

Font File Sorting

Myles C. Maxfield

Fitness Function

- Average percentage of file skipped
- Averaged across all the documents
 - 100,000 is too many documents
- GPGPU (Metal)
- More accurate than previous presentation

```

kernel void fitness(device uint32_t* generation [[buffer(0)]], device uint32_t*
    glyphSizes [[buffer(1)]], device uint8_t* urlBitmaps [[buffer(2)]], device
    uint32_t* output [[buffer(3)]], uint2 tid [[thread_position_in_grid]]) {
    uint generationIndex = tid.x;
    uint urlIndex = tid.y;
    uint32_t result = unconditionalDownloadSize + threshold;
    uint32_t unnecessarySize = 0;
    bool state = false;
    for (uint32_t i = 0; i < glyphCount; ++i) {
        uint32_t glyph = generation[glyphCount * generationIndex + i];
        uint32_t size = glyphSizes[glyph];
        bool glyphIsNecessary = ::glyphIsNecessary(urlBitmaps, urlIndex, glyph);
        if (glyphIsNecessary) {
            result += size;
            if (!state) {
                result += min(unnecessarySize, threshold);
                unnecessarySize = 0;
            }
        } else
            unnecessarySize += size;
        state = glyphIsNecessary;
    }
    output[urlCount * generationIndex + urlIndex] = fontSize - result;
}

```

argmax(fitness)

1027754

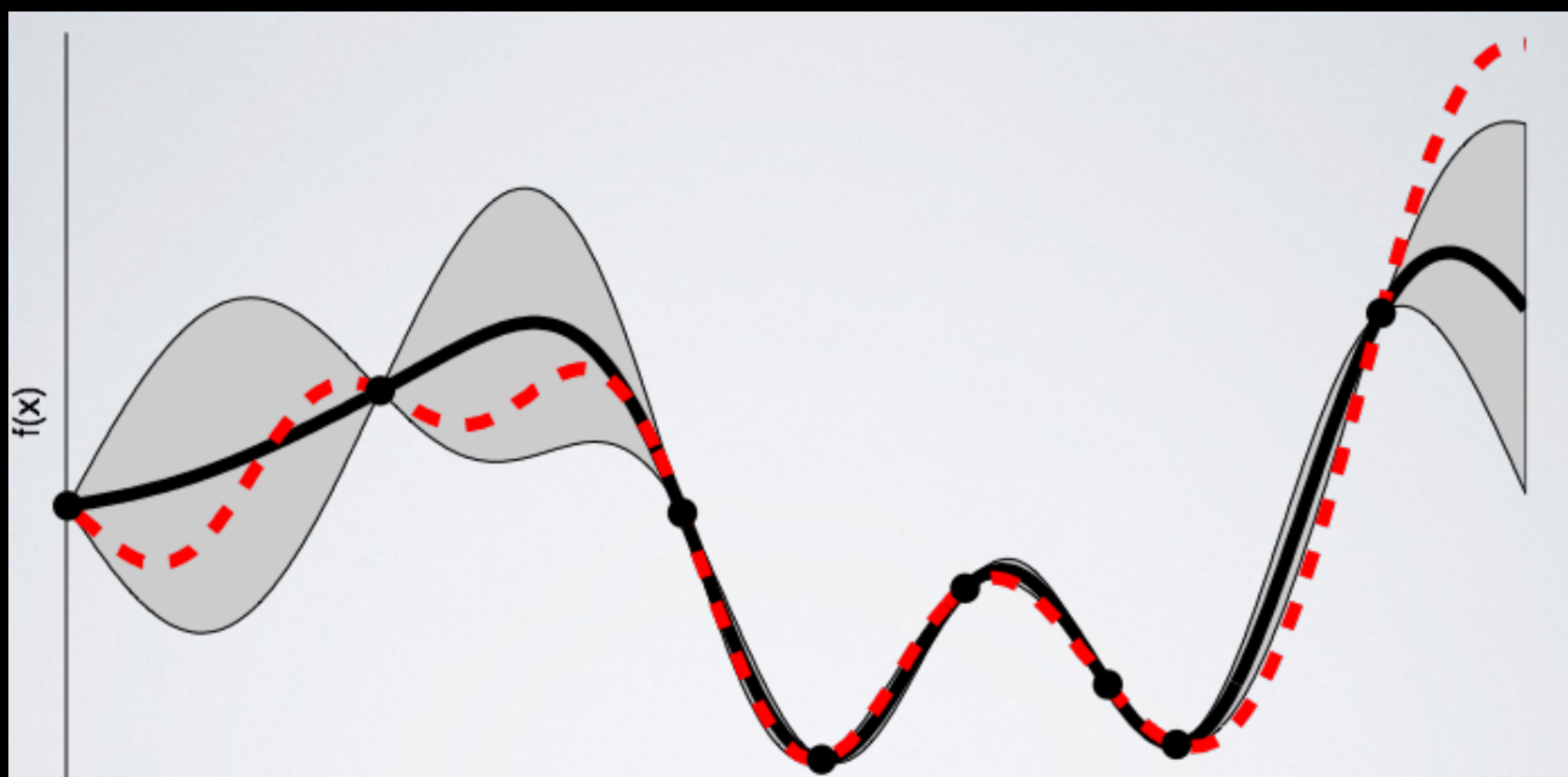
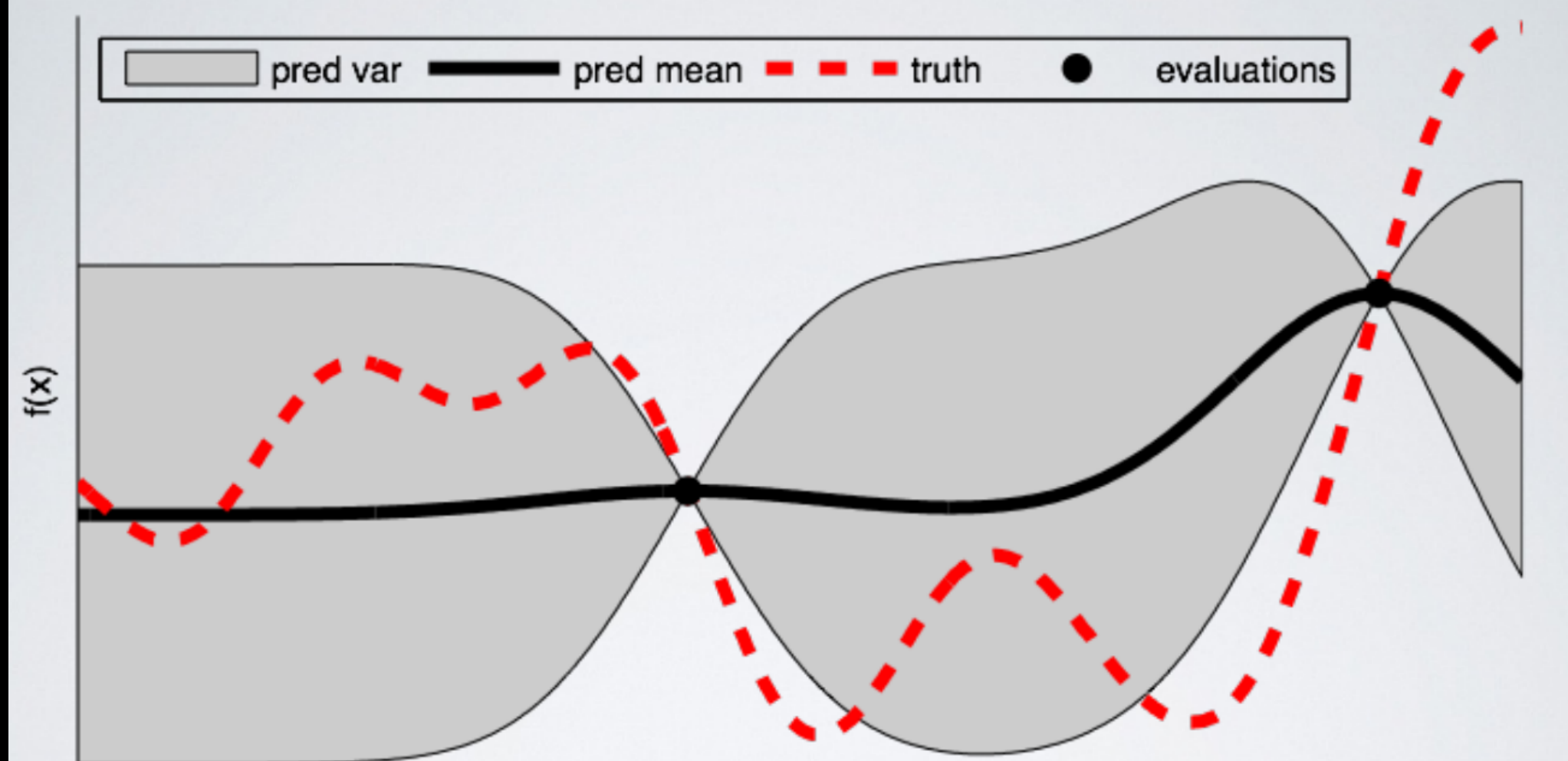
1012675

