AI Builders: Teaching Thai Students to Build End-to-End Machine Learning Projects Online

Charin Polpanumas[1,*], Peerat Limkonchotiwat[2], Benyapa Matupumanon[1], Chompakorn Chaksangchaichot[2], Natkrita Chumlek[5], Nattakarn Phaphoom[2], Pakin Siwatammarat[6], Pined Laohapiengsak[2], Sukrit Amornpornwiwat[5], Witchapong Daroontham[1], Ekapol Chuangsuwanich[3], Kowin Kulruchakorn[1], Sarana Nutanong[2], and Titipat Achakulvisut[4,*]

[1]Central Tech, Thailand
[2]School of Information Science and Technology, VISTEC, Thailand
[3]Department of Computer Engineering, Chulalongkorn University, Thailand
[4]Department of Biomedical Engineering, Mahidol University, Thailand
[5]Krungsri Nimble Company, Thailand
[6]Data Science Research Center, Faculty of Science, Chiang Mai University, Thailand
[*]Corresponding authors

***Abstract*—Teaching machine learning (ML) and artificial intelligence (AI) for middle-to-high school students is crucial to build a good sustainable research community. However, the current ML education system and ecosystem in developing countries, including most countries in South East Asia (SEA), is still far from reaching for most students. Students, especially where English is their second language, still struggle to learn ML and AI even in the global accessibility of AI classes online. Thus, teaching AI in these circumstances can be challenging yet rewarding. Here, we organized a nine-weeks online summer school, called AI Builders, aiming to teach ML and AI to Thai students. We combined existing ML and AI curricula in conjunction with end-to-end ML projects. We provided recap classes in native language, locally collected datasets, mentorship for projects, guest lectures, and career discussions in our program. Students were able to satisfactory understand ML concepts and produced meaningful projects even though some might not fully had the necessary background at the start of the program. We discussed possible improvements for future iterations.**

***Keywords*—machine learning education, K-12 education, online learning, Thai students, Southeast Asia**

I. Introduction

Thailand and most countries in South East Asia have educational problems due to various factors such as the large gap in socio-economic status, disparities, geographical location of students, or shortage of teachers in some regions [1], [2]. Even though there is global availability of online classes such as Massive open online courses (MOOC) [3], developing countries still faces the problem of low educational quality. Moreover, MOOC has extremely high dropout rate, lower than 5% complete the program [4]. The problem is especially substantial for contemporary subjects such as ML and AI due to the scarcity of teaching materials, locally collected datasets, and limited number of mentors who can discuss with students in native languages. Overall, this introduces a long-term educational problem where Thailand and SEA countries cannot provide quality ML and AI education limiting the growing needs of human resources in these domains.

Recent developments in online ML and AI education promise to bring accessibility, inclusivity, and democratize scientific education [5]. There are multiple platforms which provide ML and AI classes freely or with minimal cost barriers such as deeplearning.ai, Fast.ai, edX, and Coursera. Most platforms provide free online classes but with passive learning. The COVID pandemic in 2020 has pushed some organizations to explore the online training format [6], [7]. Many summer schools and online conferences have started exploring online or hybrid formats such as Neuromatch Academy (NMA) [8], Neuromatch Conference [9], [10], World Wide Neuro (WWN) [11], Conference on Neural Information Processing Systems (NeurIPS), The International Conference on Learning Representations (ICLR), and Association for the Advancement of Artificial Intelligence (AAAI) to name a few. There are various benefits of teaching online including increase in diversity, inclusivity, and accessibility [12]. They also lower the complexity of in-person organization and the cost of physical spaces. With appropriate social engineering, online formats can still allow engagement similar to offline formats. For example, NMA [8] applies ML to group students with similar interests together. Some conferences such as AAAI and ICLR used avatar spaces such as Gather Town and Topia to replace in-person meetings, poster sessions, and discussions. The rise of social engineering approaches and online platforms promise to bring better online meetings and teaching.

There have been proposals on how to teach AI and ML concepts to elementary and high school students [13], [14]. Some initiatives explore teaching ML and AI to high school students in an offline format [13]. However, one obstacle is that it requires students to be present in the same physical space. A lot of aspects are still left unexplored when it comes to teaching high school students in an online format, especially for non-English speaking countries where ML education is less prevalent. Students still lack awareness of global education inhibiting their ability to self-study the available online courses. They also have limited views on how to apply ML or AI to solve their problem. Current availability of online ML classes and conferences alone do not properly suit the training of middle-to-high school students. Therefore, providing proper ML education while teaching them problem solving skills promises to mitigate the above problems.

In this paper, we discuss our experience organizing AI Builders, a summer school aiming to teach ML to middle-to-high school students in Thailand. It is a project-based nine-week online summer school, aiming to teach students ML, AI, and problem solving skills through projects. We discuss components including teaching format, curriculum, datasets, and mentorship for student projects. We made our summary materials, recap videos, and guest lectures openly available. Overall, students came out of the summer school with a good understanding of ML concepts and were able to build meaningful projects. Finally, we discuss potential improvements and considerations for future endeavors.

