ITMO

Technologies and Infrastructure for Big Data

Michael Grigoriev
Zakharov Denis

Goal

Topic: Automating anime image collection from several resources for future model training.

Category: img

Category: neko

Category: kitsune

Category: smile

Data resources

NekosBest	https://nekos.best/api/v2/
WaifuPics	https://api.waifu.pics/
CatBoys	https://api.catboys.com/
Waifium	https://api.waifu.im/search/

Tech Stack

Data Scraping

ITMO

```go
func wrap(resource string, rd, cd map[string]string) {
	num := 100
	if resource == "res_name" {
		num = 500
		for i := 0; i < num; i++ {
			scrape("img", rd[resource])
		}
	fmt.Println(resource, "img", "scraped")
	return
}
```

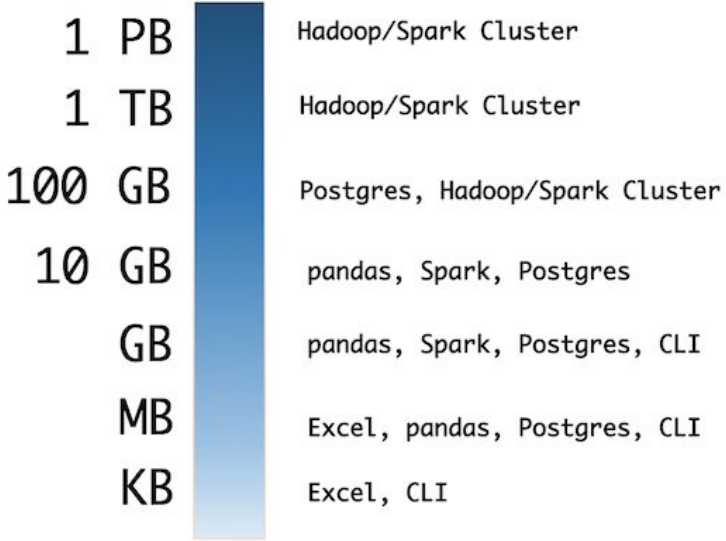
```go
func scrape(wCategory, res string) {
	url := res + wCategory
	apiResp, eGet := http.Get(url)
	if eGet != nil {
		log.Fatal(eGet)
	}
	defer apiResp.Body.Close()

	picUrl := ""

	switch res {
	case "https://api.waifu.pics/":
		pic := WaifuPics{}
	eJSON := json.NewDecoder(apiResp.Body).Decode(&pic)
	if eJSON != nil {
		log.Fatal(eJSON)
	}
	picUrl = pic.Url
}
```

Selection and implementation of Big Data

According to research Big Data framework research it would be reasonable to choose Spark for our project.



Processing

```
def buildPySpark(root_dir, sc):
    raw_train, rdd_test = [], []
    ...
    for target_lbl, file in enumerate(os.listdir(root_dir)):
        if os.path.isdir(d):
            ...
            for sub_file in os.listdir(d):
                im = io.imread(f"{d}/{sub_file}")
                ...
    # went through all files
    raw_train = np.array(raw_train)
    raw_test = np.array(raw_test)

    np.save("../../test_backup.npy", raw_test)
    np.save("../../train_backup.npy", raw_train)
```

Loading

```python
def load_backup(name="rdd_train_backup"):
    nfile = np.load(f"../../{name}.npy")
    pdf = pd.DataFrame(nfile.T)

    # Enable Arrow-based columnar data transfers
    spark.conf.set("spark.sql.execution.arrow.pyspark.enabled", "true")

    # Create a Spark DataFrame from a pandas DataFrame using Arrow
    df = spark.createDataFrame(pdf)

    return df
```

Data Storing

	DataFrame	RDD	DataSet
Language	Python	Python Scala Java	Java Scala
Fault Tolerant	Yes	Yes	Yes
Schema	Yes	No	Yes
Optimized	Catalyst	Not Supported	Catalyst
Distributed data manip	High	Low	High

Analysis of the collected and stored data

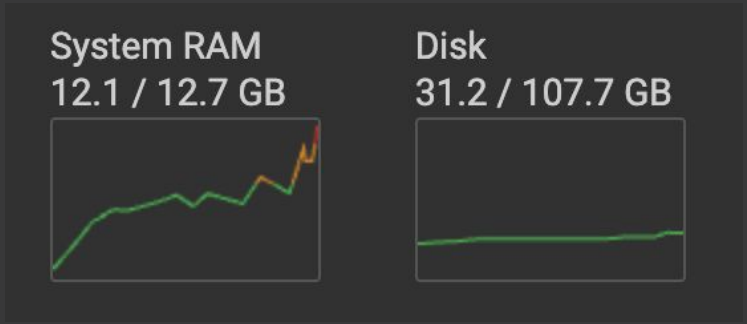
Volume – 6 GB.