Strategies

- Hyperparameter Optimization
- Fasttext
- Gradient Descent
- Genetic Algorithm
- Simulated Annealing
- Particle System (not included)

Strategies

- Hyperparameter Optimization
- Fasttext
- Gradient Descent
- Genetic Algorithm
- Simulated Annealing
- Particle System (not included)



24%|██████

| 24/100 [01:45<09:20, 7.37s/it, best loss: 0.0]

Strategies

- Hyperparameter Optimization
- **Fasttext**
- Gradient Descent
- Genetic Algorithm
- Simulated Annealing

fastText

Library for efficient text classification and representation learning

[GET STARTED](#)

[DOWNLOAD MODELS](#)

What is fastText?

FastText is an open-source, free, lightweight library that allows users to learn text representations and text classifiers. It works on standard, generic hardware. Models can later be reduced in size to even fit on mobile devices.

Download pre-trained models

Open "https://fasttext.cc" in a new tab

```
010100100
101001001
010010101
001011101
110101110
101001011
```

```
010100100
101001001
010010101
001011101
110101110
101001011
```

Resources

- English word vectors
- [Word vectors for 157 languages](#)
- Wiki word vectors
- Aligned word vectors
- Supervised models
- Language identification
- Datasets

Word vectors for 157 languages

We distribute pre-trained word vectors for 157 languages, trained on *Common Crawl* and *Wikipedia* using fastText. These models were trained using CBOW with position-weights, in dimension 300, with character n-grams of length 5, a window of size 5 and 10 negatives. We also distribute three new word analogy datasets, for French, Hindi and Polish.

Download directly with command line or from python

In order to download with command line or from python code, you must have installed the python package as [described here](#).