II. Objectives and Teaching Philosophy

We aim to equip middle and high school students with the skill sets to develop an end-to-end machine learning product to solve the problems they consider important in their own contexts. We define end-to-end as the entire process from problem definition, setting metrics and baselines for success, data collection and cleaning, exploratory data analysis, modeling and error analysis, to prototype deployment.

We take a top-down, hands-on, tool-agnostic approach in teaching the course. We follow the approach of "learning the whole game" [15] popularized by Fastai courses [16]. Essentially, we introduce the whole project examples and then break down our teaching into parts. We leverage the Fastai Practical Deep Learning for Coders (version 4) [16] and the Fastai framework [17] to teach students to build working, near state-of-the-art deep learning models in image classification, recommendation, and text classification. Then, we gradually introduce them to each component of the models such as data loaders, pretrained neural network architectures, transfer learning, loss functions, and optimization with stochastic gradient descent. During the learning process, we emphasize hands-on experience by requiring each student to deliver a working prototype as their capstone project. In fact, the majority of our mentors' time is allocated to consultation for these capstone projects. Lastly, even though we use the Fastai framework to teach machine learning, we allow students the freedom to choose their tools of choice for their capstone projects. This is not only due to practical reasons as there are some use cases the Fastai framework does not cover such as signal processing and speech synthesis, but also the fact that we aim to foster expertise in training and using machine learning models, not expertise in any specific tool or framework.

III. Program structures

There are crucial components which we incorporate to make school engaging and allow students to learn new concepts. In this section we describe components for organizing our school ranging from mentors, selection processes, curriculum, communication platforms, datasets, and post-program engagement.

A. Mentors and program manager

Our mentors were recruited volunteers who are local machine learning practitioners or researchers. They took turns to perform weekly lesson recaps for all students and auditors (one hour) and followed by consulting students' capstone projects (one hour). Since this was our first iteration, we had 13 mentors where we divided into six groups to mentor up to five students per group. Each mentor contributed approximately five hours per week for the program, broken down into two hours to study the lesson materials, two hours to attend weekly meetings (recaps and project consultation), and one hour for project consultation. Auditors were an experimental capacity created to test whether we can scale the program beyond our one-mentor-to-two-student approach. Lastly, a program manager was responsible for communication from the program to students and auditors, broadcasting the lesson recaps on Facebook Lives, setting up guest lecture events, moderating the community on Discord and other administrative tasks.

B. Applications

The applications were free of charge. We received applications on Airtable form within a 3 weeks period (Figure 1A).

An application included a take-home test and short statements. We expected students and auditors to be ready for Fastai Practical Deep Learning for Coders (version 4) [16]; therefore, we followed its prerequisites of knowing how to code in Python and having taken high school math courses. We required all applicants to complete a take-home test evaluating three main skills (i) basic Python programming, (ii) linear algebra with numpy and (iii) data manipulation with pandas library. To maximize applicants' opportunities, we provided pre-course workshops for those who might not have a strong programming background. Applicants were also asked to write two mandatory short statements (in Thai or English, 200 words) on why they want to participate and why they deserve to be in the program, as well as one optional short statement of the project they want to work on. Only the take-home tests and mandatory short statements were used for screening.

We received a total of 378 applications. Most students were from high schools (66.2%) and the rest (33.8%) were from middle school (Figure 2A). We got 58% of applicants from Bangkok (the capital city of Thailand), 10% from Bangkok perimeters, and the rest from 39 provinces in Thailand (Figure 2B,C). We also asked them to select their interests as multiple choices. Most students had interests in image processing (68.0%), followed by signal processing (49.7%) and social goods (47.6%) (Figure 2D). 72.8% of the applicants were able to complete the take-home tests.

