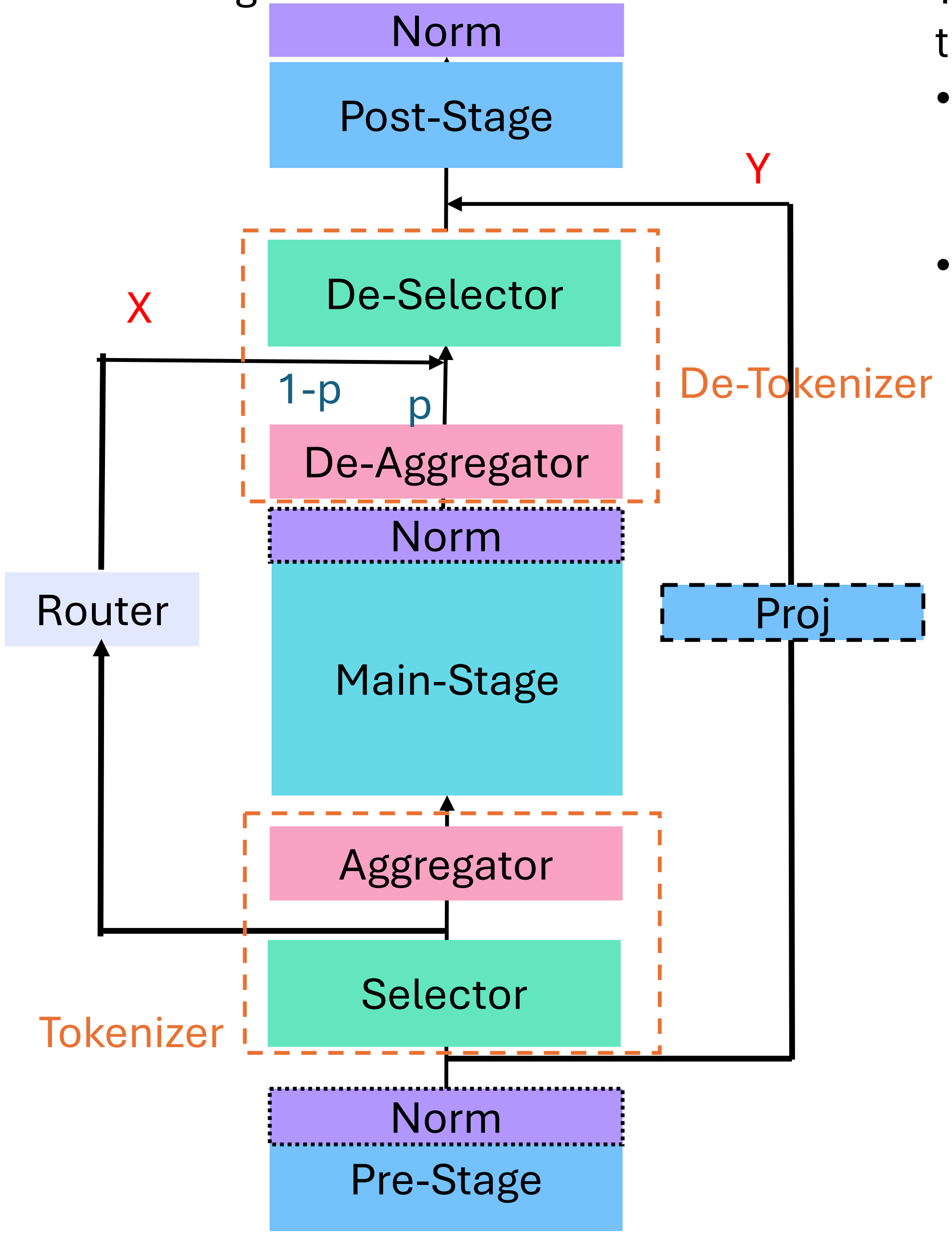
Dec27 generalization



This design puts the p-weights earlier rather than later (X rather than Y). Why this design?

- The architecture design is much more symmetric, and every branch has its own interpretation (see below). The tokenizer, router, and de-tokenizer layers together form the core modules that we propose.
- Consider what the architecture looks like in the ideal case when p goes to 1.
    - In this case, the residual happens at point Y compared to the previous design (where the p multiplication happens at point Y), in which case the residual happens at point X.
    - In this case, the Deselector layers can be interpreted as **detokenization**, *without* supervision from the original token identities. This is in line with our intuition where detokenization should largely be possible without information from the higher resolution stage.
    - The X and Y branches play two separate roles.
        - The X-branch is purely for routing and learning the p values. The 1-p multiplication on the residual should help encourage stronger signals to learn p.
        - The Y-branch is interpreted as the U-net connection that provides a signal from the finer grain resolution. This is completely independent of the tokenizer/routing modules.
        - Remember previously, the 1 – p weighting was problematic because it encouraged p away from 1 (because some residual signal was needed). The separate residual Y branch should