Command line Python

```
$ ./download_model.py en # English
Downloading https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.en.300.bin.gz
(19.78%) [=====> ]
```

Once the download is finished, use the model as usual:

```
$ ./fasttext nn cc.en.300.bin 10
Query word?
```

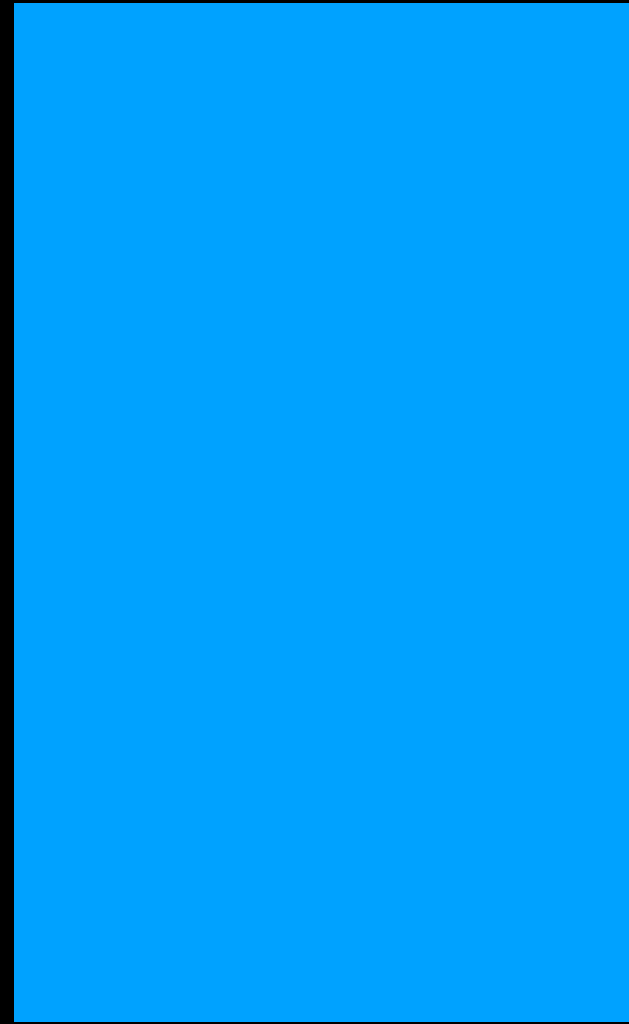
Adapt the dimension

The pre-trained word vectors we distribute have dimension 300. If you need a smaller size, you can use our dimension reducer. In order to use that feature, you must have installed the python package as [described here](#).

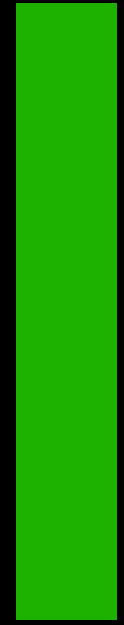
For example, in order to get vectors of dimension 100:

