18%		2		1	
6	ESFP	4	2	50%	2				
7	ESTJ	157	36	23%	18	11	2	5	
8	ESTP	11	3	27%		1	1	1	
9	INFJ	3	0	0%					
10	INFP	1	0	0%					
11	INTJ	2	1	50%				1	
12	INTP	0	0	0%					
13	ISFJ	7	1	14%	1	1			
14	ISFP	1	0	0%					
15	ISTJ	74	12	16%	3	6		2	
16	ISTP	13	1	8%	1				
	Total	**302**	**64**		**27**	**24**	**3**	**10**	**64**
	Total		**21%**		**42%**	**38%**	**5%**	**16%**	

LN3-French updated data revealed the success rate of different PTs in the following order:

ENFP, ESFP, ENFJ, INTJ, ESTP, ENTJ, ESTJ, ESFJ, ISTJ, ISFJ, ISTP, INTP (no data), ISFP, INFP, ENTP, INFJ.

ESFP and ENFJ and INTJ had 50% success rate however ESFP had more students with the same LS preference of GI as compared to ENFJ with one student with GI preference and one with GD preference. INTP, ISFP, INFP, ENTP, and INFJ had zero percent success rates but INTP had no student data and ISFP, INFP, and ENTP had one student, however INFJ had three students.

Table 13: Russian PTLS Probability Distribution Chart (LN5-Russian)

LN5-RUSSIAN									
					LS				
					G-I	G-D	P-I	P-D	
PT#	**PT**	**ST**	**Success**	**SR%**	**1**	**2**	**3**	**4**	
1	ENFJ	3	3	100%	2			1	
2	ENFP	4	3	75%	2	1			
3	ENTJ	9	7	78%	4	1	2		
4	ENTP	8	6	75%	5			1	
5	ESFJ	9	5	56%		4		1	
6	ESFP	3	3	100%	1	2			
7	ESTJ	84	56	67%	17	23	3	13	
8	ESTP	8	6	75%	2	3	1		
9	INFJ	1	1	100%			1		
10	INFP	3	3	100%	1	1		1	
11	INTJ	7	2	29%	1		1		
12	INTP	1	1	100%				1	
13	ISFJ	9	6	67%	3	1	1	1	
14	ISFP	1	1	100%		1			
15	ISTJ	55	38	69%	7	24		7	
16	ISTP	9	7	78%	5	2			
	Total	**214**	**148**		**50**	**63**	**9**	**26**	**148**
	Total		**69%**		**34%**	**43%**	**6%**	**18%**	

LN5-Russian updated data revealed the success rate of different PTs in the following order:

ESFP, ENFJ, INFP, ISFP, INTP, INFJ, ISTP, ENTJ, ENTP, ESTP, ENFP, ISTJ, ESTJ, ISFJ, ESFJ, INTJ.

Here to be noted that ESFP, ENFJ, INFP, ISFP, INTP, INFJ had 100% success rate. However, further analysis showed that ESFP, INFP, and ENFJ had three students as compared to INFJ and ISFP. Also, ESFP had GD and GI preference which was 34% and 43% respectively both PTs were global learner preferences. Similarly INTP was preferred over INFJ because the data revealed that PI learning preference had far lower success rate as compared to the PD or GI and GD learners. Both the ESTJ and ISFJ had 67% success rate but ESTJ was given priority over ISFJ because in ESTJ 84 PT preference student data was available, however in ISFJ only 9 PT preference student data was available. In light of the Argumentation theory (Grossi, 2010) and available data, it was safer to give ESTJ precedence over ISFJ.

Table 14: Personality Type Preference Chart

PT Preference			
	AD-LN1	**RU-LN5**	**FR-LN3**
1	ISFP	ESFP	ENFP
2	ENTJ	ENFJ	ESFP
3	ENFP	INFP	ENFJ
4	INTP	ISFP	INTJ
5	INFP	INTP	ESTP
6	ISFJ	INFJ	ENTJ
7	INTJ	ISTP	ESTJ
8	ESTJ	ENTJ	ESFJ
9	ESFP	ENTP	ISTJ
10	ISTP	ESTP	ISFJ
11	ESFJ	ENFP	ISTP
12	ESTP	ISTJ	INTP
13	ISTJ	ESTJ	ISFP
14	ENFJ	ISFJ	INFP
15	INFJ	ESFJ	ENTP
16	ENTP	INTJ	INFJ

By analyzing the table above, it clearly showed that in Arabic the most successful PT was ISFP, in Russian it was ESFP and in French it was ENFP. It also showed that ISFP was at the fourth position and in French ISFP was in the 13th position. This PT success rate clearly showed which PT preference was more successful in which language. By looking at this chart the decision makers could easily place the students with that PT preference in the specific languages. If there was no other option but let's say an ENTP had to be placed in the Arabic language program then the trainers and teachers would be aware of that student facing challenges and could assist the student in advance (ahead of time) by teaching the PT preferences of successful Personality Types, i.e., whose success rate was extremely well in that language.

Here an overall CLN rate of success was also given for comparison. CLN could also be used in conjunction with the AD, FR, and RU.

Table 15: Combined Language PTLS Probability Distribution Chart

CLN = L1+L3+L5									
					LS				
					G-I	G-D	P-I	P-D	
PT#	PT	ST	Success	SR%	1	2	3	4	
1	ENFJ	7	4	57%	3	0	0	1	
2	ENFP	6	4	67%	2	2	0	0	
3	ENTJ	18	11	61%	6	2	2	1	
4	ENTP	11	4	36%	4	0	0	0	
5	ESFJ	36	8	22%	1	5	0	2	
6	ESFP	10	6	60%	3	2	0	1	
7	ESTJ	305	95	31%	33	38	5	19	
8	ESTP	28	10	36%	4	3	2	1	
9	INFJ	4	0	0%	0	0	0	0	
10	INFP	6	3	50%	1	1	0	1	
11	INTJ	11	3	27%	2	0	1	0	
12	INTP	7	4	57%	1	2	0	1	
13	ISFJ	18	7	39%	2	3	1	1	
14	ISFP	3	2	67%	0	2	0	0	
15	ISTJ	157	40	25%	10	21	0	9	
16	ISTP	25	7	28%	5	2	0	0	
	Total	652	208		77	83	11	37	208
	Total		32%		37%	40%	5%	18%	

CLN (AD, RU, and FR) data revealed the success rate of different PTs in the following order:

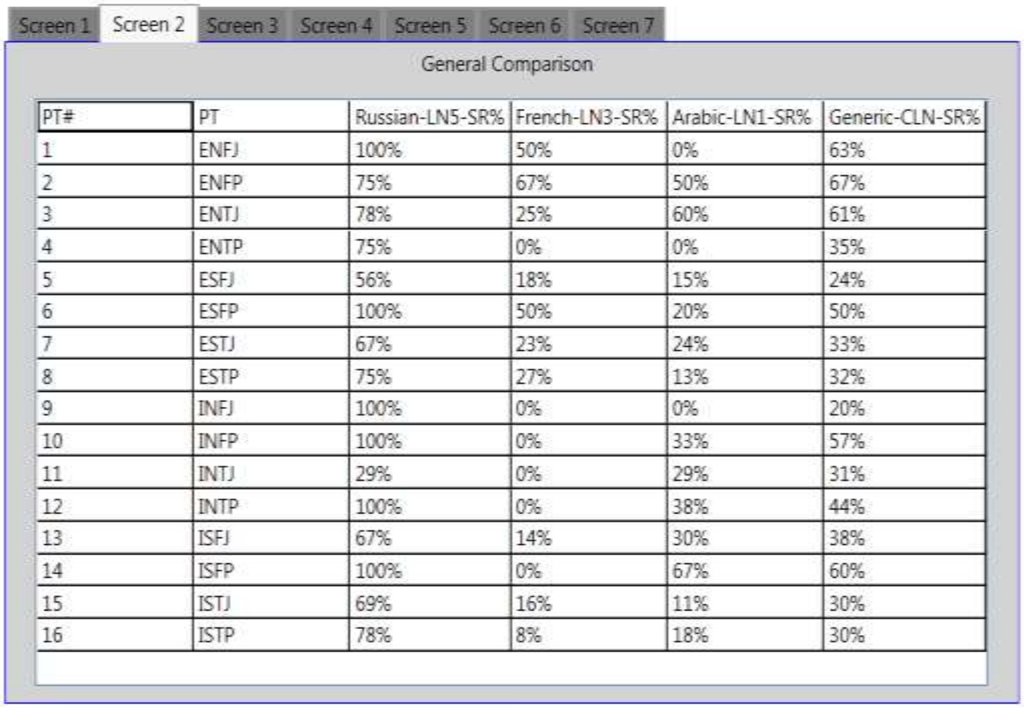
ENFP, ISFP, ENTJ, ESFP, ENFJ, INTP, INFP, ISFJ, ESTP, ENTP, ESTJ, ISTP, INTJ, ISTJ, ESFJ, INFJ.

With CLN, Table-PTC2 was shown for comparative purposes.

Table 16: Personality Type Preference Chart 2

PT Preference				
	AD-LN1	**RU-LN5**	**FR-LN3**	**CLN**
1	ISFP	ESFP	ENFP	ENFP
2	ENTJ	ENFJ	ESFP	ISFP
3	ENFP	INFP	ENFJ	ENTJ
4	INTP	ISFP	INTJ	ESFP
5	INFP	INTP	ESTP	ENFJ
6	ISFJ	INFJ	ENTJ	INTP
7	INTJ	ISTP	ESTJ	INFP
8	ESTJ	ENTJ	ESFJ	ISFJ
9	ESFP	ENTP	ISTJ	ESTP
10	ISTP	ESTP	ISFJ	ENTP
11	ESFJ	ENFP	ISTP	ESTJ
12	ESTP	ISTJ	INTP	ISTP
13	ISTJ	ESTJ	ISFP	INTJ
14	ENFJ	ISFJ	INFP	ISTJ
15	INFJ	ESFJ	ENTP	ESFJ
16	ENTP	INTJ	INFJ	INFJ

In the methodology section, the PTLS BRKEVEN Snapshot was mentioned. In PTLS BRKEVEN Snapshot, AD-LN1, FR-LN3, and RU-LN5 base data was transformed into algorithm steps to be coded and integrated through C# program in PTLS BRKEVEN Snapshot. General comparison without PT preference from high to low success rate preference was also shown in screen 2 for the decision makers or trainers.



