Behavior As A Modality

By

Yaman K Singla

With the supervision of

Prof Rajiv Ratn Shah, IIIT Delhi
Prof Changyou Chen, State University of New York at Buffalo

Submitted To:
Indraprastha Institute of Information Technology Delhi
State University of New York at Buffalo

January, 2024

We're actually much better at planning the flight path of an interplanetary rocket (rocket science) than we are at managing the economy, merging two corporations, or even predicting how many copies of a book will sell (behavior prediction). So why is it that rocket science ***seems*** *hard, whereas problems having to do with people - which arguably are much harder -* ***seem*** *like they ought to be just a matter of common sense?* - Duncan J. Watts

Also,

If the brain were so simple we could understand it, we would be so simple we couldn't. - Emerson Pugh

but

Nothing in Nature is random. A thing appears random only through the incompleteness of our knowledge (ignorance). - Baruch Spinoza

and

Timendi causa est nescire. (Ignorance is the cause of fear.)

and

What would life be if we had (only fear and) no courage to attempt anything? - Vincent Van Gogh

Acknowledgments

कार्पण्यदोषोपहतस्वभाव :
पृच्छामि त्वां धर्मसम्मूढचेता : |
यच्छ्रेय : स्यान्निश्चितं ब्रूहि तन्मे
शिष्यस्तेऽहं शाधि मां त्वां प्रपन्नम् | |BG 2 :7| |

मयि सर्वाणि कर्माणि संन्यस्याध्यात्मचेतसा |
निराशीर्निर्ममो भूत्वा युध्यस्व विगतज्वर : | |BG 3 :30| |

नैव किञ्चित्करोमीति युक्तो मन्येत तत्त्ववित् |
पश्यञ्शृण्वन्स्पृशञ्जिघ्रन्नश्नन्गच्छन्स्वपञ्श्वसन् | |
प्रलपन्विसृजन्गृह्ण्न्नुन्मिषन्निमिषन्नपि |
इन्द्रियाणीन्द्रियार्थेषु वर्तन्त इति धारयन् | |BG 5 :8-9| |

This is an attempt to capture and thank those who have shaped my journey. Albeit, due to indirect and latent relations, this list will remain non-exhaustive despite my best attempts, it is still an attempt worth making.

In no particular order (with names of the organizations where I met these giants): Rajiv Ratn Shah (IIIT-D), Changyou Chen (SUNY-Buffalo), Ranjeeta Rani (GMPS), Roger Zimmerman (National University of Singapore), Debanjan Mahata (Bloomberg), Jessy Junyi Li (University of Texas at Austin), Balaji Krishnamurthy (Adobe MDSR), Jayakumar Subramanian (Adobe MDSR), Amanda Stent (Bloomberg), Anil Seth (FIITJEE), Dhruva Sahrawat (IIIT-D), Yifang Yin (National University of Singapore), Mika Hama (Second Language Testing Institute), Payman Vafaee (Columbia University), Pankaj Bansal (Adobe), Mohit Srivastava (Adobe), Gaurav Jain (Adobe), Shubham Yadav (NSIT), Rohit Jain (NSIT), Mohd Khwaja Salik (NSIT), Pratham Nawal (NSIT), Mayank Singh (NSIT), Somesh Singh (BITS-Pilani Goa), Aanisha Bhattacharyya (Adobe MDSR), Varun Khurana (IIIT-D), Rita Yadav (GMPS), Prabha Sinha (GMPS), Neeta Pandit (GMPS), Geetha Nair (GMPS), Swami Sarvapriyananda (Ramakrishna Mission), and finally my parents and my brother, Aman Singla.

Hopefully, I can return whatever I have gathered from these people back to the earth.

Contents

Chapter 1

Introduction: The Two Cultures of Behavioral Sciences

A *modality* is defined in terms of information, such that a modality is a medium through which information is conveyed [1–3]. Similarly, a multimodal distribution is defined as having more than one peak in the probability distribution describing the nature of information. Behavior as a modality occurs in the process of communication. Communication includes all of the procedures by which one mind may affect another [4]. This includes all forms of expression, such as words, gestures, speech, pictures, and musical sounds. Communication can be seen as being composed of seven modalities (Fig. 1.1): (the communicator, message, time of message, channel, receiver, time of effect, and effect). These modalities can vary independently of each other [5–7] but carry signals about each other [8]. The message as a modality carries data from the communicator to receiver and encodes information generated by the communicator. Similarly, behavior as a modality carries data from the receiver and encodes information generated by the receiver. This is often a continuous cycle, where behavior generated in the previous cycle becomes the message of the next cycle.

Different fields deal with different parts of behavior. I will give a broad overview of these fields in the upcoming paragraphs, but two streams have emerged broadly in behavioral sciences: explanation and prediction of behavior (receiver effect) [9–11].

Historically, behavioral social scientists have sought explanations of human behavior that can provide interpretable causal mechanisms behind human functioning. A few prominent examples are Milgram's [12] and Asch's [13] experiments on persuasion, explaining the causal mechanism of obedience to authority. The approach of theorizing has worked in physical sciences where the data is plentiful, and theories make unambiguous pre-

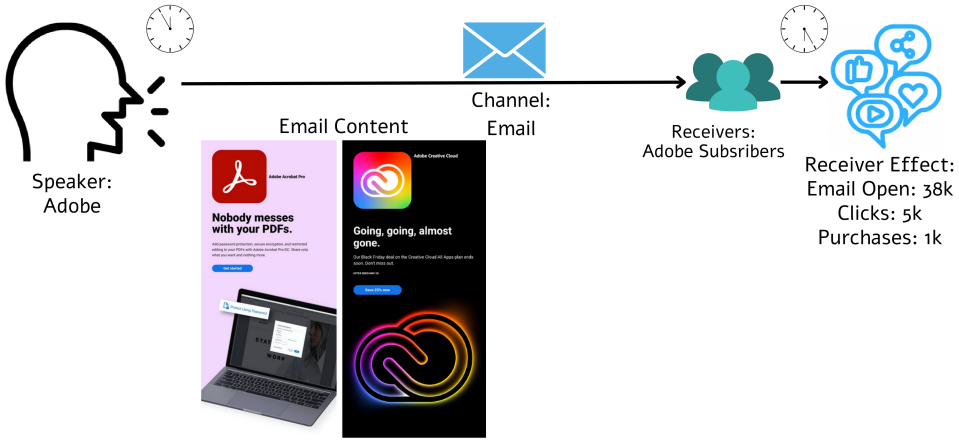


Figure 1.1: Communication process can be defined by seven factors: Communicator, Message, Time of message, Channel, Receiver, Time of effect, and Effect. Any message is created to serve an end goal. For marketers, the end goal is to bring in the desired receiver effect (behavior) (like clicks, purchases, likes, and customer retention). The figure presents the key elements in the communication pipeline - the marketer, message, channel, receivers, and finally, the receiver effect.

dictions but have not been too successful in *predicting* social outcomes in behavioral sciences [14–16]. In fact, many studies have shown that expert human opinions fare similar to non-experts (*e.g.*, predicting economic and political trends [15] and societal change: [16]), and the opinion of non-expert population is roughly the same as a random coin toss in predicting behavior (*e.g.*, predicting cascades [17] or image memorability [18]). At the same time, causal mechanisms have their own merits; most notably, they help decision-makers (often humans) to make intuitive sense of the situation and make their next decision based on it.

In parallel, due to the availability of human behavior data at scale, researchers in machine learning are showing a growing interest in traditionally behavioral science topics, such as messaging strategies leading to persuasion [19–22], information diffusion [23, 24], and most importantly, prediction and predictability of human behavior [25, 26]. Machine learning approaches bring with them the culture of (training and) testing their models on large real-world datasets and pushing the state-of-the-art in terms of predictive accuracies; at the same time, often, ML approaches can only be operated as black boxes with no direct mechanism to explain predictions [27, 28].

In the prediction community, different subfields have emerged dealing with the different parts of the problem of optimization of human behavior. For instance, advertisement personalization studies how to optimize (choose) *receiver* for a given message [29], and recommendation systems study how to

choose content from a set of pre-decided contents for a given receiver to elicit a certain effect [30]. A popular problem within the prediction community is the effect prediction problems, for example, clickthrough (CTR) prediction [31], Twitter cascade prediction [23, 24], sales prediction [25, 32], content memorability prediction [7, 33, 34], *etc.* There are also works to optimize the time of the message to elicit certain effect [7, 35]. Therefore, we see that all the factors of communication are studied independently in their own light with the aim of achieving the desired effect.

