What is the Color of Serendipity? Investigating the Use of Language Models for Semantically Resonant Color Generation

Shahreen Salim, Tanzir Pial and Klaus Mueller

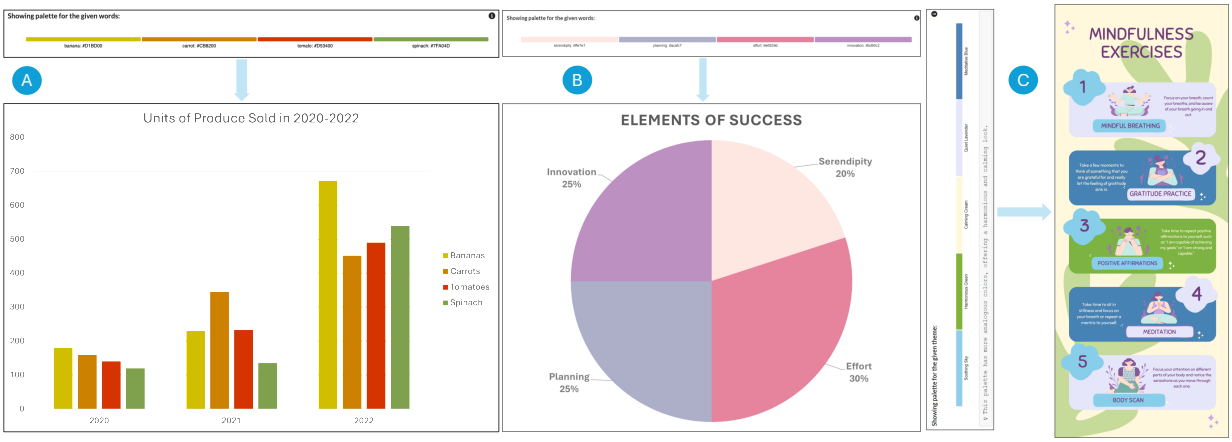

Fig. 1: Three Examples of Visualizations using Semantically Resonant Colors. The figure shows a bar chart (A), a pie chart (B), and an infographic (C) along with the colors returned by the Concept2Color interface. The bar chart, piechart, and infographics color were generated by CLIP, RoBERTa, and GPT-based models, respectively. Colors for (A) and (B) were directly generated from the input categories, whereas, the infographic colors are part of a cohesive palette returned by GPT 4 based on the theme "Mindfulness".

Abstract—Humans inherently connect certain colors with particular concepts in semantically meaningful ways that facilitate visual communication. These colors are known as semantically resonant colors. For instance, we associate "sky" and "ocean" with shades of blue, and "cherry" with red. In this paper, we investigate how language models, including Word2Vec, RoBERTa, GPT-4o mini and the vision language model CLIP generate and represent nuanced semantically resonant colors for diverse concepts. To achieve this, we utilized a large dataset of color names and concepts, tailored models for the structure of each language model, and developed an interactive web interface, **CONCEPT2COLOR**, as a use case. Additionally, we conducted experiments and a detailed analysis to assess the ability of these models to generate meaningful colors. Through these experiments, we examined how factors such as model design, training data and context affect the color output. Our findings reveal the capabilities and limitations of language models in processing and generating semantically resonant colors for concepts, thus contributing insights into how they depict semantic color-concept connections. These insights have implications for data visualization, design, and human-computer interaction, where leveraging effective semantic color generation can enhance communication and user experience.

Index Terms—Tabular Data, Text/Document Data, Datasets, Methodologies, Software Prototype, Domain Agnostic, Color Machine Learning Techniques

1 INTRODUCTION

The role of color in human communication and cognition is profound, extending far beyond mere aesthetics. Colors can convey deep conceptual connections and emotional significance, enabling complex ideas to be communicated visually. Humans often associate specific colors with certain concepts, a phenomenon deeply rooted in cognitive processes and cultural experiences. For example, the calming and serene association of the color blue is largely attributed to its prevalence in natural elements, such as the sky and ocean, reflecting color psychology [52]. In many Eastern cultures, red symbolizes good fortune and happiness, as seen in weddings and New Year celebrations [52].

Such colors that convey meaning beyond their visual appearance can be defined as "Semantically Resonant" colors. These associations span from concrete objects (e.g., cherry being red) to abstract emotions (e.g., "seeing red" as anger), and even to complex ideas like "serendipity," which may evoke varied colors based on personal or cultural context. We define semantic resonance as a color meaningfully supporting a concept, either through direct association or within a broader communicative and encoding context. With the advent of advanced language models, an intriguing question arises: How do these models interpret the semantics of color when associating colors with abstract and tangible concepts? This paper explores the role of language models in generating colors from text, aiming to uncover how well these systems reflect the semantic connections between concepts and color.

Using semantically and contextually resonant colors plays a crucial role in affective visualization and design, enhancing memorability and reducing processing time [12, 32]. For instance, when data visualizations like the bar chart shown in Fig. 1, employ colors that intuitively align with the data categories they represent, it facilitates a more intuitive understanding and comparison of the data, even in the absence of legends. Conversely, the calming colors in the infographic example in Fig. 1 not only complement the content (mindfulness exercises) but also reinforce the intended emotional response (calmness). Semanti-

- *Shahreen Salim, E-mail: ssalimaunti@cs.stonybrook.edu*
- *Tanzir Pial, E-mail: tpial@cs.stonybrook.edu.*
- *Klaus Mueller, E-mail: mueller@cs.stonybrook.edu.*
- *All authors are with Stony Brook University.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

cally resonant colors have other various applications such as education, assistive technologies, branding, marketing, creative design, etc.

Despite the benefits of using conceptually meaningful colors, determining the "right" color for a given concept remains a challenge. Designers traditionally rely on experience, guidelines, or user studies to choose appropriate colors, but this process is subjective and does not scale to the countless concepts one might wish to visualize. Different people or cultures may have varying associations (consider the concept "mourning" associated with black in Western contexts but white in parts of East Asia). Moreover, many concepts—especially abstract ones—lack obvious color correspondences. Previous research leveraged multi-modal approaches (e.g., image databases [32,34,36,58,64]) and linguistic analyses (e.g., text co-occurrences [64]), but these methods often struggle with abstract terms or nuanced metaphorical associations. Moreover, the extent to which language models alone capture and reflect these semantically resonant colors remains largely unexplored. There is a need for a more generalizable approach that can handle a wide range of concepts and capture subtle semantic-color associations without extensive hand-tuning or curated external resources. Advances in language models, trained on vast textual corpora, present a novel opportunity to automate the concept-color mapping process. Models like Word2Vec [46], RoBERTa [37], CLIP [56], and GPT-based architectures [53], including both zero-shot and fine-tuned GPT-4o-mini variants, encapsulate rich cultural and commonsense knowledge potentially useful for semantically resonant color generation. The zero-shot model allows us to evaluate the extent to which general pretraining alone encodes semantically resonant color associations, while the fine-tuned version is trained on a curated set of concept–color examples to better align its outputs with human-like color mappings. By comparing these models, we assess how well each can suggest colors that "make sense" for a given input concept, and we identify which techniques most effectively capture human-like color semantics. In this paper, we explore two primary research questions: **(1)** How effectively can different language models associate semantically resonant colors with diverse concepts? and **(2)** How do factors like model architecture, training data, and concept characteristics influence the semantic-color generation process? We specifically investigate how concept abstraction and color-association ambiguity affect model agreement on representative colors, acknowledging that concrete concepts ("banana") yield high consistency, whereas abstract or ambiguous terms ("freedom","cold") produce variability in associations.

To demonstrate the ecological validity and practical utility of our approach, we introduce **Concept2Color**, an interactive system enabling users to generate and compare color suggestions from various language models for any input concept. Designers, educators, and visualization professionals can leverage Concept2Color for semantically resonant, data-driven color choices, particularly aiding in interpreting abstract or ambiguous concepts with unclear color associations. We also highlight a diverse set of real-world use cases spanning educational tools, affective visualization, branding, and product design, illustrating how Concept2Color can support nuanced, contextually resonant color assignments.

The primary objectives of our research are twofold: **(1)** to evaluate the performance of different language models in generating or associating colors with concepts and **(2)** to explore the impact of factors such as model architecture, training data, context, on the semantically resonant color generation process. Our findings contribute to a deeper understanding of language models' capabilities for generating semantically resonant colors, potentially informing the development of more nuanced and context-aware models. Our contributions are as follows:

- We built models that generate colors from textual concepts by leveraging five different language models.
- We analyzed and evaluated the impact of various factors on the generated colors.
- We designed and developed Concept2Color, a tool that generates semantically resonant colors from concepts, along with necessary user control and practical visualization examples.

Our paper is organized as follows: Sec. 2 reviews related work, and Sec. 3 presents our color generation approach. Section 4 provides both quantitative and qualitative evaluation, while Sec. 5 introduces the Concept2Color interface. Section 6 showcases real-world use cases, and Sec. 7 and 8 discuss implications, limitations, and future work.

2 Background and Related Work

This section reviews background on semantically resonant color generation, prior work on language models associating colors with concepts, and previous efforts integrating language models with visual tasks.