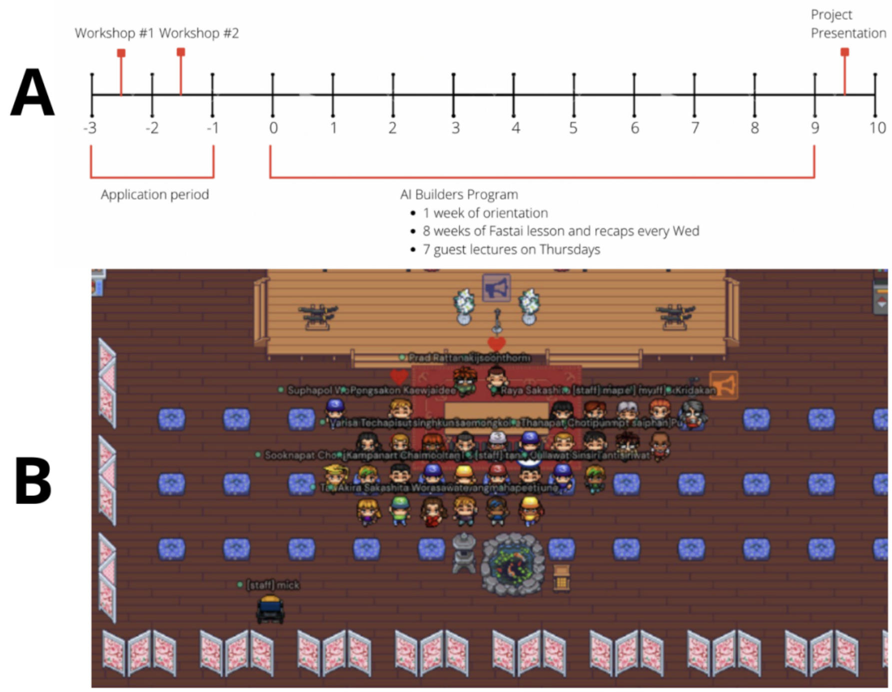


Fig. 1: Program timeline and platform. A. Program timeline. We had 3 weeks registration with precourse Python sessions. We then perform a selection after the registration ends followed by 9-week classes of Fast AI recap and deployment. Finally, we have final project presentations in the last week of program. B. Platform for teaching and mentor sessions. Each week, we arrange a recap class in Gather Town followed by a mentor session in a separated room.

C. Selection processes

We conducted a manual screening and narrowed down to 130 (34.3%) applications where students have satisfactory testing performance i.e. applicants got more than two out three of the main evaluated skills. We then asked each group of mentors to review the applications. The mentors are instructed to choose up to five students into their groups as if they were hiring a junior candidate to work in their teams, and up to five others as auditors. Finally, we had 25 students, who get project mentors, and 21 auditors.

D. Curriculum

To achieve our goal of giving students the problem-solving and machine learning skills necessary to tackle their real-world problems, we used Fastai Practical Deep Learning for Coders (version 4) [16] as the backbone of our training program. Students underwent the nine-weeks curriculum that includes video lectures from Fastai, weekly lesson recaps in native language, and project consultation. Finally, students who passed give the project presentation at the end of the program. Auditors were an experimental addition created to test whether we can scale the program beyond our one-mentor-to-two-student approach. Auditors were allowed to follow the Fastai course and to attend the one-hour weekly lesson recaps but were not expected to deliver the projects and thus were not paired with any mentor. However, they were encouraged to develop the projects on their own with mentor suggestions. Their projects, if submitted, would be evaluated in the same manner as students. We loosely modeled our course in the flipped classroom format [18] except for the first orientation class. Each week consisted of the following steps. First, students were instructed to watch a lecture and leave their questions in an anonymous online forum on Slido. Then, we provided a lesson recap in Thai and answer the submitted questions; this session usually took one hour. After that, students would go into separate project consultation sessions with their respective groups and mentors. Mentors would work with students to discuss the scope of the projects, break down problems, set goals, and assign tasks for next week.

Besides regular classes, we also provided a career session and weekly guest lectures by local researchers and practitioners to discuss their projects. These include classes such as practical data science tips, applying ML for detecting preserved animals in Thailand, the relationship between physics and ML, Natural Language Processing (NLP), and how to apply for undergraduate programs abroad. We made all the course materials, lectures and guest lectures freely available on our website [1] and Facebook page [2]. We also provided an in-class Kaggle Competition using a locally collected dataset and additional deployment class which focus on deploying ML models using web tools including Streamlit [3] and Heroku.

[1]https://vistec-ai.github.io/ai-builders/
[2]https://www.facebook.com/aibuildersx
[3]https://streamlit.io/