PT#	PT	Russian-LN5-SR%	French-LN3-SR%	Arabic-LN1-SR%	Generic-CLN-SR%
1	ENFJ	100%	50%	0%	63%
2	ENFP	75%	67%	50%	67%
3	ENTJ	78%	25%	60%	61%
4	ENTP	75%	0%	0%	35%
5	ESFJ	56%	18%	15%	24%
6	ESFP	100%	50%	20%	50%
7	ESTJ	67%	23%	24%	33%
8	ESTP	75%	27%	13%	32%
9	INFJ	100%	0%	0%	20%
10	INFP	100%	0%	33%	57%
11	INTJ	29%	0%	29%	31%
12	INTP	100%	0%	38%	44%
13	ISFJ	67%	14%	30%	38%
14	ISFP	100%	0%	67%	60%
15	ISTJ	69%	16%	11%	30%
16	ISTP	78%	8%	18%	30%

Figure 6: PTLS BRKEVEN Snapshot Program Screen 2

Individual PT preferences with respect to the language could be analyzed through PTLS BRKEVEN Snapshot as mentioned in Chapter 3, the methodology section.

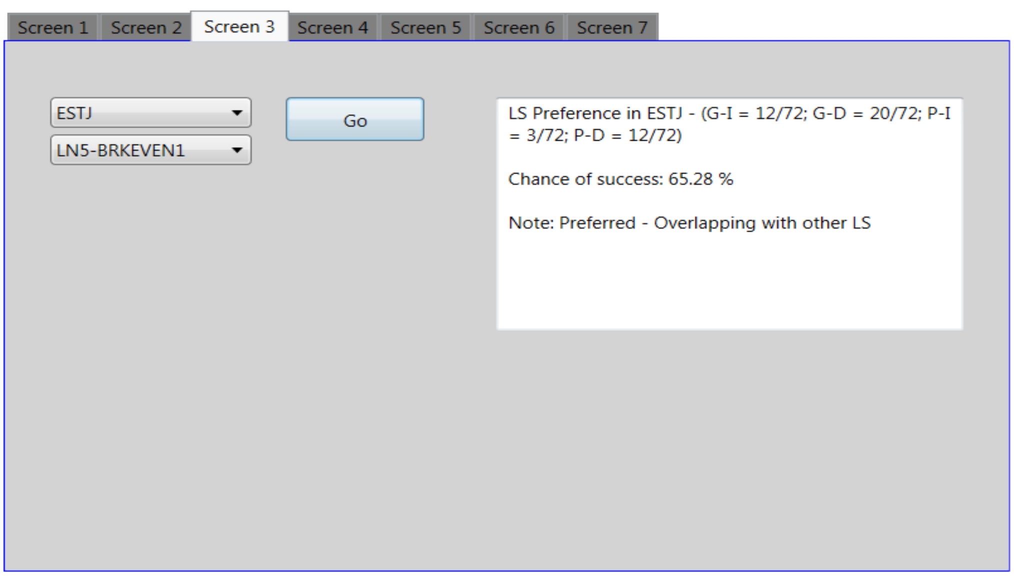


Figure 7: PTLS BRKEVEN Snapshot Program Screen 3

Screen 4 showed the comparison in a much convenient way of the same PT in different languages or different PTs in same language as shown below:

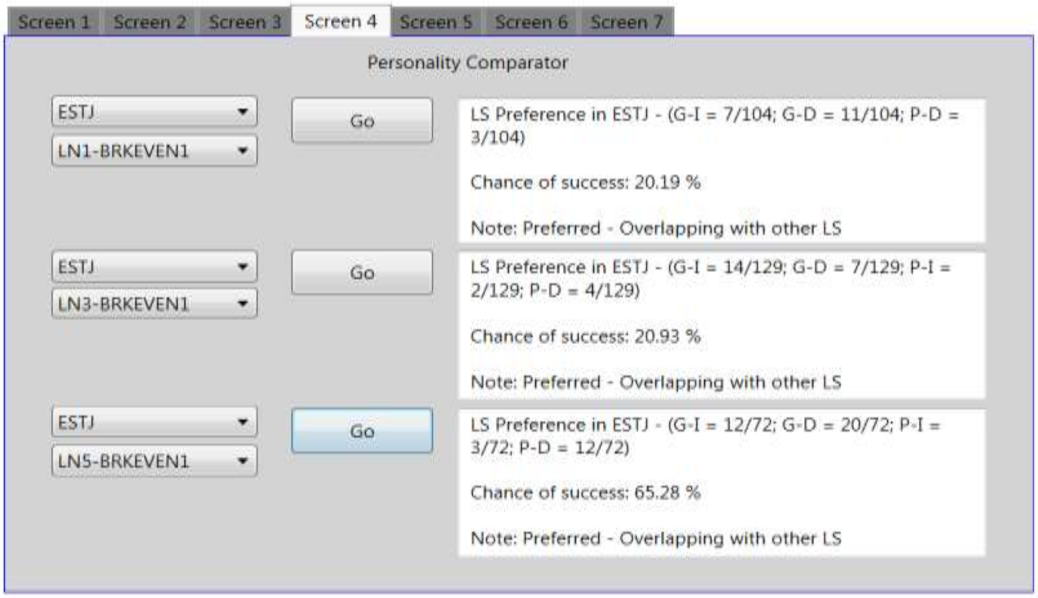


Figure 8a: PTLS BRKEVEN Snapshot Program Screen 4

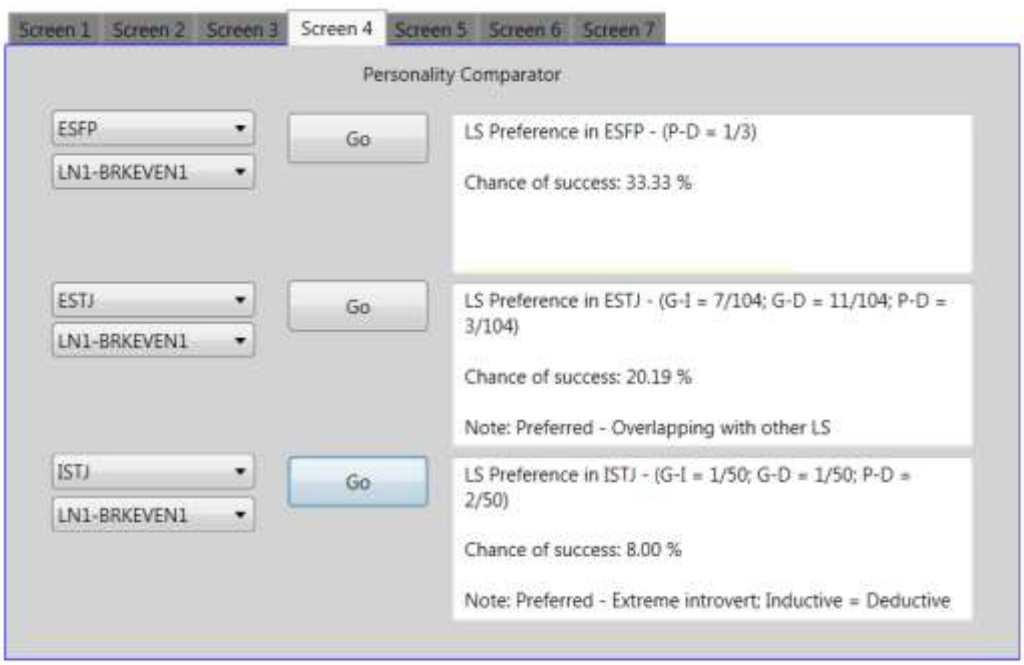


Figure 8b: PTLS BRKEVEN Snapshot Program Screen 4

Hybrid Model

In Hybrid Model Phase 1 and Phase 2, results were combined through Cobb-Douglas utility function. In AD, success rate of ENTJ in Phase 1 was 100%, in Phase 2 the ENTJ success rate was 60%. In Hybrid Model, Phase 1 and Phase 2 were combined through Cobb-Douglas utility function: Hybrid Model Output = (Phase 1 results) $^{\wedge.40}$ * (Phase 2 results) $^{\wedge.60}$

Hybrid Model result showed 74% success rate or in other words 7 out of 10 students would be successful. Actual results showed 67% success rate, which interpreted as seven out of ten students were successful or two out of three students successful.

In AD, in the test data ESFJ had five students and the Hybrid Model accuracy was 24%, actual success rate was 20%, P1 predictive success rate was 50% and P2 predictive success rate was 15%. Hybrid Model success rate was much accurate with reference to the actual success rate. Overall, when the numbers were sufficient the Hybrid Model predictive success rate was closer and accurate to the actual success rate. In ISTJ there were seven students; P1 success rate was 36%, P2 success rate was 11%, Hybrid Model Success rate was 28% and the actual success rate was 29%.

In Russian, there were 12 ESTJ students, P1 phase showed 100% success rate, P2 showed 67% success rate, Hybrid Model success rate was 79% and the actual success rate was 75%. In French, P1 success rate of ESTJ was 45%, P2 success rate was 23%, however Hybrid Model success rate was 30%, and the actual success rate was 32%.

Screen 6 in PTLS Snapshot Program combined the results from Phase 1 and Phase 2 through Cobb-Douglas function. The PTLS Snapshot program also calculated the difference in actual and predicted results caused by the external factors. PTLS

Snapshot program further looked into the difference in actual and predicted results caused by external factors such as teacher effect; teacher effect was introduced in this model and PTLS Snapshot program but the in-depth teacher effect was for the future research.

In Phase 2, the success rate of each personality type was checked in each language. Initially, the success rate of each of the PT within specific language was calculated with the base data. Then whenever new data was added, the success rate of the base data was compared with the success rate of the updated data. If the success rate of the updated data had improved i.e., better than the base rate; such as in LN5-PT, ENTP, the updated success rate was 75% as compared to the base data success rate of 67%. This increase of success rate was considered boost here. When the success rate went down such as in ENTJ in LN5 the success rate went down from 88% to 78%; this decline of 10% in success rate was named as drag. With boost or drag, one could calculate the impact on success rate with respect to the updated data. The boost showed that this specific PT was doing well and proper measures had been taken with reference to the external factor, which contributed to that boost. Drag showed that the specific PT type was not doing well or an external factor contributed to that decline. As mentioned in the methodology section, in this research the major external factor for this boost and drag was contributed to the teacher effect. Currently the data was limited and the boost and drag were shown in the picture with respect to the available data. For some PT types data was less than 15 data entries. Once enough data was available i.e., at least 30 data entries for each PT type then the boost and drag in the success rate would be more normalized.