2.1 Color Semantics and Color-Concept Associations

Color semantics refer to intrinsic meanings attributed to colors (e.g., sea green → healing, light red → passion) [14,52]. In contrast, color-concept associations quantify how strongly concepts link to different colors, producing a distribution across color space [61]. While semantics capture shared meanings, associations reflect individual and contextual variability. Both influence object recognition [71], preferences [55], and visual reasoning in visualization [32,63]. Concrete and abstract concepts alike evoke color associations [32,58], though these are often context-dependent [61]. Design research, such as Kobayashi's Color Image Scale [31], offers curated mappings between colors and psychological descriptors (e.g., "elegant", "provocative"). These lexicons use fixed mappings, while data-driven models generalize to broader vocabularies. Large language models (LLMs) offer a new lens for studying these mappings. Haber et al. [23] show that LLM embeddings capture fine-grained distinctions in polysemous terms. Mukherjee et al. [49] used GPT-4 in a zero-shot setting taking concepts as inputs to predict association ratings over 58 discrete UW-71 colors. In contrast, we directly regress RGB values utilizing text embeddings, and our fine-tuned CLIP model outperforms GPT-4 variants on held-out data (Tab. 2).

2.2 Color Design in Data Visualization and HCI

Effective color design is essential in data visualization and human-computer interaction (HCI), enhancing aesthetics, comprehension, and decision-making [12,20]. Stone [70] proposed a perceptual model that adjusts CIELAB based on patch size and crowdsourced data to improve color distinctiveness in practical settings. Building on such foundations, tools like ColorCook [67], Colorgorical [22], and Palettailor [39] offer data-driven palette selection. Semantically resonant colors can further aid chart interpretation [32], memorability [10, 13], and contextual understanding [50,62]. Our work advances semantically resonant color generation by exploring how language models can support color design in visualization and HCI.

2.3 Multimodal Approaches to Semantic Color Generation

The prevalent approach for color generation using textual descriptions has traditionally adopted a multimodal strategy. Early methods leveraged online image databases (e.g., Google Images, Flickr) to identify color patterns associated with specific tags [32–34,36,58,64], offering comprehensive visual cues that facilitate mapping between language and color. In pursuit of refining these methods, recent studies have shifted towards utilizing self-supervised learning [29] and generative adversarial networks (GANs) [43] to enhance the multimodal color generation process. For instance, Havasi et al. [24] combined the Open Mind Common Sense Network with color data to devise a statistical model for semantic color selection, while Setlur and Stone [64] exploited natural language corpora (e.g., Google N-gram) to improve query ranking for image search. Researchers have also explored palette generation from text [9,26,35,64,66]. GenColor system [28] generate concept-aligned palettes by isolating dominant colors from synthesized images.[1] We explore the power of text embeddings aligned with visual embeddings through CLIP [56], a vision-language model trained on large-scale image-text pairs. By fine-tuning CLIP's text encoder

[1]While text-to-palette generation was not our primary focus in our Concept2Color interface (see Sec. 5), we incorporate a dedicated text-to-palette module that uses GPT-based models to create cohesive palettes from a given theme, complementing our core concept-to-single-color generation.

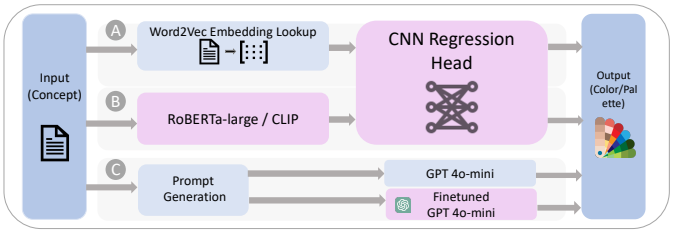

Fig. 2: Pipeline for generating colors and palettes from textual concepts. A, B, and C mark the components of Shallow-embedding models (Word2Vec), Deep transformer-based encoding model (RoBERTa) and Transformer based vision model (CLIP), and transformer-based generative models (GPT-4), respectively. The purple-colored components have trainable parameters. For GPT-based models, we provide functionalities to generate palettes along with individual colors.

for text-to-color generation, we investigate how language embeddings grounded in visual representations can enhance the semantic alignment between concepts and generated colors.

2.4 LLMs for Color Generation and Visualization

The inverse task of generating color names from a given color have been explored in [26, 60]. Other works have explored color association with word-level semantics [6, 45, 48]. Kawakami et al. [30] first approached color generation using only textual data by developing LSTM-based models that process color names to generate corresponding colors. Subsequent research [3, 38] has investigated mapping the text embedding space to a color embedding space using linear transformations with LLMs. LLMs have also demonstrated the potential to enhance visual data presentation by leveraging linguistic properties and user intent. Systems like ChartGPT [72] translate abstract linguistic inputs into precise visual representations, streamlining the visualization creation process and better capturing user intent. Furthermore, LLMs bridge linguistic descriptions with perceptual experiences, suggesting their relevance for creating more user-friendly visualization tools [44]. The development of systems such as C2Ideas exemplifies the seamless integration of linguistic insights into the design process, thereby enhancing user satisfaction and overall workflow [27]. Preliminary observations indicate that LLMs can adapt their color outputs based on a specified persona or alignment. For example, when prompted to adopt a more creative persona, models may generate bolder, unconventional color suggestions, while highly aligned models typically produce safer, broadly acceptable outputs. Although we do not delve deeply into persona effects in this paper, we acknowledge this factor as an intriguing aspect for future investigation [75].

3 Generating Semantically Resonant Colors

Similar colors are often associated to semantically related concepts, especially when the concepts are abstract. For example, red is associated with related concepts like danger and anger. In the past decade, many works in natural language processing [18, 46, 53] investigated establishing relationship between texts through creating semantically meaningful embeddings. It then makes sense to take advantage of these pre-trained embeddings and models to create textual embeddings rich with semantic information, which can be used by a regression model next to predict colors.

Figure 2 outlines our pipeline for generating semantically resonant colors from words and phrases. This approach leverages three classes of language models—(A) shallow embedding (Word2Vec [46]), (B) transformer-based encoders (RoBERTa [37], CLIP text encoder [56]), and (C) a transformer-based generative model (GPT-4o mini [54]). Both (B) and (C) are fine-tuned end-to-end for color prediction. Specifically, we treat text-to-color generation as a multivariate regression problem: an input text is embedded, passed through a regression head, and mapped to RGB values. We optimize the mean squared error loss combined over the three output channels.

3.1 Input and Output

Our method accepts single words or multi-word phrases in English as input which we refer to interchangeably as concept or color-name. Both colorable / concrete concepts, (e.g., "chalk", "corn") and non-colorable / abstract concepts, (e.g., "serendipity", "blind love") can be given as input. The model outputs a RGB triplet $\mathbf{c} = (R, G, B)$ within the range [0, 255]. When predicted colors appear muted, we optionally enhance their chroma via a vibrancy multiplier in CIELCh space. This feature is available through our *Concept2Color* interface, allowing users to tailor the color output to their preferences.

Color Name Dataset

We utilized the *Color Names* dataset [21], a comprehensive collection of over 30,282 uniquely named colors, each mapped to an RGB and HEX value. This dataset aggregates user-submitted color names and their corresponding hexadecimal codes, and is maintained as a community-driven project to explore the relationship between language and color. Each entry is a tuple $\{(C_i, c_i)\}$ where C_i is a concept and c_i is its associated RGB value. Concept lengths range from 1 to 6 words, with both median and mean lengths around 2, and a standard deviation of 0.55. Additional statistics on the dataset can be found in the supplemental materials.[2]

3.2 Encoder Models with Regression Head

We embed an input concept to a sequence of word/token embeddings using different models and then pass it to the Regression Head (RH) to infer the RGB values.

Shallow Embedding Model: Word2Vec

As a first step, we start with Word2Vec [46], a word embedding model. Word2Vec [46] learns static vector embeddings by placing semantically similar words in close proximity in the embedding space. Our dataset consists of many one or two word texts for which Word2Vec may turn out to be the ideal choice. We use the GoogleNews-vectors-negative300 version of word2Vec that has 300 dimension vectors for 3 million words and phrases with a total of 900 million frozen parameters, trained on the google news dataset of 100 billion words. Word2vec is less suited than modern large language models for complex or long-form text tasks, but it is deemed suitable for our text-to-color generation task, especially as a reasonable baseline against which we can compare the performance of the more advanced models. With a vocabulary size of 3 million, it contains representative embeddings of many common phrases. In fact its embedding table has multiple times more parameters (900 million) than some of the earlier large language models like BERT or RoBERTa.

For the proposed Word2vec-based model, each concept C undergoes pre-processing steps (e.g., decomposition into a sequence of words, standardization to lowercase, etc), and each word of the concept is converted to 300-dimensional word2vec embeddings, and all sequences are padded to a fixed length L. Out-of-vocabulary words default to an average embedding. Next, this sequence of L word embeddings of size $L \times 300$ is fed into a CNN-based regression head (RH). Alcorn [5] proposed training a Convolutional Neural Network (CNN) to estimate the relationship between pre-trained Word2vec embeddings and their associated colors, effectively bridging the gap between linguistic representations and color perception. We replicate this setting as a baseline.

Deep Transformer-based Encoder Model: RoBERTa

Though semantically meaningful, Word2Vec embeddings cannot capture contextual information since the embeddings are static. In contrast to Word2Vec, the first generation large language models like RoBERTa [37] employs a transformer architecture to capture deep contextual relationships in the output embedddings. For multi-word concepts such as "golden sunsets melt into dusk", we expect RoBERTa's

[2]We considered other publicly available color name datasets, including paint swatches [65], standard HTML color names, and user-defined colors [1,8,47,69]. We ultimately selected this dataset for its broad lexical and visual distribution. While not used here, alternative datasets may suit other goals.

contextual token embeddings to be superior for color generation. We fine-tune the RoBERTa-large model (355M parameters) for text-to-color prediction. A tokenized concept is processed by RoBERTa, whose output token embedding sequence is consumed by the RH. Fine-tuning proceeds with small learning rates to adapt the model's language understanding to color prediction.