Effect (or behavior) over a content can also enable us to understand about the content, the communicator, the receiver, or the time. Therefore, efforts have also been made to extract information about the content itself from the behavior it generates. For instance, using keystroke movements [36] and eye movements to improve natural language processing [8, 37]. Similarly, the fields of human alignment and reinforcement learning with human feedback (RLHF) try to use human behavioral signals of likes, upvotes, downloads, and annotations of a response's helpfulness to improve content generation - both text [7, 38–41] and images [6, 42–44].

In the more traditional social science and computational social science cultures, research is carried out to discover causal effects and model them. For instance, propaganda and mass communication studies [45–48] try to understand the culture, time, authors, recipients in a non-invasive manner using the messages exchanged, and persuasion studies [49, 50] where the persuasion strategy present in the content is identified and correlated with (un)successful efforts of persuasion.

A common theme that runs through both research cultures in behavioral sciences is the intent to control behavior. Explanation and prediction are intermediate steps to control and hence optimize behavior. Optimizing behavior means to fulfill the communicator's objectives by controlling the other six parts of the communication process (Fig. 1.1). Due to the problem space being large, the solution needs a general understanding of human behavior as opposed to being domain-specific.

The characteristic that marks the digital age is the prevalence of human behavioral data in huge repositories. This data is *big* (allowing to model heterogeneity), *always-on* (allowing to look in the past as well as live measurements), observational (as opposed to reactive), but also *incomplete* (does not capture all that is happening everywhere everytime in a single repository) and *algorithmically confounded* (generated as a byproduct of an engineering process with a goal) [27]. While the predictive culture has tried to make use of some of this data in the form of social media datasets like Twitter [51, 52] and Instagram [53], Google trends [25, 54], Wikipedia [55–57], shopping websites [58, 59] and other data sources [26, 60, 61], these efforts are limited, in the sense of being dependent on one or a few chosen platforms, able to answer a limited set of questions, and restricted by access to private data. We want a model that can understand (predict and ex-

plain) *human behavior in general* as opposed to modeling a particular effect (retweet prediction) on a particular platform (*e.g.* Twitter) for a certain type of users. This problem carries parallels with the problem being solved in the natural language processing (NLP) community, where supervised models in NLP are limited by the amount of supervision available and being able to answer one question (for which the supervised model was trained). The problem was solved by developing Large Language Models (LLMs), which are general purpose models capable of *understanding language*, and hence can solve natural language tasks like sentiment analysis, question answering, email generation, and language translation in zero-shot (*i.e.* without needing any explicit training for that task) [62–66].

Figure 1.2: Levels of content analysis. The figure lists tasks and their sample outputs arranged in a hierarchy [4]. This is roughly based on levels of language. Notably, humans are good at predicting the first three levels but not the last level [15–18].

Similarly, how do we develop a model capable of understanding behavior *in general*? With the intent to answer this question, we take motivation from LLMs, where the idea is to train a model on a data-rich task. The task chosen to train LLMs is the next-word prediction, and the dataset is the text collected from the entire internet. The next-word prediction task is a data-rich task that can be trained on the huge text repositories from the internet. The intuition is that two approaches have always worked for neural networks: larger model sizes and more data for training [62, 64, 65, 67]. Going from a few million tokens of text [64, 67] to a trillion tokens [63, 68] leads to an increase in the transfer learning capability leading to performance improvements over a wide variety of natural language tasks.

The digital revolution has provided us with huge repositories of data. We leverage the human behavior repositories available on the internet for this general-purpose human behavior model. The format of this data is the general communication model shown in Fig. 1.1 consisting of communicator, message, time of message, channel, receiver, time of effect, and effect. Due to the incomplete nature of behavioral repositories, all the factors are usually not always available. However, a subset is always available, and we show that the data scale, along with a large model, helps make a general behavior understanding model [5]. We call this model, Large Content and Behavior Model (LCBM). We show that LCBM can predict behavior, explain it, and generate a message to bring about certain behavior [5–7].

Are general LLMs unable to solve behavioral problems? A question that arises is whether LLMs, which already learn trillions of text tokens, are able to understand and predict behavior. We investigate that question over several large models, including GPT-3.5 [63], GPT-4 [69], Llama-13 B and LLama-7B [68], and find that they do not seem to have any behavioral capabilities. The reason for this is that large language models only include one factor (message) out of the 7-factor communication model (Fig. 1.1) while considering other parts as "noise" (for instance, see [70, 71]). This systematic purge of communicator, receiver, channel, time, and, most importantly, behavior causes the models not to develop any behavioral capabilities (Level-C of Shannon and Weaver [4]). As an example, Llava [72], a recent large language and vision model (LVM) trained by connecting a vision encoder with a language model, shows that after training on a few hundred thousand instructions, the language model can now "see", and is able to answer questions on the images. However, the questions all lie in the first two levels of content analysis shown in Fig. 1.2. The reason is that the instructions used to align the image encoder with the downstream LLM all lie in the first two levels while ignoring the last two. In the upcoming chapters, we explore how we can train a general behavior model and how including back the other factors of communication in training data help in understanding human behavior.

Outline for the upcoming chapters: Following the two traditions of behavioral sciences, in this work, in Chapter-2, I start with a more traditional approach to behavior explanation, where I cover the first works on extracting persuasion strategies in advertisements (both images and videos) [20, 22]. The contributions of these works include constructing the largest set of generic persuasion strategies based on theoretical and empirical studies in marketing, social psychology, and machine learning literature and releasing the first datasets to enable the study and model development for the same. These works have been deployed to understand the correlation between the kinds of marketing campaigns and customer behavior measured by clicks, views, and other marketing key performance indicators (KPIs).

Following this, in Chapter-3, I delve into the more modern approach of behavior prediction and leveraging the huge repositories of behavior data available. First, we propose models to integrate behavior with relatively smaller language models like BERT [62], and show that the resultant models can understand content better than the base models [8]. Then, we propose an approach to integrate behavior and content together as part of a single model. We call these models Large Content and Behavior Models (LCBM) [5]. We show that these models can predict and explain behavior. Next, in Chapter-4, we show that using these models we can generate messages to elicit certain behavior resulting in behavior optimization. We show this both for the domain of text, by taking the illustrative case of memorability and generating content that is more memorable in long-term [7], and images, by

generating images that are more performant, *i.e.*, can result in more likes and downloads [6].

Chapter 2

Explaining Behavior: Persuasion Strategies

Modeling what makes an advertisement persuasive, *i.e.*, eliciting the desired response from consumer, is critical to the study of propaganda, social psychology, and marketing. Despite its importance, computational modeling of persuasion in computer vision is still in its infancy, primarily due to the lack of benchmark datasets that can provide persuasion-strategy labels associated with ads. Motivated by persuasion literature in social psychology and marketing, we introduce an extensive vocabulary of persuasion strategies and build the first ad corpus (both image and video) annotated with persuasion strategies. We then formulate the task of persuasion strategy prediction with multi-modal learning. The image dataset also provides image segmentation masks, which labels persuasion strategies in the corresponding ad images on the test split. We publicly release our code and dataset at `https://midas-research.github.io/persuasion-advertisements/`. This chapter is based on two papers I published along with collaborators [20, 22].

Marketing communications is the mode by which companies and governments inform, remind, and persuade their consumers about the products they sell. They are the primary means of connecting brands with consumers through which the consumer can know what the product is about, what it stands for, who makes it, and can be motivated to try it out. To introduce meaning into their communication, marketers use various rhetorical devices in the form of persuasion strategies such as **emotions** (*e.g.*, Oreo's "Celebrate the Kid Inside", humor by showing Ronald McDonald sneaking into the competitor Burger King's store to buy a burger), **reasoning** (*e.g.*, "One glass of Florida orange juice contains 75% of your daily vitamin C needs"), **social identity** (*e.g.*, Old Spice's "Smell like a Man"), and **impact** (*e.g.*, Airbnb showing a mother with her child with the headline "My home is funding her future") (Refer to Fig. 2.3 to see these ads). Similarly, even for marketing the same product, marketers use different persuasion

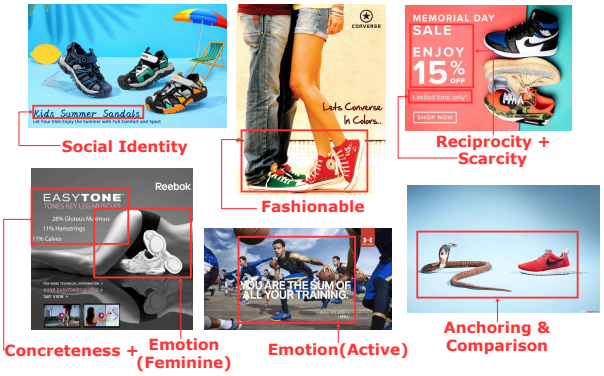


Figure 2.1: Different persuasion strategies are used for marketing the same product (footwear in this example). The strategies are in red words and to be defined by us in the paper.