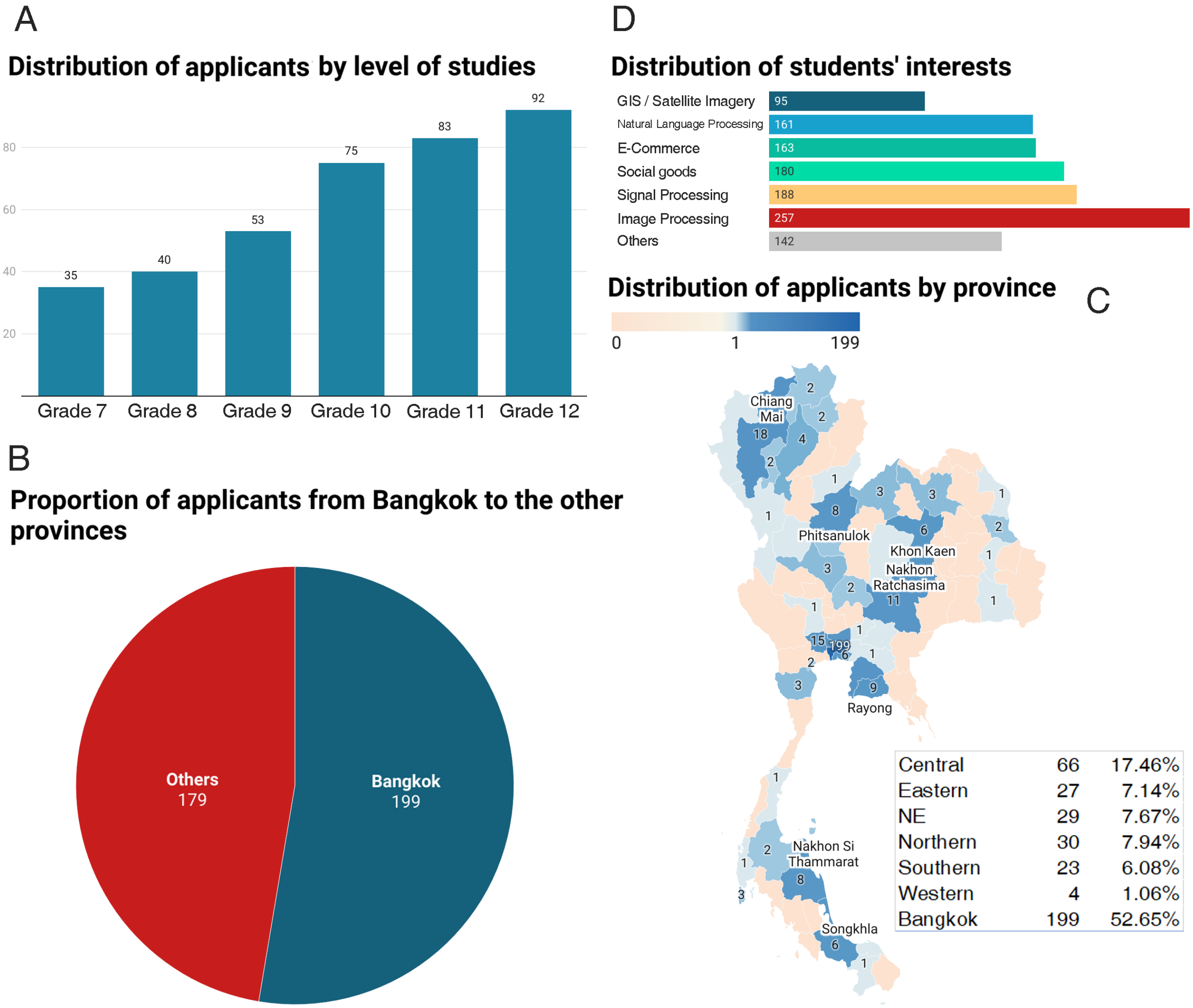


Fig. 2: Statistics of applications. A. Distribution of applications by level of studies. Here we show the number of applications by their study level. B. Distribution of applications in Bangkok and other provinces. 58% of applications are from Bangkok, 10% from its vicinity, and the rest are from other provinces. C. Distribution of applications outside Bangkok color-coded by their region. We show the distribution of applications in other cities color-coded by their regions i.e. North, Northeastern, East, South, West. D. Distribution of students' interests. Most students are interested in image processing followed by signal processing and social goods. .

E. Communication platforms

Communication is an important key for organizing online teaching. We had several layers of communication platforms. We used official communication with email, in-class communication in Gather Town (Figure 1B), out-of-class communication with participants in Discord, and public communication through our Facebook page. Official communication such as acceptance letters, program communication, and project evaluation was always conducted by email. Our weekly class and mentorship for the project were held in Gather Town. It allowed us to perform the lesson recap with every participant in the large hall followed by project consultation in separate rooms without moving out of the platform. Outside of class, we used Discord as our main communication channel with participants and Facebook page for engaging with the public. Discord allows students to work on projects or study together online. We noticed that students generally use the Discord channel for group listening to Anime songs during their programming or studying sessions. They could also reach out to mentors if they needed help with programming questions. Aside from Discord, we also discussed and shared student projects with local communities such as Thai Natural

Language Processing (Thai NLP) [4] and Data Science BKK [5]. Our platform cost was all from Gather Town, which costed $120 per month (educational discount) for two months to host 60 users. Facebook live and Discord chat were free.

F. Datasets

To interact with and contribute to the local ML community, we incorporated multiple local datasets to our program for our teaching materials and for student projects. For example, we used the Mak Pin Lom dataset [6], a gallery of local plant seeds collected by a team in Northeastern Thailand, for the mid-term in-class Kaggle competition where the seed type can be defined from their shapes (Figure 3).

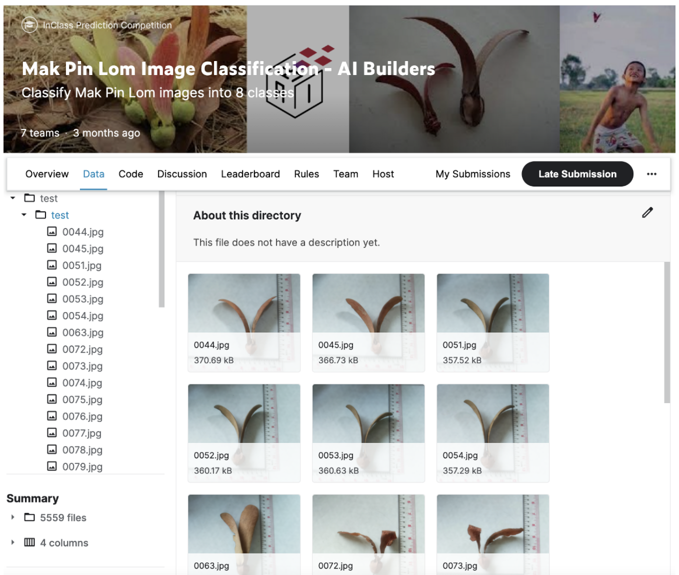


Fig. 3: Examples of in-class Kaggle competition and student final projects. We use the of Mak pin lom dataset, collected by a research team in Northeastern Thailand, for an in-class Kaggle competition.