< — — 300 — — >

< — — 4,000 — — >



X



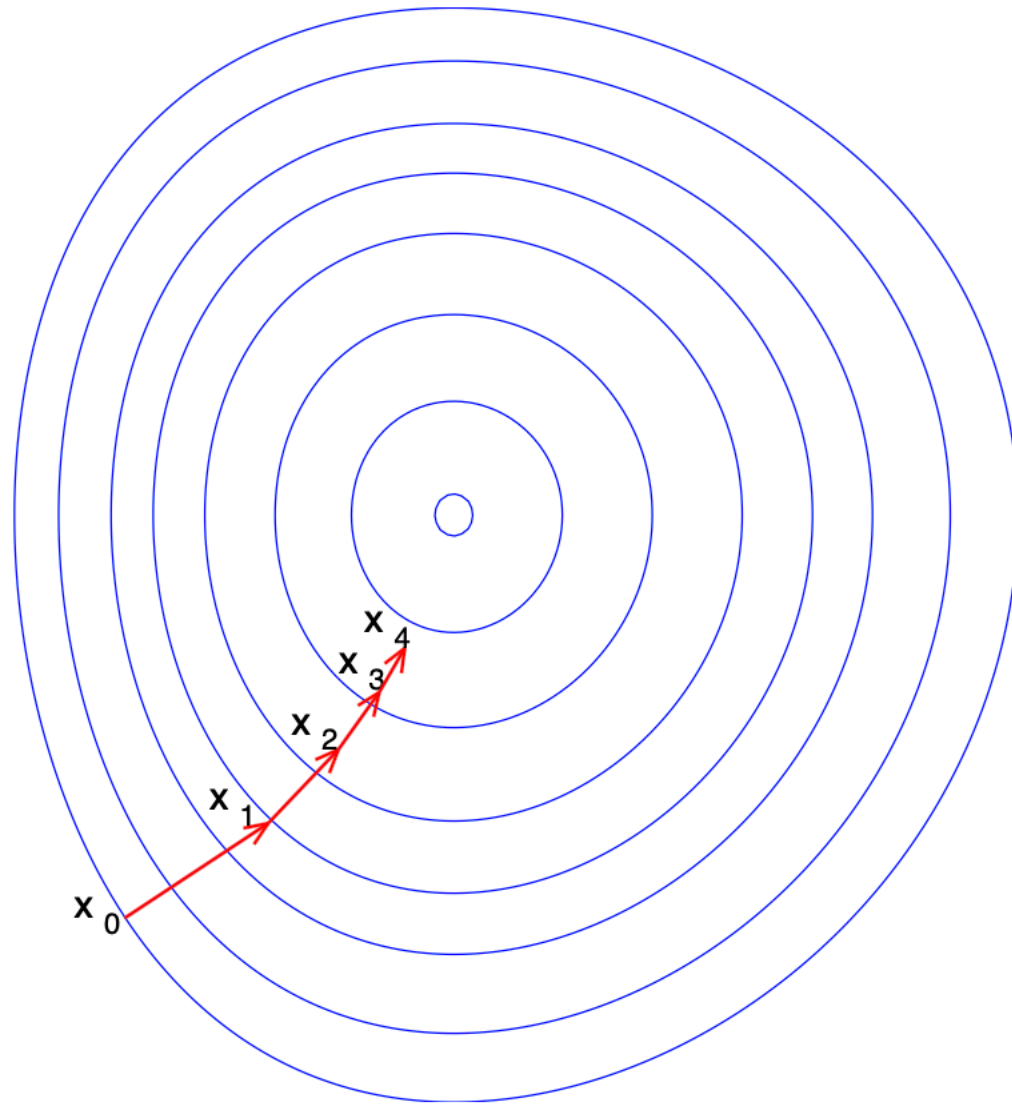
=



81%|██████████ | 81/100 [00:37<00:11, 1.66it/s, best loss: 0.0]

Strategies

- Hyperparameter Optimization
- Fasttext
- Gradient Descent
- Genetic Algorithm
- Simulated Annealing



$$(\boldsymbol{p}) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(\boldsymbol{p}) \\ \vdots \\ \frac{\partial f}{\partial x_n}(\boldsymbol{p}) \end{bmatrix}$$

$$f'(a) = \lim_{h \rightarrow 0} \frac{f(a+h) - f(a)}{h}$$

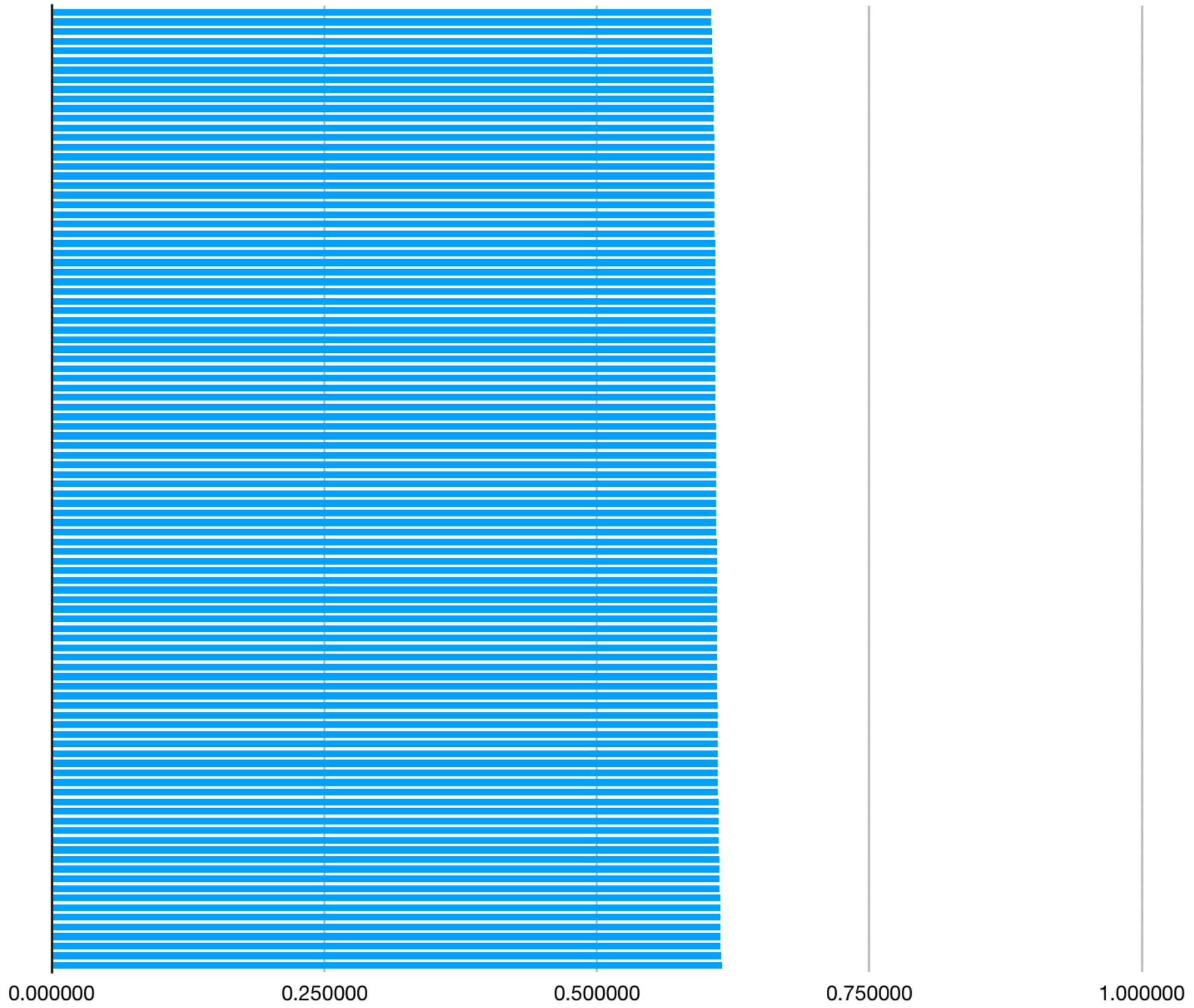


Strategies

- Hyperparameter Optimization
- Fasttext
- Gradient Descent
- Genetic Algorithm
- Simulated Annealing