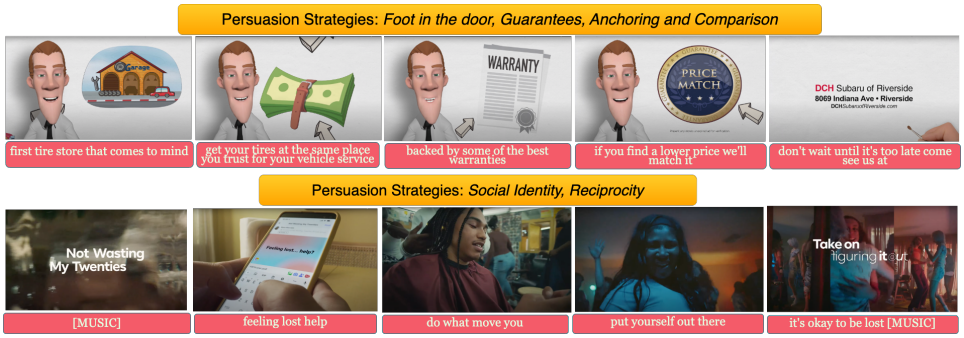
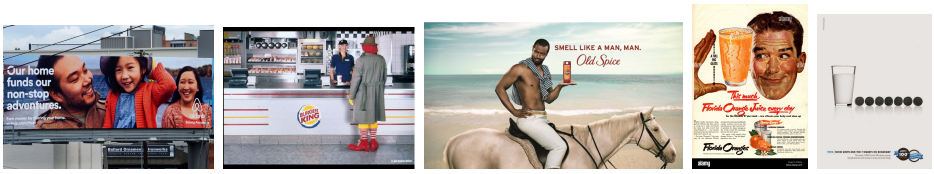


Figure 2.2: Examples of videos with their annotated persuasion strategies. Relevant keyframes and ASR captions are shown in the figure, along with the annotated strategies. These two videos can be watched at `https://bit.ly/3Ie3JG0`, `https://bit.ly/3OgtLwj` .

strategies to target different demographies (see Fig. 2.1). Therefore, recognizing and understanding persuasion strategies in ad campaigns is vitally important to decipher viral marketing campaigns, propaganda, and enable ad-recommendation.

Studying rhetorics of this form of communication is an essential part of understanding visual communication in marketing. Aristotle, in his seminal work on rhetoric, underlining the importance of persuasion, equated studying rhetorics with the study of persuasion* [73]. While persuasion is studied extensively in behavioral sciences, such as marketing [74, 75] and psychology [76, 77], computational modeling of persuasion in computer vision is still in its infancy, primarily due to the lack of benchmark datasets that can

*"Rhetoric may be defined as the faculty of discovering in any particular case all of the available means of *persuasion*" [73]



(a) (b) (c) (d) (e)

Figure 2.3: Various rhetoric strategies used in advertisements

provide representative corpus to facilitate this line of research. In the limited work that has happened on persuasion in computer vision, researchers have tried to address the question of which image is more persuasive [78] or extracted low-level features (such as emotion, gestures, and facial displays), which indirectly help in identifying persuasion strategies without explicitly extracting the strategies themselves [79]. On the other hand, decoding persuasion in textual content has been extensively studied in natural language processing from both extractive, and generative contexts [19, 21, 80]. This forms the motivation of our work, where we aim to identify the persuasion strategies used in visual content such as advertisements.

The systematic study of persuasion began in the 1920s with the media-effects research by Lasswell [48], which was used as the basis for developing popular models of persuasion, like the Elaboration Likelihood Model (ELM) [77], Heuristic Systematic Model (HSM) [50], and Hovland's attitude change approach [76]. Laswell in this research broke down communication into five factors by defining communication as an act of *who* said it, *what* was said, in *what* channel it was said, to *whom* it was said, and with what *effect* it was said. Later, this model was used as the basis for developing popular models of persuasion, like Elaboration Likelihood Model [77], Heuristic Systematic Model [50], and Hovland's attitude change approach [76]. Amongst these, the most widely accepted model of persuasion theory is the Elaboration Likelihood Model (ELM).

These models of persuasion posit a dual process theory that explains attitude and behavior change (persuasion) in terms of the following major factors: stimuli (messages), personal motivation (the desire to process the message), capability of critical evaluation, and cognitive busyness. These factors could be divided into cognitive, behavioral, and affective processes of attitude change. Thus, a person may begin liking a new political candidate because she just donated $100 to the campaign (behavior-initiated change), because the theme music in a recently heard commercial induced a general pleasantness (affect-initiated change), or because the person was impressed with the candidate's issue positions (cognitive initiated change). Similarly, if a person already likes a political candidate he may agree to donate money to the campaign (behavioral influence), may feel happiness upon meeting the

candidate (affective influence), and may selectively encode the candidate's issue positions (cognitive influence) [77].

ELM posits that when facing a message from a persuader, the persuadee reacts by using the two information processing channels: central processing or peripheral processing. When the persuadee processes information centrally, the cognitive responses, or elaborations, will be much more relevant to the information, whereas when processing peripherally, the individual may rely on heuristics and other rules of thumb when elaborating on a message. The factors which influence how and how much one will elaborate the persuasive message is given by the message type, personal motivation, and other factors presented in the ELM. Being at the high end of the elaboration continuum, people assess object-relevant information in relation to schemas that they already possess, and arrive at a reasoned attitude that is supported by information [81].

In this chapter, we build on these psychological insights from persuasion models in sociology and marketing and study the message strategies that lead to persuasion. We codify, extend, and unify persuasion strategies studied in the psychology and marketing literature into a set of 20 strategies divided into 9 groups (see Fig. 2.4, Table 2.2): *Authority and Credibility*, *Social Identity and Proof*, where cognitive indirection in the form of group decisioning and expert authority is used for decisions, *Value and Impact Formulation* where logic is used to explain details and comparisons are made, *Reciprocity*, *Foot in the door*, *Overcoming Resistance* where social and cognitive consistency norms are harnessed to aid decision-making, *Scarcity*, *Anthropomorphism* and *Emotion* where information is evaluated from the lenses of feelings and emotions. In addition to introducing the most extensive vocabulary for persuasion strategies, we make a superset of persuasion strategies presented in the prior NLP works, which introduced text and domain-specific persuasion tactics, thus making large-scale understanding of persuasion across multiple contexts comparable and replicable.

Constructing a large-scale dataset containing persuasion strategies labels is time-consuming and expensive. We leverage active learning to mitigate the cost of labeling fine-grained persuasion strategies in advertisements. We first introduce an attention-fusion model trained in a multi-task fashion over modalities such as text, image, and symbolism. We use the action-reason task from the Pitts Ads dataset [82] to train the model and then annotate the raw ad images from the same dataset for persuasion strategies based on an entropy based active learning technique.

To sum up, our contributions include:

1. We construct the largest set of generic persuasion strategies based on theoretical and empirical studies in marketing, social psychology, and machine learning literature.
2. We introduce the first dataset for studying persuasion strategies in advertisements. This enables initial progress on the challenging task of auto-

Figure 2.4: Persuasion strategies in advertisements. Marketers use both text and vision modalities to create ads containing different messaging strategies. Different persuasion strategies are constituted by using various rhetorical devices such as slogans, symbolism, colors, emotions, allusion.

matically understanding the messaging strategies conveyed through visual advertisements. We also construct a prototypical dataset containing image segmentation masks annotating persuasion strategies in different segments of an image.

3. We formulate the task of predicting persuasion strategies with a multi-task attention fusion model.

4. We conduct extensive experiments on the released corpus, showing the effect of different modalities on identifying persuasion strategies, correlation between strategies and topics and objects with different strategies.

2.1 Related Work

How do messages change people's beliefs and actions? The systematic study of persuasion has captured researchers' interest since the advent of mass influence mechanisms such as radio, television, and advertising. Work in persuasion spans across multiple fields, including psychology, marketing, and machine learning.

Persuasion in Marketing and Social Psychology: Sociology and communication science has studied persuasion for centuries now starting from the seminal work of Aristotle on rhetoric. Researchers have tried to construct and validate models of persuasion. Due to space constraints, while we cannot cover a complete list of literature, in Section 2.2, we list the primary studies which originally identified the presence and effect of various persuasion tactics on persuadees. We build on almost a century of this research and crystallize them into the persuasion strategies we use for anno-

tation and modeling. Any instance of (successful) persuasion is composed of two events: (a) an attempt by the persuader, which we term as the persuasion strategies, and (b) subsequent uptake and response by the persuadee [83, 84]. In this work, we study (a) only while leaving (b) for future work. Throughout the rest of the paper, when we say persuasion strategy, we mean the former without considering whether the persuasion was successful or not.

