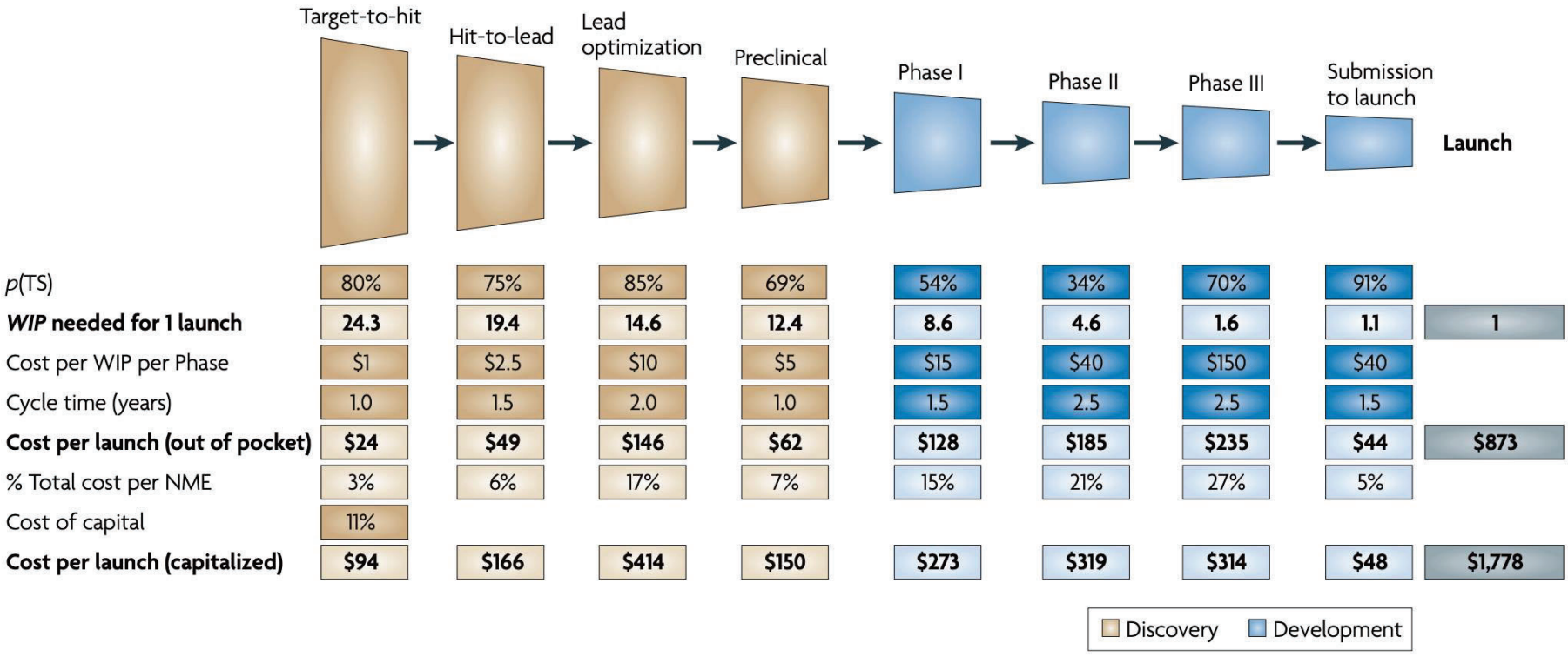
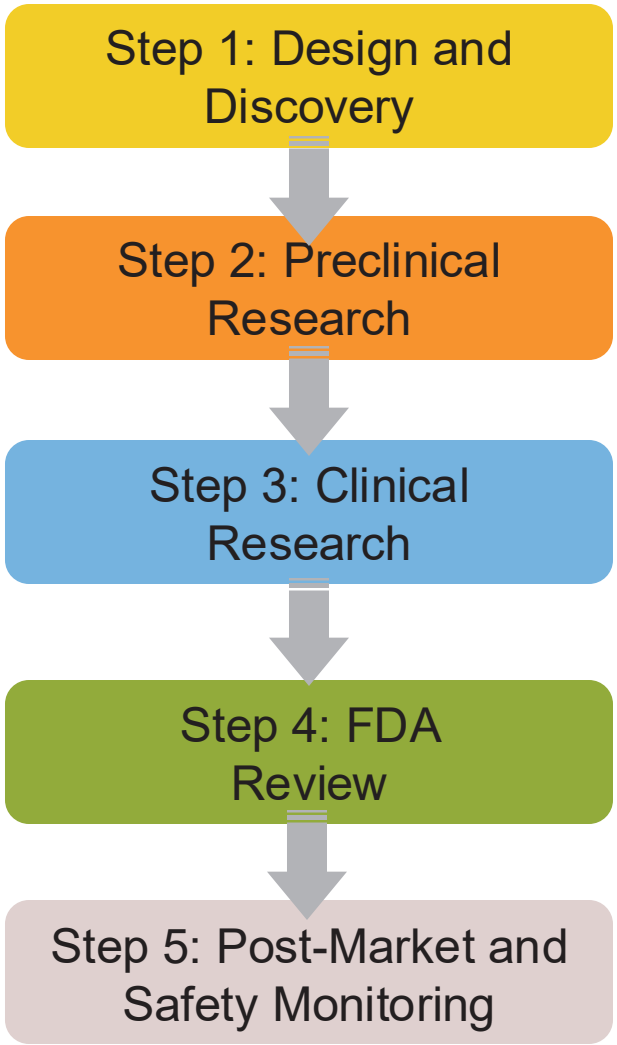
Phases of drug discovery from initial stage (target-to-hit) to final stage (launch)