One of the strongest feature of LLMs is that they can be fine-tuned using datasets as small as a few thousands as evidenced by BERT [18], GPT-1 [57] despite having millions of parameters. This is achieved by starting from well-established weights for millions of parameters learnt during pre-training which are then fine-tuned to a specific task like text-to-color generation. Employing conservative learning rates, as discussed in Section 3.2, ensures the model does not overfit. This strategy preserves the pretrained features to some extent while simultaneously allowing the model to learn task-specific nuances.

Transformer-based Vision-Language Model: CLIP

Both Word2Vec and RoBERTa produces semantically meaningful embeddings but the color informations are not necessarily encoded in the embeddings. Their understanding of color depends on how much colors of concepts are discussed in the text used in their pre-training, which may be little or much depending on the concept. We hypothesized a multi-modal model that can encode meaningful embeddings in both image and text space, would have already encoded color information in the text embeddings by utilizing the images. CLIP [56] jointly pretrains its transformer-based text and image encoders on large-scale image–text pairs using a contrastive loss, thereby aligning visual and textual embeddings within a shared space. Notably, its text encoder is trained from scratch to capture explicit visual-textual associations, rendering it particularly suitable for text-to-color generation. In contrast to LLMs like RoBERTa—which learn color semantics indirectly from text—CLIP directly learns to align phrases (e.g., "green leaf") with their corresponding visual representations (e.g., an image of a green leaf).

CLIP is not a generative model, it is designed to assess the similarity between images and texts via creating embeddings in a shared space. We first considered using a generative model (e.g., diffusion model [17]) which can take as input the text embedding of a concept from CLIP's text encoder, and then generate an image representing the color of the concept. We would then finetune the CLIP text encoder and the generative model end-to-end using our dataset. This is a common pipeline used for text to image generation. Since our target image can be as small as 1 pixel with the R, G, B values, we opted not to use a large generative model. Instead we employ the same RH we use for Word2Vec and RoBERTa here to keep our pipeline simpler to train and use. The CLIP text encoder (428M parameters) generates visually informed text embedding of a given concept which we then pass on to the RH to generate R, G, B values. We opt not to utilize the image encoder of CLIP for the sake of simplicity and also for keeping our pipeline consistent across different models.

Regression Head (RH)

The regression head (RH) maps token embeddings from text encoders to RGB triplets. We tested multiple RH architectures such as linear layers, multi-layer perceptrons, one-dimensional Convolutional Neural Networks (CNN). A 1D CNN with global max pooling consistently achieved the best performance across Word2Vec, RoBERTa, and CLIP. Formally, RH is defined as:

$$RH(E_C) = Linear(GMP(L_{Conv1D}(E_C)))$$

where E_C is the token embedding sequence for concept C produced by a text encoder in the previous step and GMP represents a global max pooling operation.

The convolutional layer employs 128 one-dimensional filters (kernel size = 1), followed by GMP and a linear projection to RGB space. A filter transforms each token embedding independently, after which GMP selects the highest-activated feature per filter. This design emphasizes the most semantically salient tokens—e.g., for "Burnt Red" we expect filters to predominantly activate on "red" while a subset may respond to "burnt" enabling nuanced color modulation. We expect this to be the case for many concepts in our dataset where one token should ideally dominate the final color and others should act as modifiers.

Table 1: Comparison of fine-tuning strategies for transformer-based color generation. RoBERTa performs best with a constant learning rate, whereas CLIP achieves lowest error using encoder learning rate decay. Frozen CLIP embeddings yield competitive performance indicating its visual linguistic understanding.

Model	Frozen Embeddings	Learning Rate Decay	CIEDE-2000 ↓
RoBERTa	Yes	-	18.53 ±2.14
	No	Yes	18.31 ±2.16
	No	No	**17.0** ±0.15
CLIP	Yes	-	16.52 ±0.16
	No	Yes	**16.35** ±0.16
	No	No	16.77±0.15

While the transformers in the text encoders can theoretically learn such token interactions via self-attention, fine-tuning them with aggressive learning rates risks "catastrophic forgetting" [41] of pre-trained weights / linguistic knowledge. We observed this empirically: transformers paired with linear layers without convolutional constraints produced unstable color mappings, leading us to adopt the CNN-based RH to maintain the integrity of the pre-trained representations.

Fine-tuning Strategies

For RoBERTa and CLIP, we tested multiple fine-tuning strategies to come up with the most optimal setting for each of them. Table 1 details the performance of both the models for different strategies using the CIEDE-2000 metric discussed in Sec. 4.1. We tested the following:

(i) Freezing the encoder and training only the RH. This mirrors Word2Vec's static embeddings training, forcing the RH alone to capture task-specific semantics. We see in Tab. 1 that frozen CLIP embeddings perform much better than frozen RoBERTa embeddings. This is expected since the frozen CLIP embeddings are already enriched with visual information as CLIP was designed to align text and image embeddings in the same space, but RoBERTa's frozen embeddings only have textual information.

(ii) Jointly fine-tuning the encoder and RH together with a conservative learning rate (e.g., $l_{enc} = l_{RH} = 10^{-5}$). This yielded the best performance for RoBERTa showing the need to tune its encoder parameters to learn color information associated to a concept.

(iii) Applying an aggressive learning rate for the RH (e.g., $l_{RH} = 10^{-3}$) coupled with an exponentially decaying rate of 0.7 for encoder layers, where later layers (closer to the RH) train faster than earlier ones. This setting produced the best performance for CLIP, yielding a balance between retaining the visually enriched embeddings and learning weights specifically for color generation. This shows we can still benefit from minor updates to CLIP's deeper encoder weights for the text-to-color generation task.

3.3 Generative LLM: GPT 4o-mini

After exploring encoder-based LLMs, we explore color generation using decoder-based LLMs, specifically GPT 4o-mini and a fine-tuned version of it. GPT's architecture is more suited to text-to-text generation problems as opposed to a regression problem like text-to-color generation. Nonetheless, GPT has become sufficiently commonplace that it will be one of the first options a person may think of using while figuring out a color for a concept in real life. This warrants its inclusion in our set of models for experimentation and comparisons.

Prompts

We craft zero-shot prompts that instruct GPT to respond as a "color expert" with the following system message:

> "You are a color expert who can generate a color based on the meaning and context of a given theme and how a human would visualize that word or phrase in color.
>
> Generate a color for the word or phrase C_i based on its meaning and context, as visualized by a human. Provide only the color name and RGB values in a Python list of dictionary objects, with each dictionary containing: {'name': color name, 'r': R value, 'g': G value, 'b': B value}."

Upon receiving responses from GPT we employ regular expressions to identify any deviations from the expected format.

Fine-tuning GPT 4o-mini

We use the OpenAI API to fine-tune GPT 4o-mini on the color-names dataset framing the task as text-to-text generation problem where the target R, G, B values are treated as text. It optimizes for next token prediction using cross entropy loss–not ideal for a numerical regression problem such as text-to-color generation. Nonetheless, GPT 4o-mini is the largest model we employed for this task and it can provide valuable insight into how good generative models are for regression tasks. We use the default options on the OpenAI API for the learning rate.

4 Evaluation

In this section, we comprehensively evaluate color generation across different language models, demonstrating their ability to generate semantically resonant colors. We also discuss the effectiveness, challenges, and insights gained from each model.

4.1 Model Training and Performance

We primarily use CIEDE2000 [40] as the evaluation metric for model performances. It captures the perceptual difference between two colors quantitatively with CIEDE2000 values less than 1 indicating color differences imperceptible to the human eye. We compute the mean CIEDE2000 color difference between the predicted color $\hat{c}_i$ and the actual color c_i given the concept C_i. We also report Mean Absolute Error (MAE) and R^2 in Tab. 2.

Training Encoder Models. We first exclude 10% of the color-names dataset as our final test set and use the remaining 90% for training and cross-validation. We use the test set only once in the very end to get final model test scores. We conduct a 10-fold cross validation on the training set for Word2Vec, RoBERTa, and CLIP. We do not use a fixed number of epochs, instead we employ early stopping patience = 5, i.e., if the model does not improve for 5 consecutive epochs based on the validation metric of CIEDE2000 on the validation set, we terminate the training. All models required < 15 epochs in all folds. We run the cross-validation multiple times to figure out the optimal set of hyper-parameters. The cross-validation results are reported in Tab. 2.

Training GPT Models. For GPT 4o-mini, we finetune the model using the entire training set once. The temperature setting T decides how non-deterministic GPT will be. We experiment with temperature setting of 0 to make the model completely deterministic, predicting always the RGB values it believes to be most probable. We also used the default temperature setting of 1 for both the models to strike a balance between creativity and coherence in the generated color suggestions. With T=1, we ran inference 5 times for each sample and took the average as the final prediction. Table 2 shows the temperature setting of 0 to provide better performance according to CIEDE2000 in both models. The supplemental materials can be referred to for a discussion on how the predictions by GPT vary with T=1 for the same concept.

Comparative Analysis. Our best-performing model, based on CLIP, achieves +13.69 improvement in MAE over GPT with zero shot (T = 1). Notably, all variants of the GPT models get negative or near-zero test R^2 signifying the fine-tuned GPT models have over-fit on the training set while the zero-shot GPT models are not much better than predicting the mean of the training set. This shows that the text-to-text generation objective is not suitable for a numeric regression task like color generation. Unfortunately, that is the only fine-tuning option OpenAI provides at the moment.