Velocity – from scraper.

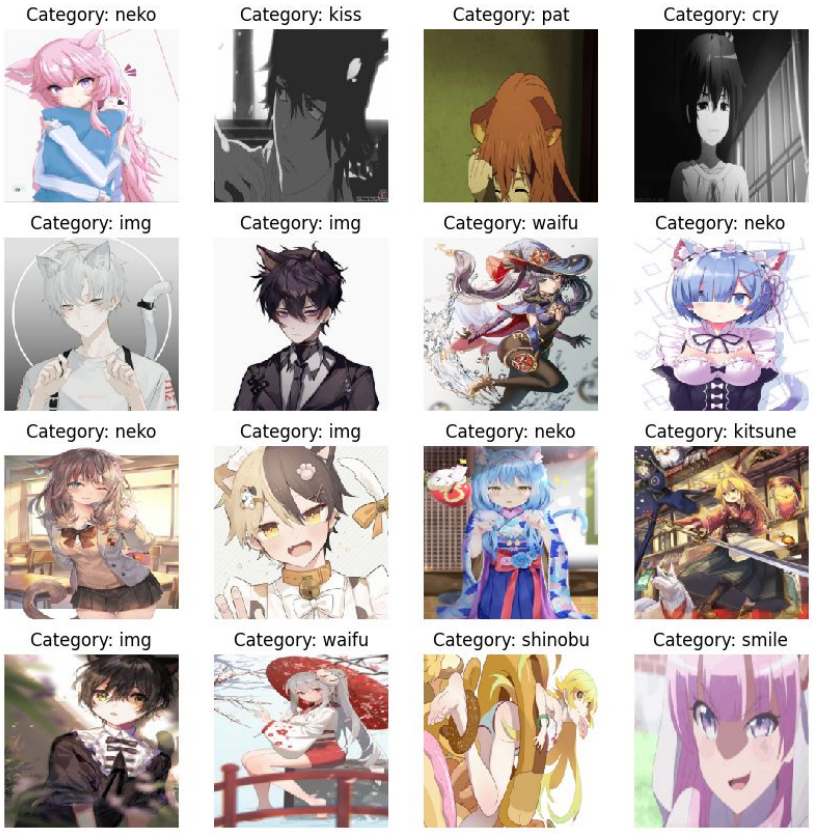
Variety – GIF, JPEG, JPG, PNG.

Variability – chosen processing tools, cannot, work with GIF format.

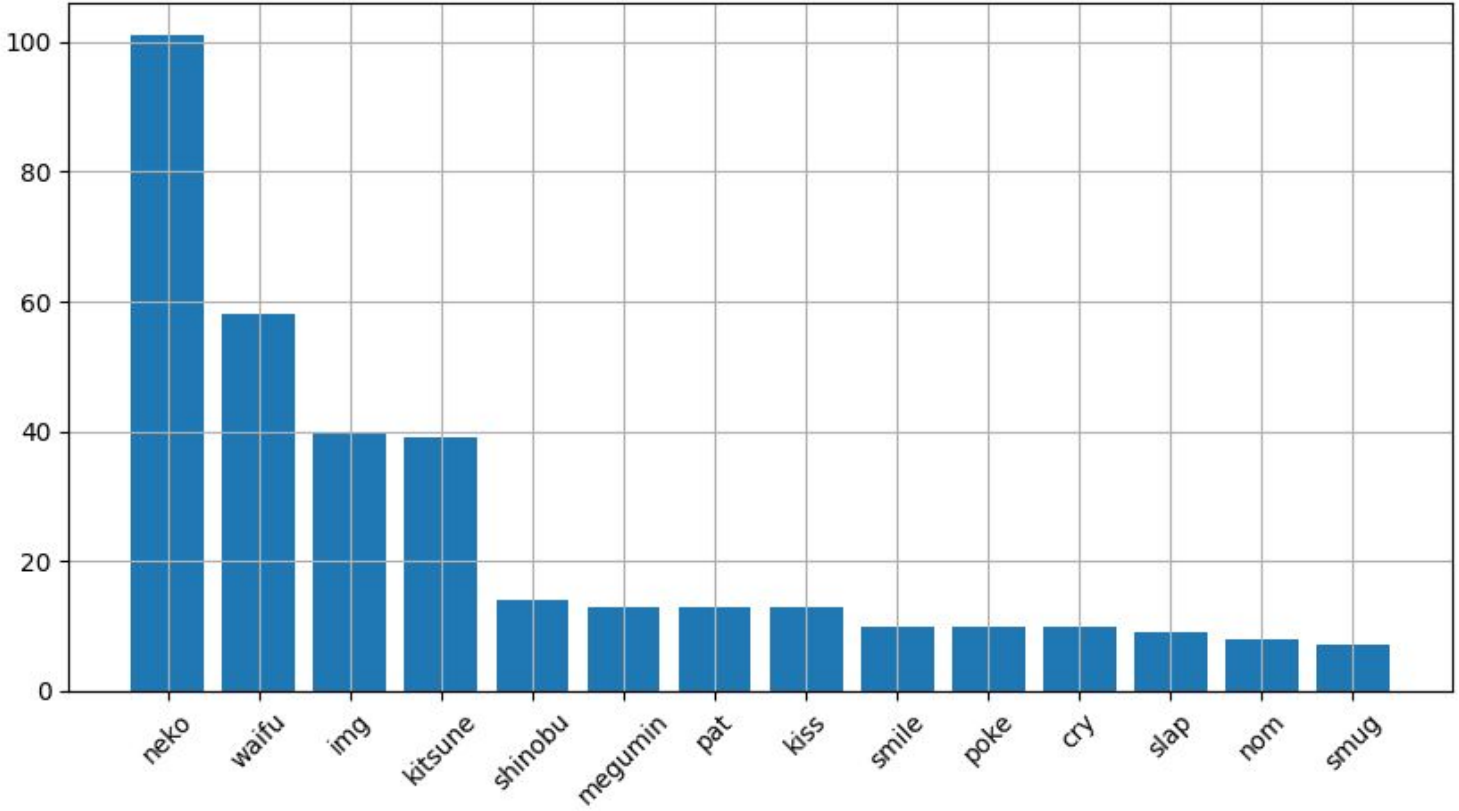
Value – gathered data is unstructured and bonded with only label tag.



