pub mod validatior {

pub mod cli\_validator {

use crate::cli\_args::{ARGS, MODEL\_SPECS};

pub fn validate\_input\_args() {

assert\_eq!(MODEL\_SPECS.len(), ARGS.inter\_op\_parallelism.len());

assert\_eq!(MODEL\_SPECS.len(), ARGS.intra\_op\_parallelism.len());

//TODO for now we, we assume each model's output has only 1 tensor.

//this will be lifted once tf\_model properly implements mtl outputs

//assert\_eq!(OUTPUTS.len(), OUTPUTS.iter().fold(0usize, |a, b| a+b.len()));

}

pub fn validate\_ps\_model\_args() {

assert!(ARGS.versions\_per\_model <= 2);

assert!(ARGS.versions\_per\_model >= 1);

assert\_eq!(MODEL\_SPECS.len(), ARGS.input.len());

assert\_eq!(MODEL\_SPECS.len(), ARGS.model\_dir.len());

assert\_eq!(MODEL\_SPECS.len(), ARGS.max\_batch\_size.len());

assert\_eq!(MODEL\_SPECS.len(), ARGS.batch\_time\_out\_millis.len());

}

}

}