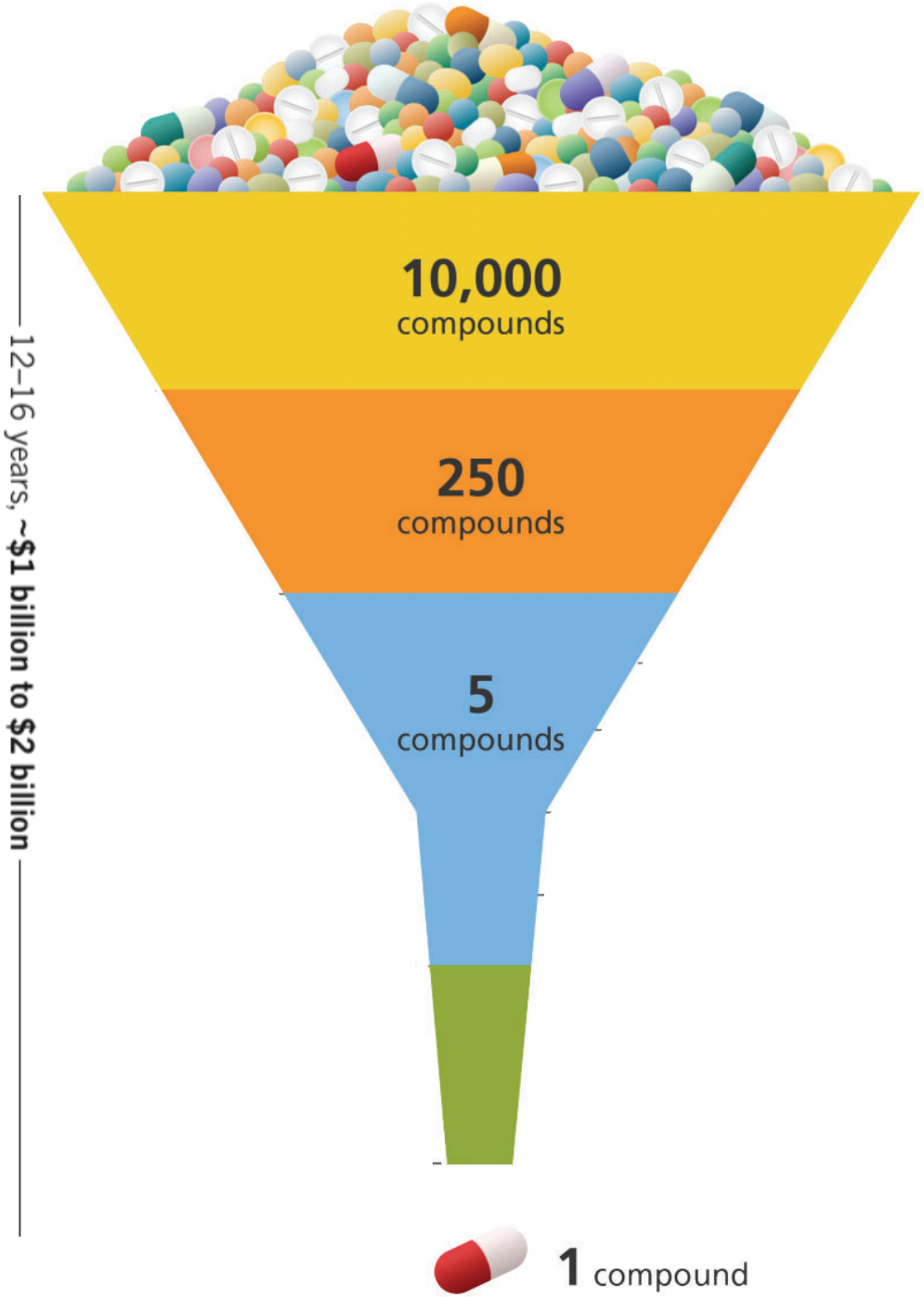


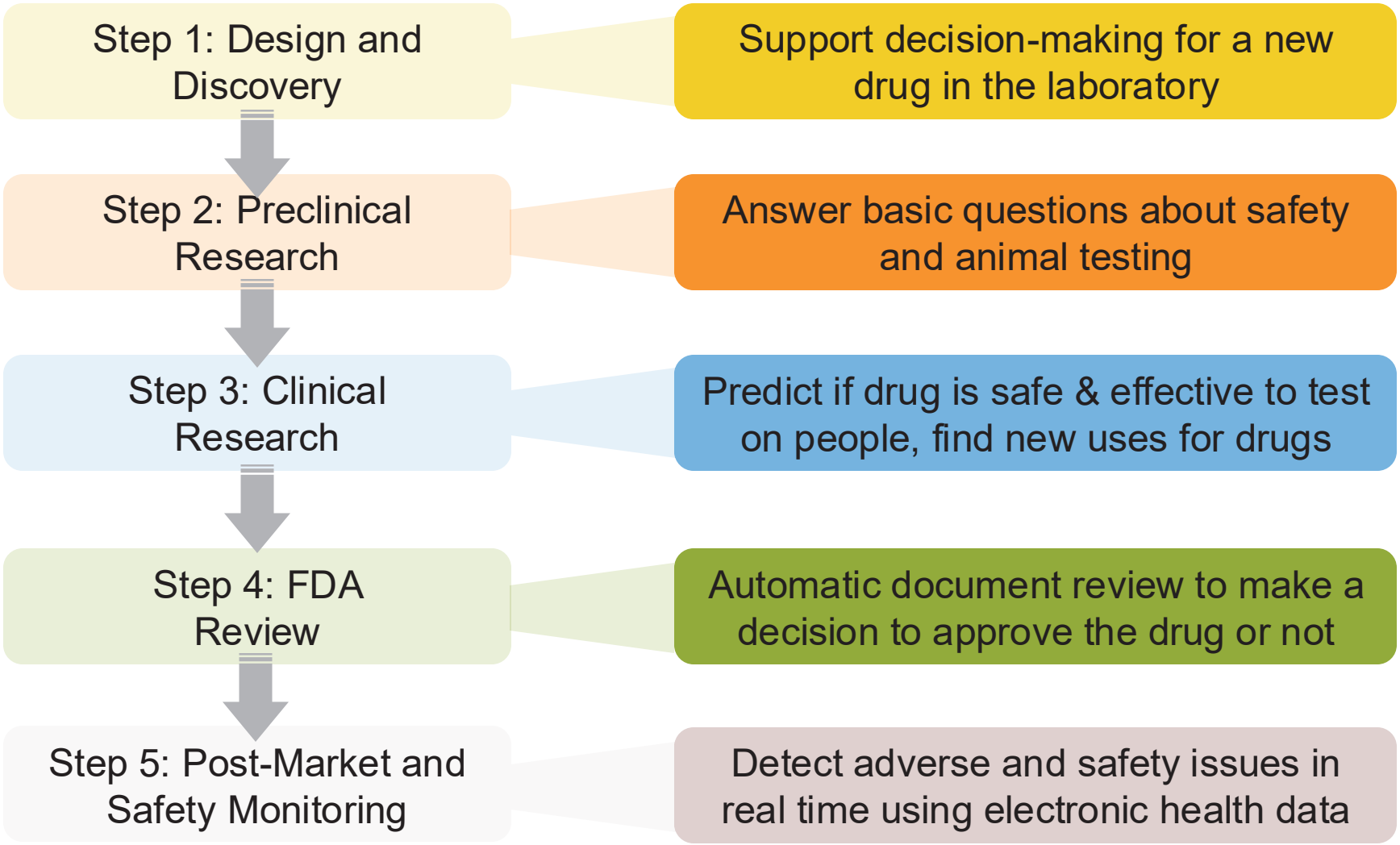
p(TS) – probability of successful transition from one stage to the next; NME – new molecular entity; WIP – work in process