Persuasion in Machine Learning: Despite extensive work in social psychology and marketing on persuasion, most of the work is qualitative, where researchers have looked at a small set of messages with various persuasion strategies to determine their effect on participants. Computational modeling of persuasion is still largely lacking. In the limited work in computational modeling of persuasion, almost all of it is concentrated in the NLP literature, with only very few works in computer vision. Research on persuasion in NLP under the umbrella of argumentation mining is broadly carried out from three perspectives: extracting persuasion tactics, studying the effect of constituent factors on persuasion, and measurement of persuasiveness nature of content. A few examples of research studies that annotate persuasive strategies in various forms of persuader-persuadee interactions like discussion forums, social media, blogs, academic essays, and debates are [83, 85, 86]. We use these and other studies listed in Section 2.2 to construct our vocabulary of persuasion strategies in advertisements.

Other studies focus on factors such as argument ordering [87, 88], target audience [89], and prior beliefs [90] for their effect in bringing about persuasion. Studies such as [91, 92] also try to measure persuasiveness and generate persuasive content. The generation of persuasive (textual) messages has been studied [93] and, in particular, a novel ML method for learning user model tailored persuasion strategy has also been proposed [94, 95].

As one of the first works in the limited work in the computer vision domain, Joo *et al.* [79] introduced syntactical and intent features such as facial displays, gestures, emotion, and personality, which result in persuasive images. Their analysis was done on human images, particularly politicians, during their campaigns. Their work on political campaigners is more restrictive than general product and public-service advertisements. Moreover, they deal with low-level features such as gestures and personality traits depicted through the face, which are important for detecting persuasion strategies but are not persuasion strategies themselves. Recently, Bai *et al.* [78] studied persuasion in debate videos where they proposed two tasks: debate outcome prediction and intensity of persuasion prediction. Through these tasks, they predict the persuasiveness of a debate speech, which is orthogonal to the task of predicting the strategy used by the debater. Other similar works which discuss persuasiveness of images and videos are [96, 97].

2.2 Generic Taxonomy of Persuasion Strategies

This section introduces the generic taxonomy of persuasive strategies, their definitions, examples, and connections with prior work. Representative literature from a) SPM: Social Psychology and Marketing, b) ML: Machine Learning

1. Authority and Credibility: SPM:[12, 77, 98–102] ML:[80, 83, 103–105]
   (a) **Guarantees**: Guarantees reduce risk and people try out such products more often.
   (b) **Authority**: Authority indicated through expertise, source of power, third-party approval, credentials, and awards
   (c) **Trustworthiness**: Trustworthiness indicated honesty and integrity of the source through tropes like years of experience, "trusted brand", numbers and statistics
2. Social Identity and Proof: SPM:[106–110] ML: [83, 91, 103, 105, 111–115]
   (a) **Social Identity**: *Normative* influence, which involves conformity with the positive expectations of "another", who could be "another person, a group, or one's self" (includes self-persuasion, fleeting attraction, alter-casting, and exclusivity)
   (b) **Social Proof**: *Informational influence* by accepting information obtained from others as evidence about reality, *e.g.*, customer reviews and ratings
3. Reciprocity: SPM:[99, 116–119] ML:[80, 83, 87, 91, 103]
   (a) **Reciprocity**: By *obligating* the recipient of an act to repayment in the future, the rule for reciprocation begets a sense of future obligation, often unequal in nature
4. Foot in the door: SPM: [99, 120, 121] ML:[86, 122, 123]
   (a) **Foot in the door**: Starting with small requests followed by larger requests to facilitate compliance while maintaining *cognitive coherence*.
5. Overcoming Resistance: SPM:[124–126] ML:{None}
   (a) **Overcoming Resistance**: Overcoming resistance (reactance) by postponing consequences to the future, by focusing resistance on realistic concerns, by forewarning that a message will be coming, by acknowledging resistance, by raising self-esteem and a sense of efficacy.
6. Value and Impact Formulation: SPM:[127–132] ML:[133, 134]
   (a) **Concreteness**: Using concrete facts, evidence, and statistics to appeal to the logic of consumers
   (b) **Anchoring and Comparison**: A product's value is strongly influenced by what it is compared to.
   (c) **Social Impact**: Emphasizes the importance or bigger (societal) impact of a product

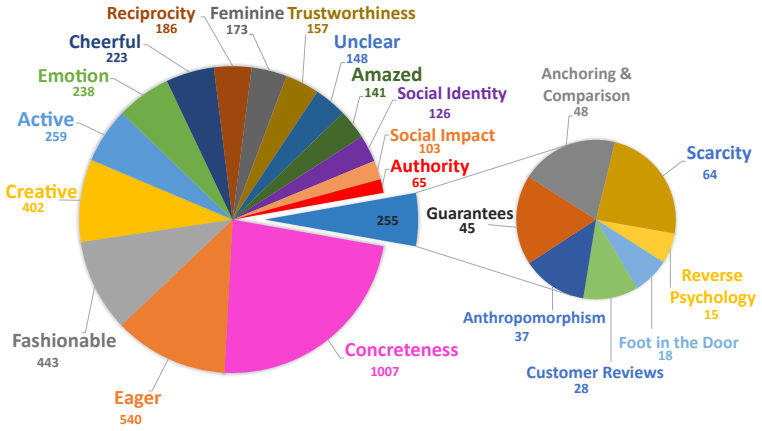


Figure 2.5: Distribution of Persuasion Strategies in the image persuasion strategy dataset. The top-3 strategies are Concreteness, Eager, and Fashionable.

7. Scarcity: SPM: [135–138] ML:[80, 87, 112]
   (a) **Scarcity**: People assign more value to opportunities when they are less available. This happens due to psychological reactance of losing freedom of choice when things are less available or they use availability as a cognitive shortcut for gauging quality.
8. Anthropomorphism: SPM:[110, 139, 140] ML:{None}
   (a) **Anthropomorphism**: When a brand or product is seen as human-like, people will like it more and feel closer to it.
9. Emotion: Aesthetics, feeling and other non-cognitively demanding features used for persuading consumers SPM:[77, 141, 142] ML:[85, 104, 105, 112, 115, 133, 143]
   (a) **Amazed**
   (b) **Fashionable**
   (c) **Active, Eager**
   (d) **Feminine**
   (e) **Creative**
   (f) **Cheerful**
   (g) **Further Minor**
10. **Unclear**: If the ad strategy is unclear

2.3 Persuasion Strategy Corpus Creation

2.3.1 Persuasion Strategy Dataset For Image Advertisements

To annotate persuasion strategies on image advertisements, we leverage raw images from the Pitts Ads dataset. It contains 64,832 image ads with labels of topics, sentiments, symbolic references (*e.g.* dove symbolizing peace), and

reasoning the ad provides to its viewers (see Fig 2.10 for a few examples). The dataset had ads spanning multiple industries, products, services, and also contained public service announcements. Through this, they presented an initial work for the task of understanding visual rhetoric in ads. Since the dataset already had a few types of labels associated with the ad images, we used active learning on a model trained in a multi-task learning fashion over the reasoning task introduced in their paper. We explain the model and then the annotation strategy followed in §2.4.

To commence training, we initially annotated a batch of 250 randomly selected ad images with persuasion strategies defined in Section 2.2. We recruited four research assistants to label persuasion strategies for each advertisement. Definitions and examples of different persuasion strategies were provided, together with a training session where we asked annotators to annotate a number of example images and walked them through any disagreed annotations. To assess the reliability of the annotated labels, we then asked them to annotate the same 500 images and computed Cohen's Kappa statistic to measure inter-rater reliability. We obtained an average score of 0.55. The theoretical maximum of Kappa, given the unequal distribution, is 0.76. In such cases, Cohen [144] suggested that one should divide kappa by its maximum value $\mathbf{k}/\mathbf{k_{max}}$, which comes out to be 0.72. This is a *substantial* agreement. Further, to maintain labeling consistency, each image was double annotated, with all discrepancies resolved by an intervention of the third annotator using a majority vote.

The assistants were asked to label each image with no more than 3 strategies. If an image had more than 3 strategies, they were asked to list the top 3 strategies according to the area covered by the pixels depicting that strategy. In total, we label 3000 ad-images with their persuasion strategies; and the number of samples in train, val, and test split are 2500, 250, and 250, respectively[†]. Fig. 2.5 presents the distribution of persuasion strategies in the dataset. It is observed that concreteness is the most used strategy in the dataset, followed by eagerness and fashion. The average number of strategies in an ad is 1.49, and the standard deviation is 0.592. We find that scarcity (92.2%), guarantees (91.1%), reciprocity (84.4%), social identity (83.3%), and cheerful (83%), are the top 5 strategies, which occur in groups of 2 or 3. We observe that the co-occurrence of these strategies is due to the fact that many of them cover only a single modality (*i.e.*, text or visual), leaving the other modality free for a different strategy. For example, concreteness is often indicated by illustrating points in text, while the visual modality is free for depicting, say, emotion. See Fig. 2.6 for an example, where the image depicting *Authority* also has concreteness strategy in it. Similarly, feminine emotion is also depicted in Fig. 2.1, along with concreteness.