Table 2: Comparison of 10-fold cross-validation (CV) and test-set results across three metrics. Word2Vec has the highest number of parameters, but they are frozen by design. CLIP outperforms all models. The parameter count for GPT models is not publicly available.

Model, # of parameters	MAE ↓		R^2 ↑		CIEDE2000 ↓	
	Dev (CV)	Test	Dev (CV)	Test	Dev (CV)	Test
Baseline (Mean)	-	56.26	-	0.0	-	24.50
Word2Vec, 900M	40.49 ±0.48	40.92	0.40 ±0.01	0.39	17.61 ±0.11	17.80
Zero-shot GPT (T=0)	-	50.92	-	-0.07	-	21.38
Zero-shot GPT (T=1)	-	51.40	-	0.03	-	24.08
Fine-tuned GPT (T=0)	-	53.11	-	-0.21	-	19.97
Fine-tuned GPT (T=1)	-	58.35	-	-0.17	-	27.77
$\text{RoBERTa}_{\text{Large}}$, 355M	38.42 ±0.49	38.4	0.43 ±0.01	0.41	17.0 ±0.15	17.0
CLIP, 428M	**37.63** ±0.65	**37.71**	**0.47** ±0.01	**0.46**	**16.35** ±0.16	**16.4**

All the encoder models significantly outperformed the mean baseline across all three metrics, indicating their ability to capture meaningful relationships between concepts and colors. Among these, CLIP achieved the best performance, followed by RoBERTa, with Word2Vec ranking third. Both RoBERTa and CLIP benefit from dynamic contextual embeddings enabled by the transformer architecture compared to Word2Vec's static embeddings. The improvement shown by the vision-language model CLIP over RoBERTa can be attributed to the visual-linguistic relationship it has learnt during pre-training.

4.2 Comparison with Models Enhanced by Google Images

Heer and Stone [26] created a color dictionary based on the XKCD dataset using probabilistic models and handcrafted rules. Building upon this, Setlur and Stone [64] introduced the concept of colorability and proposed a multimodal approach to retrieve representative colors for 36 highly colorable terms using Google Image Search. In this section, we compare their image retrieval-based results with colors generated directly from text by our models.

Figure 3 shows a strip plot of CIEDE2000 distances between model-generated and reference XKCD colors. Overall, GPT-4 models performed best, with the finetuned variant achieving the lowest median error, followed closely by its zero-shot counterpart. Both models outperformed the image-based approach of Setlur and Stone [64], despite relying solely on textual input. The CLIP and RoBERTa models showed comparable performance to Setlur and Stone, with RoBERTa exhibiting slightly more variability. Word2Vec produced the largest perceptual errors, indicating weak alignment with human color naming.

The distribution of errors is notably non-normal, warranting the use of non-parametric statistical tests. A Friedman test revealed significant differences in model performance across all six models (χ^2=42.54, p <0.0001). This result held even when excluding Word2Vec (χ^2=9.94, p=0.041), and when comparing only Setlur and Stone, CLIP, and RoBERTa (χ^2=7.72, p=0.021), suggesting robust differences among models even within narrower performance ranges.

To identify which model pairs differ significantly, we conducted Wilcoxon signed-rank tests with Bonferroni correction. These tests showed that Word2Vec's predictions differed significantly from all other models (p <0.001). In contrast, no statistically significant differences were observed between the top-performing models (GPT-4 variants, CLIP, RoBERTa) or between these models and Setlur and Stone's image-based approach. These findings reinforce the strength of large language models—particularly GPT-4—in generating perceptually accurate colors for semantically rich, familiar terms. The strong performance of zero-shot GPT-4 suggests that such terms are likely well-represented in its training corpus. While finetuning offered a modest performance gain, the lack of significant difference suggests that pretrained language models already encode strong word-color associations.

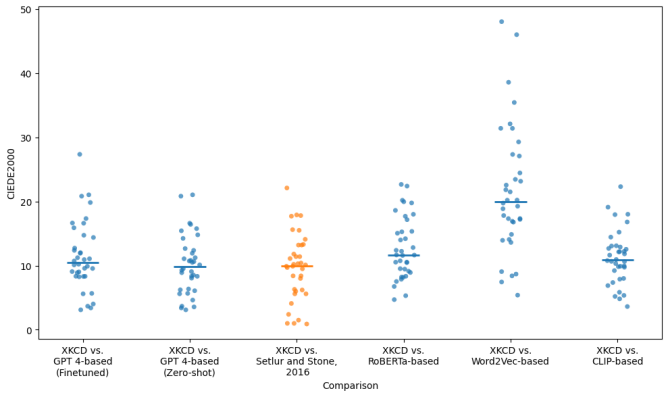

Fig. 3: CIEDE2000 distance between 36 XKCD colors and model-generated colors. Lower values indicate closer perceptual similarity. The zero-shot GPT yields the best performance, possibly owing to the XKCD color terms being well represented in its training corpus.

4.3 Embedding vs Perceptual Distance

We hypothesize that the fine-tuned RoBERTa and CLIP models learn to encode color information in the embeddings they generate. This is not true for the static Word2Vec embeddings that capture semantic relationship but very little color information. To qualitatively test this hypothesis, we present Fig. 4, where each heatmap visualizes the absolute difference between the embedding distance and the perceptual color distance for word pairs. A deeper red hue indicates a larger mismatch between the two distances, while lighter shades denote stronger alignment. The key idea is – distances in only the embedding spaces encoding color information should be highly correlated with the perceptual color distance.

In the Word2Vec model (left panel), we observe many deep red cells, suggesting frequent misalignment between embedding and perceptual distances. For example, the pair "banana" and "grape" (both fruits) is semantically similar, leading to a low embedding distance but colored yellow and red/green respectively, leading to a high perceptual difference. In contrast, both RoBERTa (middle panel) and CLIP (right panel), which were fine-tuned for color generation, show a lighter shade for the pair "banana" and "grape", suggesting these fruits' embeddings have shifted away from each-other due to encoding color information. This trend of having more near-white cells, reflecting better alignment between semantic and perceptual similarities, is evident in general for both RoBERTa and CLIP.

This contrast highlights how fine-tuning with color-specific objectives helps models like RoBERTa and CLIP generate embeddings that are more perceptually grounded. The increased number of white cells in these models demonstrates that they are better able to align embedding similarity with perceptual similarity, essential for generating meaningful colors for a concept.

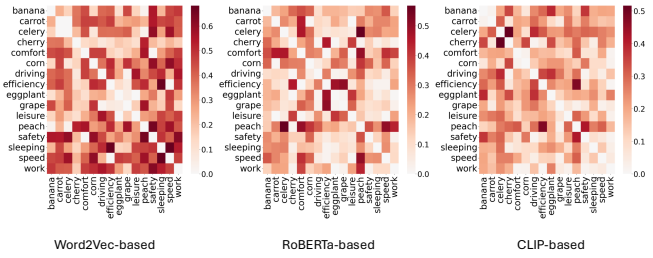

Fig. 4: Heatmaps demonstrating the difference between normalized Embedding Distance and Perceptual Distance of pairwise words. Shades of red denote the disparity of embedding distance and perceptual distance. Word2Vec has an average embedding and perceptual distance difference of 0.34 while the fine-tuned RoBERTa and CLIP have 0.2 and 0.18 respectively, highlighting a strong embedding-color perception correlation.

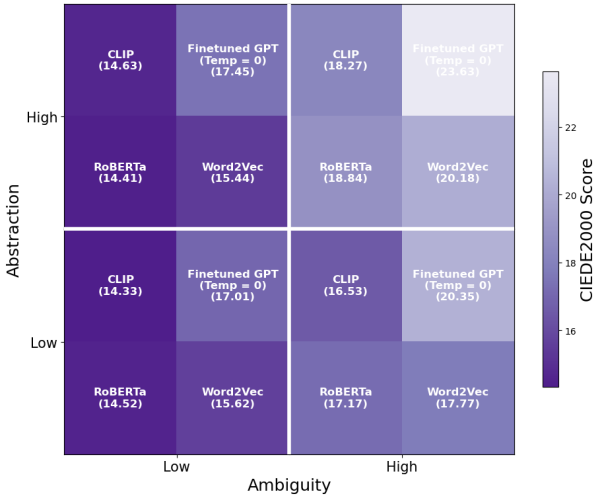

Fig. 5: Model performances on the four subsets of test set, stratified by ambiguity and abstractness. All model performances deteriorate as ambiguity and abstractness of concepts increase.

4.4 Abstract and Ambiguous Concepts.

Concepts vary widely in how consistently they are associated with specific colors. Two factors that shape these associations are abstraction (how concrete or grounded a concept is) and the ambiguity caused by many competing color associations that a concept might evoke. We operationalize and quantify both properties to better understand how they affect model performance and human perception.

Abstraction of Concept. We define abstraction as the degree to which a concept is concrete or sensory-based, following prior work in psycholinguistics. To quantify, we use concreteness ratings from Brysbaert et al. [16] and Muraki et al. [51]. For multi-word expressions, we first check if an entry exists in Muraki et al.'s dataset; if not, we compute a part-of-speech–weighted average of individual word concreteness scores using Brysbaert et al.'s norms. Nouns and adjectives are given higher weight, as they tend to carry more semantic content.