Drug-like chemical space
10^{60} chemical compounds

Drugs available
to humans
~10^4 •

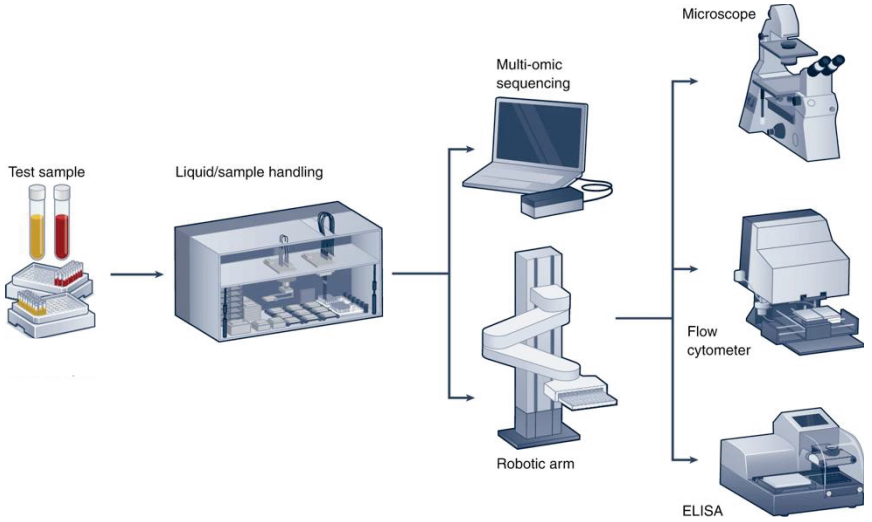

Step 1: Design and Discovery
Step 2: Preclinical Research
Step 3: Clinical Research
Step 4: FDA Review
Step 5: Post-Market and Safety Monitoring


10,000 compounds
250 compounds
5 compounds
1 compound
12–16 years, ~$1 billion to $2 billion


Step 1: Design and Discovery
Support decision-making for a new drug in the laboratory
Step 2: Preclinical Research
Answer basic questions about safety and animal testing
Step 3: Clinical Research
Predict if drug is safe & effective to test on people, find new uses for drugs
Step 4: FDA Review
Automatic document review to make a decision to approve the drug or not
Step 5: Post-Market and Safety Monitoring
Detect adverse and safety issues in real time using electronic health data

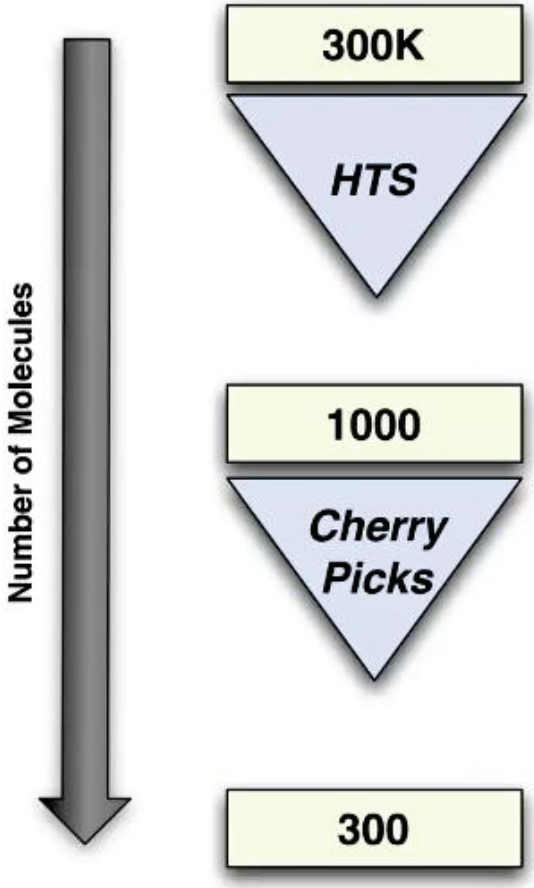
High throughput screening (HTS)

- Test thousands to hundreds of thousands of compounds in one or more assays
  - Biochemical, genetic, and pharmacological assays
- Integrate with robotics for self-driving lab
- **Goal:** Rapidly identify novel modulators of biological systems
  - Cellular basis of diseases
  - Therapeutic agents



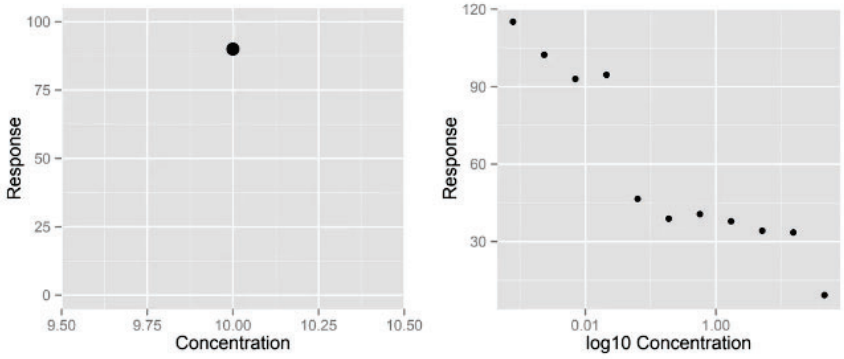
Goals of high throughput screening

- Rapidly screen large collections of compounds (chemical libraries)
- Efficiently identify active compounds
  - Test them in slower, accurate, expensive screens
- Use the data to learn what types of compounds tend to be active