Both base and updated files were uploaded in the PTLS BRKEVEN Snapshot program and it automatically showed the drag or boost of a specific PT with the updated or current data. In the above picture PT13 – ISFJ in LN5 (RU) showed the boost of 10%. The PTLS BRKEVEN Snapshot program showed it as follows in screen 5.

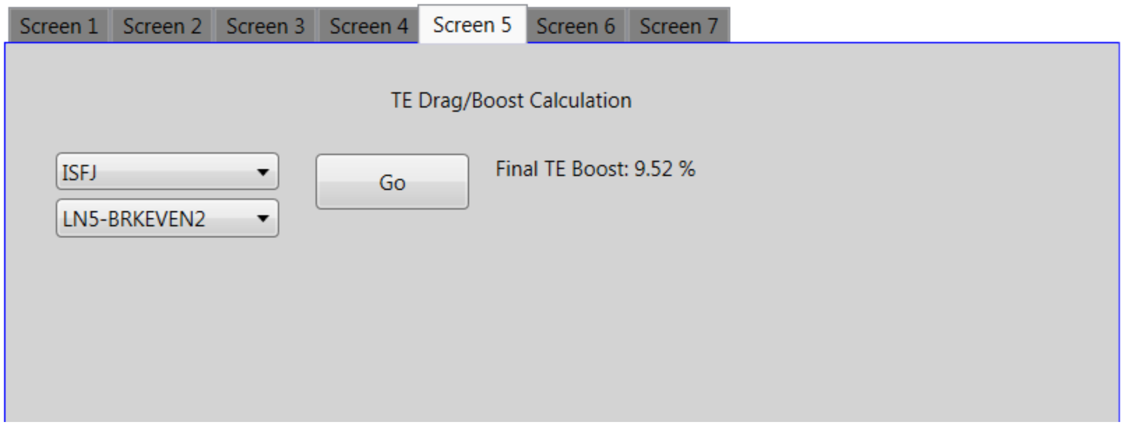


Figure 10a: PTLS BRKEVEN Snapshot Program Screen 5

Screen 5 calculated the difference in success rate of a specific PT within a specific language. If the new success rate (updated/current) was better than the previous success rate (base/point of reference) then that was a boost. If the new success rate (updated/current) was worse than the previous success rate (base/point of reference) then that was a drag. If there was no difference between the BRKEVEN1 SR% and BRKEVEN2 SR% i.e., if the success rate was same then there was no further processing.

Corrective action

Advantage of the boost or drag after the comparison of predictive and actual results could provide insight to give immediate corrective action as soon as possible to help students succeed. This boost and drag could be calculated by Cobb-Douglas function, where the following equation was introduced:

$$Y = (\text{TE-Experience})^{.25}(\text{TE-Qualifications})^{.35}(\text{TE-Effort})^{.40}$$

The following tables were already provided in the methodology section. For convenience, the tables are also provided below:

Table 3: TE Experience Chart

TE Experience Chart		
	Experience	**Face value assigned for calculations**
1	Two years or less $X \leq 2$	.19
2	6 years or less but more than 2 years $2 < X \leq 6$	.39
3	10 years or less but more than 6 years $6 < X \leq 10$	.59
4	15 years or less but more than 10 years $10 < X \leq 15$	.79
5	More than 15 years $X > 15$	.99

Table 4: TE Qualifications Chart

TE Qualifications Chart		
	Qualifications	**Face values assigned for calculations**
1	HS/HS+	.19
2	BA/BS	.39
3	BA/BA in language/target language/linguistics/teaching	.59
4	MA/MSc.	.79
5	MA/MSc. or above (PhD). OR MA/MSc. in language/target language/linguistics/teaching.	.99

In the screen 7 of the PTLS BRKEVEN snapshot program once the values assigned from the chart are entered as input the Teacher Effort is calculated as shown in the following snapshot, which was continuation of the previous snapshot.

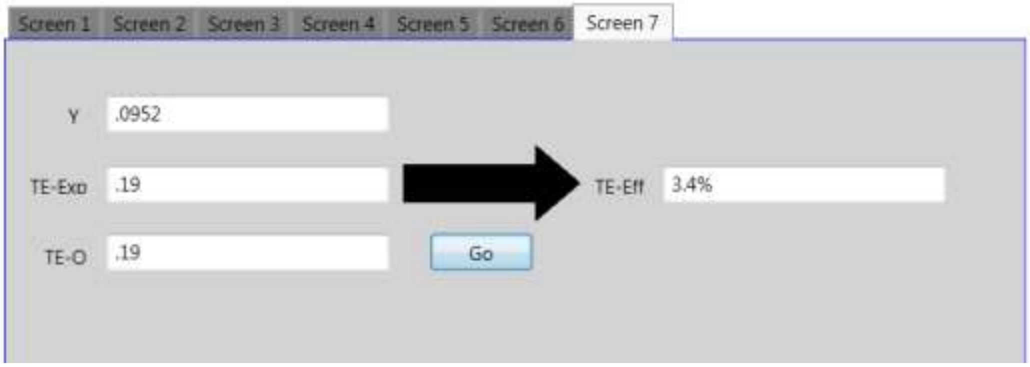


Figure 11: PTLS BRKEVEN Snapshot Program Screen 7

When enough data for each personality type would be available then the effort levels could be determined; but an effort chart was also introduced in this research for future reference.

Table 17: TE Effort Chart

	Effort	Result Interpretation
1	None to very low	0 to 20% of the Boost/Drag $X \leq 20$
2	Very low to medium low	More than 20%, up to 50% $20 < X \leq 50$
3	Medium low to High	More than 50%, up to 75% $50 < X \leq 75\%$
4	High to medium High	More than 75%, up to 90% $75 < X \leq 90\%$
5	Medium High to Exceptional	More than 90% $X > 90$

The above mentioned snapshot showed that the updated success rate (SR) was higher than the base success rate and the boost was 9.52% in ISFJ personality type in LN5 (Russian language). If teacher experience was 0.19 which according to the TE-Experience chart was two years or less and if the TE-Q (Teacher Qualification) was 0.19, which according to the TE-Q scale was HS, and then the TE-Eff (Teacher effort) was 3.4% of the 9.52% boost. However, the boost & drag with respect to the Cobb-Douglas function was introduced in this research but this research was limited to predictive modeling that gave the decision makers a better insight to enable the decision makers and trainers to predict in a much improved way to recommend student placement with respect to their PTLS preferences and to assist students to succeed in their language programs with respect to their specific PTLS preferences. Cobb-Douglas function with respect to calculating the external factors and analyzing the sub segments within that external factor was for the future research; current available data was not enough for all 16 personality types, if there were only three students in a personality type who were successful the success rate was 100% and in the updated data if the two new students did not succeed; the updated success a rate would be 60%, which would give 40% drag. Just with two students such a big drag or boost would not provide accurate results at this time. However, once the data would be enough then with gradual changes in the success rate, one could monitor and evaluate properly.

The findings in this chapter with reference to literature review and methodology served as the basis for a summary of this research project and the interpretation of the data, conclusion, and recommendations for future studies in Chapter 5.

CHAPTER 5: CONCLUSIONS & RECOMMENDATIONS

The purpose of this chapter was to determine the findings, recommendations, and implications of PTLS preferences with respect to different programs with reference to predictive modeling i.e., how to predict which PTLS preference would perform better in a specific program when it came to student placement and training.

The literature review suggested that students with different personality types and learning styles preferences perceive and learn things differently (Boeree, 1997; ME Ehrman et al., 2002; Leaver, 1997; Lightbown & Spada, 2006). In this research, focus was on the predictive analysis of students' success with reference to their PTLS preferences. Research was conducted with the available data to check if students with particular PTLS preferences perform better and if one could predict which PTLS preference was more successful in a specific language program. Moreover, literature review also suggested, if a particular PTLS preference was not performing well in a specific program of study, then how could one assist those students to succeed (Leaver, 1997; Leaver et al., 2005).

Sample for this study comprised of the unidentifiable data from students in three language programs Arabic, French, and Russian languages. The following research questions were studied through Support Vector Machine and Decision Tree classifiers via Machine Learning and computing through probability classification:

1. Do different personality types and learning styles preferences impact the results differently in various programs of study?

2. Can the decision makers improve their decision-making process by combining different ways of predictive analysis to recommend students of different PTLS preferences for placement with an end-result of a better rate of success?
3. How can the predictive analysis become effective with limited data?
4. Predictive analysis rarely provides100% accurate results. Is there any method which can determine the external factors or the impact of external factors with reference to predictive modeling?

The conceptual framework and theory that guided this study involved computing through Machine Learning and probability classification.

Summary of results

Phase 1 Results

Primary focus in this phase was PTLS data which included personality type and learning style preferences, overlapping personality type and learning styles preferences with other personality types and learning styles, and age with respect to the OPI (Oral Proficiency Interview) scores. There were 16 different personality types and in learning style preferences only two of the features were emphasized i.e., particular vs global learner and inductive vs deductive learner. Some additional features were also available from the data, which were used to determine if there was any significant impact on the results by using additional features. Six different iterations were run with different data combinations to check any significant impact on the results with reference to model vs prediction accuracy; results and explanation of the results within each language was given with Table 2A-AD, 2B-RU, and 2C-FR in Chapter 4. Detailed results are given in Index-A in the end. Scenario with all the combined data of all the languages was also run to check model and prediction accuracies to check if that might have any significant impact on the model or prediction rate accuracy. Prediction rate accuracy refers to the prediction accuracy with respect to the actual results. Each of the language test data was also run with the other two languages to check if any of the students that would not succeed in the current language program succeed in any of the other languages.

Arabic Results

In Arabic language, 228 data entries were available to formulate an SVM and DT classifier to check the predictions from 58 test data entries. SVM and DT classifiers had 81.1% and 79.8% model accuracy; prediction accuracy was 67% with the SVM and 66% with the DT classifier. From the test data, two of the students who did not succeed in Arabic would succeed in French and 35 of the students who would not succeed in Arabic would succeed in Russian.

Russian Results

In Russian language, 178 data entries were available to formulate an SVM and DT classifier to check the predictions from 36 test data entries. SVM and DT classifiers had 66.3% and 60.1% model accuracy; prediction accuracy was 83% with the SVM and 86% with the DT classifier. From the test data, one of the students who did not succeed in Russian would succeed in the Arabic language program.