Visualization: Stored Data

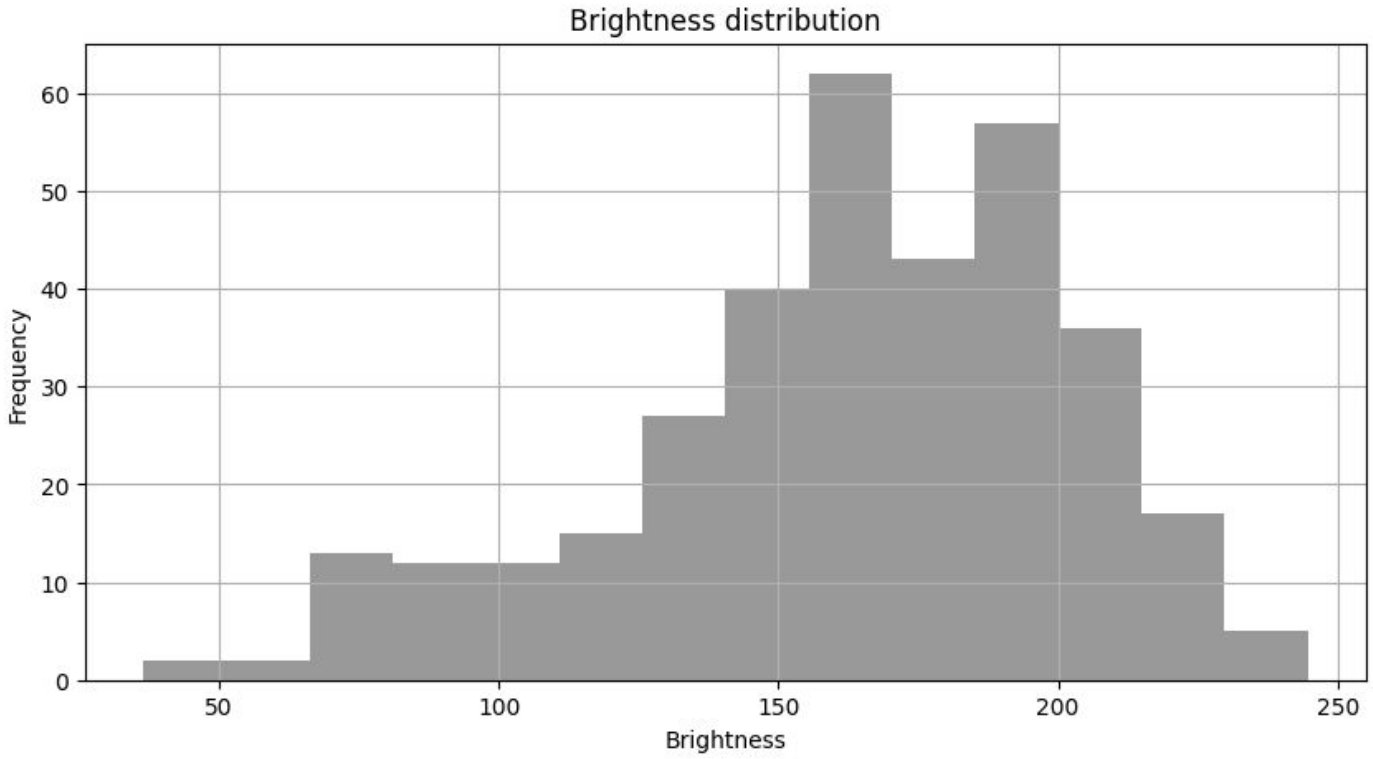


Visualization: Label Distribution

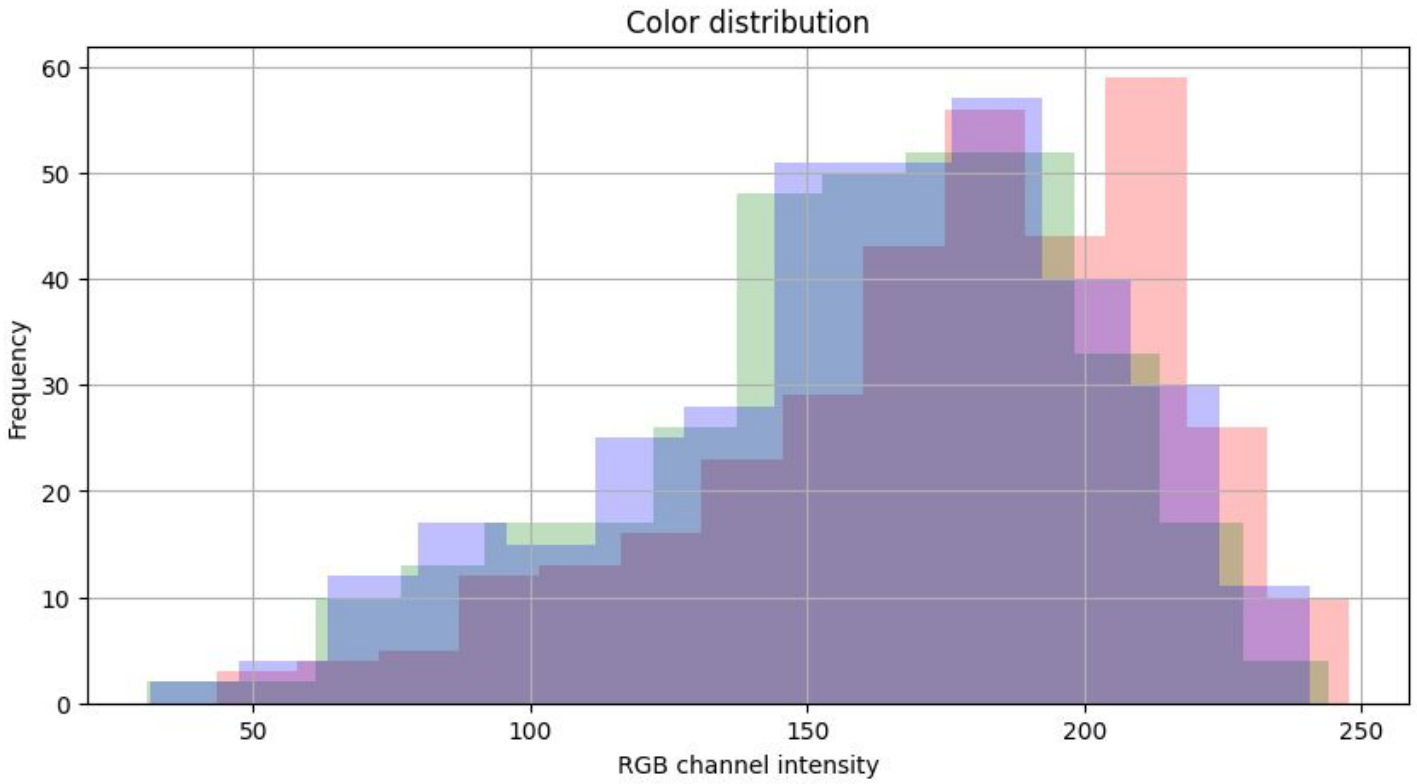


Visualization: Duplicate identification
using perceptual hash (*imagehash lib*)

Visualization: Brightness distribution



Visualization: Color distribution



Links to NSFW

Scraper:

https://github.com/Dormant512/weeb-scrape/blob/main/weeb-scrape.go

Data Processing:

https://colab.research.google.com/drive/1jh-pJwGYIK5TVyYD8rYik3waYinHZY6f?usp=sharing

THANK YOU FOR YOUR TIME!

ITsMOre than a UNIVERSITY