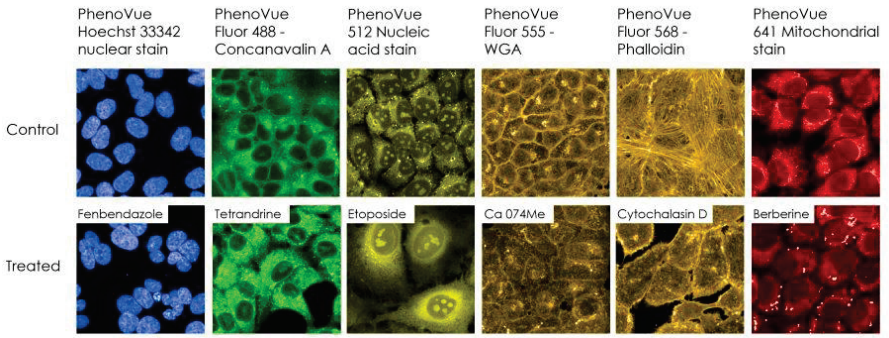


HTS data types

- Categorical: active/inactive or toxic/nontoxic
- Continuous: single-point or dose-response
- Multiple readouts:
  - Might read at different wavelengths or time points
  - More complex when dealing with images



Single-point vs. dose-response readouts



Cell painting for phenotypic drug discovery

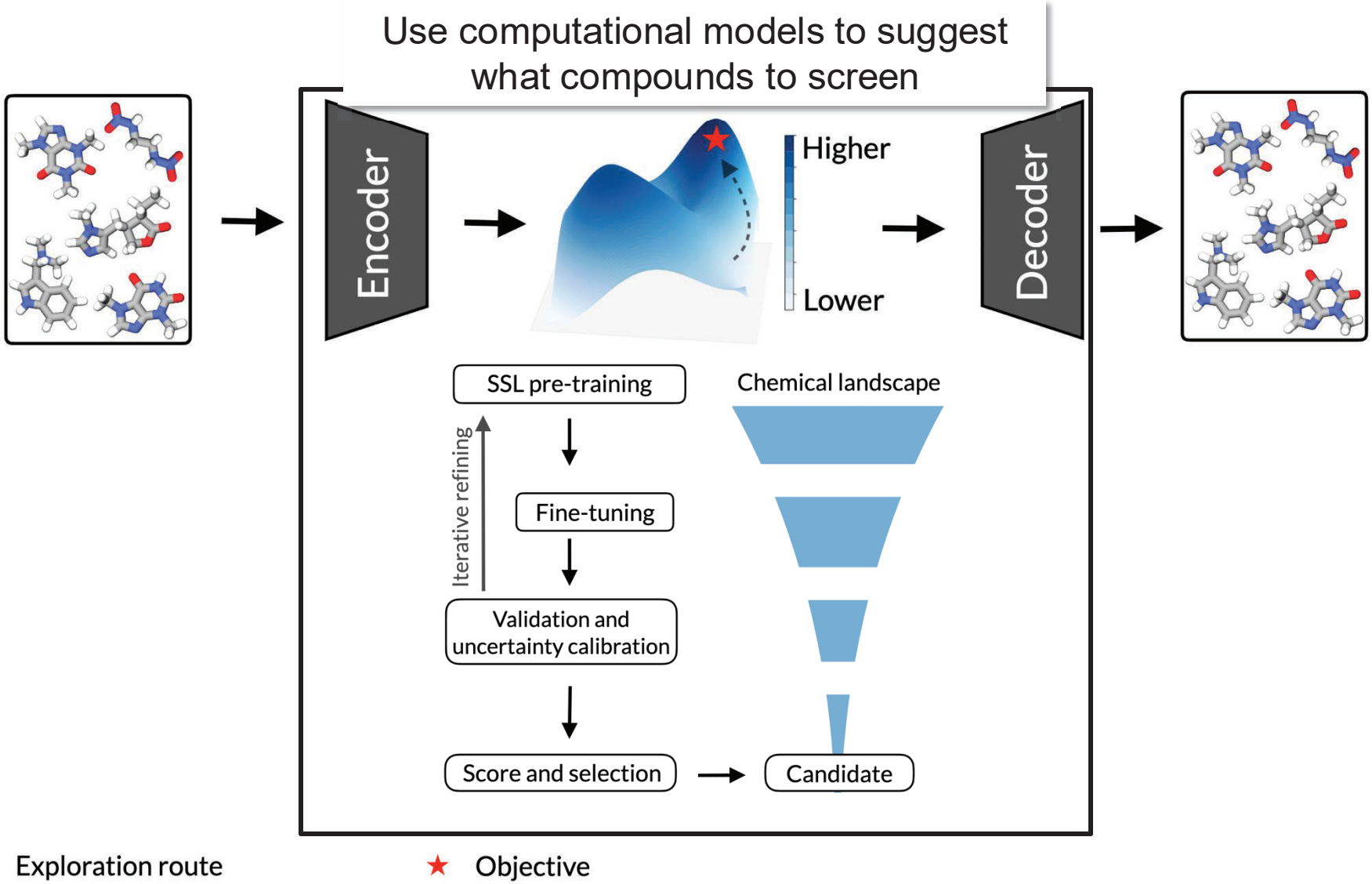
HTS: Machine learning setup

- HTS tests the activity of molecules:

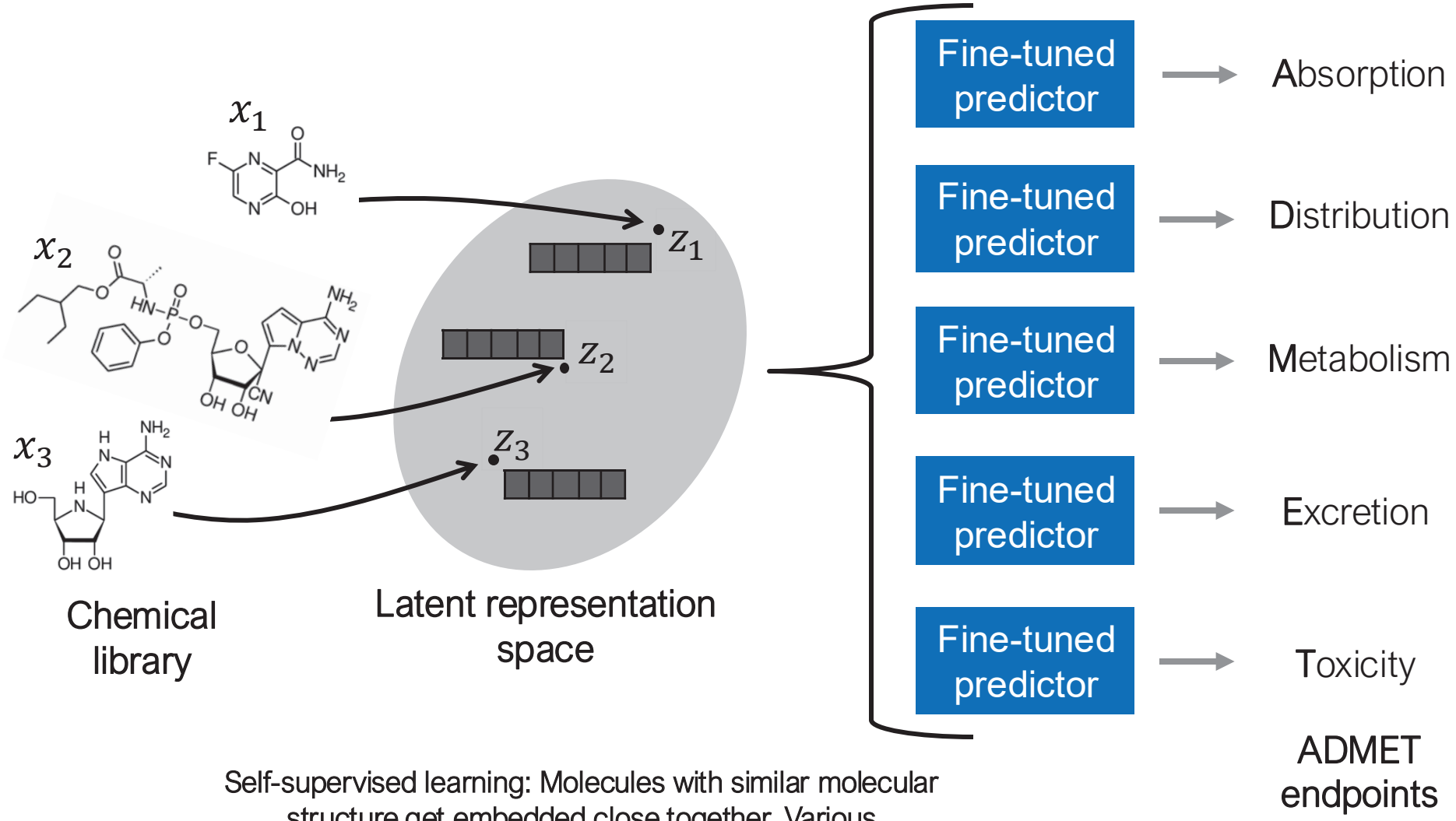
$$Activity = f\left(Structure\right)$$

- We need to describe the molecular structure
  - Various discrete or real-valued descriptors
  - Surfaces (3D)
  - Binary fingerprints
  - Learned molecular embeddings