French Results

In French language, 246 data entries were available to formulate an SVM and DT classifier to check the predictions from 56 test data entries. SVM and DT classifiers had 80.9% and 78.9% model accuracy; prediction accuracy was 68% with the SVM and 66% with the DT classifier. From the test data, one of the students who did not succeed in

French would succeed in Arabic and 32 of the students who would not succeed in French would succeed in Russian.

CLN Results

In CLN, 652 data entries were available to formulate an SVM and DT classifier to check the predictions from 150 test data entries. SVM and DT classifiers had 68.7% and 67.5% model accuracy; prediction accuracy was 53% with the SVM and 55% with the DT classifier.

Phase 2 Results

In Phase 2 part one results, the outcome of the comparative analysis of the rate of success (SR %) of each PTLS within each language was done and then the success rate of each PTLS was compared with all the languages.

Table 18: LN-PT Rate of Success Chart

PT	RU-LN5-SR%	RU-LN3-SR%	FR-LN1-SR%
ENFJ	100%	33%	0%
ENFP	75%	100%	0%
ENTJ	88%	0%	57%
ENTP	67%	0%	0%
ESFJ	50%	15%	13%
ESFP	100%	50%	33%
ESTJ	65%	21%	20%
ESTP	71%	30%	18%
INFJ	0%	0%	0%
INFP	100%	0%	0%
INTJ	33%	0%	20%
INTP	100%	0%	50%
ISFJ	57%	20%	33%
ISFP	100%	0%	100%
ISTJ	61%	14%	8%
ISTP	80%	10%	20%

After calculating the rate of success% of each PT in every language, base data revealed the most successful personality type from high to low preference within each language as follows:

Table 14: Personality Type Preference Chart

PT Preference	AD-1	RU-1	FR-1
	AD-LN1	RU-LN5	FR-LN3
1	ISFP	ESFP	ENFP
2	ENTJ	ENFJ	ESFP
3	ENFP	INFP	ENFJ
4	INTP	ISFP	ESTP
5	INFP	INTP	ESTJ
6	ISFJ	ENTJ	ISFJ
7	INTJ	ISTP	ESFJ
8	ESTJ	ENFP	ISTJ
9	ESFP	ESTP	ISTP
10	ISTP	ENTP	INTJ
11	ESFJ	ESTJ	INTP
12	ESTP	ISTJ	ENTP
13	ISTJ	ISFJ	ISFP
14	ENFJ	ESFJ	INFP
15	INFJ	INTJ	ENTJ
16	ENTP	INFJ	INFJ

Here, AD-1, RU-1, and FR-1 indicated the base score and AD-2, RU-2, FR-2 would be for the updated scores i.e., whenever the new data was added in the base data. Table PTC 1 revealed that in Arabic language, PT with the highest rate of success was ISFP; ISFP was at fourth place in Russian and at thirteenth place in French.

In Russian language, PT with the highest rate of success was ESFP; where ESFP was at ninth place in Arabic and second place in French.

In French language, PT with the highest rate of success was ENFP; where ENFP was at the third place in Arabic and eighth place in Russian.

Table 16: Personality Type Preference Chart 2

PT Preference						
	AD-2	**AD-1**	**RU-2**	**RU-1**	**FR-2**	**FR-1**
1	ISFP	ISFP	ESFP	ESFP	ENFP	ENFP
2	ENTJ	ENTJ	ENFJ	ENFJ	ESFP	ESFP
3	INTP	ENFP	INFP	INFP	ENFJ	ENFJ
4	ISFJ	INTP	ISFP	ISFP	INTJ	ESTP
5	ESFP	INFP	INTP	INTP	ESTP	ESTJ
6	ESTJ	ISFJ	INFJ	ENTJ	ENTJ	ISFJ
7	ISTP	INTJ	ISTP	ISTP	ESTJ	ESFJ
8	INTJ	ESTJ	ENTJ	ENFP	ESFJ	ISTJ
9	ESTP	ESFP	ENTP	ESTP	ISTJ	ISTP
10	ESFJ	ISTP	ESTP	ENTP	ISFJ	INTJ
11	ISTJ	ESFJ	ENFP	ESTJ	ISTP	INTP
12	ENFJ	ESTP	ISTJ	ISTJ	INTP	ENTP
13	ENFP	ISTJ	ESTJ	ISFJ	ISFP	ISFP
14	INFJ	ENFJ	ISFJ	ESFJ	INFP	INFP
15	INFP	INFJ	ESFJ	INTJ	ENTP	ENTJ
16	ENTP	ENTP	INTJ	INFJ	INFJ	INFJ

In this phase, whenever the new data was available, the new data was added with the base data to make it the updated data. In this way one could check the consistency in the success rate of each PT within that specific environment.

Table PTC 2 revealed that in Arabic language, with the updated data, PT with the highest rate of success was still ISFP; ISFP was also the most successful PT in the base data. In the updated data ISFP was still at the fourth place in Russian and at the thirteenth place in French.

In Russian language, in the updated data, PT with the highest rate of success was still ESFP; where ESFP was at fifth place in the Arabic and it was ninth place in Arabic with the base data, and ESFP was still in the second place in French. ESFP fluctuation in Arabic was contributed to the small numbers i.e., in the base data one out of three passed, which was 33% success rate but in the updated data only one passed out of five, which was 20% success rate within ESFP; when the numbers were below ten even a small fluctuation could show a big difference in the success rate. Whereas when the ESTJ data was looked at, the fluctuation was not as drastic as ESFP. In ESTJ in the base data there were 104 students and 21 of those students were successful, which was 20% success rate; however in the updated data there were 125 students and 30 were successful, which was 24% success rate and the difference was 4%. As more data would be added and updated, success rate overview would become clearer for each language and the big fluctuations would start normalizing (stabilizing).

In French language, in the updated data, PT with the highest rate of success was ENFP, which was the same as in the base data. ENFP was third place in Arabic but it was 13th place in Arabic in the base data; however it was to be noted that this was due to the low student numbers in that personality type; a small fluctuation could bring a big change. In Russian ENFP success rate was at the eighth place and in the base data it was at the eleventh place.

My contribution

My novel contribution is a Hybrid Model, where the aggregate of two phases through Cobb-Douglas utility function provides a better accuracy rate as compared to an individual phase. Additionally, I devised a method using the Cobb-Douglas utility function that calculates external factor i.e., teacher effort affecting results. Hybrid Model provides better insight for the decision makers to recommend or place students in different programs. Hybrid Model predicted the success rate, which was very close to the actual success rate. There were some limitations due to the limited available data in some personality types. When the test data was more than 10 data entries then the predicted results were much accurate to the actual results. Cobb-Douglas function was used to combine the inputs from Phase 1 and Phase 2 to show the aggregate.

Table 19a: Hybrid Model Key Results

Hybrid Model Predictions		
AD - Arabic language Phase1 – ESFJ success rate – 50% Phase2 - ESFJ success rate - 15% Hybrid Model Prediction – 24% Actual success rate 20%	RU - Russian language Phase1 – ESTJ success rate – 100% Phase2 - ESTJ success rate – 67% Hybrid Model Prediction – 79% Actual success rate 75%	FR - French language Phase1 – ESTJ success rate – 45% Phase2 - ESTJ success rate - 23% Hybrid Model Prediction - 30% Actual success rate 32%
Hybrid Model Placement Preference		
AD - ISFP then ENTJ	RU - ESFP then ENFJ	FR - ENFP then ESFP

Table 19b: Hybrid Model Results Overview

	PT	Test Data-ST#	P1Comb-Success	P2Success	HYBRID MODEL	ActualSuccess	#Students actually passed/test-data
ARABIC	ENTJ	3	100%	60%	74%	67%	2
	ESFJ	5	50%	15%	24%	20%	1
	ESTJ	21	43%	24%	30%	29%	6
	ISFJ	4	50%	30%	37%	25%	1
	ISFP	2	25%	67%	45%	50%	1
	ISTJ	7	36%	11%	18%	29%	2
RUSSIAN	ESTJ	12	100%	67%	79%	75%	9
	ISTJ	11	100%	69%	80%	82%	9
	ISTP	4	100%	78%	86%	75%	3
FRENCH	ESFJ	4	50%	18%	27%	25%	1
	ESTJ	28	45%	23%	30%	32%	9
	ISFJ	2	25%	14%	18%	50%	1
	ISTJ	11	41%	16%	23%	27%	3

A new program was developed in C sharp to analyze the data for recommendations to assist the decision makers in placing or recommending students to different programs based on their PTLS preferences. Prediction capability for success or failure was enhanced by using the aggregate of two different methods, which was not possible by using only one method before. Moreover, by using Argumentation Theory

(Grossi, 2010), and Attribution Theory (Kelley & Michela, 1980), through Cobb-Douglas function external factor such as teacher effect was factored in and the concept of drag and boost was introduced to give corrective action to enhance the success rate by keeping in view teacher experience, qualifications, and effort.

$Y = (\text{TE-Experience})^{.25}(\text{TE-Qualifications})^{.35}(\text{TE-Effort})^{.40}$

Enough data would be needed for the future research to continue validating the teacher effect when calculating boost or drag; where boost was the increased rate of success and drag was the decreased rate of success. Though the teacher effort has to be further analyzed in the future research but some instances were checked with the limited data to validate the teacher effect and effort as shown in Table 20.

Table 20: Teacher Effort Overview

T-19	**T-3**
Six ESTJ students	Four ESTJ students
TE Exp-4yrs	TE Exp-14yrs
TE Q - High School + T-TRG	TE Q- BA in Arabic Language + T-TRG
All students passed – two got ILR 2	Two students failed – none got ILR2
(.39Exp/.19Q) .2% of 4% (Double Effort)	(.79Exp/.59Q) .1% of 4%
In case 20% boost = 13.8%	In case of 20% boost = 3.3%
T-12	**T-9**
Three ESTJ students	Four ESTJ students
TE Exp-4yrs	TE Exp-6yrs
TE Q - BA in Arabic language	TE Q - BA Basic Teaching
All students received ILR level 1+	All student received ILR level 1+
.39Exp/.59Q	.39Exp/.59Q
If 20% drag then 5.1% lack of effort	If 20% drag then 5.1% lack of effort
If 20% boost then 5.1% effort	If 20% boost then 5.1% effort

Significant findings and future applications

Q: Do different personality types and learning style preferences impact the results differently in various programs of study?

A: Results from the P1 and P2 indicate that different personality types and learning style preferences impact the results differently in various programs of study. The highest success rate of PT in Arabic is of ISFP, in Russian it is ESFP, and in French it is ENFP.

