What You Can Do

1. **Condition on a Single Property:**
   If your goal is to condition on a band gap value, change your command to use the model **dft_band_gap**:

bash

Copy

```
export MODEL_NAME=dft_band_gap

export RESULTS_PATH=results/

mattergen-generate $RESULTS_PATH --pretrained-name=$MODEL_NAME --batch_size=16 --num_batches=1 --properties_to_condition_on="{'dft_band_gap': 3.5}"
```

2. **Joint Conditioning:**
   If you need to condition on energy above hull, use the appropriate model—for instance, **chemical_system_energy_above_hull** if you're also providing a chemical system:

bash

Copy

```
export MODEL_NAME=chemical_system_energy_above_hull

export RESULTS_PATH=results/

mattergen-generate $RESULTS_PATH --pretrained-name=$MODEL_NAME --batch_size=16 --num_batches=1 --properties_to_condition_on="{'chemical_system': 'Li-O', 'energy_above_hull': 0.05}"
```

3. **Custom Joint Conditioning:**
   If you truly need to condition on both dft_band_gap and energy_above_hull together, you'd have to either train a custom model with both conditioning fields or use an available joint conditioned model that supports those exact keys. As of the released models, the joint models are:
   - dft_mag_density_hhi_score (joint on magnetic density and HHI score)
   - chemical_system_energy_above_hull (joint on chemical system and energy above hull)
     There isn't a model pre-trained for joint conditioning on dft_band_gap and energy_above_hull.

In Summary

The assertion error is a safeguard: it ensures that you only pass keys that the chosen model was trained to condition on. To resolve the error, choose the appropriate pre-trained model (by setting --pretrained-name) that matches your conditioning prompt. If you wish to condition on a property like band gap, use the dft_band_gap model instead of mattergen_base.

For battery material development, several complementary approaches can be valuable:

1. **MatterGen & MatterSim Workflow:**
   MatterGen is especially powerful for generating novel candidates by directly "imagining" crystal structures that meet your design criteria. For battery anodes, cathodes, and electrolytes, you can fine-tune MatterGen on property-labeled datasets (e.g. targeting high ionic conductivity for electrolytes or optimized redox potentials for cathodes) and then use MatterSim to rapidly relax and evaluate these candidates. This integrated AI workflow helps you explore materials space beyond existing databases.

2. **High-Throughput Screening with Databases:**
   Complementary to generative approaches, using high-throughput screening from databases like the Materials Project, OQMD, or NOMAD can help identify known candidates. Combining these with machine learning property predictors (for stability, conductivity, etc.) further refines your candidate list.

3. **Graph-Based Models (e.g. DeepMind's GNoME):**
   Some recent methods, like DeepMind's GNoME, use graph neural networks to predict the stability of candidate structures on a massive scale. These tools can serve as an alternative or complementary screening approach, especially for battery-relevant chemistries.

4. **Custom AI/ML Pipelines:**
   You can develop your own ML models (using frameworks such as PyTorch or TensorFlow) tailored to battery performance metrics by training on curated datasets. For instance, a generative model (diffusion model, VAE, or GAN) fine-tuned with battery-specific property data can target features like electrode potential, ionic conductivity, or thermal stability.

In Practice

- **For Anodes & Cathodes:**
  Focus on models (or prompts) that target parameters such as high electronic conductivity, specific redox potentials, and structural stability. MatterGen can be conditioned on properties like formation energy or band gap to propose candidates with balanced performance.

- **For Electrolytes:**
  Emphasize ion mobility and chemical stability. Fine-tune your generative model using data on ionic conductivity and electrochemical stability windows, then validate candidates with simulations or experimental collaboration.

Recommended Approach

For a research pipeline, a good strategy is to use MatterGen for candidate generation under battery-specific prompts and integrate it with simulation tools (like MatterSim or DFT workflows) for validation. This combination leverages the speed of AI generation and the accuracy of computational screening, ultimately accelerating discovery for battery anodes, cathodes, and electrolytes.