Palette Prophet

Kristian R. Møllmann[1] *Kristoffer Marboe*[2] *Kasper N. K. Hansen*[3]

{[1]s194246, [2]s194249, [3]s194267}@student.dtu.dk

Labelling

The original data consisted of 6,000+ unlabelled paintings scraped from the Danish online marketplace Den Blå Avis (DBA).

We randomly picked out 1,000 paintings to hand label using two methods: ELO and scale.

The **ELO** rating system was originally invented as a chess-rating system and is, in this project, used to determine preference. Labeller is presented with two paintings and needs to choose which they prefer.

The **scale** rating system is a simple nine-point scale where 1 corresponds to atrocious, and 9 corresponds to magnificent.

To this end, we created two `tkinter` `python` apps for easy annotations; see Figure 1. Each labeller completed 2,500 matchups in the ELO setting, continuously updating the ELO ratings and saving the matchup history. For the scale system, each painting was processed once.

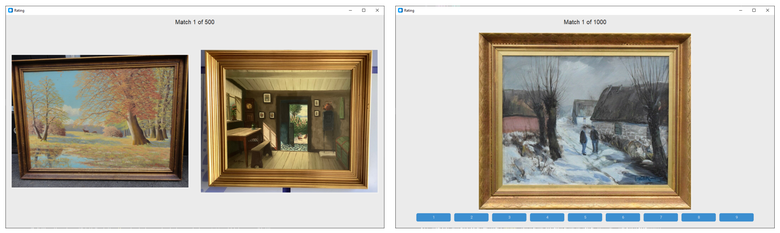

Figure 1. Left: Interface for data labelling using the ELO system. Right: Interface for data labelling on a 9-point scale.

Simulating ELO ratings

ELO is widely used for rating systems but requires numerous matches for accuracy, as limited data could misclassify paintings due to ”bad” matchups. To enhance our model and reduce manual effort, we explored two methods to extend the ELO ratings effectively.

Logical match-ups uses a rule-based system to simulate match-ups using transitive properties of the match-up history. Say we have three paintings: A, B, and C, and the history shows that the labeller preferred A over B and B over C. In that case, the labeller should also prefer A over C:

$$\text{If } A > B \text{ and } B > C \text{ then } A > C.$$

Using this approach, we artificially increased the number of matchups manyfold, thus reducing manual labour.

Match classification using CLIP's [1] visual transformer model to embed the images into a 512-dimensional vector, hopefully capturing the essential features of the images. The simple MLP classifier takes two embedded images and predicts the winner. To make the model invariant to the order of the images, it has the structure seen in Figure 2.

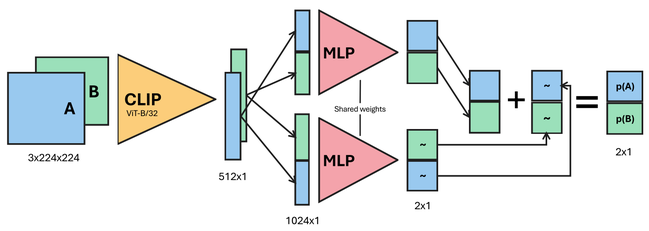

Figure 2. Match predictor model structure. The MLP structure is $1024 \to 128 \to 16 \to 2$ with ReLU and 50% dropout.

Table 1. 10 fold cross-validated CLIP match model results.

	Kasper		Kristian		Kristoffer	
	CE	Accuracy	CE	Accuracy	CE	Accuracy
CLIP+MLP	.625 ± .006	.65 ± .01	.573 ± .011	.70 ± .02	.639 ± .017	.64 ± .01

The 2500 matches are split into 8:1:1 by images, such that test images only occur in matches in the test set and validation images also do not occur in the training data.

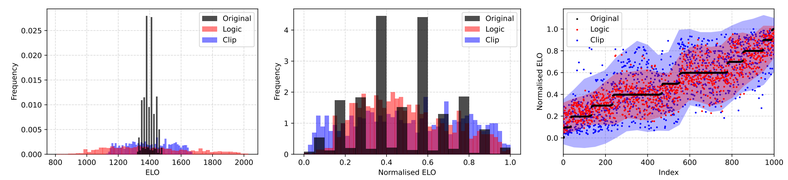

Figure 3. Left: Unnormalised ELO scores. Middle: Normalised ELO scores which our preference models are trained on. Right: Actual ELO scores for each image sorted by *original* ELO depending on the method used. The shaded areas denote the binned mean and stds for *Logic* and *Clip*, respectively.

Data

Post-labelling, we obtained four datasets per member for 1,000 paintings:

Dataset	Matches	Range
Scale	(1,000)	1 - 9
Original ELO	2,500	1,300 - 1,500
Simulated ELO using CLIP	12,500	1,100 - 1,700
Simulated ELO using logical	276,000	300 - 2,300

Table 2. Maximum number of matches and ELO range across all three labellers.

A model is to be trained to predict scale/ELO for each dataset to compare the merits of each method. To compare the methods, 9-fold cross-validation was used on the datasets split into a train, test, and validation set, corresponding to 80%, 10%, and 10%.

Pros and cons of scale and ELO ratings.