Moreover, we encouraged students to create their own datasets under open source license or use local ML datasets. After the program, students produced and worked on various datasets including a Chinese-Thai machine translation dataset [19], a Thai OCR dataset [20], a Thai image captioning dataset back-translated from Flickr-8k dataset [21], Thai text-to-speech dataset [22], Thai instrument (Ranat-Ek) recordings for music generation, and 48-classes Thai food image dataset [7]. Other open source datasets that were not specific to the local community include farm-vs-wild salmon classification dataset [8] and artwork commission price estimation dataset [9]. These datasets allowed students to create projects that relate to their backgrounds and interests. Additionally, they also contributed to the growing Thai ML community.

[4] https://www.facebook.com/groups/thainlp/
[5] https://www.facebook.com/groups/dsbkkgroup/
[6] https://www.kaggle.com/c/makpinlom/
[7] https://www.kaggle.com/somboonthamgemmy/foodydudy
[8] https://github.com/kangkengkhadev/salmon_prediction
[9] https://github.com/pradrattana/anime_commission_suggested_price

G. Post program engagement

To increase public awareness, we hosted a public presentation session for students who passed the evaluation following by a series of student projects on our Facebook page after the program. We also helped pair students with researchers and industry partners in Thailand for students who would like to continue working on their projects.

IV. Evaluation

We evaluated our program by the student's ability to understand course materials, their attendance, and their project.

A. Student's understanding

After the program, we asked students to rate their understanding after recap classes on a scale of one to five. According to optional after-class surveys across all eight lessons, we saw that they understand the content more than the original lectures (paired t-statistic (53) = 3.8517, $p < 0.001$). The average ratings of understanding the course materials rose from 4.26 to 4.64 out of 5 according to surveys at the end of each class. Recapping classes in native language provided a deeper understanding of course materials and made courses more engaging. All students who were assigned to groups with mentors attend the recap classes. However, only three auditors attended all 8 weeks of lesson recaps.

B. Evaluation criteria for program completion

We wanted final projects to be easily reproducible and easily understood by a wide audience. Therefore, we asked them to submit a Github repository, short essays describing their projects, seven minutes presentations followed by three minutes questions, and a web application or simple deployment. Mentors evaluated each project based on the following criteria:

- Problem statement (15%) - The project solves a real-world problem in the contexts of the students.
- Metrics and baselines (15%) - The project has well-defined metrics that determine what success in solving the problem looks like. It also benchmarks those metrics with existing ML and non-ML solutions.
- Data collection and cleaning (15%) - The project has a thorough and reproducible process to develop the data.
- Exploratory data analysis (20%) - The project shows understanding about the data for solving the problem through visualization and other means of exploration.
- Modeling and error analysis (20%) - The project has properly trained models with train-validation-test splits. Error analysis is sufficiently performed.
- Prototype deployment (15%) - The project can be used by a person with little-to-no technical expertise such as a Streamlit web app deployed on free-tier Heroku instances. It does not need to be production-ready.

We asked mentors to score each project and calculate an average of the scores. The average score was then reviewed in an all-hands meeting of mentors to calibrate between-group scoring biases. Students who got over 70% are eligible for the certificate of completion. Most students, 20 out of 26,

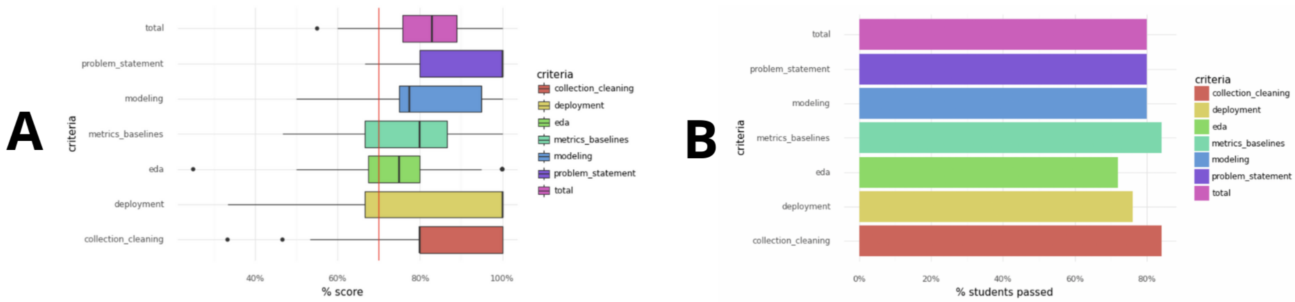


Fig. 4: Project evaluation scores by evaluation criteria. A. Score distribution for each evaluation criterion. B. Proportion of students who passed the 70% threshold for each criterion.

who submitted the project passed the evaluation criteria where 19 projects were students with mentors and 1 from auditors. Students who did not pass our criteria got low scores in exploratory data analysis and deployment (Figure 4).