■ Random Fitness

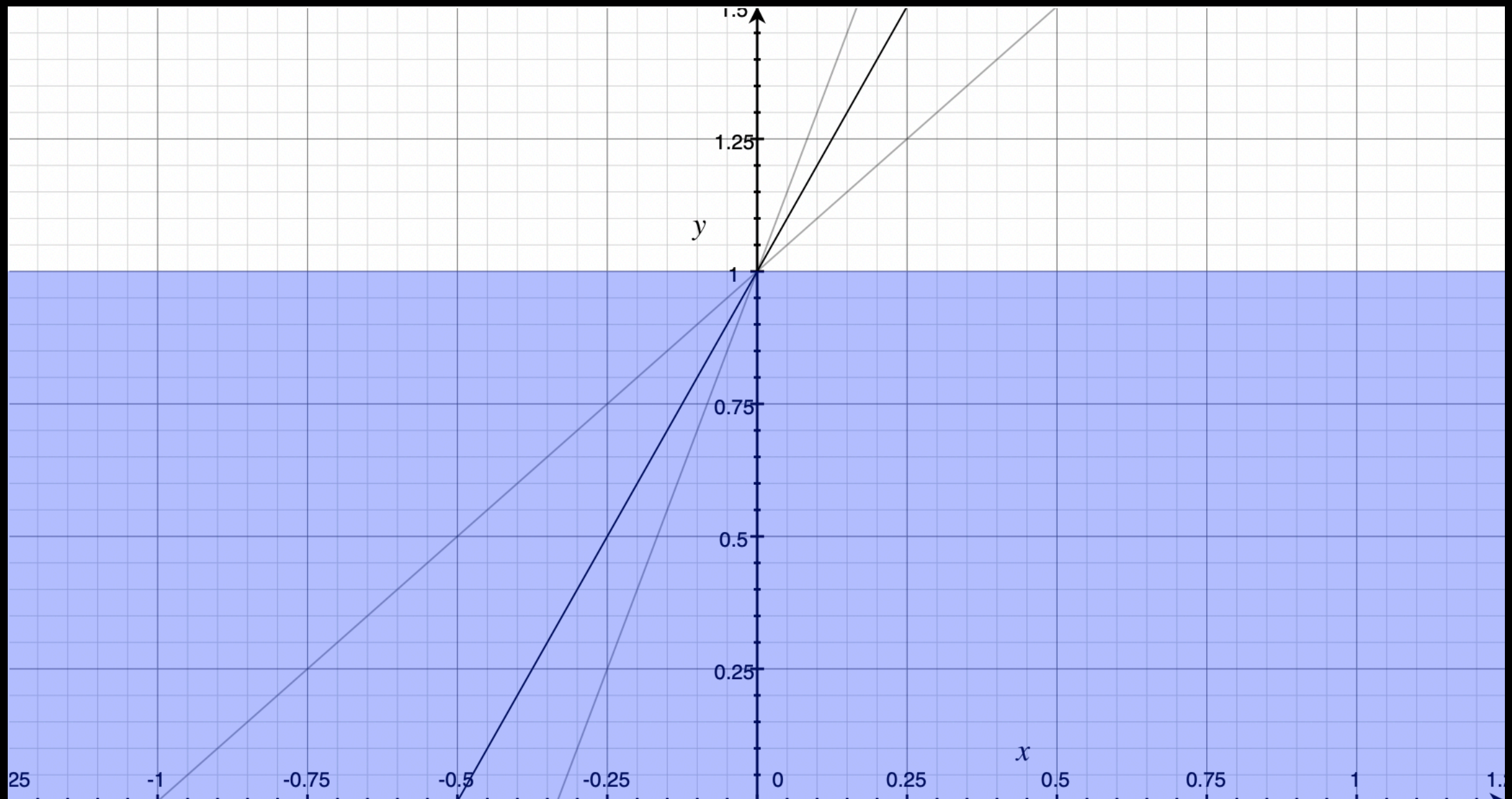




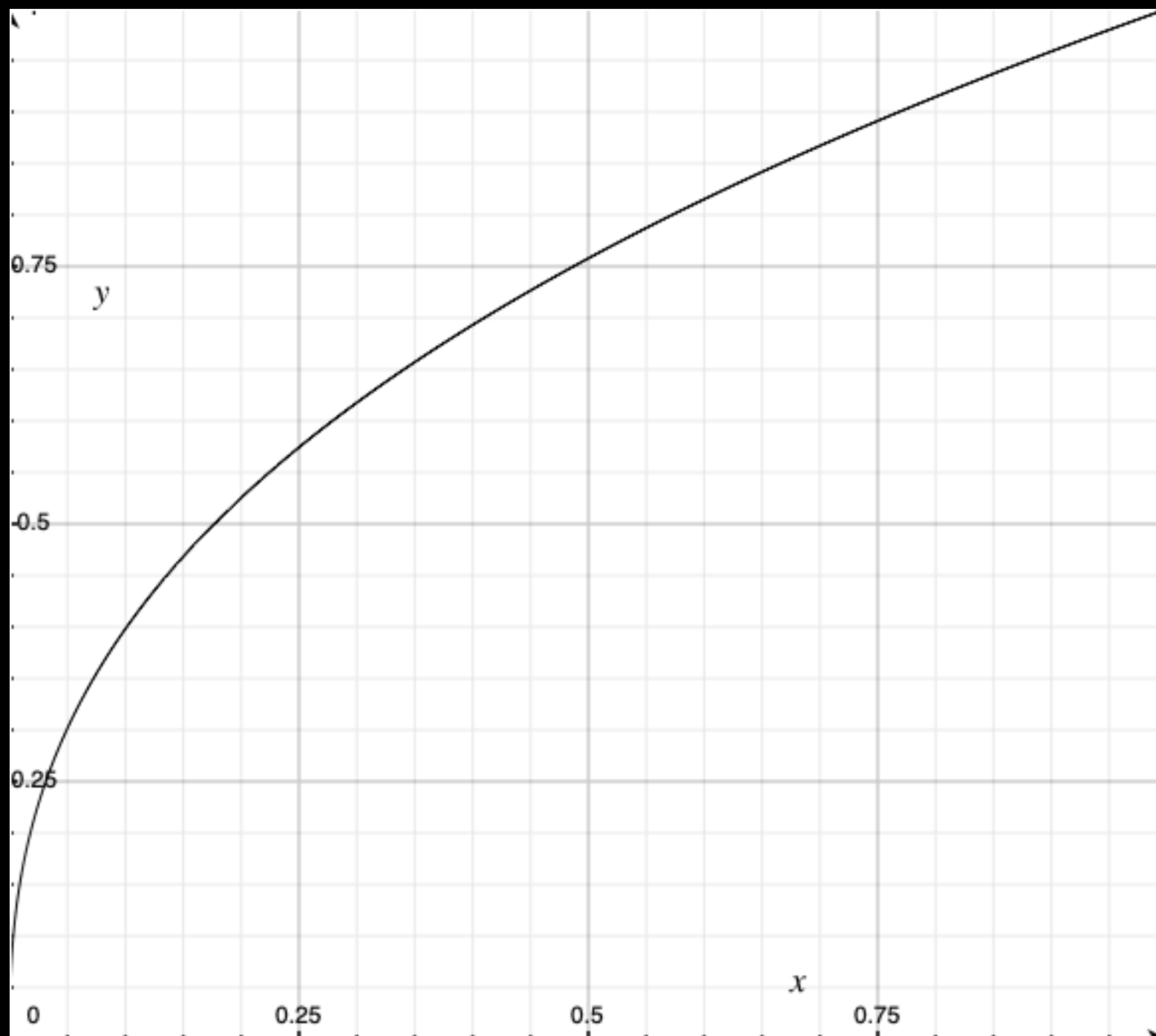
Strategies

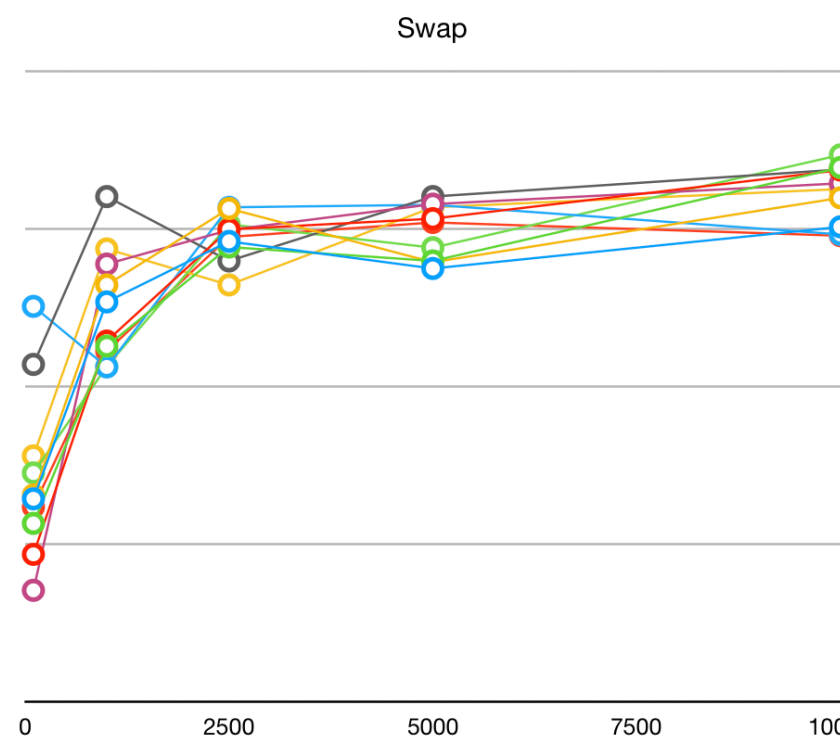