[†]Table 2.4 shows the detailed distribution of the number of strategies in ads

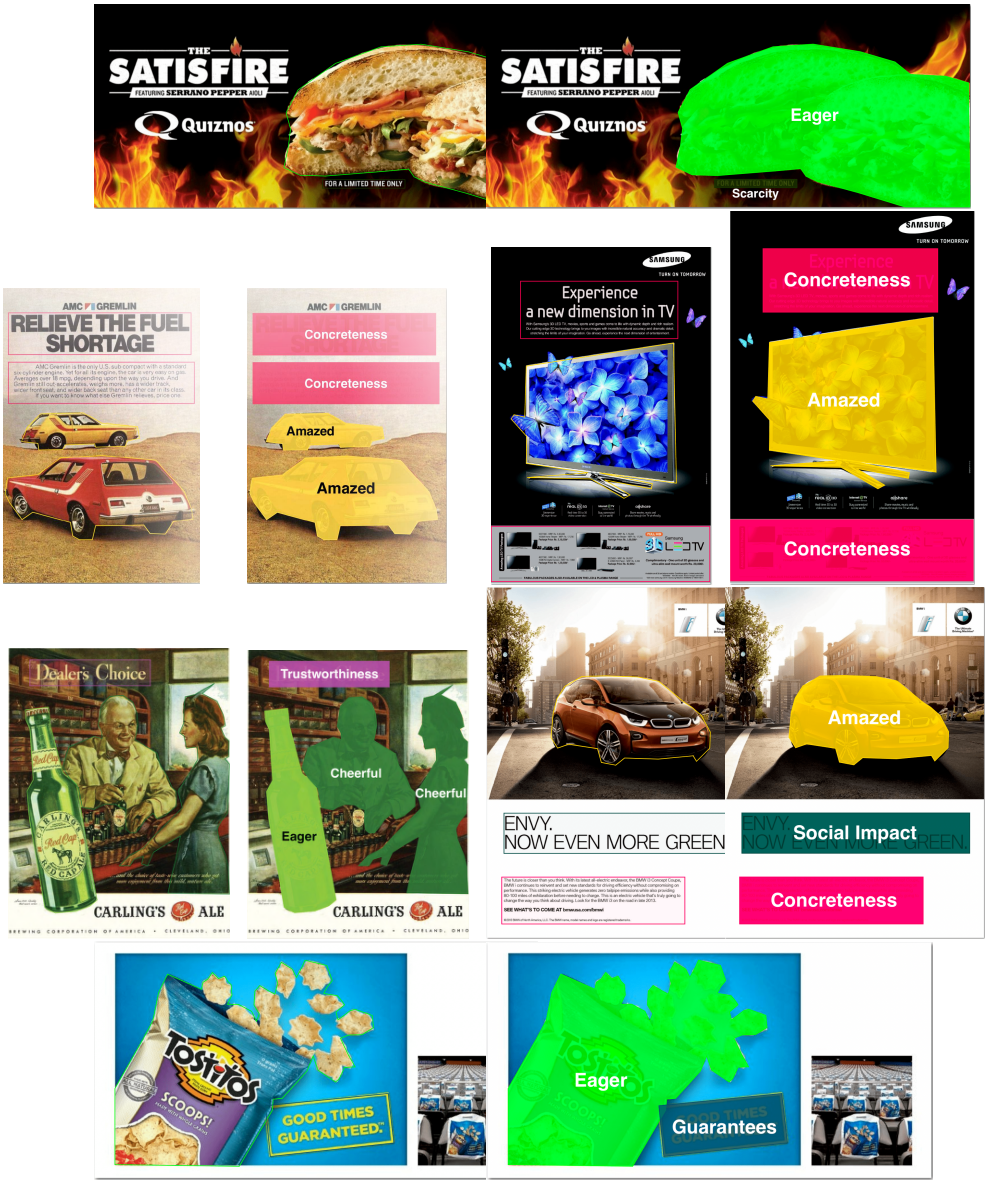


Figure 2.6: Image with a segmentation mask depicting the strategies *Emotion:Cheerful*, *Emotion:Eager* and *Trustworthiness*.

Next, we calculate the Dice correlation coefficient[‡] statistics for pairs of co-occurring persuasion strategies. The top-5 pairs are eager-concreteness (0.27), scarcity-reciprocity (0.25), eager-cheerful (0.19), amazed-concreteness (0.17), and eager-reciprocity (0.17). We find that these correlation values are

[‡]The Dice Coefficient is defined as: $2 * |X \cap Y| \ / \ (|X| + |Y|)$, where X and Y are two sets; a set with vertical bars on either side refers to the cardinality of the set, i.e. the number of elements in that set; and $\cap$ refers to the intersection of two sets.

(a) (b) (c)

Figure 2.7: Advertisements containing humans and concreteness

not particularly high since marketers seldom use *common pairings* of messaging strategies to market their products. The visual part mostly shows eager strategy in ads; therefore, we find that the text modality becomes free to show other strategies. That is why primarily text-based concreteness, cheerfulness, and reciprocity strategies are present with the visual-based eager strategy in the text modality. Also, primarily vision-based amazement, eagerness, and scarcity (short-text) strategies co-occur with text-based reciprocity and concreteness (*e.g.*, Fig. 2.1).

Next, we calculate the correlation between image topics and objects present with persuasion strategies. We see that the emotion:feminine and emotion:fashionable strategies are most often associated with beauty products and cosmetics (corr=0.4256, 0.2891). This is understandable since most beauty products are aimed at women. We see that the fast-food and restaurant industries often use eagerness as their messaging strategy (corr = 0.5877,0.3470). We find that the presence of humans in ads is correlated with the concreteness strategy (see Fig 2.7 for a few examples) (corr=0.3831). On the other hand, vehicle ads use emotion:amazed and concreteness (corr=0.5211,0.2412) (see Fig:2.8 for detailed correlations).

Similar to a low correlation in co-occurring strategies, we find that product segments and their strategies are not highly correlated. This is because marketers use different strategies to market their products even within a product segment. Fig. 2.1 shows an example in which the footwear industry (which is a subsegment of the apparel industry) uses different strategies to market its products. Further, for a batch of 250 images, we also label segmented image regions corresponding to the strategies present in the image. These segment masks were also double-annotated. Fig. 2.6 presents an example of masks depicting parts of the image masked with different persuasion strategies in a drink advertisement.

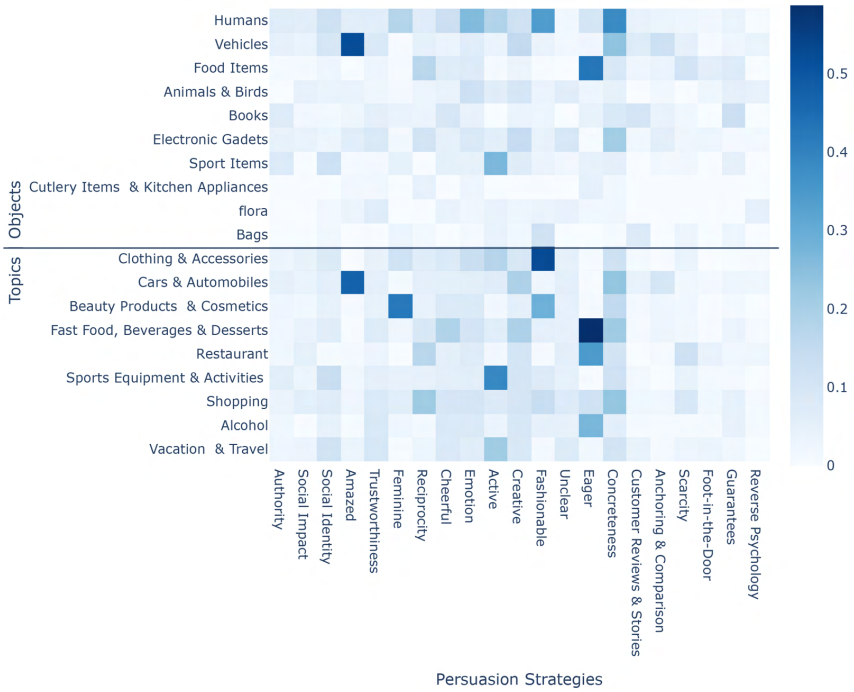


Figure 2.8: Dice correlation between topics and strategies. Topics are taken from the Pitts Ad dataset and further similar topics are combined to get these values.