C. *Final projects*

Student final projects ranged from Thai text-to-speech synthesis, wild-vs-farm salmon classification, object detection using Light Detection and Ranging (LiDAR) point cloud [23], stock prediction, generation of Thai instrument sounds (Ranat Ek), Thai food classification, Thai-Chinese machine translation, and enhanced Thai optical character recognition (OCR) (Figure 5). 8 out of 25 students used Fastai in their projects. Some projects cannot be done with Fastai such as speech and music synthesis, machine translation, and object detection using LiDAR dataset. This indicates that we need to prepare other tools and tutorials for more diverse use cases. In the future, we consider incorporating the HuggingFace library and course [24] for NLP use cases and NVIDIA NeMo library [10], which has speech processing and synthesis capabilities.

V. Discussions

Here, we presented AI Builder, a nine-week project-based summer school aiming to teach Thai students ML and AI. Most students were able to understand ML concepts and produced meaningful ML projects by the end of our school. We discussed program structure, class format, projects, and evaluation process.

Although we adopt Fastai [16] for our main curriculum, we noticed that the Fastai framework is only well-designed for a set of specific use cases such as image classification, recommendation, and text classification. It also limited students' project development. The course introduces image classification early but discussed tabular data and NLP in later chapters. Students who work on image classification can work on their projects much earlier whereas those who work on NLP do not learn their tools until the last week. This is reflected in submitted projects where almost all image-based projects use Fastai while NLP and other projects do not. One possibility is to redesign the curriculum to be track-based, teaching students common core knowledge then separate them into tracks depending on data and projects. We can introduce fundamental ML such as gradient descent and basic neural network during the first few weeks then allow them to choose tracks according to the types of data such as image, tabular data, NLP, and speech or signal. Additionally, we can group mentors and match with students based on their interests [8].

A lack of sufficient social interactions among students can still be improved. All of the feedback we got from the optional after-program surveys mentioned that having opportunities to socialize would improve the program. While a few participants take advantage of the Discord server to actively socialize, the majority of the participants are not quite active. Guest lectures and a mid-term in-class Kaggle competition are held as optional events to foster social interactions. However, we did not receive much participation outside of those who have already been active on Discord. We plan to implement more ice-breaking activities such as group discussion or short project proposal. We also consider having them work in pairs or small groups during our school.

There are rooms to improve efficiency of the program. For example, we had one person manually going over and scored all take-home application exams. Automated code can greatly speed up our screening process as well as preventing human errors. For effectiveness, we observe that students who did not pass the projects either did not have a clear project proposal or changed their project drastically mid-program. We hypothesize that a clear project proposal is a good predictor of project success where we may incorporate it for selection in our next iteration. However, more data is needed to understand the success of the students.

Geographic diversity of applications is another issue we want to improve. Currently, we received more applications from Bangkok and its perimeter than all other provinces combined. This is likely that we only advertise through only a selection of Facebook pages and community groups. In the future, we will work towards more effective channels to gain more students by reaching out to local schools or universities in different regions. We hope these initiatives can bring more applications from provinces outside Bangkok.

Computational resources are one of the major obstacles for some projects. Currently, students use free-tier notebooks such as Google Colaboratory or Kaggle to train their models. However, some projects need more resources to finish in

[10]https://github.com/NVIDIA/NeMo

a reasonable timeline. For instance, Chinese-Thai machine translation models requires training on a large instance with a graphic processing unit (GPU) provided by a mentor. In future iteration, we plan to provide students more access to computational resources from our partners and sponsors.

Scaling the program while maintaining the teaching quality is one of the biggest challenges in our next iteration. The mentor-to-student ratio of 1:2 seems to be the limit for mentors with the current setup. Due to our mentor limitations, we can only take in 25 students and 21 auditors out of 130 applicants who passed the take-home test. Recruiting more mentors for the program and providing additional stipend could mitigate this issue. Content availability does not seem to be the bottleneck since we make our course materials and videos publicly available. Still, we could not retain attendance for most auditors. One possibility is that auditors can watch the recap classes anytime since they do not have to attend the mentor sessions. Those who attended all lessons reported discussing with mentors, other participants, or guest lectures as the main reasons. One solution is to assign the auditors to mentors which allow them to be more engaged and thus attend more classes.

Finally, we want to build a sustainable school and community. We need to support students after completion as well. We connected some students, whose projects had clear next steps towards a research publication or industry use cases, with local researchers and practitioners through our partner organizations. Although most university admission in Thailand uses a centralized process based on standardized tests, some universities are considering more portfolio-based admission. We are actively exploring universities and research partners where our projects can be considered as additional materials for university admission. We hope that students can have a foundation to contribute the Thai and global ML community in the future.

VI. Conclusions

We presented AI Builder, a nine-week project-based summer school aiming to teach Thai students ML and AI. Most students were able understand ML concepts and completed their projects by the end of the program. Many projects provided meaningful contributions to Thai ML community in terms of datasets, models, and code. We discussed potential improvement for our future iterations.