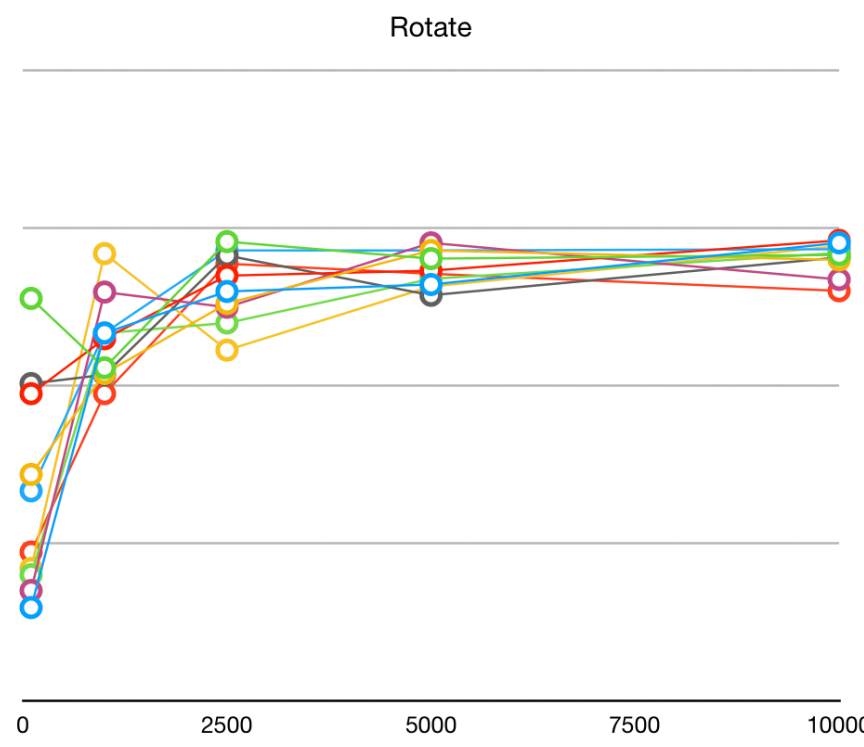
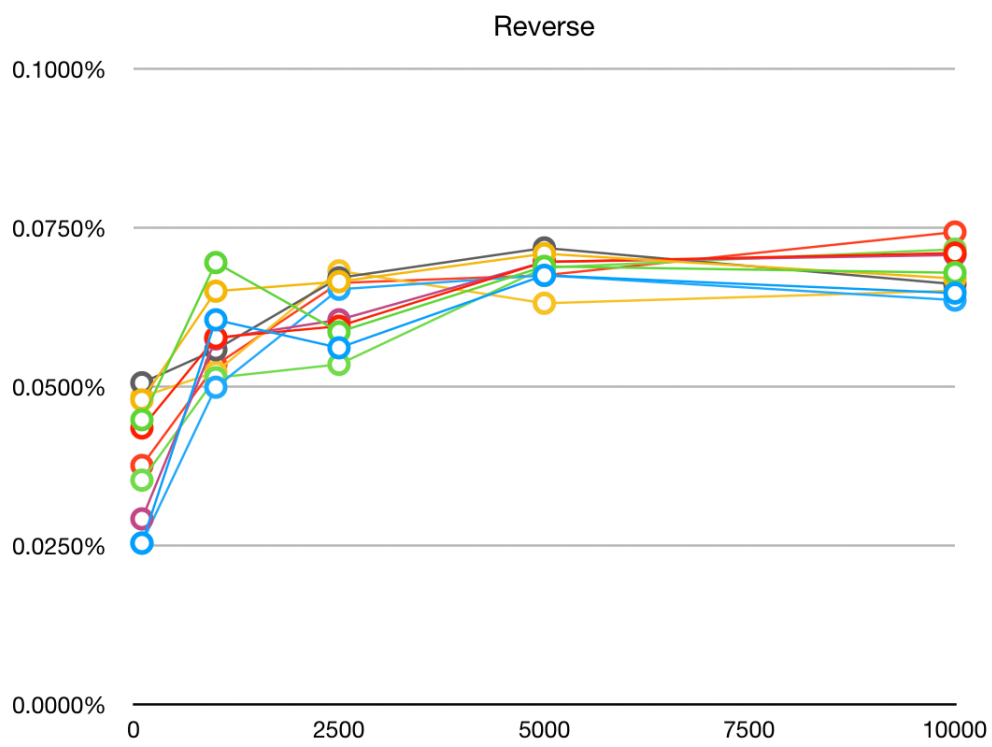
- Hyperparameter Optimization
- Fasttext
- Gradient Descent
- Genetic Algorithm
- Simulated Annealing





