Naming Survey Protocol

***Title**: "I see models being a whole other thing": An Empirical Study of Pre-Trained Model Naming Conventions and A Tool for Enhancing Naming Consistency*

***Recruitment**: In the study, we will recruit the users of Hugging Face PTMs, including PRO and normal accounts.*

***Compensation**: We will provide financial compensation to 3rd-party team participants. We will incentivize survey participants through a $10 gift card.*

Research Questions

Survey RQs:

- **RQ1**: How is PTM naming different from traditional software package naming?
- **RQ2**: What elements should be included in a PTM identifier?
- **RQ3**: How do engineers identify naming inconsistencies?

Discussion:

What **improvements** can be made to model registry infrastructures (e.g. Hugging Face), to enhance searchability and reuse of model names?

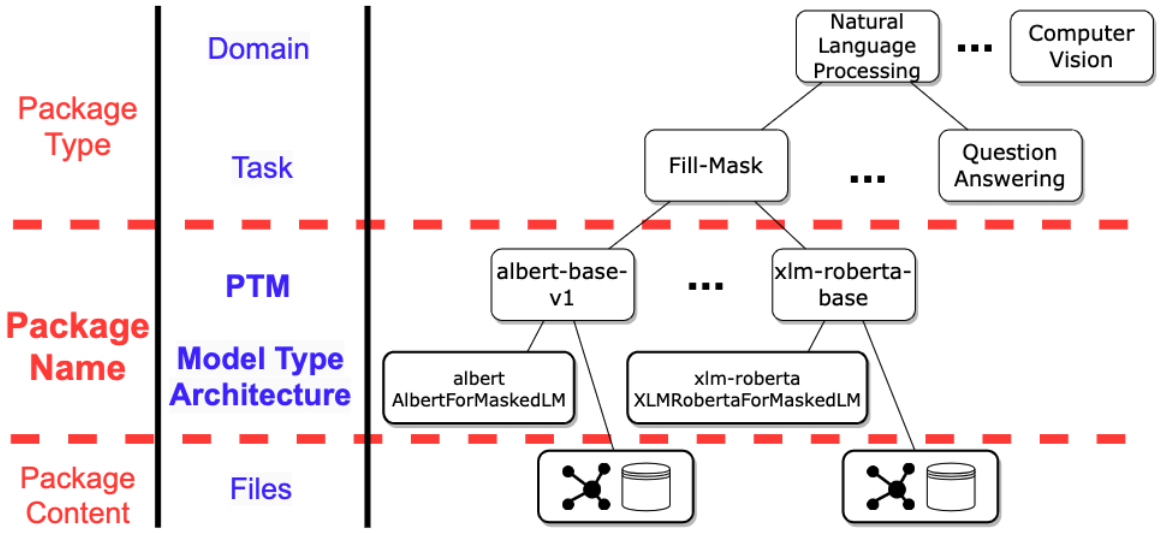
Questions

Demographic Questions

1. How many years have you worked on ML?
    a. 1 - 2 years
    b. 3 - 5 years
    c. 6 - 10 years
    d. 11 - 20 years
    e. > 20 years
2. How many years have you worked on SE?
    a. 1 - 2 years
    b. 3 - 5 years
    c. 6 - 10 years
    d. 11 - 20 years
    e. > 20 years
3. What is the size of your organization?
    a. Small (1 - 50 employees)
    b. Medium (51 - 250 employees)
    c. Large (251 - 1000 employees)
    d. Very large (1001+ employees)
4. What deployment contexts do you work on?
    a. Web application
    b. Desktop
    c. Cloud and data center
    d. IoT/embedded systems
    e. Mobile devices
5. How many pre-trained model (PTM) packages have you used from model registries: (e.g. *Hugging Face, Pytorch hub):*
    a. 0
    b. 1 - 5
    c. 5 - 10
    d. 10 - 20
    e. > 20
6. How many PTM packages have you created (contribute + give names) in model registries (e.g. Hugging Face, Pytorch hub):
    a. 0
    b. 1 - 5
    c. 5 - 10
    d. 10 - 20

e. > 20

(Definition of PTM Naming Provided here)



The user-controllable names are PTM identifier, model type and architecture. In this study, we define a PTM **package name** as the combination of a package ***identifier*** (e.g., albert-base-v2, facebook-llama/Llama-2-7b-chat-hf) and the ***model type*** or ***architecture*** indicated in the metadata (e.g., albert, AlbertForMaskedLM)

Comparison to Trad. Software

1. (RQ1) How is PTM naming similar/different from naming traditional software packages such as those on NPM or PyPi? Why do you think that is? (Text box)