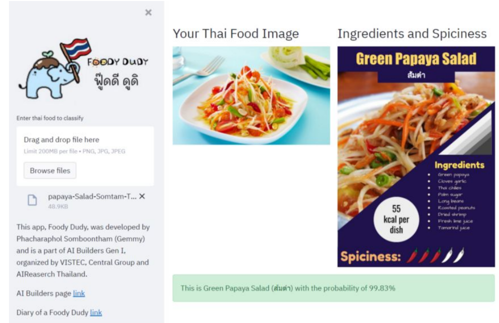
Acknowledgment

We would like to thank the guest lecturers including Kanchit Jeng Rongchai, Attapol Thamrongrattanarit-Rutherford, Thiparat Chotibut, Thanachart Ritbumroong, Virot Chiraphadhanakul, and web personality Uncle Engineer. We thank those who contributed to the pre-course resources including Abhabongse Janthong and Nattapol Vannaboot. We thank Kanchit Jeng Rongchai and Yingboon Chongsomchai for the Mak Pin Lom dataset. We thank the staffs from Central Technology Organization who helped run the program including Kridakan Phinidchai and Pajaree Muensuwan. We thank our website contributors, developer Tulakan Ruangrong, and logo designer Phannisa Nirattiwongsakorn.

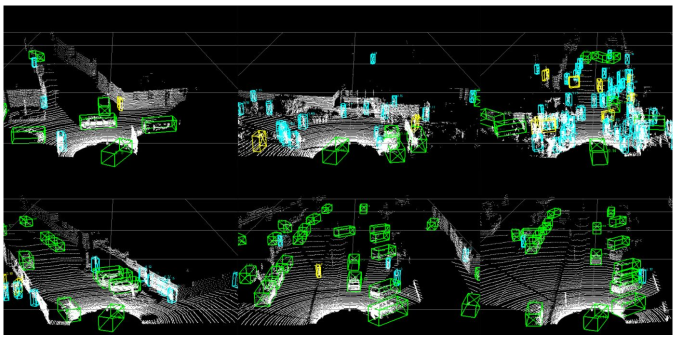
References

[1] A. Booth, "Education and economic development in southeast asia: Myths and realities," *ASEAN Econ. Bull.*, vol. 16, no. 3, pp. 290–306, 1999.
[2] A. S. Sadiman, "Challenges in education in southeast asia," in *international seminar on 'Towards Cross Border Cooperation between South and Southeast Asia: The importance of India's North East playing bridge and buffer role', Kaziranga, India*, 2004, pp. 16–19.
[3] S. Palvia, P. Aeron, P. Gupta, D. Mahapatra, R. Parida, R. Rosner, and S. Sindhi, "Online education: Worldwide status, challenges, trends, and implications," 2018.
[4] W. Feng, J. Tang, and T. X. Liu, "Understanding dropouts in moocs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 517–524.
[5] T. J. Blayone, W. Barber, M. DiGiuseppe, E. Childs *et al.*, "Democratizing digital learning: theorizing the fully online learning community model," *International Journal of Educational Technology in Higher Education*, vol. 14, no. 1, pp. 1–16, 2017.
[6] O. B. Adedoyin and E. Soykan, "Covid-19 pandemic and online learning: the challenges and opportunities," *Interactive Learning Environments*, pp. 1–13, Sep. 2020.
[7] T. K. F. Chiu, "Student engagement in K-12 online learning amid COVID-19: A qualitative approach from a self-determination theory perspective," *Interactive Learning Environments*, pp. 1–14, May 2021.
[8] T. van Viegen, A. Akrami, K. Bonnen, E. DeWitt, A. Hyafil, H. Ledmyr, G. W. Lindsay, P. Mineault, J. D. Murray, X. Pitkow *et al.*, "Neuromatch academy: Teaching computational neuroscience with global accessibility," *Trends in cognitive sciences*, 2021.
[9] T. Achakulvisut, T. Ruangrong, I. Bilgin, and others, "Point of view: Improving on legacy conferences by moving online," *Elife*, 2020.
[10] T. Achakulvisut, T. Ruangrong, P. Mineault, T. P. Vogels, M. A. K. Peters, P. Poirazi, C. Rozell, B. Wyble, D. F. M. Goodman, and K. P. Kording, "Towards democratizing and automating online conferences: Lessons from the neuromatch conferences," *Trends Cogn. Sci.*, vol. 25, no. 4, pp. 265–268, Apr. 2021.
[11] P. A. Bozelos and T. P. Vogels, "Making (neuro) science accessible world-wide: Online seminars for the globe," *Elife*, 2020.
[12] S. Sarabipour, "Research culture: Virtual conferences raise standards for accessibility and interactions," *Elife*, vol. 9, p. e62668, 2020.
[13] N. Norouzi, S. Chaturvedi, and M. Rutledge, "Lessons learned from teaching machine learning and natural language processing to high school students," *AAAI*, vol. 34, no. 09, pp. 13 397–13 403, Apr. 2020.
[14] T. K. F. Chiu, H. Meng, C.-S. Chai, I. King, S. Wong, and Y. Yam, "Creation and evaluation of a pretertiary artificial intelligence (AI) curriculum," *IEEE Trans. Educ.*, pp. 1–10, 2021.
[15] D. N. Perkins, *Making Learning Whole: How Seven Principles of Teaching Can Transform Education*. John Wiley & Sons, Oct. 2010.
[16] S. Gugger and J. Howard, *Deep Learning for Coders with Fastai and PyTorch: AI Applications Without a PhD*. O'Reilly Media, Jun. 2020.
[17] J. Howard and S. Gugger, "Fastai: A layered API for deep learning," *Information*, vol. 11, no. 2, p. 108, 2020.
[18] J. Bergmann and A. Sams, *Flip Your Classroom: Reach Every Student in Every Class Every Day*. International Society for Technology in Education, Jun. 2012.
[19] L. Lowphansirikul, C. Polpanumas, A. T. Rutherford, and S. Nutanong, "A large English–Thai parallel corpus from the web and machine-generated text," *Lang. Resour. Eval.*, Mar. 2021.
[20] K. Chaimooltan, "TH-National-Document-OCR," Jul. 2021.
[21] M. Hodosh, P. Young, and J. Hockenmaier, "Framing image description as a ranking task: Data, models and evaluation metrics," *J. Artif. Intell. Res.*, vol. 47, pp. 853–899, Aug. 2013.
[22] C. Wutiwiwatchai, S. Saychum, and A. Rugchatjaroen, "An intensive design of a thai speech synthesis corpus," in *in International Symposium on Natural Language Processing (SNLP 2007)*, 2007.
[23] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 697–12 705.
[24] T. Wolf, L. Debut *et al.*, "Transformers: State-of-the-Art natural language processing," 2020.

