



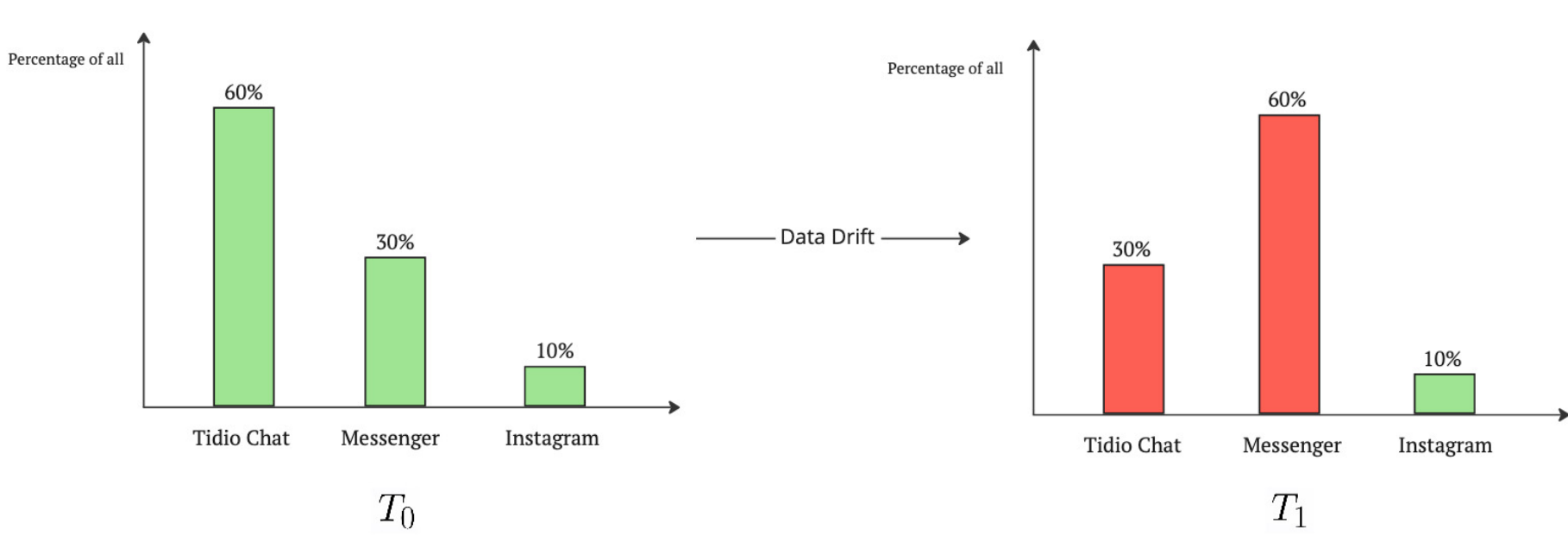
## Why?

Model performance tends to degrade after you’ve deployed a model. It may occur because the model is constant over time and the input data changes. In other words, it is clear that if the model is built on data that changes, there will be a moment when the model should change accordingly. Besides the need for retraining, one also should be aware of the current model condition in the production environment - both in terms of accuracy and latency/system metrics.