Q: Can the decision makers improve their decision-making process by combining different ways of predictive analysis to recommend students of different PTLS preferences for placement with an end-result of a better rate of success?

A: Yes, the decision makers can improve their decision-making process by combining different ways of predictive analysis to recommend students of different PTLS preferences for placement by keeping in mind the objective of a better rate of success. If management has to decide to place six different students of different personality types in the Arabic language among the three languages (FR, RU, AD) and students are the following PT:

ESTJ, ENFP, ISTJ, ESTJ, ESFP, ENFJ.

By looking at the data and the Table-PTC1, it is preferred to recommend the students in the following order for the Arabic language:

ESFP, ESTJ, ESTJ, ISTJ, ENFJ, ENFP.

If management has to decide to place two of the students in French, two in Russian, and two in Arabic; by looking at the Table-PTC1 student allocation priority wise will be as follows:

French: ENFP, ISTJ

Russian: ESFP, ENFJ,

Arabic: ESTJ, ESTJ

Q: How can the predictive analysis become effective with limited data?

A: Predictive analysis can become more efficient by regular updating of the current data. Predictive analysis can become more effective with even limited data in the Hybrid Model. Initially, the data was checked with the SVM and DT classifiers for success. However, when the data was checked with the PTLS Snapshot Program, then the increased rate of success of a PTLS could be determined by comparative analysis with the success rates of other PTLS success rates. PTLS with a higher success rate would get precedence over the PTLS with lower success rate. Whenever the new data would be available, the base data (the previous data) would be updated and the updated success rate of each PTLS would be realigned. In this way management had a better insight in placing a PTLS student where the rate of success of that PTLS student was higher as compared to other PTLS.

Q: Predictive analysis rarely provides100% accurate results; is there any method which can determine the external factors or the impact of external factors with reference to predictive modeling?

A: In Phase 2 drag and boost concept was also introduced. By comparing the success rate of each personality type in the base data (the previous data) with the updated data one could compute the difference of success rate in each personality type. If the previous success rate was higher than the current success rate then it would be a drag (decline in the performance). If the previous success rate was lower than the current success rate i.e., there was an improvement in the success rate then it was named as a boost (improved performance). If the rate of success of a particular PTLS preference was the same, i.e., if there was no change, then there wasn’t any drag or boost.

Currently, in this project data used in modeling was of 24 months. PTLS Snapshot program had the current and previous (base) data. Difference in the previous and current data showed a boost or drag (if there was any). Success rates of a specific PTLS would show consistency over time; however, if a drastic change in the success rate was revealed from the difference in the previous and newly updated data then this drastic change was contributed due to some external factor. Here the external factor was mainly contributed to the teachers/trainers because in the specific environment of this research, teachers' influence students' success the most.

Teacher affect was further integrated into three factors: Teacher Experience, Teacher Qualifications, and Teacher Effort. Teacher effort indicated teachers' passion, motivation, and creativity/innovation in teaching. Cobb-Douglas utility function was introduced and also programmed in the PTLS Snapshot program for calculations. Based on the logic of argumentation and attribution theories as discussed in the literature and methodology section of this research, values for teacher experience and qualifications are assigned. Drag and boost was then compared with reference to teacher experience and qualifications to calculate teacher effort. Previous results are compared with the current results, where the difference between the current and previous results shows the success rate. Where drag is considered a decline in the success rate and boost is considered as an increase in the success rate. Drag and boost findings are highlighted in the Table 20 below.

Table 21: Drag and Boost Findings

PT #	PT	LN5-Base	LN5-Upda	Impact/Diff.	LN3-Base	LN3-Update	Impact/Dif	LN1-Base	LN1-Updat	Impact/Di
1	ENFJ	100%	100%	0%	33%	50%	17%	0%	0%	0%
2	ENFP	75%	75%	0%	100%	67%	-33%	0%	50%	50%
3	ENTJ	88%	78%	-10%	0%	25%	25%	57%	60%	3%
4	ENTP	67%	75%	8%	0%	0%	0%	0%	0%	0%
5	ESFJ	50%	56%	6%	15%	18%	2%	13%	15%	2%
6	ESFP	100%	100%	0%	50%	50%	0%	33%	20%	-13%
7	ESTJ	65%	67%	1%	21%	23%	2%	20%	24%	4%
8	ESTP	71%	75%	4%	30%	27%	-3%	18%	13%	-5%
9	INFJ	0%	100%	100%	0%	0%	0%	0%	0%	0%
10	INFP	100%	100%	0%	0%	0%	0%	0%	33%	33%
11	INTJ	33%	29%	-5%	0%	0%	0%	20%	29%	9%
12	INTP	100%	100%	0%	0%	0%	0%	50%	38%	-13%
13	ISFJ	57%	67%	10%	20%	14%	-6%	33%	30%	-3%
14	ISFP	100%	100%	0%	0%	0%	0%	100%	67%	-33%
15	ISTJ	61%	69%	8%	14%	16%	2%	8%	11%	3%
16	ISTP	80%	78%	-2%	10%	8%	-2%	20%	18%	-2%
ST #		178	214	36	246	302	56	228	286	58
SR		66%	69%	3%	19%	21%	2%	19%	22%	3%

Limitations

Due to data limitations, it is advised to ignore +/- 5% change in success rate at this time. Also, to note that where data entries of a specific PTLS are less than 15, even a small fluctuation will end up showing a big difference.

Future recommendations

Teachers spend most of the time with the students during their language training. If teachers are well trained and well qualified, then teachers can tailor their instruction to student needs and students PTLS preferences. Teachers can also train the students in facilitating their students to use their PTLS preferences to overcome students' learning weaknesses. Actual results are checked with the predicted results. If the actual results are better than the predicted results than it is called a boost. If the actual results are worse than the predicted results than it is called a drag. This difference in predicted to actual result i.e., the boost or drag effect is contributed to the external factor i.e., the teacher effect. Cobb-Douglas utility function can be used to show the relationship between the various inputs within the external factor i.e., teacher effect. Cobb-Douglas is a concept from economics that shows the relationship between two or more inputs and the amount of the output that can be produced by those inputs such as explained in the article Human-Capital Investments and Productivity (Black & Lynch, 1996). In the book Managing Complexity: Practical Considerations in the Development and Application of ABMs to Contemporary Policy Challenges on page 51, “The Utility Function” is mentioned; “utility function can incorporate relevant theories and factors which show some kind of a relationship between variables, and it can instantiate different theories by adjusting parameters” (Hadzikadic, O’Brien, & Khouja, 2013). Cobb-Douglas utility function was used to develop an Agent-Based Model (ABM), Actionable Capability for Social and Economic Systems model (ACSES model) because it is easily expandable to include additional preferences or values or motivations if they are important for a theory (page 52)(Hadzikadic et al., 2013). Specific version of the Cobb-Douglas utility function is given in the form of equation (2) on page 55 in the Managing Complexity: Practical Considerations in the Development and Applications of ABMs to Contemporary Policy Challenges, “U = (1-L)WL (1-C)WC(1-I)WI(1-E)WE(1-V)WV(1-F)WF(1-R)WR; where L is loyalty to leader, C is coercion, I is ideology, E is economic welfare, and R is the repression and social influence for defying repression and the weights Wx for motivation x, give the relative importance of the different motivations to the agent and the relative effect they have on U” (Hadzikadic et al., 2013). On the same pattern by using Argumentation Theory (Grossi, 2010), and Attribution Theory (Weiner, 1972) (Kelley & Michela, 1980), through Cobb-Douglas function teacher effect can be factored in by considering teacher experience, qualifications, and effort. Y = (TE-Experience)$^{.25}$(TE-Qualifications)$^{.35}$(TE-Effort)$^{.40}$

Table 3: TE Experience Chart

TE Experience Chart		
	Experience	**Face value assigned for calculations**
1	Two years or less $X \leq 2$	.19
2	Six years or less but more than two years $2 < X \leq 6$	.39
3	Ten years or less but more than six years $6 < X \leq 10$	.59
4	15 years or less but more than ten years $10 < X \leq 15$	.79
5	More than 15 years $X > 15$	.99

Table 4: TE Qualifications Chart

TE Qualifications Chart		
	Qualifications	**Face values assigned for calculations**
1	HS/HS+	.19
2	BA/BS	.39
3	BA/BA in language/target language/linguistics/teaching	.59
4	MA/MSc.	.79
5	MA/MSc. or above (PhD). OR MA/MSc. in language/target language/linguistics/teaching.	.99

If the drag is high which impacted the results, then one can look into training the teachers and at the difference in results after teacher training. This research can also lead to any future research to devise models that may be utilized in specific instances where the PTLS permutation for a particular student is not the best fit for the specific program. In such cases, that kind of a model may suggest different learning strategies for achieving better results in training (just in case the student has no other option to avoid that training program). This may also decrease the student turnover/failure rate, enhance learning retention, and may lead to life-long/continuous learning. This is because various personality types and learning styles prefer different learning strategies. One may also calculate different scenarios where the cost savings, timelines of courses, and learning efficiency of students can be connected.

The impact of technology on each segment within the external factor can also be calculated. If new technology is introduced such as new smartboard technology, which is helping the teacher in teaching and saving time, then an increment value can be added to the assigned values for a set scale within the TE experience category and vice-versa. Also, the impact of professional development can be analyzed within the TE qualifications on a similar pattern. Afterward, the feedback from the students about teacher's quality of instruction and experience with respect to the results can be verified with the boost or drag result.

Enough data is needed to continue validating the teacher effect when calculating boost or drag; where boost is the increased rate of success and drag is the decreased rate of success.

REFERENCES

Adhatrao, K., Gaykar, A., Dhawan, A., Jha, R., & Honrao, V. (2013). Predicting Students' Performance using ID3 and C4. 5 Classification Algorithms. *arXiv preprint arXiv:1310.2071*.

Black, S. E., & Lynch, L. M. (1996). Human-capital investments and productivity. *The American Economic Review, 86*(2), 263-267.

Boeree, C. G. (1997). Carl Jung. *Personality Theories*, 13-14.

Chan, S. (2001). Complex Adaptive Systems. 1-9.

Clawson, J. G. (1997). Adult learning theory.

Davis, R. A. (1989). *Personality: Its use in selecting candidates for US Air Force undergraduate pilot training*. Retrieved from DTIC Document Research Report No. AU-ARI-88-8, Air University Press, Maxwell Air Force Base, Alabama.