Naming practices

1. (RQ2) Which naming convention do you prefer when reusing PTM from model registries like Hugging Face? [Henninger 1994] (*multi-checkbox*)
    a. Named by task: high-level category (e.g. question-answering)
    b. Named by application: what a PTM does (e.g. fake-news-detector, text2image-prompt-generator)
    c. Named by implementation: what a PTM is (e.g. bert-base-uncased)

    d. Named by “Implementation + task” (e.g. *Llama-2-7b-chat-hf*)
    e. Named by “Implementation + application” (e.g. *distilroberta-base-finetuned-fake-news-detection*)
    f. Others (text box)
2. (RQ2) Here is a list of PTM naming elements. Check each box if you think it would be important to include that element in a name. - Selected Choice
    a. Architecture
    b. Model size
    c. Dataset
    d. Model versioning
    e. Language
    f. Task
    g. Adaptation method
    h. Training regime
    i. Application goal
    j. Number of (hidden) layers (e.g. L-12, H-128)
    k. Number of parameters
    l. Dataset characteristics
    m. Others (please specify)
3. (RQ2) Machine learning models are often adapted for improved performance or specific needs. These modifications might involve changing the architecture, training regime, and dataset. What kinds of modifications might necessitate a new model type/architecture for the model ("distilBERT"), as opposed to continuing to hyphenate the name ("albert-v2-50M") (For example, DistilBERT and Bert are separate model architectures on HuggingFace) - Selected Choice
    a. Input/Output layers
        i. Modified tensor shape in Input/Output layers
        ii. Addition/deletion of layers in Input/Output layers
    b. Main body of the architecture
        i. Modified layers in the main body of the architecture (e.g. dropout rates, activation functions)
        ii. Addition/deletion of layers in the main body of the architecture
    c. Modified trianing regime of the PTM
    d. Changed training dataset
    e. Other (*text box*)

Naming Challenges

4. (RQ2) In your experience, do the PTMs available in model registries accurately describe their behavior/content? What discrepancies have you experienced? Please explain
    a. Yes (short answer for each attribute)
        i. Architecture (e.g. bert, resnet):
        ii. Model size (e.g. base, large, 50, 101)
        iii. Dataset (e.g. squad, imagenet)
        iv. Model versioning (e.g. v1, v2)
        v. Language (e.g. en, English, Arabic)
        vi. Task (e.g. question-answering, qa)
        vii. Adaptation method (e.g. finetine, distill, fewshot)
        viii. Training regime (e.g. pretrain, sparse)
        ix. Number of (hidden) layers (e.g. L-12, H-128)
        x. Number of parameters (e.g. 100M, 8B)
        xi. Dataset characteristics (e.g. case, uncased, 1024-1024)
        xii. Others (please specify)
    b. No (Which component/factor could be inaccurate based on your experience?)
5. (RQ3) In your regular practice, do you think you would notice if a PTM had an incorrect name? Check the box if you think YOU WOULD NOTICE if the naming element were incorrect. - Selected Choice
    a. Architecture
    b. Model size
    c. Dataset
    d. Model versioning
    e. Language
    f. Task
    g. Reuse method
    h. Training process
    i. Number of (hidden) layers (e.g. L-12, H-128)
    j. Number of parameters
    k. Dataset characteristics
6. If you think you would typically notice one or more of these naming elements being incorrect, please tell us what process you follow to do so (e.g. reading model card, reading source, etc.)

ONNX Questions

1. Which framework do you use for model development?
    - PyTorch
    - TensorFlow
    - JAX/FLAX
    - MLX
    - Other
2. Do you use ONNX as part of your model development and deployment process?
3. Other than ONNX, do you use other interoperability tools? (check all that apply) - Selected Choice
    - MMdnn
    - NNEF
    - Other
4. For what purposes do you use ONNX?
    - Framework-to-framework Model Conversion (e.g., converting from a model from TensorFlow to PyTorch)
    - Model Conversion for Deployment (e.g., converting to ONNX for deployment using ONNXRuntime or TensorRT.
    - Other (please specify)
5. If you use ONNX for framework-to-framework conversion, could you please describe your use case further and why do you use ONNX?
6. Do you ever deploy directly from a deep learning framework such as PyTorch or TensorFlow? If so, what do you consider when choosing between deploying from a DL framework vs. via a tool like ONNX?
7. Do you commonly encounter problems while working with ONNX models? - Selected Choice
    - Crashes (e.g., Model does not convert to ONNX.)
    - Performance Differences (e.g., the accuracy of the ONNX model does not match the original model)
    - Other (please specify)
8. If you encounter such problems, how do you address them?

Recruitment message

Subject: Survey on Machine Learning Model Naming Conventions

Content:

Dear Hugging Face contributor,

I hope this email finds you well. I am studying engineering practices when re-using pre-trained models (PTMs).

I would appreciate your help in collecting some data. I am conducting a survey to understand engineers' perspectives on PTM naming conventions and on interoperability tools.

- My survey takes ~5 minutes. All data will be anonymized to ensure privacy.
- You would be compensated for your time with a $10 Amazon gift card.

Here is the link to the survey: [link]

If you have colleagues who might have relevant opinions, please feel free to forward to them.

This study is supported by the ANONYMOUS SOURCE, and has been approved by our institution's IRB (#xxxx-xxx).

Questions? Contact me (Author [email]).

Thank you for your consideration.

Best regards,

Author