### Data Drift

$P(x)$  changes

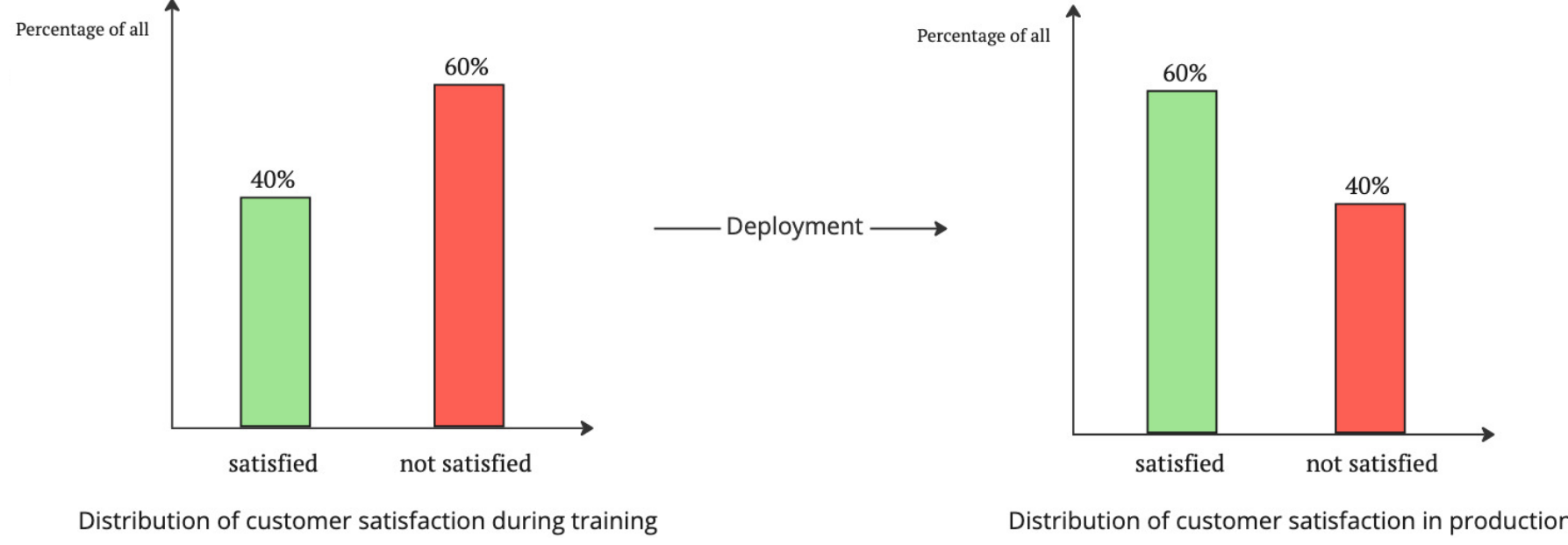
The state of the data that our model is built on changes.



### Domain Shift

Sample doesn't precisely approximate  $P(x,y)$

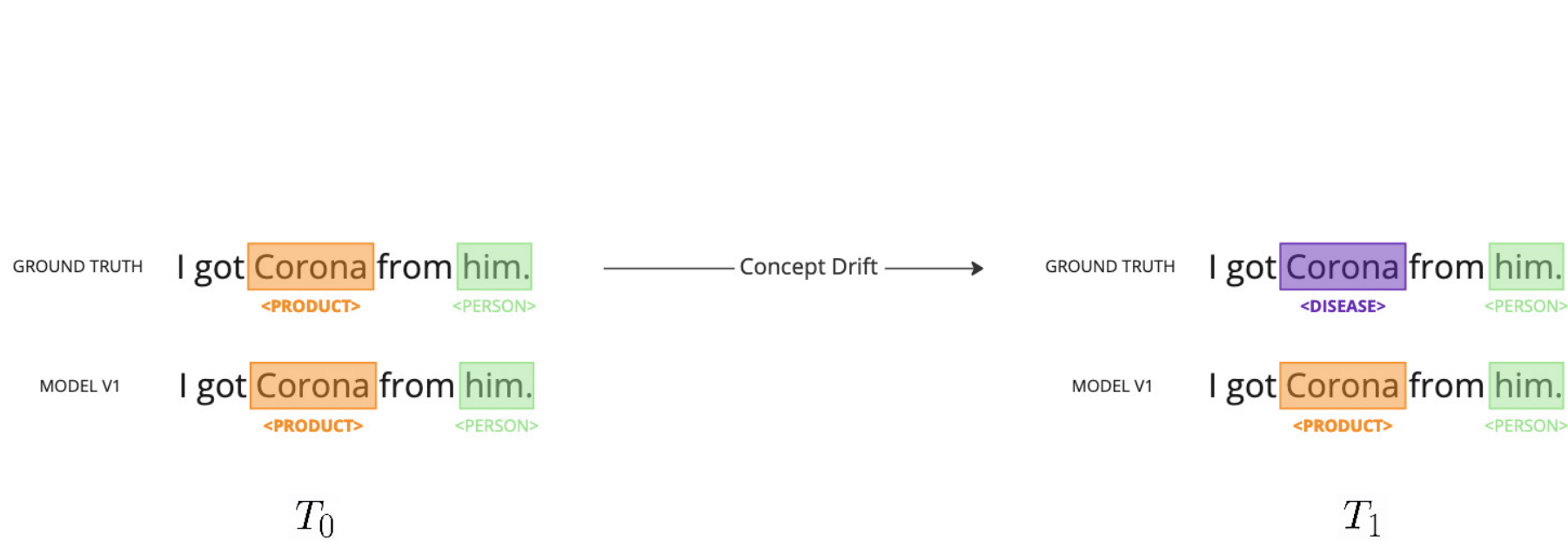
Training data distribution does not reflect the production distribution.