DLAB Test Score Range. (2014). Retrieved from http://dlabprep.com/dlab-test-score-range/

DOLGIN, D., & GIBB, G. (1988). *PERSONALITY-ASSESSMENT IN AVIATION SELECTION-PAST, PRESENT, AND FUTURE.* Paper presented at the AVIATION SPACE AND ENVIRONMENTAL MEDICINE.

Ehrman, M., Leaver, B., & Skekhtman, B. (2002). E & L learning style questionnaire v. 2.0. *Retrieved May, 20*, 2010.

Ehrman, M., & Leaver, B. L. (2003). Cognitive styles in the service of language learning. *System, 31*(3), 393-415.

Getchell, A. (2008). Agent-based modeling. *Physics, 22*(6), 757-767.

Grossi, D. (2010). *On the logic of argumentation theory.* Paper presented at the Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1.

Hadzikadic, M., O'Brien, S., & Khouja, M. (2013). *Managing Complexity: Practical Considerations in the Development and Application of ABMs to Contemporary Policy Challenges* (Vol. 504): Springer.

Hambali Moshood, A. Comparative Analysis of Decision Tree Algorithms for Predicting Undergraduate Students' Performance in Computer Programming.

Helmreich, R. L., & Wilhelm, J. A. (1987). Human performance in aerospace environments: The search for psychological determinants.

Herzog, M. (2007). An overview of the history of the ILR language proficiency skill level descriptions and scale. *Interagency Language Roundtable*.

Kelley, H. H., & Michela, J. L. (1980). Attribution theory and research. *Annual review of psychology, 31*(1), 457-501.

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Teechniques. (Informatics 31 (2007) 249-268 249), 20.

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques.

Kuhn, M. (2008). Building Predictive Models in R Using the Caret Package. *Journal of Statistical Software, 28*(5), 1-26.

Leaver, B. L. (1997). *Teaching the whole class* (4th ed.): Corwin Press, Inc. A Sage Publication Company.

Leaver, B. L., Ehrman, M., & Shekhtman, B. (2005). *Achieving success in second language acquisition* (1st ed.). UK: Cambridge University Press.

Lightbown, P., & Spada, N. M. (2006). *How languages are learned* (3rd ed.): Oxford University Press, USA.

Liskin-Gasparro, J. E. (2003). The ACTFL proficiency guidelines and the oral proficiency interview: A brief history and analysis of their survival. *Foreign Language Annals, 36*(4), 483-490.

Martin, J. (1992). A Comparison of Personality Type and Learning Style of Elementary Education Majors, Math Majors, and Math Professors: Cultures in Conflict.

Meyer, D., & Wien, F. T. (2015). Support vector machines. *The Interface to libsvm in package e1071*.

Mitchell, T. M. (2006). *The discipline of machine learning* (Vol. 9): Carnegie Mellon University, School of Computer Science, Machine Learning Department.

Neville, P. G. (1999). Decision trees for predictive modeling. *SAS Institute Inc, 4*.

Okeke Ogochukwu, C., & Ezenwegbu Nnamdi, C. Application of Data-mining Technique and Intelligent System on Demography Analysis in Nigeria.

Oxford, R. L. (1990). *Language learning strategies: What every teacher should know*: Newbury House Publishers.

Oxford, R. L. (2013). *Teaching & Researching: Language Learning Strategies*: Routledge.

Petersen, C. R., & Al-Haik, A. R. (1976). The Development of the Defense Language Aptitude Battery (Dlab. *Educational and Psychological Measurement, 36*(2), 369-380.

Safavian, S. R., & Landgrebe, D. (1990). A survey of decision tree classifier methodology.

Salzberg, S. L. (1994). C4. 5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning, 16*(3), 235-240.

Taskar, B., Segal, E., & Koller, D. (2001). *Probabilistic classification and clustering in relational data.* Paper presented at the International Joint Conference on Artificial Intelligence.

Tovar, C. C. a. D. (2007). Communicative Language Teaching. *Dialog on Language Instruction, Volume 18 Number 1 & 2*, 69-75.

Wang, X., & Fu, Y. (2013). *Some characterizations of the Cobb-Douglas and CES production functions in microeconomics.* Paper presented at the Abstract and Applied Analysis.

Weiner, B. (1972). Attribution theory, achievement motivation, and the educational process. *Review of educational research, 42*(2), 203-215.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., . . . Philip, S. Y. (2008). Top 10 algorithms in data mining. *Knowledge and information systems, 14*(1), 1-37.

Wurster, C. D. (1993). *Myers-Briggs Type Indicator: A cultural and ethical evaluation*. Retrieved from DTIC Document Executive Research Project S86, The Industrial College of the Armed Forces National Defense University.

APPENDIX A: PRE-HYBRID MODEL TRIALS & ERRORS

Before this research methodology and devising the concept of Hybrid Comparative Predictive Modeling; initial research had trials and errors which included looking at the algorithms and classifiers other than SVM and Decision Trees. However, the classifier accuracy achieved through trained data was much lower than the SVMs and Decision Trees. Moreover, in some instances even if the model accuracy was above 80%, the prediction differences were more than 25% in some cases. The reasons to select SVMs and decision trees are already in the literature review. Initially, data of thirteen languages as one unit was considered, but some language data contained hardly 25 to 30 data entries. When the classification learners were used in calculating one fit for all the languages, accuracy rate was much lower and optimized placement could not be achieved in each language i.e., one PT cannot fit all languages across the board. A person good in Russian cannot perform or learn the same way in another language like Arabic, French, Chinese, or Korean.

Other programs and methods such as Complex Systems and Agent-Based Modeling (Chan, 2001; Getchell, 2008) and Agent Based Modeling vs Equation Based Modeling was also examined. After reviewing the dynamics of the project, this is more of a classification and Machine Learning problem instead of a complex adaptive system problem. All the trial and errors of that research led to the proposed theory of this Hybrid Predictive Model, which was experimented, developed, and validated in this thesis. For this project, initially, R programming was looked at as a preferred tool and Caret package in R was used for classification, which contained numerous tools for developing predictive models using the rich set of models available in R (Kuhn, 2008). In R programming, algorithms such as ID3 and C4.5 were considered and Random Forest was used for predictions. However, MATLAB was a more convenient, efficient, and faster tool in the environment where the research project was conducted and it will be more convenient and adaptable to the users afterwards.

APPENDIX B: DATA NOTES

Phase 1 Data calculations folder (separate folder)

(Folder contains the following)

a. Final-AD MASTER DATA LIST
    a. All the data coding and data processing for ML through SVM and DT
b. Final-FR MASTER DATA LIST
    a. All the data coding and data processing for ML through SVM and DT
c. Final-RU MASTER DATA LIST
    a. All the data coding and data processing for ML through SVM and DT
d. Final-CLN MASTER DATA LIST
    a. All the data coding and data processing for ML through SVM and DT

Phase 2 Data calculations folder (separate folder)

(Folder contains the following)

a. LN-Comparison1-Base
b. LN-Comaprison2-Update
c. LN-General Comparison
d. AD-FR-RU-CLN-Model-Pred Accuracy (comparisons)
e. TE-Calculations

APPENDIX C: PT SNAPSHOT PROGRAM NOTES

PT Snapshot Program folder (separate folder)

a. PT Source Code folder

b. PTLS Snapshot Program

APPENDIX D: DETAILED SVM & DT RESULTS

ARABIC RESULTS
(Summarized in Table 7)

Iteration 1-AD_ML PTLS-DLAB – 228 Entries

Model number 1.1
Status: Trained
Accuracy: 76.8%
Prediction speed: ~1300 obs/sec
Training Time: 8.6159 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 78.1%
Prediction speed: ~2700 obs/sec
Training Time: 1.3931 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 79.8%
Prediction speed: ~2800 obs/sec
Training Time: 1.2966 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 79.4%
Prediction speed: ~1700 obs/sec
Training Time: 3.8182 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 77.6%
Prediction speed: ~2200 obs/sec
Training Time: 4.1665 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 71.9%
Prediction speed: ~2600 obs/sec
Training Time: 49.431 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 80.7%
Prediction speed: ~2300 obs/sec
Training Time: 1.5255 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2

Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 80.3%
Prediction speed: ~2100 obs/sec
Training Time: 1.407 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.7
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 81.1%
Prediction speed: ~2600 obs/sec
Training Time: 1.3667 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 2-AD_ML PTLS+DLAB – 92 Entries

Model number 1.1
Status: Trained
Accuracy: 80.4%
Prediction speed: ~770 obs/sec
Training Time: 1.7793 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 80.4%
Prediction speed: ~820 obs/sec
Training Time: 1.5785 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 81.5%
Prediction speed: ~790 obs/sec
Training Time: 1.6241 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 80.4%
Prediction speed: ~800 obs/sec
Training Time: 1.7223 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained

Accuracy: 80.4%
Prediction speed: ~810 obs/sec
Training Time: 1.7797 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 70.7%
Prediction speed: ~840 obs/sec
Training Time: 1.701 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 80.4%
Prediction speed: ~800 obs/sec
Training Time: 1.6323 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 83.7%
Prediction speed: ~770 obs/sec
Training Time: 1.6191 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 83.7%
Prediction speed: ~910 obs/sec
Training Time: 1.6212 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19

Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 3-ADef_ML-DLAB – 228 Entries

Model number 1.1
Status: Trained
Accuracy: 76.8%
Prediction speed: ~2000 obs/sec
Training Time: 1.6391 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 76.8%
Prediction speed: ~1800 obs/sec
Training Time: 1.6011 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 76.8%
Prediction speed: ~1900 obs/sec
Training Time: 1.5246 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 77.6%
Prediction speed: ~1700 obs/sec
Training Time: 1.7573 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 79.4%
Prediction speed: ~1600 obs/sec
Training Time: 1.7513 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 73.7%
Prediction speed: ~1700 obs/sec
Training Time: 1.7353 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 81.1%

Prediction speed: ~1800 obs/sec
Training Time: 1.7808 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 80.7%
Prediction speed: ~1800 obs/sec
Training Time: 1.7248 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.1
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 81.1%
Prediction speed: ~1700 obs/sec
Training Time: 1.7542 secs

Classifier

Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 24
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 4-ADef_ML+DLAB – 92 Entries

Model number 1.1
Status: Trained
Accuracy: 69.6%
Prediction speed: ~850 obs/sec
Training Time: 1.9632 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 69.6%
Prediction speed: ~870 obs/sec
Training Time: 1.3824 secs