slope * Δ fitness + 1 > rand()





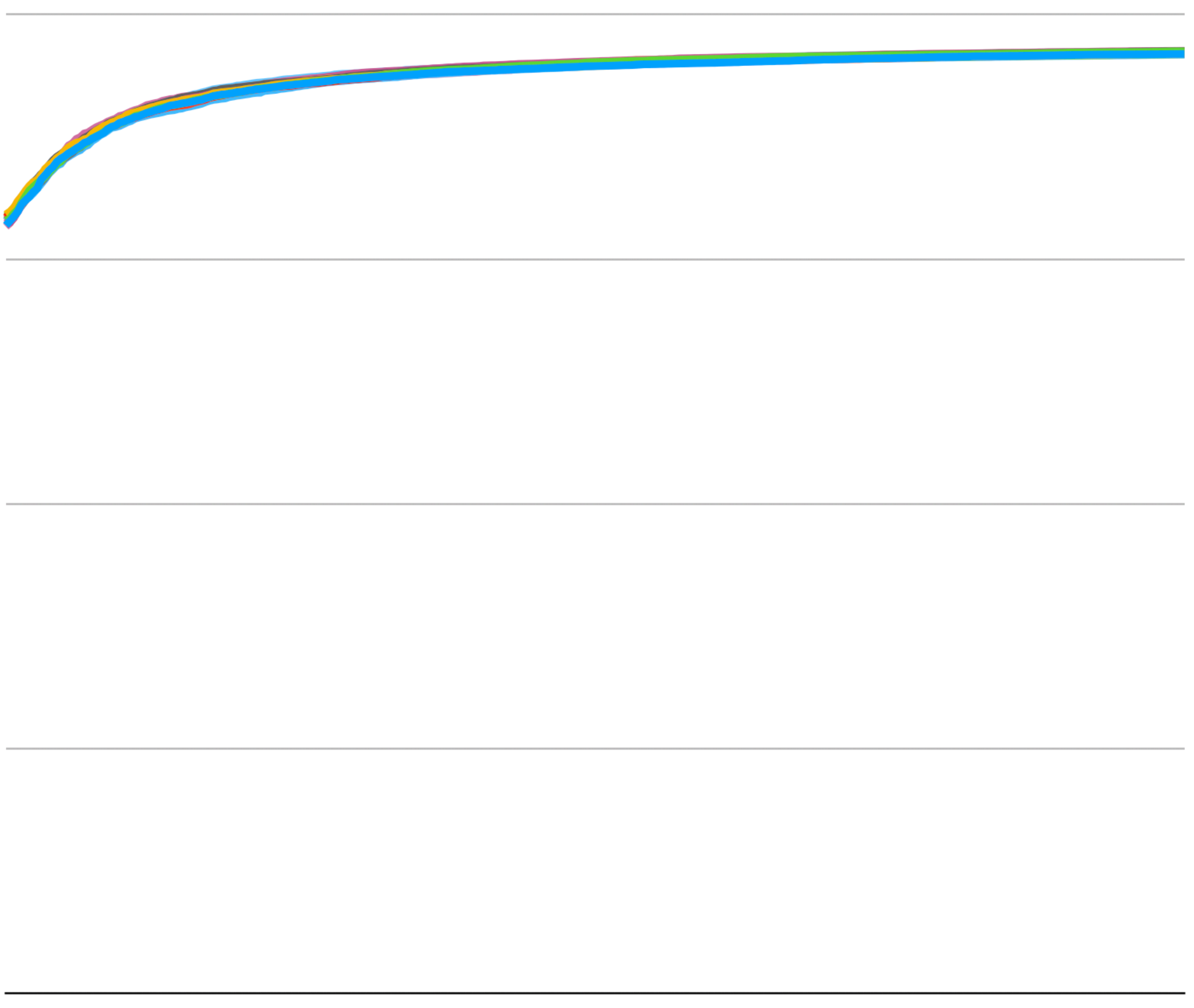
0.80000

0.60000

0.40000

0.20000

0.00000



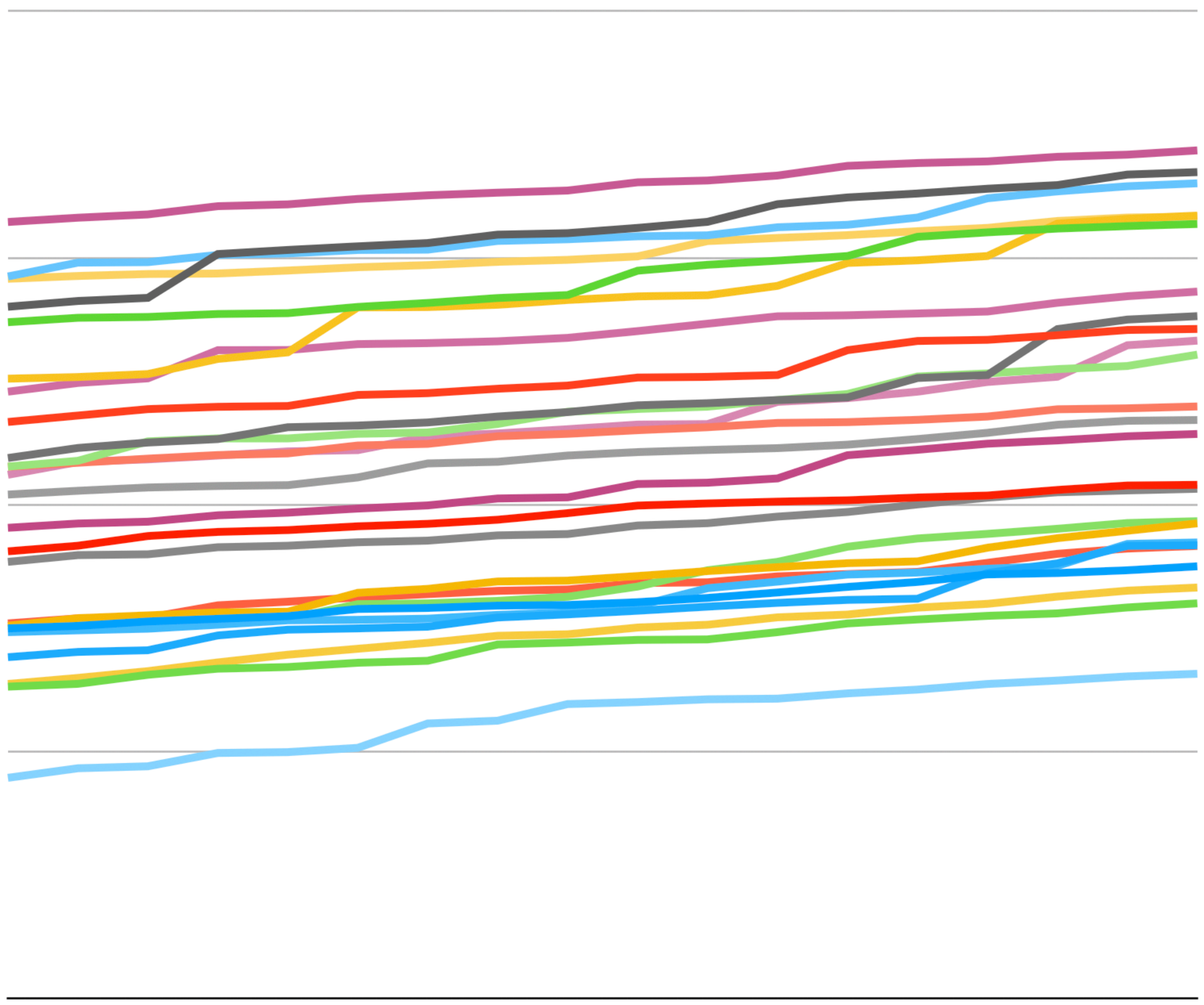
0.771

0.77

0.768

0.767

0.765



78%

Future Work

- Pleateaus?
- Compare final orderings
- Optimize particle system
- New fonts
- Compression
- Contribute

W3C®