In-silico screening and optimization of molecular structure



Molecular property prediction

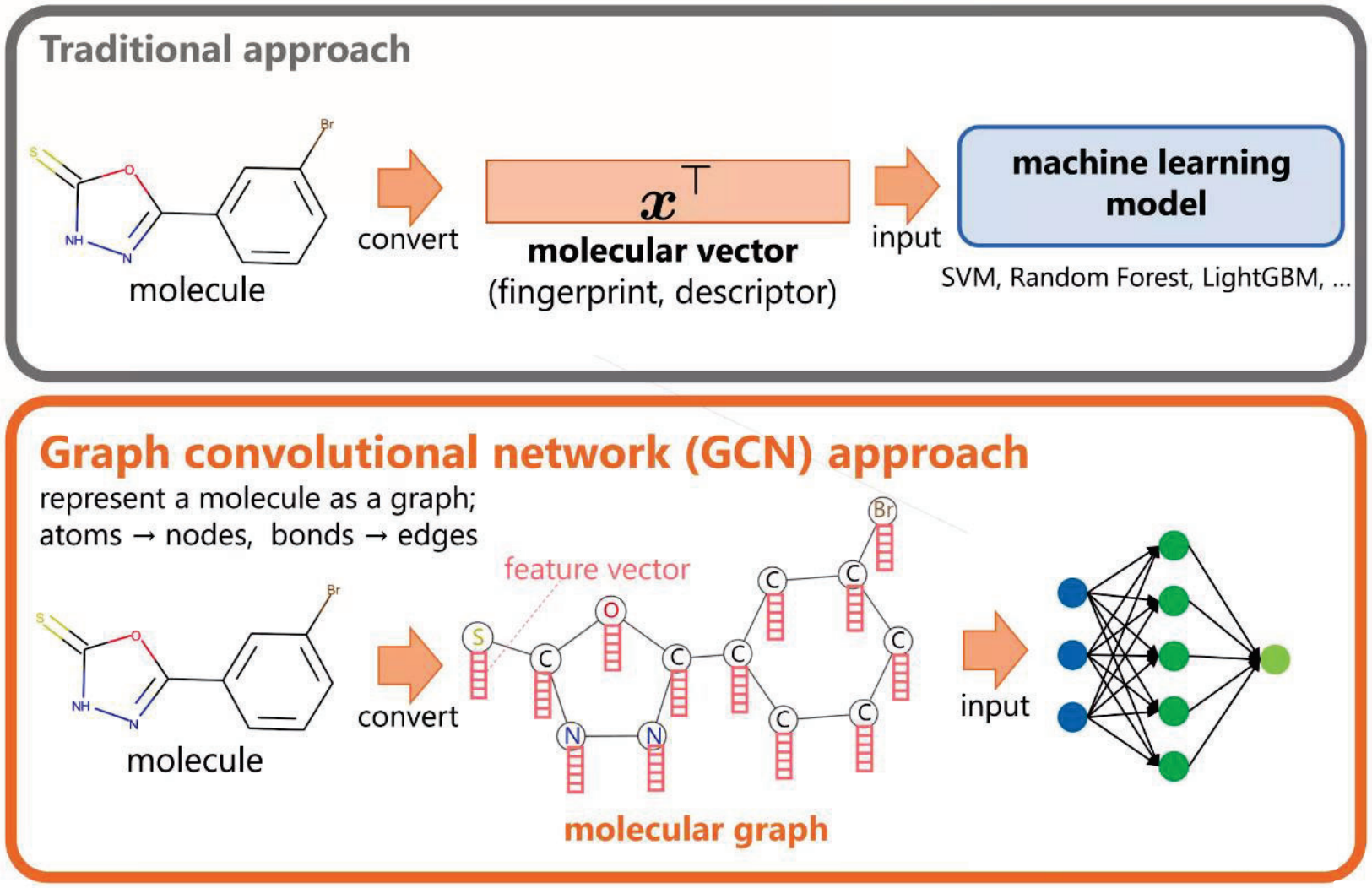


Self-supervised learning: Molecules with similar molecular structure get embedded close together. Various representations: Neural fingerprints, Attentive fingerprints, SMILES descriptors, Graphormer, Transformer-M, and others

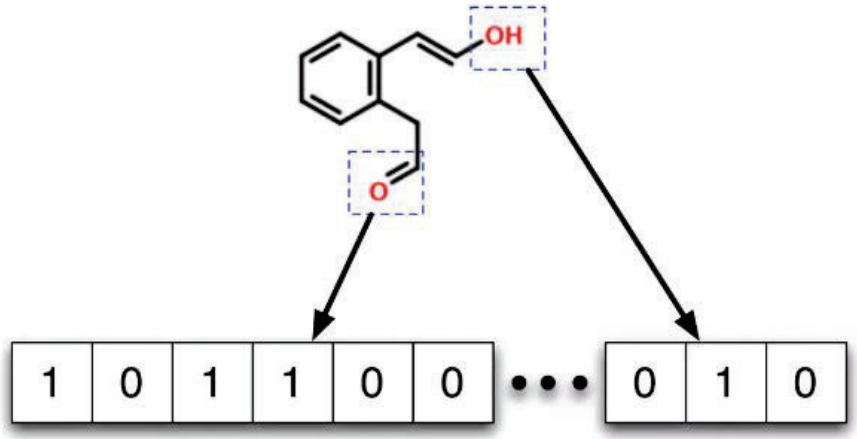
What can we use molecular representations for?