2.3.2 Persuasion Strategy Dataset For Video Advertisements

For this task, we collected 2203 video advertisements from popular brands publicly available on the web. We use the following 12 strategies as our target persuasion strategy set: *Social Identity, Concreteness, Anchoring and Comparison, Overcoming Reactance, Reciprocity, Foot-in-the-Door, Authority, Social Impact, Anthropomorphism, Scarcity, Social Proof,* and *Unclear*. We use non-experts human annotators to label this dataset (as compared to expert humans for the image ads dataset). In order to make the class labels easier to understand for non-expert human annotators, we make a list of 15 yes/no type-questions containing questions like "*Was there any expert (person or company) (not celebrity) encouraging to use the product/brand? Was the company showcasing any awards (e.g., industrial or government)? Did the video show any customer reviews or testimonials?*" (complete list in Table 2.1).

Each human annotator watches 15 videos such that each video gets viewed by at least two annotators and answers these questions for each video. Based on all the responses for a video, we assign labels to that video. We remove videos with an inter-annotator score of less than 60%. After removing those, we get a dataset with 1002 videos, with an average length

of 33 secs and a distribution as shown in Fig. 2.9. This dataset is then used for the persuasion strategy identification task.

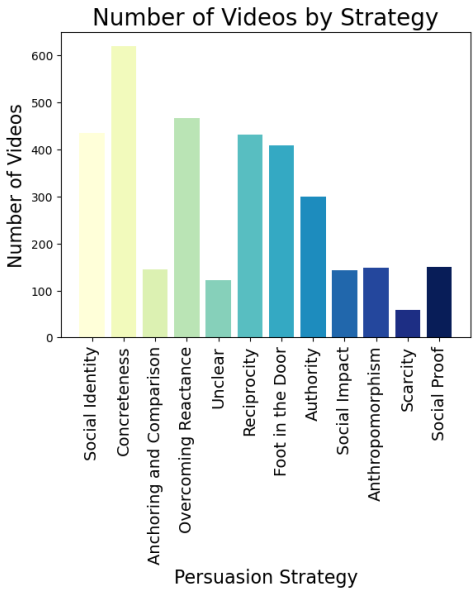


Figure 2.9: Distribution of persuasion strategies in our video persuasion strategy dataset

2.4 Modeling: Persuasion Strategy Prediction

2.4.1 Modelling Persuasion Strategy For Image Advertisements

The proposed Ads dataset $\mathcal{D}$ annotated with the persuasion strategies comprises of samples where each sample advertisement a_i is annotated with a set of annotation strategies S_i such that $1 \leq |S_i| \leq 3$. The unique set of the proposed persuasion strategies $\mathcal{P}$ is defined in Table 2.2. Given a_i, the task of the modeling is to predict the persuasion strategies present in the input ad. As we observe from Fig. 2.4, advertisements use various rhetoric devices to form their messaging strategy. The strategies thus are in the form of multi-modalities, including images, text and symbolism. To jointly model the modalities, we design an attention fusion multi-modal framework, which fuses multimodal features extracted from the ad, *e.g.*, the ad image, text present in the ad extracted through the OCR (Optical Character Recognition), regions of interest (ROIs) extracted using an object detector, and embeddings of captions obtained through an image captioning model (see Fig. 2.11). The information obtained through these modalities are firstly embedded independently through their modality specific encoders followed

Question	Strategy	Question	Strategy
Was there any expert (person or company) (not celebrity) encouraging to use the product/brand?	Authority	Did the video show any normal customers (non-expert, non-celebrity) using the product?	Social Identity
Did the video showcase any awards or long usage history of the product/brand?	Authority	Did the video show any customer reviews or testimonials?	Social Proof
Was the product/brand comparing itself with other competitors or existing solutions?	Anchoring and Comparison	Were any number/statistics mentioned?	Concreteness
Did the video talk about any specific features or provide information about the product/brand?	Concreteness	Were there any mention of any offers on the brand/product?	Reciprocity
Were the offers limited or available for a short period of time?	Scarcity	Was the product/brand told to be free or available on a discount?	Foot in the Door, Reciprocity
Was the brand/product described as simple, easy to use, or can start using with minimal resistance?	Overcoming Reactance, Foot in the Door	Was the brand/product talking about bigger societal impact?	Social Impact
Did the brand provide any guarantees that might help reduce the risk of people trying out the product?	Overcoming Reactance	Did the video provide any resources, tips, guides, or tools related to the product?	Reciprocity
Is the brand or product portrayed as human-like?	Anthropomorphism		

Table 2.1: The questions we asked to the non-expert annotators to help them identify persuasion strategy contained in the video advertisement.

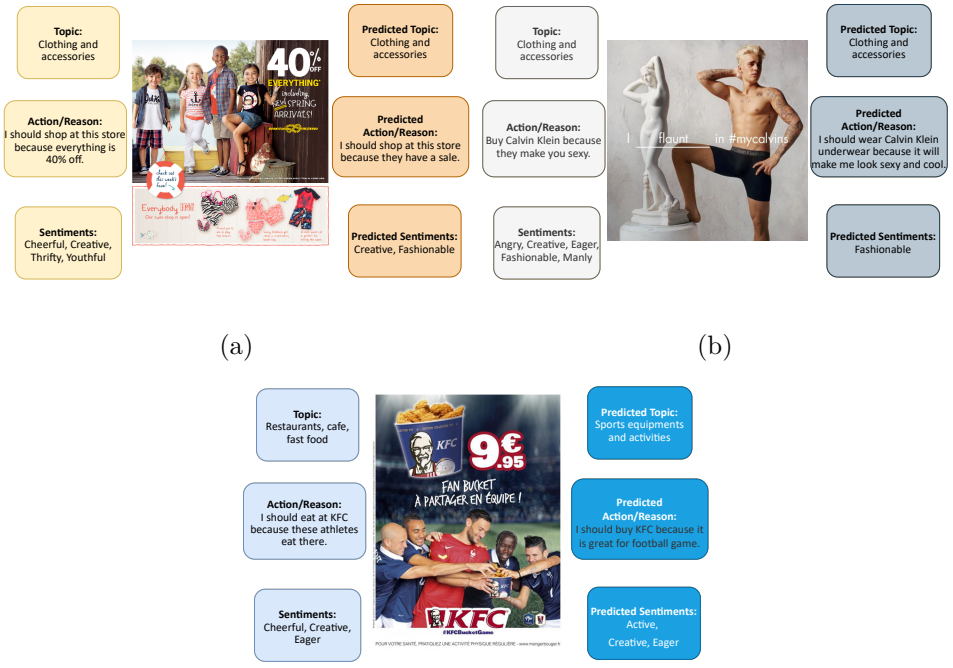


Figure 2.10: Some samples from the Pitts Ads dataset along with the ground truth and predicted action-reason statement, topic and sentiments.

by a transformer-based cross-attention module to fuse the extracted features from different modalities. The fused embeddings from the attention module are then used as input for a classifier that predicts a probability score for each strategy $p \in \mathcal{P}$. The overall architecture of the proposed model is illustrated in Fig.2.11. In the following, we describe each step in the prediction pipeline in detail.

2.4.1.1 Feature Extractors

In order to capture different rhetoric devices, we extract features from the image, text, and symbolism modalities.

Image Feature: We use the Vision Transformer [145] (ViT) model for extracting image features from the entire input image. The model resizes the input image to size 224×224 and divides it into patches of size 16×16. The model used has been pre-trained on the ImageNet 21k dataset. We only use the first output embedding, which is the CLS token embedding, a 768 dimension tensor, as we only need a representation of the entire image. Then, a fully connected layer is used to reduce the size of the embedding, resulting in a tensor of dimension 256.

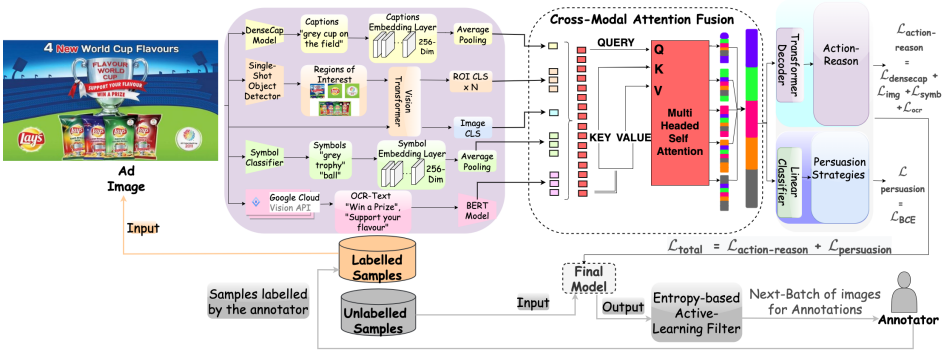


Figure 2.11: Architecture of the Persuasion Strategy Prediction model. To capture the different rhetoric devices, we extract features for the image, text, and symbolism modalities and then apply cross-modal attention fusion to leverage the interdependence of the different devices. Further, the model trains over two tasks: persuasion strategies and the reasoning task of action-reason prediction.