Scale

Pros	Cons
+ Fixed range	- Class imbalance
+ Fast labelling	- Biased

ELO

Pros	Cons
+ Evenly distributed	- Dynamic range
+ Unbiased	- Slow labelling

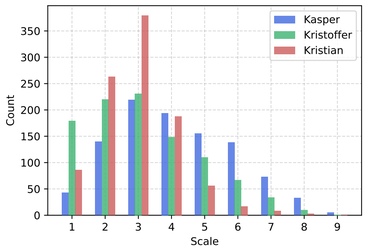

Figure 4. Distribution of scale ratings for all group members.

CLIP

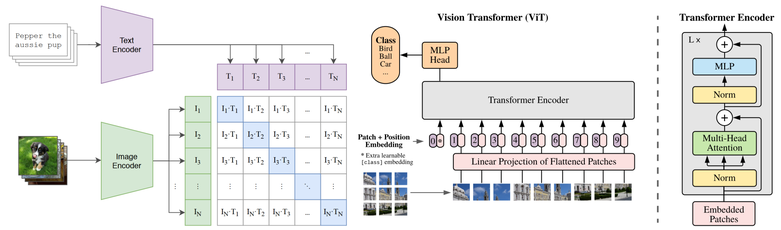

Figure 5. Left: CLIP dual encoder setup with contrastive loss [1]. Right: Visual transformer (ViT) structure used for the visual encoder in CLIP [2].

Predicting User Preferences

A basic painting score regression setup based on CLIP is used to predict user preferences; see Figure 6. The model structure is based on preliminary experiments.

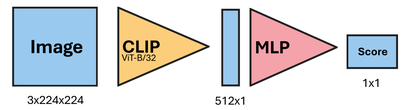

Figure 6. Basic model for predicting a painting's score based on CLIP's image feature extractor. The MLP structure is $128 \to 32 \to 1$ with ReLU and 70% dropout.

Dataset	MSE			Kendall's τ Coefficient		
	Kasper	Kristian	Kristoffer	Kasper	Kristian	Kristoffer
Scale	.036 ± .004	.012 ± .001	.028 ± .002	.46 ± .43	.58 ± .02	.42 ± .02
Original	.027 ± .002	.036 ± .002	.062 ± .004	.42 ± .02	.42 ± .03	.30 ± .02
CLIP	**.009** ± .001	**.011** ± .001	**.011** ± .001	**.79** ± .01	**.79** ± .00	**.78** ± .01
Logical	.071 ± .010	.054 ± .004	.076 ± .005	.40 ± .03	.37 ± .03	.26 ± .01

Table 3. Cross-validated test MSE and Kendall's τ coefficient ± one standard deviation. The best value is highlighted in bold.

Predicting User Preferences II

Since *true* labels are difficult to define, *worst*, *middle* and *best* images from 1000 unseen images are used to fairly compare the models. All images selected by all models were ranked on a 9-point scale in random order by the group members.

Table 4. Mean scale-score for each group of 9 images (bottom, middle, top) from 1000 unseen images ranked by the models.

	Kasper			Kristian			Kristoffer		
	Worst	Middle	Best	Worst	Middle	Best	Worst	Middle	Best
Scale	1.4 ± .96	3.9 ± 1.3	5.9 ± 2.1	**1.2** ± .42	2.2 ± 1.0	**6.9** ± .87	1.4 ± .50	3.6 ± 1.3	**5.6** ± 2.0
Original	1.3 ± .67	4.4 ± 1.2	**7.4** ± .96	1.4 ± .50	2.8 ± 1.8	6.8 ± .92	1.4 ± .68	3.1 ± 1.1	4.9 ± 1.4
CLIP	**1.2** ± .42	3.2 ± 1.1	6.6 ± 1.8	1.2 ± .42	2.9 ± .99	6.8 ± .92	**1.2** ± .42	3.9 ± 1.9	5.3 ± .94
Logical	1.4 ± .96	5.2 ± 1.6	6.2 ± 1.9	1.9 ± .47	3.0 ± 1.3	6.2 ± 1.5	1.6 ± .50	2.9 ± 1.2	5.2 ± 1.6

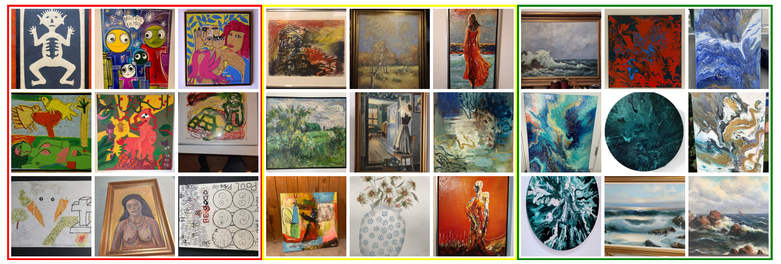

Figure 7. Left: 9 lowest-rated images. Middle: 9 middle most images. Left: 9 highest-rated images. All according to CLIP's predictions for Kristian's preferences.

Conclusion - Further Research

- ELO simulation methods did not seem to improve the prediction of user preferences.
- A larger preference dataset would be nice, but labelling is time-consuming.
- Develop an appropriate method of comparing different models.
- Develop a diffusion model for generating paintings with user preferences.

References

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, vol. abs/2103.00020, 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.