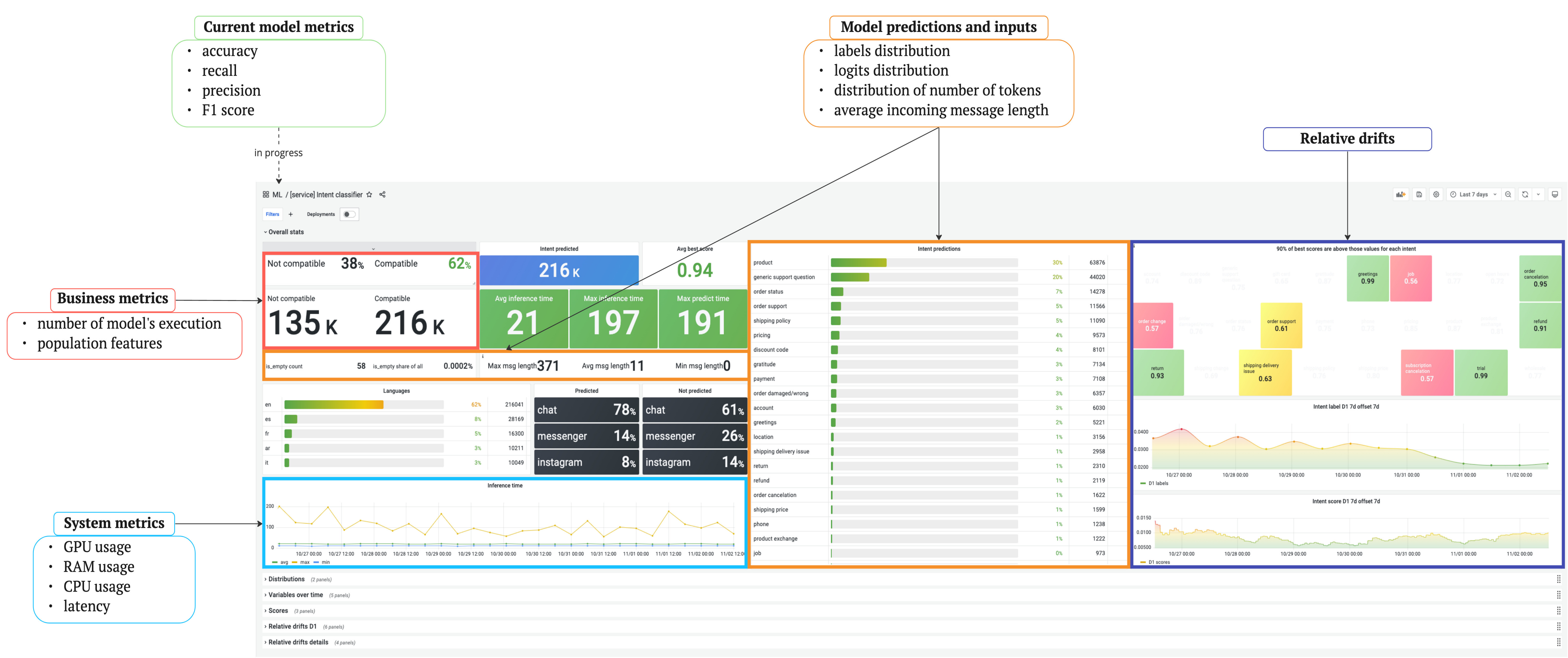
### Concept Drift

$P(y | x)$  changes

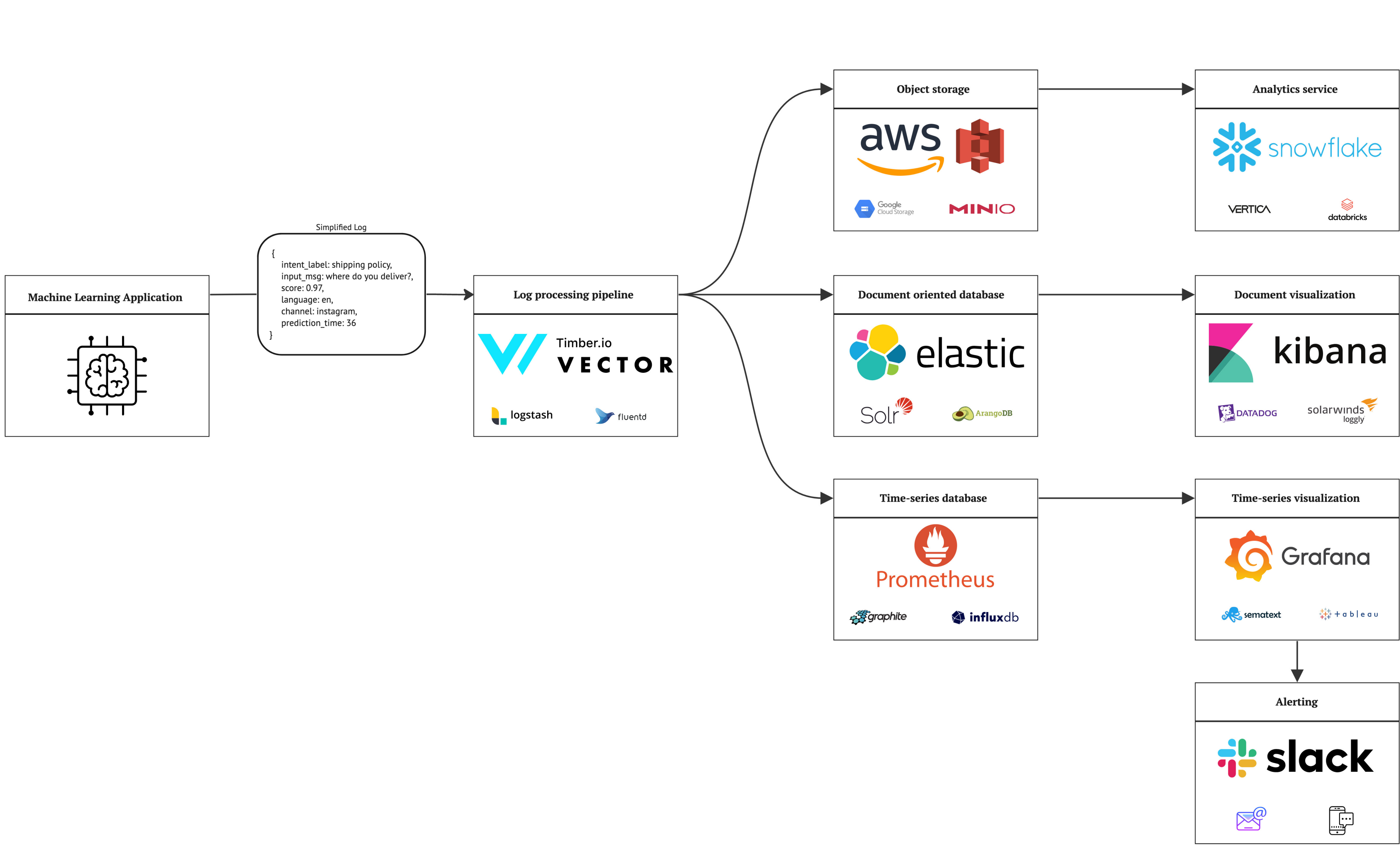
Relationship changes, not the input, i.e. the underlying meaning of the input data changes.



## What?



## How?



- **Log processing pipeline** - collect, transform logs, create metrics and route logs
- **Object storage** - collect information on inputs, intermediate results and outputs of the model (infinite retention)
- **Document oriented database** - collect application logs together with information on inputs, intermediate results and outputs of the model (with retention)
- **Time-series database** - collect, store and perform transformations on metrics
- **Analytics service** - collect, store and perform complex on-demand transformations on data collected in *object storage*
- **Document visualization** - provide simple visualization on text documents collected in *document oriented database*
- **Time-series visualization and alerting** - provide complex visualization on metrics collected in *time-series database* and alerting based on setup thresholds