Regions of Interest (RoIs) from Detected Objects and Captions: Ad images contain elements that the creator deliberately chooses to create *intentional impact* and deliver some *message* in addition to the ones that occur *naturally* in the environment. Therefore, it is important to identify the composing elements of an advertisement to understand the creator's intention and the ad's message to the viewer. We detect and extract objects as regions of interest (RoIs) from the advertisement images. We get the RoIs by training the single-shot object detector model [146] on the COCO dataset [147]. We compare it with the recent YOLOv5 model [148]. We also extract caption embeddings to detect the most important activity from the image using a caption generation mode. We compare DenseCap [149] and the more recent BLIP [150] for caption generation.

OCR Text: The text present in an ad presents valuable information about the brand, such as product details, statistics, reasons to buy the product, and creative information in the form of slogans and jingles that the company wants its customers to remember and thus making it helpful in decoding various persuasion strategies. Therefore, we extract the text from the ads and use it as a feature in our model. We use the Google Cloud Vision API for this purpose. All the extracted text is concatenated, and the size is restricted to 100 words. We pass the text through a BERT model and concatenate the embeddings for those 100 words. Similar to image embeddings, an FC layer is used to convert embeddings to 256 dimensions. The final embedding of the OCR is a tensor of dimension 100×256.

Symbolism: While the names of the detected objects convey the names or literal meaning of the objects, creative images often also use objects for their symbolic and figurative meanings. For example, an upward-going arrow represents growth or the north direction or movement towards the up-

ward direction depending on the context; similarly, a person with both hands pointing upward could mean danger (*e.g.*, when a gun is pointed) or joy (*e.g.*, during dancing). In Fig. 2.4, in the creative Microsoft ad, a symbol of a balloon is created by grouping multiple mice together. Therefore, we generate symbol embeddings to capture the symbolism behind the most prominent visual objects present in an ad. We use the symbol classifier by Hussain *et al.* [82] on ad images to find the distribution of the symbolic elements present and then convert this to a 256 dimension tensor.

2.4.1.2 Cross-Modal Attention

To capture the inter-dependency of multiple modalities for richer embeddings, we apply a cross-modal attention (CMA) layer [151] to the features extracted in the previous steps. Cross-modal attention is a fusion mechanism where the attention masks from one modality (*e.g.* text) are used to highlight the extracted features in another modality (*e.g.* symbolism). It helps to link and extract common features in two or more modalities since common elements exist across multiple modalities, which complete and reinforce the message conveyed in the ad. For example, the pictures of the silver cup, stadium, and ball, words like "Australian", "Pakistani", and "World Cup" present in the chips ad shown in Fig. 2.11 link the idea of buying *Lays* with supporting one's country's team in the World Cup. Cross attention can also generate effective representations in the case of missing or noisy data or annotations in one or more modalities [151]. This is helpful in our case since marketing data often uses implicit associations and relations to convey meaning.

The input to the cross-modal attention layer is constructed by concatenating the image, RoI, OCR, caption, and symbol embeddings. This results in a 114×256 dimension input to our attention layer. The cross-modal attention consists of two layers of transformer encoders with a hidden dimension size of 256. The output of the attention layer gives us the final combined embedding of our input ad. Given image embeddings E_i, RoI embeddings E_r, OCR embeddings E_o, caption embeddings E_c and symbol embeddings E_s, the output of the cross-attention layer E_{att} is formulated as:

$$\text{Enc}(X) = \text{CMA}([E_i(X), E_r(X), E_o(X), E_c(X), E_s(X)]) \,,$$

where $[\ldots, \ldots]$ is the concatenation operation. For the advertisement in Fig. 2.11, we observed that the caption "grey cup on the field" attends to OCR text (containing words like "win") and ViT features of the RoI (of "cup" and "field").

2.4.1.3 Persuasion Strategy Predictor

This module is a persuasion strategy predictor, which processes the set of feature embedding Enc(X) obtained through cross-modality fusion. Specifically, Enc(X) is passed through a self-attention layer as:

$$o_1 = softmax(\text{Enc}(X) \otimes W_{self-attn})^\intercal \otimes \text{Enc}(X) \tag{2.1}$$

where Enc(X) is of the dimension 114×256, $W_{self-attn} \in \mathcal{R}^{256\times 1}$, $\otimes$ denote tensor multiplication and o_1 denotes the output of self attention layer, which is further processed through a linear layer to obtain $o_{|\mathcal{P}|}$ to represent the logits for each persuasion strategy. We apply sigmoid over each output logit such that the i^{th} index of the vector after applying sigmoid denotes p_i - the probability with which i^{th} persuasion strategy is present in the ad image. Our choice of using sigmoid over softmax is motivated by the fact that multiple persuasion strategies can be present simultaneously in an ad image. Consequently, the entire model is trained in an end-to-end manner using binary cross-entropy loss $\mathcal{L}_s$ over logit for each strategy:

$$\mathcal{L}_s = [-y_i \log(p_i) - (1 - y_i) \log(1 - p_i)] \tag{2.2}$$

where, y_i is 1 if i^{th} persuasion strategy is present in the ad and 0 otherwise. It can be observed in Table 2.2 that our model achieves an accuracy of 59.2%, where a correct match is considered if the strategy predicted by the model is present in the set of annotated strategies for a given ad. Further, we perform several ablations where we exclude each modality while retaining all the other modalities. We note that for each modality, excluding the modality results in a noticeable decrease in accuracy, with significant decreases observed when excluding DenseCap ($\sim 3.6\%$) and OCR ($\sim 4.4\%$). Further, we observe that using DenseCap for obtaining caption embeddings, and SSD for object detection works better than BLIP and YOLOv5, respectively (see Table 2.3). We also explore using focal loss [152] in place of cross-entropy loss to handle class imbalance but observed that it led to degradation instead of improvements (top-1 acc.[§] of 56.4% *vs* 59.2% using cross-entropy). We also train the model of Hussain *et al.* [82] for strategy prediction through a similar configuration as ours (along with action-reason generation using an LSTM branch). We find that their top-1 and top-3 accuracy is 52.4% (vs. 59.2% ours) and 75.7% (vs. 84.8% ours), which is lesser compared to our model.

[§] *Top-1 Accuracy*: It is defined as the fraction of images, where the highest predicted strategy is present in the ground-truth strategies. *Top-3 Accuracy* : It is defined as the fraction of images, where any of the top-3 highest predicted strategies is present in the ground-truth strategies.

Models	Top-1 Acc.	Top-3 Acc.
Our Model	**59.2**	**84.8**
w/o DenseCap	55.6	80.8
w/o Symbol	58.8	81.6
w/o DenseCap & Symbol	55.2	80.8
w/o OCR	54.8	82
w/o Symbol, OCR & DenseCap	58	78.8
w/o Action-Reason Task	56.4	80.4
Random Guess	6.25	18.75

Table 2.2: Effect of different Modalities and Tasks on the accuracy and performance of the strategy prediction task.

2.4.1.4 Multi Task Learning

One of the key opportunities for our persuasion strategies data labeling and modeling task was the presence of additional labels already given in the base Pitts Ads dataset. In that, authors had given labels about the reasoning task. For the reasoning task, the annotators were asked to provide answers in the form "I should [Action] because [Reason]." for each ad. In other words, they asked the annotators to describe *what the viewer should do and why*, according to the ad. Similar to the reasoning task, persuasion strategies provide various cognitive, behavioral, and affective reasons to try to elicit the motivation of the ad viewers towards their products or services. Therefore, we hypothesize that these natural language descriptions of *why the viewers should follow* the ad will be informative in inferring the ad's persuasion strategy.

We formulate obtaining action-reason statement as a sequence generation task where the model learns to generate a sentence $Y^g = (y_1^g, \ldots, y_T^g)$ of length T conditioned on advertisement X by generating the sequence of tokens present in the action-reason statement. To achieve this, we use a transformer decoder module that attends on the features $\text{Enc}(X)$ as shown in Fig. 2.11. The annotated action-reason statement is used to train the transformer decoder as an auxiliary task to strategy prediction through the standard teacher forcing technique used in Seq2Seq framework. Please refer to the Supplementary for more architectural details about the action-reason generation branch. As shown in Table 2.2, generating action-reason as an auxiliary task improves the strategy prediction accuracy by 2.8%. We

Model Used	Top-1 Accuracy	Top-3 Accuracy	Recall
Model with DenseCap & SSD	59.2	84.8	74.59
Model with BLIP & YOLOv5	58.4	83.8	71.58

Table 2.3: Comparison of caption and object detection models. We noticed that BLIP while being more recent and trained on a larger dataset, generates more informatory captions for background objects which DenseCap successfully ignores.

evaluate the performance on action-reason generation on following metrics: BLEU-1, BLEU-2, BLEU-3, BLEU-4, METEOR, ROUGE, CIDER, SPICE and observed a score of 53.6, 42.0, 33.1, 25.7, 26.3, 48.4, 42.8, 8.9 respectively.