Ambiguity of Color. Concepts exhibit color ambiguity due to real-world chromatic variability (e.g., apples being green and red), subjective associations (e.g., sadness linked to blue or grey), or polysemy (e.g., apple the fruit vs the company) . While related to the notion of colorability presented in [64], our focus is conceptually the inverse. We propose a measure to quantify the color ambiguity of concepts. We compute word2vec embeddings for $k = 10$ color terms (e.g., *red*, *yellow*, etc.) and for the words in the target concept. We then calculate the association between a word w and the i-th color-term, $a_{i,w}$, via softmax on the cosine similarities. We hypothesize a word has high color ambiguity if its association score with more than one color-term is high. The ambiguity of a word w is then defined as the normalized Shannon's entropy:

$$\text{Ambiguity}(w) = \frac{-\sum_{i=1}^{k} a_{i,w} \log(a_{i,w})}{\log(k)} \quad (1)$$

The ambiguity of a concept is the average ambiguity of its constituent words. Lower values indicate stable color associations, while higher values suggest ambiguity.

To validate this approach, we evaluate 20 objects identified by Tanaka et al. [71] as having high (e.g., taxi, fire engine) or low (e.g., dog, lamp) color diagnosticity. Our ambiguity scores achieve a high Spearman rank correlation of -0.77 ($p < 1.07 \times 10^{-5}$) with Tanaka et al.'s diagnosticity classification. Finally, we determine a threshold equal to the mean value computed from the 10 high- and 10 low-diagnosticity objects to classify any concept as ambiguous or non-ambiguous.

Model Performance across Varying Levels of Abstraction and Ambiguity. To assess the influence of abstraction and ambiguity on color prediction, we partition the test set into four subsets based on the binary attributes of abstraction and ambiguity. While the overall

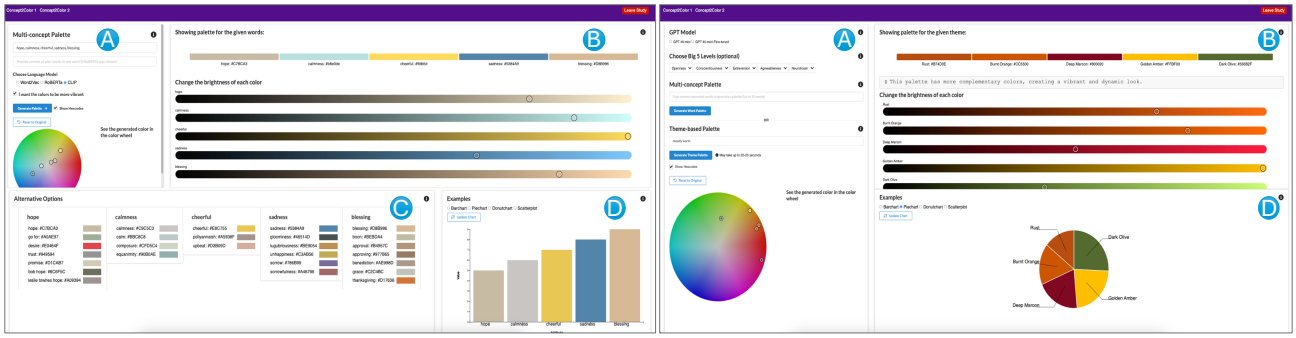

Fig. 6: Two segments of the Concept2Color interface. A, B, C, and D mark the input module, palette review and adjustment module, concept alternative module, and example visualization modules, respectively. The left panel shows Concept2Color 1 powered by Word2Vec, RoBERTa, and CLIP; the right panel shows Concept2Color 2 powered by GPT-40-mini (finetuned and zero-shot).

ranking of model performance remains consistent across these subsets, as shown in Fig. 5, all models achieve peak performance on the low-ambiguity, low-abstraction subset (deeper hue in all models). In contrast, performance systematically degrades as either abstraction or ambiguity increases, indicating that higher levels of these attributes are directly associated with greater prediction difficulty (lighter hues on the right).

5 The Concept2Color Interactive Interface

Building on the insights from our evaluation of language models for text-to-color generation, we developed the Concept2Color interface to translate these capabilities into a practical, user-friendly tool. The interface allows users to generate semantically meaningful color palettes from natural language input, drawing on the strengths and behaviors of the models analyzed in previous sections. Designed to support domains such as data visualization, graphic design, and web design, Concept2Color enables users to generate, refine, and preview concept-driven palettes interactively.

In the following we present the guidelines that drove our interface design, its architecture and modules, and usability study results.

5.1 Design Guidelines

To align the interface with our goals, we established the following design guidelines:

DG1: Tailoring to Model Strengths. The interface should leverage the distinct capabilities of each underlying language model.

DG2: Flexible Color Adjustments. Users should be able to refine generated palettes while maintaining semantic relevance.

DG3: Semantically Aligned Alternatives. The interface should suggest related concept-based alternatives to foster creative exploration.

DG4: Practical Visualization Examples. Incorporating real-world visualizations can improve the usability and relevance of generated colors.

5.2 Interface Architecture and Modules

The Concept2Color interface is a web-based system with a Python backend and a D3.js frontend. Following **DG1**, the system is split into two segments: the first leverages models Word2Vec, RoBERTa, and CLIP, while the second is optimized for generative models (GPT-4o-mini, both finetuned and zero-shot). Each segment comprises a combination of the following modules:

Multi-Concept Input Module (Fig. 6-A, left & right). This module allows users to enter comma-separated concepts into a textbox. A radio button interface lets users select from the available models. Additionally, users can opt for more vibrant color outputs via a checkbox. Input validation ensures a maximum of 10 concepts. This module supports **DG1** by aligning input structure with model type and supports **DG2** by allowing users to preview and iterate on color generation.

Theme-Based Input Module (Fig. 6-A, right). Unique to the GPT segment, this module lets users specify a single high-level theme (e.g., "sunset") to generate cohesive palettes where all colors are semantically related to the theme. This supports **DG1** by leveraging GPT's strength in holistic, theme-driven generation.

Palette Review and Adjustment Module (Fig. 6-B, left & right). To support **DG2**, this module presents the generated colors as labeled swatches, with hex codes for each. Users can manipulate the brightness of individual colors via sliders and view the palette on a color wheel. A reset button reverts colors to their original state. For theme-based palettes, we also show information on palette harmony to support designers in assessing color balance.

Concept Alternatives Module (Fig. 6-C, left). To facilitate **DG3**, this module displays alternative concepts and their associated colors for each original input, based on semantically related terms retrieved via the WordsAPI [2]. Clicking an alternative updates the palette in real-time. A reset button allows users to return to the original selection. This encourages exploration while maintaining semantic alignment.

Example Visualizations Module (Fig. 6-D, left & right). In line with **DG4**, this module provides example visualizations, including bar charts, pie charts, doughnut charts, and scatter plots. Users can switch between chart types and refresh them with adjusted palettes using an "Update Chart" button. This supports practical applications and allows users to preview the visual effect of the selected colors.

5.3 Interface Evaluation

We conducted an in-person usability study to assess the effectiveness and overall usability of Concept2Color. Our evaluation focused on both quantitative measures using the System Usability Scale (SUS) [73] and qualitative feedback to understand user satisfaction and identify areas for improvement.

Participants. Eight participants (age 25–34) voluntarily took part in the study. All held at least a Bachelor's degree, and there was an equal gender distribution. Most participants identified as Asian or Asian American. Full demographic details are available in the supplement.

Procedure. A research team member first introduced the system and explained its functionality. Participants completed a brief demographic questionnaire and were asked to interact with the interface through a set of predefined tasks, including generating palettes using both concept and theme-based inputs, adjusting palette brightness, and exploring example visualizations. Participants then completed the SUS questionnaire and answered open-ended questions about their experience.

Usability Results. The interface received a high average SUS score of 85.93, indicating excellent usability. As shown in Fig. 7, most users strongly agreed that the interface was easy to use, well-integrated, and quick to learn. Participants reported high confidence in using the system and low perceived need for support.

Qualitative Feedback and Suggestions. Participants described the interface as intuitive and visually appealing. The input structure, color feedback, and visualization modules were particularly praised. Theme-based palettes were frequently cited as "cohesive" and "aesthetic", especially for inputs like "ocean", "sunset", or "forest".

Several participants highlighted the usefulness of the color adjustment sliders, describing them as "fine-grained and responsive". There

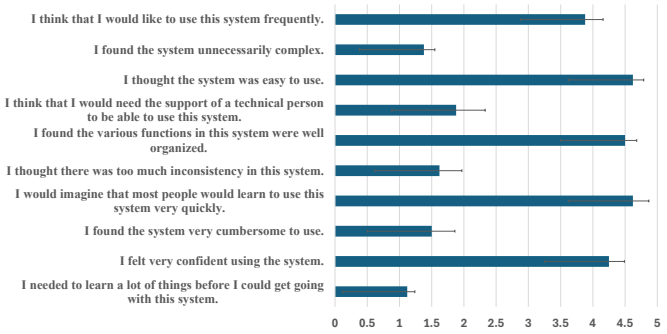

Fig. 7: Concept2Color interface mean usability ratings on the SUS scale. Error bars represent standard error. (1=Strongly Disagree, 5=Strongly Agree).

was some variation in the perceived usefulness of the example charts: while some found them essential for imagining real-world use, others felt the charts could be more tailored to design use-cases beyond data visualization. Suggestions for improvement included incorporating more nuanced synonym suggestions in the alternatives module, adding tooltips or explanations for palette harmony, and, expanding model options, including multilingual support. Overall, participant feedback confirmed the value of the system.