- Search
  - Given a potent active molecule, find similar ones (or dissimilar but also potent)
- Prediction of various endpoints
  - Given a set of active and inactive molecules, build a model to predict which members from a chemical library will be active
- Clustering
  - Given a set of molecules, do they cluster into structurally different groups?

Two strategies for producing molecular representations



Fingerprint representations



- Lots of types of fingerprints
- Keyed fingerprints indicate the presence or absence of a structural feature
- Length can vary from 166 to 4096 bits or more
- Fingerprints usually compared to each other using the Tanimoto metric

Towards neural fingerprints

Algorithm 1 Circular fingerprints

Input: molecule, radius R, fingerprint length S
Initialize: fingerprint vector $\mathbf{f} \leftarrow \mathbf{0}_S$
for each atom a in molecule
  $\mathbf{r}_a \leftarrow g(a)$ ▷ lookup atom features
for $L = 1$ to R ▷ for each layer
  for each atom a in molecule
    $\mathbf{r}_1 \dots \mathbf{r}_N = \text{neighbors}(a)$
    $\mathbf{v} \leftarrow [\mathbf{r}_a, \mathbf{r}_1, \dots, \mathbf{r}_N]$ ▷ concatenate
    $\mathbf{r}_a \leftarrow \text{hash}(\mathbf{v})$ ▷ hash function
    $i \leftarrow \text{mod}(r_a, S)$ ▷ convert to index
    $\mathbf{f}_i \leftarrow 1$ ▷ Write 1 at index
Return: binary vector $\mathbf{f}$

Algorithm 2 Neural graph fingerprints

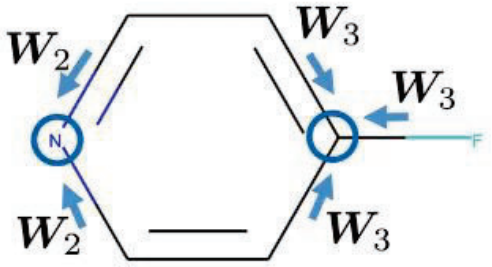
Input: molecule, radius R, hidden weights $H_1^1 \dots H_R^5$, output weights $W_1 \dots W_R$
Initialize: fingerprint vector $\mathbf{f} \leftarrow \mathbf{0}_S$
for each atom a in molecule
  $\mathbf{r}_a \leftarrow g(a)$ ▷ lookup atom features
for $L = 1$ to R ▷ for each layer
  for each atom a in molecule
    $\mathbf{r}_1 \dots \mathbf{r}_N = \text{neighbors}(a)$
    $\mathbf{v} \leftarrow \mathbf{r}_a + \sum_{i=1}^{N} \mathbf{r}_i$ ▷ sum
    $\mathbf{r}_a \leftarrow \sigma(\mathbf{v} H_L^N)$ ▷ smooth function
    $\mathbf{i} \leftarrow \text{softmax}(\mathbf{r}_a W_L)$ ▷ sparsify
    $\mathbf{f} \leftarrow \mathbf{f} + \mathbf{i}$ ▷ add to fingerprint
Return: real-valued vector $\mathbf{f}$

Figure 2: Pseudocode of circular fingerprints (*left*) and neural graph fingerprints (*right*). Differences are highlighted in blue. Every non-differentiable operation is replaced with a differentiable analog.

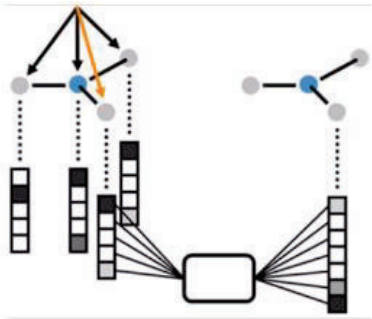
Duvenaud et al., NeurIPS 2015

Neural fingerprint representations

1) **Neural graph fingerprints**
   - Generate molecular fingerprints with a neural network
   - Update atom features using only adjacent atoms
   - Use different weights for node degrees



2) **Molecular graphs**
   - Update atom features by convolutional and pooling layers using adjacent atoms



- **They did not consider property of edges (bonds)**
- **They did not consider atoms other than 1-neighbor**

Duvenaud et al., NeurIPS 2015; Altae Tran et al., ACS Central Science 2017