48-class 14k-image Thai food classification dataset



Thai food classification web application built with Streamlit



Point cloud object detection using KITTI dataset

	Method	Modality	Speed	Easy	Car Moderate	Hard
BEV	PointPillars	LiDar	62	88.35	86.1	79.83
	Mine	LiDar	~60-65	89.66	87.17	84.38
3D	PointPillars	LiDar	62	79.05	74.99	68.3
	Mine	LiDar	~60-65	86.46	77.28	74.65

Benchmark results for point cloud object detection

BLEU	zh->th_syllable	zh->th_word	th->zh
Google Clone Translate API	23.903	13.699	21.448
AIforThai (xiaofan)	6.74	4.271	1.564
Ours	15.945	8.425	15.535

Benchmark results of Chinese (zh) - Thai (th) and Thai-Chinese machine translation using word- and syllable-level BLEU

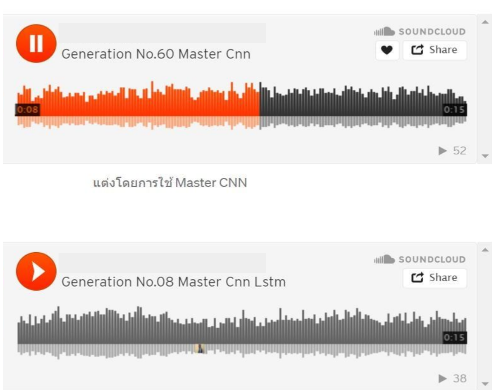
ตัวอย่างประโยคแปล ไทย→จีน

(สุ่มจาก test set ด้วย seed = 42)

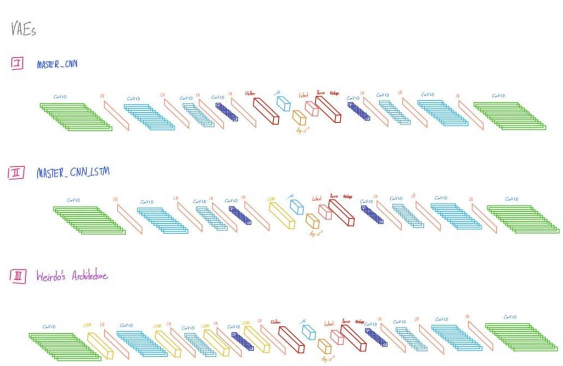
[Source] ฉันประเมินหรือเปล่า ว่าอะไรจะให้ผลที่น่าพึงพอใจที่สุด เหมือนกับที่มิลล์จะทำ?
[Target] 我有没有像米尔会做的那样 衡量最大的享受? [line 981]
[Ours] 我评估了吗?什么是最满意的结果,就像马尔科姆所做的那样?"
[AI for Thai] 你知道吗?""我想给你写一份""+""就行了。
[Google] 我评估了吗？什么会产生最令人满意的结果？就像米尔会做的那样？

[Source] จะว่าเป็นความฝัน ก็คงเห็นจะไม่เข้าที
[Target] 把这种现象视作梦的一种是行不通的 [line 1009]
[Ours] 这将是一个梦想,永远不会出现
[AI for Thai] 你就知道吧。
[Google] 不管是不是梦我可能不会看到它

Examples of Thai-Chinese translation compare to other benchmark such as AI for Thai and Google translate



Thai Instrument (Ranat Ek) Sound Generation with Variational Autoencoder



Architectures implemented include CNN, CNN-LSTM, and a custom architecture

Fig. 5: A showcase of the student final projects. We show some of the student final projects including Thai food classification, Thai-Chinese machine translation, 3D Object detection using LIDAR point cloud data, and generation of Thai instrument (Ranat-Ek) sound.