6 Use Cases

We applied our text-to-color generation models across diverse use cases in data visualization, UI design, and content creation—domains where color helps convey meaning, guide attention, and enhance user experience [59]. Using five model architectures within the Concept2Color interface, we show how semantically guided color selection supports contextually meaningful design. These examples demonstrate practical value across workflows and are detailed, along with prompts and outputs, in the supplemental materials.

6.1 Data Visualization and Storytelling

Color plays a central role in effective data storytelling by highlighting meaning and guiding visual attention [12]. Traditional tools often rely on manual color choices, which can lead to inconsistency, cognitive overload, or even unintended bias. Concept2Color streamlines this process by automatically generating semantically meaningful colors from natural language labels, enabling more coherent, contextually aligned narratives. The example in Fig. 8-A shows an application for a financial or business dashboard, where we used Concept2Color to assign meaningful colors to abstract categories: 'Revenue growth" → light green, "Market risk" → brick red, "Government regulations" → blue, and "Consumer trends" → pink, all within a pie chart. In contrast, Fig. 1-A presents a bar chart colorized by Concept2Color using more tangible categories.

6.2 Infographics Design

Infographics also benefit from this approach, as illustrated in Fig. 8-B. Unlike rigid, keyword-based highlighting, Concept2Color reflects the nuanced semantics of each term, assigning colors that reinforce both tone and message. In the infographic, stress-related phrases like "High Workload" and "Lack of Support" are mapped to red-orange hues that convey urgency, while terms such as "Reduced Absenteeism" and "Higher Retention" are paired with calming blues and teals to reinforce a sense of stability and improvement. The phrase "Benefits of a Mentally Healthy Workplace" is rendered against a vibrant yellow background—an energizing color that draws attention and highlights the positivity of the message without distracting from the overall narrative.

6.3 User Interface Design

Color plays a pivotal role in user interfaces (UI), influencing user experience, mood, and readability [25]. Our method enables adaptive UI design by dynamically adjusting interface elements based on semantic meaning or user sentiment. This is especially relevant for mood-adaptive applications and context-aware notifications. Prior work in affective computing has explored emotion-aware interfaces [12]; our models extend this by enabling open-domain color mapping for any word or phrase, without relying on predefined lists. For example, in Fig. 8-C we used Concept2Color to color-code message types in an email inbox UI: "Urgent alerts" (e.g., "Security Warning") → red, "Updates" (e.g., "Shipping Update") → grey, "Positive emails" (e.g., "Payment received") → green, and "Warnings" (e.g., "Low balance") → orange. In practice, a fine-tuned model like RoBERTa can analyze incoming messages and apply contextually appropriate colors to highlight urgency, helping users scan key information more efficiently. Lightweight models like Word2Vec can run on mobile devices for real-time, low-latency color adaptation. This boosts ecological validity: as users naturally express emotions and priorities through language, a UI that responds in the same color language feels more engaging.

6.4 Education and Cognitive Science (Memory Aids and Knowledge Visualization)

Text-to-color mappings can enhance education by using color as a cognitive aid. Research shows that color-coded materials improve recall and reduce cognitive load [19]. Our models automate this by dynamically assigning semantically meaningful colors to key terms in textbooks or e-learning content. For example, an e-learning platform could highlight different types of cells in relevant colors, reinforcing memory through semantically meaningful visual associations (see Fig. 8-D for an example we created with the help of Concept2Color). This supports cognitive theories that multi-sensory encoding—combining verbal and visual cues—strengthens memory [19]. Unlike manual color schemes, Concept2Color scales across curricula, ensuring consistency while adapting to new terms and offers a scalable, data-driven way to improve retention, comprehension, and semantic organization in education.

6.5 Creative Design and Semantic Search

Mapping language to color unlocks new possibilities in design, branding, and digital art. Concept2Color automates color selection based on semantic meaning, supporting creative workflows where color reflects intent. Designers can input keywords like "innovative," "trustworthy," or "eco-friendly" to generate cohesive palettes aligned with brand messaging. Beyond branding, artists can use text-to-color mapping for expressive purposes—such as children's storybooks, poetry visualizations, or synesthetic art (see Fig. 8-E for an example)—assigning colors that match the emotional tone or cultural meaning. For instance, positive words may appear in warm tones (gold, orange), while somber words use cool hues (blue, gray), allowing dynamic, sentiment-driven color schemes. Text-driven color search also enhances creative exploration. Designers can query terms like "vintage" or "futuristic" to retrieve relevant palettes, bridging abstract concepts with visual output. Tools like Adobe's Generative Recolor [4] already point to this trend. Concept2Color's semantic search extends such capabilities with context-aware mappings powered by advanced language models, adapting to emerging language and domain-specific terms.

6.6 Brand Identity and Marketing

Color plays a crucial role in brand identity and marketing, shaping consumer perception and emotional engagement [42]. Concept2Color provides a data-driven method for selecting brand colors based on textual descriptors, aligning messaging with visual identity. Branding agencies often define company personalities through keywords (e.g., "youthful," "energetic," "disruptive"), which our models can instantly translate into colors. Prior research shows that aligning brand colors with perceived traits (e.g., green for eco-friendly, blue for security) boosts consumer trust and purchase intent [42] (see Fig. 8-F). Additional applications include sentiment-driven social media management, where customer reviews are color-tagged, e.g., red for urgent complaints, gold for praise, gray for neutral feedback, and text analytics in advertising. Ultimately, Concept2Color brings the affective power of color into marketing and branding, enhancing visual consistency, engagement, and creativity.

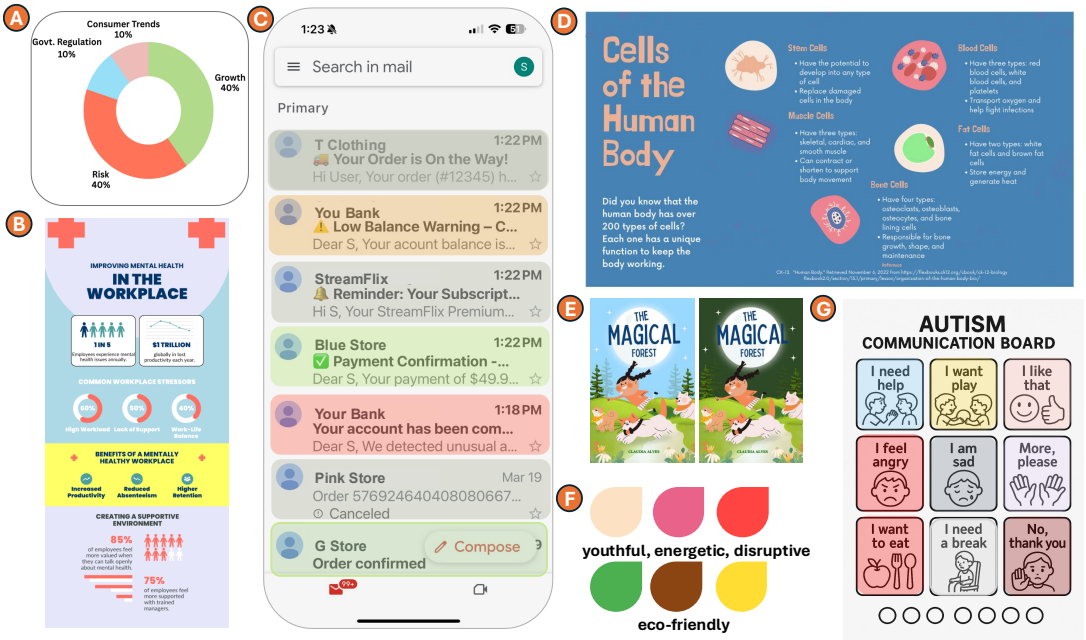

Fig. 8: Real-world applications of Concept2Color in diverse domains. All colors were generated using Concept2Colorthe mockups were produced by the authors using applications such as Canva. (A) Financial pie chart with semantically meaningful colors (RoBERTa). (B) Workplace mental health infographic (GPT-4o mini fine-tuned). (C) Email UI with semantic message highlighting (CLIP). (D) Educational biology poster using theme-based palette (GPT-4o mini zero-shot). (E) Storybook cover design with mood-aligned palette (Word2Vec). (F) Brand identity palettes for keywords like "youthful" and "eco-friendly" (GPT-4o mini fine-tuned). (G) Autism communication board with emotion-linked colors (GPT-4o mini zero-shot).

6.7 Assistive Technologies

In accessibility and assistive technology, text-to-color mappings offer an additional communication channel, especially for users with cognitive or linguistic differences. Color cues can enhance attention and comprehension, particularly for neurodivergent individuals [11]. For example, educators often use color-coded visuals to support children with autism spectrum disorder (ASD). Our system extends this by dynamically coloring text based on emotional or functional meaning, helping ASD users interpret language cues. A communication aid might display "I need a break" on a calming blue background, while distress messages appear in soft red—mirroring tools like emotion badges [7] (see Fig. 8-G). Users with dyslexia or language-processing disorders may benefit from consistent color-coding of names or sentiment-tinted tooltips for complex vocabulary. Unlike static tools, our models support a broad vocabulary, including idioms and slang. Context-aware color cues help reduce cognitive load and ambiguity.

7 Discussion, Limitations, and Future Work

Our study investigates how language models generate semantically resonant colors, revealing their capacity to model links between linguistic meaning and visual perception—echoing findings that language and color processing are intertwined in the brain [68]. The ability of language models to reproduce intuitive color associations suggests they have learned such mappings through large-scale text training.

