```mermaid
PART D: hgraph2graph
Training Process

Input SMILES String
  ↓
setup Data Representation
  ↓
Extract Fragments (Rings/Bonds)
  ↓
Build Junction Tree (Hierarchical)
  ↓
Build Atom-level Graph
  ↓
Generate Alignments (Map atoms to clusters)
  ↓
process Encoder (hgap_encoder.py)
  ↓
Compute Atom/Bond Embeddings (Graph MPN)
  ↓
Propagate Messages (Tree-GRU)
  ↓
Compute mu and log_var
  ↓
Sample Latent Vector z (Reparameterization)
  ↓
process Decoder (hgap_decoder.py)
  ↓
Coarse-to-Fine Order ┄┄→ Predict Tree Topology (BFS Node/Edge)
  ↓
Predict Attachment Points (Orientation)
  ↓
Predict Graph Reconstruction (Final Bonds)
  ↓
group Objective & Optimization
  ↓
Training
  ├─ True → Apply Teacher Forcing
  │           ↓
  │         Calculate Reconstruction Loss
  │           ↓
  │         Calculate KL Divergence Loss
  │           ↓
  │         Loss Computed?
  │           ├─ True → Backpropagate with Adam
  │           │           ↓
  │           │         Update Checkpoint
  │           │           ↓
  │           └─ False → evaluate Validation Accuracy
  │                       ↓
  │                     (back to group Objective & Optimization)
  └─ False → Trained Model Ready
```