Classifier
Preset: Medium Tree

Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 81.5%
Prediction speed: ~840 obs/sec
Training Time: 1.1527 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 82.6%
Prediction speed: ~770 obs/sec
Training Time: 1.6039 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 75.0%
Prediction speed: ~750 obs/sec
Training Time: 1.5531 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 75.0%
Prediction speed: ~720 obs/sec
Training Time: 1.5934 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 83.7%
Prediction speed: ~820 obs/sec
Training Time: 1.5935 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 83.7%
Prediction speed: ~800 obs/sec
Training Time: 1.5955 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 83.7%
Prediction speed: ~740 obs/sec

Training Time: 1.5675 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 25
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 5-ADef_ML-DLAB-R – 228 Entrees

Model number 1.1
Status: Trained
Accuracy: 71.5%
Prediction speed: ~2000 obs/sec
Training Time: 1.5014 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 75.9%
Prediction speed: ~2200 obs/sec
Training Time: 1.5071 secs

Classifier

Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 79.8%
Prediction speed: ~1900 obs/sec
Training Time: 1.3498 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 78.9%
Prediction speed: ~1800 obs/sec
Training Time: 1.6396 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection

All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 78.5%
Prediction speed: ~1900 obs/sec
Training Time: 1.6781 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 74.6%
Prediction speed: ~2000 obs/sec
Training Time: 1.6445 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 81.1%
Prediction speed: ~1700 obs/sec
Training Time: 1.646 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.4
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 81.1%
Prediction speed: ~1900 obs/sec
Training Time: 1.8866 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.7
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained

Accuracy: 81.1%
Prediction speed: ~2100 obs/sec
Training Time: 1.6689 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 6-ADef_ML+DLAB-R – 92 Entrees

Model number 1.1
Status: Trained
Accuracy: 79.3%
Prediction speed: ~870 obs/sec
Training Time: 1.2934 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 79.3%
Prediction speed: ~890 obs/sec

Training Time: 1.2234 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 80.4%
Prediction speed: ~910 obs/sec
Training Time: 1.2204 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 81.5%
Prediction speed: ~630 obs/sec
Training Time: 3.9158 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One

Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 75.0%
Prediction speed: ~810 obs/sec
Training Time: 1.5676 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 76.1%
Prediction speed: ~810 obs/sec
Training Time: 1.552 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 83.7%
Prediction speed: ~770 obs/sec
Training Time: 2.9325 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 83.7%
Prediction speed: ~790 obs/sec
Training Time: 1.4671 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 83.7%
Prediction speed: ~920 obs/sec
Training Time: 1.3859 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

RUSSIAN RESULTS
(Summarized in Table 8)

Iteration 1-RU_ML PTLS-DLAB – 178 Entrees

Model number 1.1
Status: Trained
Accuracy: 55.6%
Prediction speed: ~1000 obs/sec
Training Time: 8.3252 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 58.4%
Prediction speed: ~1900 obs/sec
Training Time: 1.4725 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 60.1%
Prediction speed: ~2300 obs/sec
Training Time: 1.2048 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 64.6%
Prediction speed: ~1400 obs/sec

Training Time: 3.3887 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 58.4%
Prediction speed: ~1900 obs/sec
Training Time: 1.7816 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 59.0%
Prediction speed: ~1600 obs/sec
Training Time: 2.3193 secs

Classifier
Preset: Cubic SVM

Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 61.8%
Prediction speed: ~2000 obs/sec
Training Time: 1.4208 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 65.7%
Prediction speed: ~2000 obs/sec
Training Time: 1.4359 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.7
Box constraint level: 1
Multiclass method: One-vs-One

Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 66.3%
Prediction speed: ~1900 obs/sec
Training Time: 1.2925 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 2-RU_ML PTLS+DLAB – 131 Entrees

Model number 1.1
Status: Trained
Accuracy: 61.8%
Prediction speed: ~1400 obs/sec
Training Time: 1.2374 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 61.8%
Prediction speed: ~1600 obs/sec
Training Time: 1.1925 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 70.2%
Prediction speed: ~1200 obs/sec
Training Time: 1.2143 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 71.8%
Prediction speed: ~1300 obs/sec
Training Time: 1.7377 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 61.1%
Prediction speed: ~1400 obs/sec
Training Time: 1.3112 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 63.4%
Prediction speed: ~1400 obs/sec
Training Time: 1.2382 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 62.6%
Prediction speed: ~1100 obs/sec
Training Time: 1.3957 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 64.9%
Prediction speed: ~1500 obs/sec
Training Time: 1.2188 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian

Kernel scale: 4.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 65.6%
Prediction speed: ~1100 obs/sec
Training Time: 1.6477 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 3-RUef_ML-DLAB – 178 Entries

Model number 1.1
Status: Trained
Accuracy: 59.0%
Prediction speed: ~1700 obs/sec
Training Time: 1.3046 secs

Classifier
Preset: Complex Tree

Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 58.4%
Prediction speed: ~1600 obs/sec
Training Time: 1.3604 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 68.5%
Prediction speed: ~1700 obs/sec
Training Time: 1.1599 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 68.0%
Prediction speed: ~1700 obs/sec
Training Time: 1.2529 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 66.3%
Prediction speed: ~1700 obs/sec
Training Time: 1.2748 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 62.4%
Prediction speed: ~1600 obs/sec
Training Time: 1.3123 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 64.6%
Prediction speed: ~1600 obs/sec
Training Time: 1.3695 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 66.3%
Prediction speed: ~1000 obs/sec

Training Time: 1.9151 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.1
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 66.3%
Prediction speed: ~1700 obs/sec
Training Time: 1.3798 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 24
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 4-RUef_ML+DLAB – 131 Entries

Model number 1.1
Status: Trained
Accuracy: 59.5%
Prediction speed: ~1300 obs/sec

Training Time: 1.3169 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 59.5%
Prediction speed: ~1300 obs/sec
Training Time: 1.2288 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 67.9%
Prediction speed: ~1400 obs/sec
Training Time: 1.2035 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 61.8%
Prediction speed: ~760 obs/sec
Training Time: 1.9819 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 60.3%
Prediction speed: ~1100 obs/sec
Training Time: 1.312 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 57.3%
Prediction speed: ~1200 obs/sec
Training Time: 1.2951 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 64.9%
Prediction speed: ~1100 obs/sec
Training Time: 1.4392 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 61.8%
Prediction speed: ~1100 obs/sec
Training Time: 1.4268 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 65.6%
Prediction speed: ~1300 obs/sec
Training Time: 1.3626 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 25
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 5-RUef_ML-DLAB-R – 178 Entrees

Model number 1.1
Status: Trained
Accuracy: 57.3%
Prediction speed: ~1700 obs/sec
Training Time: 1.1927 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 57.3%
Prediction speed: ~1900 obs/sec
Training Time: 1.1813 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 62.9%
Prediction speed: ~1800 obs/sec
Training Time: 1.1243 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 62.4%
Prediction speed: ~1700 obs/sec
Training Time: 1.2899 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 54.5%
Prediction speed: ~1600 obs/sec
Training Time: 1.3336 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One

Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 55.6%
Prediction speed: ~1700 obs/sec
Training Time: 1.3151 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 64.0%
Prediction speed: ~1700 obs/sec
Training Time: 1.2722 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.4
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 65.2%
Prediction speed: ~1700 obs/sec
Training Time: 1.3662 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.7
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 66.3%
Prediction speed: ~1300 obs/sec
Training Time: 1.8876 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 6-RUef_ML+DLAB-R – 131 Entrees

Model number 1.1
Status: Trained
Accuracy: 67.9%
Prediction speed: ~1300 obs/sec
Training Time: 1.1996 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 67.9%
Prediction speed: ~1400 obs/sec
Training Time: 1.1888 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 71.0%

Prediction speed: ~1400 obs/sec
Training Time: 1.115 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 68.7%
Prediction speed: ~1200 obs/sec
Training Time: 1.2762 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 66.4%
Prediction speed: ~1200 obs/sec
Training Time: 1.3661 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic

Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 62.6%
Prediction speed: ~1200 obs/sec
Training Time: 1.3283 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 67.2%
Prediction speed: ~1200 obs/sec
Training Time: 1.2947 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 65.6%
Prediction speed: ~1200 obs/sec
Training Time: 1.2753 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 65.6%
Prediction speed: ~1100 obs/sec
Training Time: 1.3294 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

FRENCH RESULTS
(Summarized in Table 9)

Iteration 1-FR_ML PTLS-DLAB – 246 Entries

Model number 1.1
Status: Trained
Accuracy: 78.0%
Prediction speed: ~3000 obs/sec
Training Time: 1.1808 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 77.6%
Prediction speed: ~3000 obs/sec
Training Time: 1.1088 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 78.9%
Prediction speed: ~2900 obs/sec
Training Time: 1.1671 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 80.5%
Prediction speed: ~2500 obs/sec
Training Time: 1.4122 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained

Accuracy: 74.0%
Prediction speed: ~2800 obs/sec
Training Time: 55.069 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 69.9%
Prediction speed: ~2400 obs/sec
Training Time: 71.548 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 79.7%
Prediction speed: ~2500 obs/sec
Training Time: 1.3017 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 80.5%
Prediction speed: ~2400 obs/sec
Training Time: 1.3077 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.7
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 80.9%
Prediction speed: ~2400 obs/sec
Training Time: 1.7145 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19

Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 2-FR_ML PTLS+DLAB – 92 Entrees

Model number 1.1
Status: Trained
Accuracy: 80.4%
Prediction speed: ~1200 obs/sec
Training Time: 1.0533 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 80.4%
Prediction speed: ~1200 obs/sec
Training Time: 1.0427 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 79.3%
Prediction speed: ~1000 obs/sec
Training Time: 1.1727 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 84.8%
Prediction speed: ~1000 obs/sec
Training Time: 1.1964 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Model is favorite
Status: Trained
Accuracy: 75.0%
Prediction speed: ~1100 obs/sec
Training Time: 1.2197 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 73.9%
Prediction speed: ~1000 obs/sec
Training Time: 1.3219 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 83.7%

Prediction speed: ~960 obs/sec
Training Time: 1.2572 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 84.8%
Prediction speed: ~920 obs/sec
Training Time: 1.2406 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 84.8%
Prediction speed: ~950 obs/sec
Training Time: 1.2296 secs

Classifier

Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 3-FRef_ML-DLAB – 246 Entries

Model number 1.1
Status: Trained
Accuracy: 73.2%
Prediction speed: ~2600 obs/sec
Training Time: 1.3581 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 73.2%
Prediction speed: ~2500 obs/sec
Training Time: 1.2438 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20

Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 78.5%
Prediction speed: ~2400 obs/sec
Training Time: 1.134 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 79.7%
Prediction speed: ~2100 obs/sec
Training Time: 1.4364 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 72.4%
Prediction speed: ~2100 obs/sec
Training Time: 1.6165 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 70.7%
Prediction speed: ~2000 obs/sec
Training Time: 1.4216 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 80.5%
Prediction speed: ~870 obs/sec
Training Time: 1.7806 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 80.5%
Prediction speed: ~2200 obs/sec
Training Time: 1.3137 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.1
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 80.9%
Prediction speed: ~2200 obs/sec

Training Time: 1.465 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 24
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 4-FRef_ML+DLAB – 92 Entries

Model number 1.1
Status: Trained
Accuracy: 81.5%
Prediction speed: ~830 obs/sec
Training Time: 1.2788 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 81.5%
Prediction speed: ~830 obs/sec
Training Time: 1.2106 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 82.6%
Prediction speed: ~910 obs/sec
Training Time: 1.253 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 84.8%
Prediction speed: ~820 obs/sec
Training Time: 1.3709 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 83.7%
Prediction speed: ~790 obs/sec
Training Time: 1.4288 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 81.5%
Prediction speed: ~860 obs/sec
Training Time: 1.2348 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 84.8%
Prediction speed: ~870 obs/sec
Training Time: 1.4563 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 84.8%
Prediction speed: ~790 obs/sec
Training Time: 1.366 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 84.8%
Prediction speed: ~800 obs/sec
Training Time: 1.354 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 25
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 5-FRef_ML-DLAB-R – 246 Entrees

Model number 1.1
Status: Trained
Accuracy: 70.7%
Prediction speed: ~2400 obs/sec
Training Time: 1.3786 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained

Accuracy: 72.8%
Prediction speed: ~2500 obs/sec
Training Time: 1.2411 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 79.7%
Prediction speed: ~2800 obs/sec
Training Time: 1.1288 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 80.1%
Prediction speed: ~2300 obs/sec
Training Time: 1.3426 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic

Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 69.9%
Prediction speed: ~2200 obs/sec
Training Time: 1.5179 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 72.4%
Prediction speed: ~2100 obs/sec
Training Time: 1.6479 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 79.7%
Prediction speed: ~800 obs/sec
Training Time: 1.8515 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.4
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 80.9%
Prediction speed: ~2400 obs/sec
Training Time: 1.307 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.7
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 80.9%
Prediction speed: ~2300 obs/sec
Training Time: 1.4079 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 6-FRef_ML+DLAB-R – 92 Entries

Model number 1.1
Status: Trained
Accuracy: 70.7%
Prediction speed: ~930 obs/sec
Training Time: 1.1906 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 70.7%
Prediction speed: ~770 obs/sec
Training Time: 1.3442 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 76.1%
Prediction speed: ~900 obs/sec
Training Time: 1.1546 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 81.5%
Prediction speed: ~730 obs/sec
Training Time: 1.334 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 82.6%
Prediction speed: ~840 obs/sec
Training Time: 1.2564 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Model is favorite
Status: Trained
Accuracy: 81.5%
Prediction speed: ~940 obs/sec
Training Time: 1.2468 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic

Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 84.8%
Prediction speed: ~790 obs/sec
Training Time: 1.4256 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 84.8%
Prediction speed: ~900 obs/sec
Training Time: 1.2827 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 84.8%
Prediction speed: ~860 obs/sec
Training Time: 1.4016 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

CLN RESULTS
(Summarized in Table 10)

Iteration 1-CLN_ML PTLS-DLAB – 652 Entries

Model number 1.1
Status: Trained
Accuracy: 63.8%
Prediction speed: ~3500 obs/sec
Training Time: 10.608 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100

Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 66.4%
Prediction speed: ~7800 obs/sec
Training Time: 1.8167 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 67.5%
Prediction speed: ~6500 obs/sec
Training Time: 1.5481 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 68.1%
Prediction speed: ~4300 obs/sec
Training Time: 4.9934 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 67.3%
Prediction speed: ~5400 obs/sec
Training Time: 14.089 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 65.5%
Prediction speed: ~4800 obs/sec

Training Time: 236.62 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 66.4%
Prediction speed: ~4800 obs/sec
Training Time: 2.5209 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 68.7%
Prediction speed: ~5600 obs/sec
Training Time: 2.1638 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.7

Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 67.3%
Prediction speed: ~6400 obs/sec
Training Time: 1.9104 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 2-CLN_ML PTLS+DLAB – 316 Entrees

Model number 1.1
Status: Trained
Accuracy: 59.4%
Prediction speed: ~2600 obs/sec
Training Time: 1.8337 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index

Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 58.4%
Prediction speed: ~3000 obs/sec
Training Time: 1.3682 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 61.0%
Prediction speed: ~3300 obs/sec
Training Time: 1.3072 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 65.7%
Prediction speed: ~2500 obs/sec
Training Time: 1.9756 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 66.0%
Prediction speed: ~2800 obs/sec
Training Time: 4.7069 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 60.3%
Prediction speed: ~3200 obs/sec
Training Time: 19.164 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 61.3%
Prediction speed: ~2400 obs/sec
Training Time: 1.7002 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 66.3%
Prediction speed: ~2700 obs/sec
Training Time: 1.6888 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 4.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 64.1%
Prediction speed: ~2500 obs/sec
Training Time: 1.6528 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 19
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 3-CLNef_ML-DLAB – 652 Entries

Model number 1.1
Status: Trained
Accuracy: 62.1%
Prediction speed: ~4700 obs/sec
Training Time: 1.9162 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 69.6%
Prediction speed: ~5300 obs/sec
Training Time: 1.4711 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 69.3%
Prediction speed: ~5600 obs/sec
Training Time: 1.3653 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 67.6%
Prediction speed: ~980 obs/sec
Training Time: 2.8786 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 64.4%
Prediction speed: ~4300 obs/sec
Training Time: 3.02 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 61.2%
Prediction speed: ~3300 obs/sec
Training Time: 2.8916 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1

Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 67.0%
Prediction speed: ~1000 obs/sec
Training Time: 3.376 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 68.4%
Prediction speed: ~900 obs/sec
Training Time: 3.1389 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.1
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 67.2%
Prediction speed: ~860 obs/sec
Training Time: 3.1592 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 24
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 4-CLNef_ML+DLAB – 315 Entries

Model number 1.1
Status: Trained
Accuracy: 63.8%
Prediction speed: ~2600 obs/sec
Training Time: 1.7318 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 63.8%
Prediction speed: ~3100 obs/sec
Training Time: 1.4794 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 65.1%
Prediction speed: ~2600 obs/sec
Training Time: 1.6186 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 66.0%
Prediction speed: ~2100 obs/sec
Training Time: 2.1618 secs

Classifier
Preset: Linear SVM
Kernel function: Linear

Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 61.0%
Prediction speed: ~2300 obs/sec
Training Time: 1.9267 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 56.5%
Prediction speed: ~2700 obs/sec
Training Time: 1.7354 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 61.9%
Prediction speed: ~620 obs/sec
Training Time: 2.54 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 65.7%
Prediction speed: ~670 obs/sec
Training Time: 2.7475 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 6.2
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 63.5%
Prediction speed: ~810 obs/sec
Training Time: 2.1654 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 25
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 5-CLNef_ML-DLAB-R – 652 Entrees

Model number 1.1
Status: Trained
Accuracy: 61.7%
Prediction speed: ~5500 obs/sec
Training Time: 1.9613 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 67.6%

Prediction speed: ~5600 obs/sec
Training Time: 1.5474 secs

Classifier
Preset: Medium Tree
Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 65.8%
Prediction speed: ~5700 obs/sec
Training Time: 1.4567 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 68.3%
Prediction speed: ~5000 obs/sec
Training Time: 1.9525 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 65.8%
Prediction speed: ~4700 obs/sec
Training Time: 3.3602 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 59.7%
Prediction speed: ~4500 obs/sec
Training Time: 2.8683 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 66.9%
Prediction speed: ~1700 obs/sec
Training Time: 2.8131 secs

Classifier
Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.4
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 67.9%
Prediction speed: ~1200 obs/sec
Training Time: 2.869 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.7
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 67.9%
Prediction speed: ~4700 obs/sec
Training Time: 2.0431 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.

Iteration 6-CLNef_ML+DLAB-R – 316 Entries

Model number 1.1
Status: Trained
Accuracy: 59.0%
Prediction speed: ~2700 obs/sec
Training Time: 1.6879 secs

Classifier
Preset: Complex Tree
Maximum number of splits: 100
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.2
Status: Trained
Accuracy: 60.6%
Prediction speed: ~2700 obs/sec
Training Time: 1.4478 secs

Classifier
Preset: Medium Tree

Maximum number of splits: 20
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 1.3
Status: Trained
Accuracy: 60.0%
Prediction speed: ~2800 obs/sec
Training Time: 1.4783 secs

Classifier
Preset: Simple Tree
Maximum number of splits: 4
Split criterion: Gini's diversity index
Surrogate decision splits: Off

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.1
Status: Trained
Accuracy: 68.6%
Prediction speed: ~2400 obs/sec
Training Time: 1.7546 secs

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.2
Status: Trained
Accuracy: 60.6%
Prediction speed: ~2100 obs/sec
Training Time: 1.9435 secs

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.3
Status: Trained
Accuracy: 60.0%
Prediction speed: ~2200 obs/sec
Training Time: 2.0812 secs

Classifier
Preset: Cubic SVM
Kernel function: Cubic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.4
Status: Trained
Accuracy: 63.2%
Prediction speed: ~640 obs/sec
Training Time: 2.5603 secs

Classifier

Preset: Fine Gaussian SVM
Kernel function: Gaussian
Kernel scale: 1.5
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.5
Status: Trained
Accuracy: 68.3%
Prediction speed: ~680 obs/sec
Training Time: 2.7642 secs

Classifier
Preset: Medium Gaussian SVM
Kernel function: Gaussian
Kernel scale: 5.8
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Model number 2.6
Status: Trained
Accuracy: 63.5%
Prediction speed: ~940 obs/sec
Training Time: 2.2441 secs

Classifier
Preset: Coarse Gaussian SVM
Kernel function: Gaussian
Kernel scale: 23
Box constraint level: 1
Multiclass method: One-vs-One

Standardize data: true

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

yfit = c.predictFcn(X) was used to check the test data; where c represents the classifier used for the test data and X represents the test data.