Evaluation Strategies. We focused on two aspects: objective color quality, using CIEDE2000 on ≈3,000 human-annotated concept–color pairs, and system usability via a standard study. Since our dataset [21] reflects aggregated human judgments, strong model agreement on held-out test data serves as a proxy for human alignment. While this suggests early semantic resonance, validation through a full-scale human preference study remains important yet resource-intensive. A key limitation of our current evaluation is the lack of testing with professional designers, which we plan to address in future work. We aim to involve both expert (e.g., designers, artists) and non-expert participants to assess contextual appropriateness, subjective appeal, and creative potential. We also aim to compare model outputs with curated systems such as Kobayashi's Color Image Scale [31], examining alignment with historically validated affective mappings.

Dataset Biases and Model Scope. A central limitation of this work lies in the models' training data, which is largely English-language and Western-centric. This can lead to cultural and linguistic biases in color associations. Future efforts should incorporate multilingual and cross-cultural corpora to produce more inclusive and diverse associations.

From Colors to Palettes. We focused on generating single colors for concepts, but an important extension is palette generation. Models could be fine-tuned on text–palette datasets, or current outputs post-processed using color theory. This would expand the utility of semantic color generation in applications where harmony and contrast are critical.

Accessibility and Contrast. While Concept2Color produces semantically resonant hues, practical use also demands sufficient contrast and perceptual separability (e.g., "banana" vs. "carrot" in Fig. 1, or the purple contour in Fig. 8D). We plan to apply lightness or saturation adjustments to meet WCAG 2.1 thresholds [74], and include fall-back palettes optimized for legibility (e.g., ColorBrewer [15]). A contrast-checking module will help ensure outputs are both semantically aligned and visually accessible in real-world settings.

8 Conclusion

This paper presents a comprehensive investigation into how language models generate semantically resonant colors, revealing their capacity to capture complex associations between language and visual perception. By evaluating a range of models—including both pretrained and fine-tuned architectures—we show that language models can produce color associations that align with human intuition. Notably, we find that even general-purpose models like ChatGPT can yield color mappings that rival more specialized and labor-intensive image-search approaches, while downloadable models such as CLIP (when fine-tuned) offer even stronger performance, making them attractive for deployment in practical workflows.

Our analysis highlights not only the cognitive alignment between model-generated and human associations but also the limitations introduced by training data biases and model scope. To support broader exploration, we developed *Concept2Color*, an interactive tool that visualizes model outputs and enables users to generate semantically meaningful colors for a wide range of concepts. Overall, our findings contribute to a growing understanding of how language models can encode and express perceptual meaning. This work opens new avenues for future research in palette generation, cross-cultural analysis, and interdisciplinary evaluation, advancing both theoretical inquiry and practical applications in human-computer interaction and beyond.

9 Supplemental Materials

All supplemental materials are provided in the PCS Submission System as a .zip file. The description and location of all supplemental materials are provided as a separate document named "Supplemental Materials Details.pdf" inside the zipped folder.

Acknowledgments

This research was funded in part by a Stony Brook University seed grant.

References

[1] Rgb to color name mapping(triplet and hex). https://web.njit.edu/~walsh/rgb.html. Accessed: 2024-03-01. 3

[2] Wordsapi. https://www.wordsapi.com/. Accessed: 03-01-2024. 7

[3] M. Abdou, A. Kulmizev, D. Hershcovich, S. Frank, E. Pavlick, and A. Søgaard. Can language models encode perceptual structure without grounding? a case study in color. In A. Bisazza and O. Abend, eds., *Proceedings of the 25th Conference on Computational Natural Language Learning*, pp. 109–132. Association for Computational Linguistics, Online, Nov. 2021. doi: 10.18653/v1/2021.conll-1.9 3

[4] Adobe. Generative ai: Redefining productivity in creative imaging. https://www.adobe.com/uk/cc-shared/assets/roc/pdf/whitepapers-ebooks/generative-ai-report-redefining-creativity/2023-adobe-generative-ai-redefining-productivity-in-creative-imaging-uk.pdf, 2023. Accessed: 2025-03-30. 8

[5] M. A. Alcorn. Learning to name colors with word embeddings. https://github.com/airalcorn2/Color-Names, 2017. Accessed: 2024-03-01. 3

[6] J. Andreas and D. Klein. Grounding language with points and paths in continuous spaces. In *Proc. CoNLL*, pp. 58–67, 2014. doi: 10.3115/v1/W14-1607 3

[7] A. S. A. N. (ASAN). Color communication badges, 2014. Accessed: 2025-03-26. 9

[8] Back4App. Color class from all colors with name and rgb: Database hub. https://www.back4app.com/database/back4app/rgb-color-codes-and-names/color-dataset-via-api, 2024. Accessed: 2024-03-01. 3

[9] H. Bahng, S. Yoo, W. Cho, D. Park, Z. Wu, X. Ma, and J. Choo. Coloring with words: Guiding image colorization through text-based palette generation. In *Proc. ECCV*, pp. 443–459. Springer, 2018. doi: 10.1007/978-3-030-01258-8_27 2

[10] M. Bajo. Semantic facilitation with pictures and words. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 14(4):579–589, 1988. doi: 10.1037/0278-7393.14.4.579 2

[11] N. Barbazi and C. Wang. Perceiving through colors: Visual supports for children with autism. In J. Kalra, ed., *Human Factors in Aging and Special Needs. AHFE (2023) International Conference*, vol. 88 of *AHFE Open Access*. AHFE International, USA, 2023. doi: 10.54941/ahfe1003667 9

[12] L. Bartram, A. Patra, and M. Stone. Affective color in visualization. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, 11 pages, p. 1364–1374. Association for Computing Machinery, New York, NY, USA, 2017. doi: 10.1145/3025453.3026041 1, 2, 8

[13] L. H. Berry. The interaction of color realism and pictorial recall memory. ERIC Document ED334974, Auburn University, 1991. Accessed: 2024-03-01. 2

[14] C. Biggam. The semantics of colour: A historical approach. *The Semantics of Colour: A Historical Approach*, pp. 1–257, 04 2012. doi: 10.1017/CBO9781139051491 2

[15] C. A. Brewer. ColorBrewer 2.0. https://colorbrewer2.org/. Accessed: 2025-06-29. 9

[16] M. Brysbaert, A. B. Warriner, and V. Kuperman. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior Research Methods*, 46:904–911, 2014. doi: 10.3758/s13428-013-0403-5 6

[17] Z. Chang, G. A. Koulieris, H. J. Chang, and H. P. H. Shum. On the design fundamentals of diffusion models: A survey, 2025. 4

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. 3, 4

[19] I. Diachenko, S. Kalishchuk, M. Zhylin, A. Kyyko, and Y. Volkova. Color education: A study on methods of influence on memory. *Heliyon*, 8(11):e11607, 2022. doi: 10.1016/j.heliyon.2022.e11607 8

[20] S. L. Franconeri, L. M. Padilla, P. Shah, J. M. Zacks, and J. Hullman. The science of visual data communication: What works. *Psychological Science in the Public Interest*, 22(3):110–161, 2021. PMID: 34907835. doi: 10.1177/15291006211051956 2

[21] M. GitHub. Meodai/color-names: Large list of handpicked color names. https://github.com/meodai/color-names. Accessed: 2024-03-01. 3, 9

[22] C. C. Gramazio, D. H. Laidlaw, and K. B. Schloss. Colorgorical: Creating discriminable and preferable color palettes for information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):521–530, 2017. doi: 10.1109/TVCG.2016.2598918 2

[23] J. Haber and M. Poesio. Polysemy—evidence from linguistics, behavioral science, and contextualized language models. *Computational Linguistics*, 50(1):203–247, 2024. doi: 10.1162/coli_a_00500 2

[24] C. Havasi, R. Speer, and J. Holmgren. Automated color selection using semantic knowledge. In *Proc. AAAI Fall Symposium on Commonsense Knowledge*. AAAI, 2010. Accessed: 2024-03-01. 2

[25] F. Hawlitschek, L.-E. Jansen, E. Lux, T. Teubner, and C. Weinhardt. Colors and trust: The influence of user interface design on trust and reciprocity. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 590–599. IEEE, 2016. doi: 10.1109/HICSS.2016.80 8

[26] J. Heer and M. Stone. Color naming models for color selection, image editing and palette design. In *ACM Human Factors in Computing Systems (CHI)*, 2012. doi: 10.1145/2207676.2208547 2, 3, 5

[27] Y. Hou, M. Yang, H. Cui, L. Wang, J. Xu, and W. Zeng. C2ideas: Supporting creative interior color design ideation with a large language model. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, article no. 172, 18 pages. Association for Computing Machinery, New York, NY, USA, 2024. doi: 10.1145/3613904.3642224 3

[28] Y. Hou, X. Zeng, Y. Wang, M. Yang, X. Chen, and W. Zeng. GenColor: Generative color-concept association in visual design. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*, 2025. doi: 10.1145/3706598.3713418 2

[29] R. Hu, Z. Ye, B. Chen, O. van Kaick, and H. Huang. Self-supervised color-concept association via image colorization. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):247–256, 2023. doi: 10.1109/TVCG.2022.3209481 2

[30] K. Kawakami, C. Dyer, B. Routledge, and N. A. Smith. Character sequence models for colorful words. In J. Su, K. Duh, and X. Carreras, eds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1949–1954. Association for Computational Linguistics, Austin, Texas, Nov. 2016. doi: 10.18653/v1/D16-1202 3

[31] S. Kobayashi. *Color Image Scale*. Kosdansha International, 1991. 2, 9

[32] S. Lin, J. Fortuna, C. Kulkarni, M. Stone, and J. Heer. Selecting semantically-resonant colors for data visualization. *Computer Graphics Forum (Proc. EuroVis)*, 2013. doi: 10.1111/cgf.12127 1, 2

[33] S. Lin and P. Hanrahan. Modeling how people extract color themes from images. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, 10 pages, p. 3101–3110. Association for Computing Machinery, New York, NY, USA, 2013. doi: 10.1145/2470654.2466424 2

[34] A. Lindner, N. Bonnier, and S. Süsstrunk. What is the color of chocolate? – extracting color values of semantic expressions. *Conference on Colour in Graphics*, 6, 01 2012. doi: 10.2352/CGIV.2012.6.1.art00062 2

[35] A. Lindner and S. Süsstrunk. Automatic color palette creation from words. *Color and Imaging Conference*, 21, 01 2013. doi: 10.2352/CIC.2013.21.1.art00012 2

[36] A. J. Lindner, B. Z. Li, N. Bonnier, and S. Süsstrunk. A large-scale multi-lingual color thesaurus. In *Proc. Color and Imaging Conf. (CIC)*, pp. 30–35, 2012. doi: 10.2352/CIC.2012.20.1.art00006 2

[37] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *CoRR*, abs/1907.11692, 2019. 2, 3

[38] P. Loyola, E. Marrese-Taylor, and A. Hoyos-Idrobo. Perceptual structure in the absence of grounding: the impact of abstractedness and subjectivity in color language for LLMs. In H. Bouamor, J. Pino, and K. Bali, eds., *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 1536–1542. Association for Computational Linguistics, Singapore, Dec. 2023. doi: 10.18653/v1/2023.findings-emnlp.102 3

[39] K. Lu, M. Feng, X. Chen, M. Sedlmair, O. Deussen, D. Lischinski, Z. Cheng, and Y. Wang. Palettailor: Discriminable colorization for categorical data. *IEEE Transactions on Visualization and Computer Graphics*, PP:1–1, 10 2020. doi: 10.1109/TVCG.2020.3030406 2

[40] M. R. Luo, G. Cui, and B. Rigg. The development of the cie 2000 colour-difference formula: Ciede2000. *Color Res. Appl.*, 26(5):340–350, 2001. doi: 10.1002/col.1049 5

[41] Y. Luo, Z. Yang, F. Meng, Y. Li, J. Zhou, and Y. Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning, 2025. 4

[42] T. Maghraby, A. Elhag, R. Romeh, D. Elhawary, and A. Hassabo. The psychology of color and its effect on branding. *Journal of Textiles, Coloration and Polymer Science*, 21(2):355–362, 2024. doi: 10.21608/jtcps.2024.259014.1270 8

[43] P. Maheshwari, N. Jain, P. Vaddamanu, D. Raut, S. Vaishay, and V. Venkatesh. Generating compositional color representations from text. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pp. 3675–3679. ACM, 2021. doi: 10.1145/3459637.3482346 2

[44] R. Marjieh, I. Sucholutsky, P. van Rijn, N. Jacoby, and T. L. Griffiths. Large language models predict human sensory judgments across six modalities. *Sci. Rep.*, 14(1):12785, 2024. doi: 10.1038/s41598-024-72071-1 3

[45] B. McMahan and M. Stone. A bayesian model of grounded color semantics. *Transactions of the Association for Computational Linguistics*, 3:103–115, 2015. doi: 10.1162/tacl_a_00126 3

[46] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space, 2013. 2, 3

[47] M. K. . C. M. S. G. S. C. . mk.bcgsc.ca. Color names. https://mk.bcgsc.ca/colornames/. Accessed: 2024-03-01. 3

[48] S. Mohammad. Colourful language: Measuring word-colour associations. In F. Keller and D. Reitter, eds., *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*, pp. 97–106. Association for Computational Linguistics, Portland, Oregon, USA, June 2011. 3

[49] K. Mukherjee, T. T. Rogers, and K. B. Schloss. Large language models estimate fine-grained human color-concept associations, 2024. 2

[50] K. Mukherjee, B. Yin, B. E. Sherman, L. Lessard, and K. B. Schloss. Context matters: A theory of semantic discriminability for perceptual encoding systems. *IEEE Transactions on Visualization & Computer Graphics*, 28(01):697–706, jan 2022. doi: 10.1109/TVCG.2021.3114780 2

[51] E. J. Muraki, S. Abdalla, M. Brysbaert, and et al. Concreteness ratings for 62,000 english multiword expressions. *Behavior Research Methods*, 55:2522–2531, 2023. doi: 10.3758/s13428-022-01912-6 6

[52] J. Olesen. Color meanings: Symbolism and psychology of colors. https://www.color-meanings.com/, 2024. Accessed: 2024-03-20. 1, 2

[53] OpenAI. Gpt-4 technical report, 2024. 2, 3

[54] OpenAI. *GPT-4o mini: advancing cost-efficient intelligence*. OpenAI. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/, July 2024. Accessed: March 2025. 3

[55] S. Palmer, K. Schloss, and J. Sammartino. Visual aesthetics and human preference. *Annual review of psychology*, 64, 09 2012. doi: 10.1146/annurev-psych-120710-100504 2

[56] 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. In M. Meila and T. Zhang, eds., *Proceedings of the 38th International Conference on Machine Learning*, vol. 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. 2, 3, 4

[57] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. Improving language understanding by generative pre-training. 2018. 4

[58] R. Rathore, Z. Leggon, L. Lessard, and K. B. Schloss. Estimating color-concept associations from image statistics. *IEEE Transactions on Visualization & Computer Graphics*, 26(01):1226–1235, jan 2020. doi: 10.1109/TVCG.2019.2934536 2

[59] I. Santos, S. Gama, and D. Gonçalves. Cognihue: Studying the cognitive effect of color in hci. Master's thesis, Instituto Superior Técnico, 2017. 8

[60] B. Schauerte and R. Stiefelhagen. Learning robust color name models from web images. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 3598–3601, 2012. 3

[61] K. B. Schloss. Color semantics in human cognition. *Current Directions in Psychological Science*, 33(1):58–67, 2024. doi: 10.1177/09637214231208189 2

[62] K. B. Schloss, Z. Leggon, and L. Lessard. Semantic discriminability for visual communication. *IEEE Transactions on Visualization & Computer Graphics*, 27(02):1022–1031, feb 2021. doi: 10.1109/TVCG.2020.3030434 2

[63] K. B. Schloss, L. Lessard, C. S. Walmsley, and K. Foley. Color inference in visual communication: the meaning of colors in recycling. *Cognitive research: principles and implications*, 3:1–17, 2018. doi: 10.1186/s41235-018-0090-y 2

[64] V. Setlur and M. C. Stone. A linguistic approach to categorical color assignment for data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22:698–707, 2016. doi: 10.1109/TVCG.2015.2467471 2, 5, 6

[65] Sherwin Williams. Downloadable color palettes. https://www.sherwin-williams.com/property-facility-managers/color/tools/downloadable-color-palettes, 2024. Accessed: 2024-03-01. 3

[66] C. Shi, W. Cui, C. Liu, C. Zheng, H. Zhang, Q. Luo, and X. Ma. Nl2color: Refining color palettes for charts with natural language. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):814–824, 2024. doi: 10.1109/TVCG.2023.3326522 2

[67] Y. Shi, S. Chen, P. Liu, J. Long, and N. Cao. Colorcook: Augmenting color design for dashboarding with domain-associated palettes. *Proc. ACM Hum.-Comput. Interact.*, 6(CSCW2), article no. 433, 25 pages, nov 2022. doi: 10.1145/3555534 2

[68] W. T. Siok, P. Kay, W. S. Y. Wang, A. H. D. Chan, L. Chen, K.-K. Luke, and L. H. Tan. Language regions of brain are operative in color perception. *Proceedings of the National Academy of Sciences*, 106(20):8140–8145, 2009. doi: 10.1073/pnas.0903627106 9

[69] N. D. Solutions. Colors by name hex. https://www.nwdsak.com/Web-Resources/Web-Design/Colors-By-Name-Hex. Accessed: 2024-03-01. 3

[70] M. Stone, D. A. Szafir, and V. Setlur. An engineering model for color difference as a function of size. *Color and Imaging Conference*, 22(1):253–253, 2014. doi: 10.2352/CIC.2014.22.1.art00045 2

[71] J. W. Tanaka and L. M. Presnell. Color diagnosticity in object recognition. *Perception & Psychophysics*, 61(6):1140–1153, 1999. doi: 10.3758/BF03207619 2, 6

[72] Y. Tian, W. Cui, D. Deng, X. Yi, Y. Yang, H. Zhang, and Y. Wu. Chartgpt: Leveraging llms to generate charts from abstract natural language. *IEEE Transactions on Visualization and Computer Graphics*, 31(3):1731–1745, 15 pages, Mar. 2025. doi: 10.1109/TVCG.2024.3368621 3

[73] Usability.gov. System usability scale (sus). https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html. (Accessed on 03-01-2024). 7

[74] W3C. Web Content Accessibility Guidelines (WCAG) 2.1. https://www.w3.org/TR/WCAG21/, 2018. Accessed: 2025-06-29. 9

[75] M. Zhu, Y. Weng, L. Yang, and Y. Zhang. Personality alignment of large language models, 2025. 3