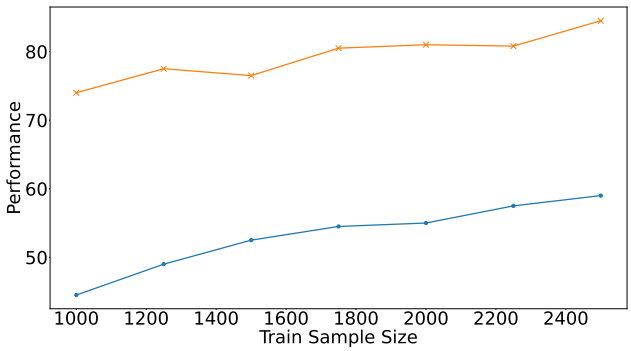


Figure 2.12: Incremental effect of introducing new data through active learning; Results for prediction of persuasion strategies on the test set

2.4.1.5 Active Learning

We use an active learning method to ease the large-scale label dependence when constructing the dataset. As in every active learning setting, our goal is to develop a learner that selects samples from unlabeled sets to be annotated by an oracle. Similar to traditional active learners [153, 154], we use uncertainty sampling to perform the sample selection. In doing so, such function learns to score the unlabeled samples based on the expected performance gain they are likely to produce and used to update the current version of the localization model being trained. To evaluate each learner, we measure the performance improvements, assessed on a labeled test set at different training dataset sizes.

	#ads with 1 strategy	#ads with 2 strategies	#ads with 3 strategies	Avg. strategies	Std. Dev.
Train-Set	1440	905	155	1.486	0.612
Val-Set	132	98	20	1.552	0.639
Test-Set	147	93	10	1.452	0.574
Total	1719	1096	185	1.49	0.592

Table 2.4: Distribution of test, train, validation, and the total dataset

At every learning step t, a set of labeled samples L_t is first used to train a model f_t. Then, from an unlabeled pool $U_t = D - L_t$, an image instance a is chosen by a selection function g. Afterwards, an oracle provides temporal ground-truth for the selected instance, and the labeled set L_t is augmented with this new annotation. This process repeats until the desired performance is reached or the set U_t is empty.

In our implementation, we instantiate the active learning selection function as the entropy of the probability distribution predicted by the model over the set of persuasion strategies for a given ad image instance a. Formally, $g = -\sum_{i=1}^{|\mathcal{P}|} p_i^n * \log(p_i^n)$, where p_i^n denotes the normalized probability with which i^{th} persuasion strategy is present in a as per the model prediction. The normalized probability p_i^n is estimated as $p_i / \sum_{j=1}^{|\mathcal{P}|} p_j$. Intuitively, ad samples with high entropy selection values indicate that the model trained on limited data has a higher degree of confusion while predicting the persuasion strategy since it is not decisively confident about predicting few strategies. Hence, we rank the unlabeled ad images in the decreasing order of difficulty according to the corresponding values of the entropy selection function and select the top-k ads in the subsequent batch for annotation followed by training. As shown in Fig. 2.12, we set k to be 250 and analyze the effect of incrementally introducing new samples selected through active learning. It can be seen that both top-1 and top-3 accuracy increases with the addition of new training data. We stop at the point when 2500 training samples are used since the model performs reasonably well with a top-1 and top-3 strategy prediction accuracy of 59.2% and 84.8% (see Fig. 2.12).

2.4.2 Modelling Persuasion Strategy For Video Advertisements

Large Language Models (LLMs) have been demonstrated to perform well for downstream classification tasks in the text domain. This powerful ability has been widely verified on natural language tasks, including text classification, semantic parsing, mathematical reasoning, *etc.* Inspired by these

advances of LLMs, we aim to explore whether they could tackle reasoning tasks on multimodal data (*i.e.* videos). Therefore, we propose a storytelling framework, which leverages the power of LLMs to verbalize videos in terms of a text-based story and then performs downstream video understanding tasks on the generated story instead of the original video. Our pipeline can be used to verbalize videos and understand videos to perform complex downstream tasks such as emotion, topic, and persuasion strategy detection.

We show the performance of our framework on fifteen distinct tasks across five datasets. Firstly, we employ a video story dataset to evaluate the story generation task. Secondly, we utilize a video advertisements dataset to assess topic and emotion classification, as well as action and reason generation. Then, the persuasion strategy dataset to evaluate the task of understanding persuasion strategies within stories, and finally, HVU and LVU for concept, user engagement, and attribute prediction. These diverse datasets allow us to evaluate the performance and capabilities of our framework thoroughly.

1. The Video story dataset [155] contains 105 videos, from four types of common and complex events (*i.e.* birthday, camping, Christmas, and wedding) and corresponding stories written by annotators. It has longer videos (average length 12.4 mins) and longer descriptions (162.6 words on average). Moreover, the sentences in the dataset are more sparsely distributed across the video (55.77 sec per sentence). *Metrics*: Following [155], we use several NLP metrics, *viz.*, BLEU-N, ROUGE-L, METEOR and CIDEr to measure the similarity between the story generated by the model and ground truth.

2. The Image and Video Advertisements [82] contains 3,477 video advertisements and the corresponding annotations for emotion and topic tags and action-reason statements for each video. There are a total of 38 topics and 30 unique emotion tags per video. Further, we have 5 action-reason statements for each video for the action-reason generation task. For our experiment, we use 1785 videos, due to other videos being unavailable/privated from Youtube.

Metrics: Following [82], for the topic and emotion classification task, we evaluate our pipeline using top-1 accuracy as the evaluation metric. Further, since [82] did not use any fixed set of vocabulary for annotations, rather they relied on annotator-provided labels, the labels are often very close (like cheerful, excited, and happy). Therefore, based on nearness in Plutchik's [156] wheel of emotions, we club nearby emotions and use these seven main categories: joy, trust, fear, anger, disgust, anticipation, and unclear. For the action-reason task, following [82], we evaluate our accuracy on the action and reason retrieval tasks where 29 random options along with 1 ground truth are provided to the model to find which one is the ground truth. Further, we also generate action and reason statements and evaluate the generation's faithfulness with the ground truth using metrics like ROUGE,

BLEU, CIDEr, and METEOR.

3. Persuasion strategy dataset: This is the dataset we contribute for understanding persuasion strategies.

Metrics: We evaluate the performance using top-1 accuracy metric. Videos have a varied number of strategies, therefore, we consider a response to be correct if the predicted strategy is present among the list of ground-truth strategies.

4. Long-Form Video Understanding (LVU): We *et al.* [157] released a benchmark comprising of 9 diverse tasks for long video understanding and consisting of over 1000 hours of video. The various tasks consist of content understanding ('relationship', 'speaking style', 'scene/place'), user engagement prediction ('YouTube like ratio', 'YouTube popularity'), and movie metadata prediction ('director', 'genre', 'writer', 'movie release year'). We *et al.* [157] use top-1 classification accuracy for content understanding and metadata prediction tasks and MSE for user engagement prediction tasks.

5. Holistic Video Understanding (HVU): HVU [158] is the largest long video understanding dataset consisting of 476k, 31k, and 65k samples in train, val, and test sets, respectively. A comprehensive spectrum includes the identification of various semantic elements within videos, consisting of classifications of scenes, objects, actions, events, attributes, and concepts. To measure performance on HVU tasks, similar to the original paper, we use the mean average precision (mAP) metric on the validation set.

Next, we explain our pipeline to solve these tasks.

2.4.2.1 Video Verbalization

To obtain a verbal representation of a video, we employ a series of modules that extract unimodal information from the multimodal video. This information is then used to prompt a generative language model (such as GPT-3.5 [63] and Flan-t5 [159]) to generate a coherent narrative from the video. The overall pipeline is depicted in Fig. 2.13. In the following, we delve into each component of the framework in details.

1. Video Metadata: Understanding the context of a story is crucial, and we achieve this by gathering information about the communicator (brand). We leverage the publicly available video title and channel name from the web. Additionally, we utilize Wikidata [160], a collaborative knowledge base that provides comprehensive data for Wikipedia, to obtain further details such as the company name, product line, and description. This information helps us comprehend the story elements and establish connections with the brand's business context. For non-advertisement videos, we skip this step and retrieve only the video title.

2. Text Representation of Video Frames: We extract two types of textual information from video frames. Firstly, we capture the literal text

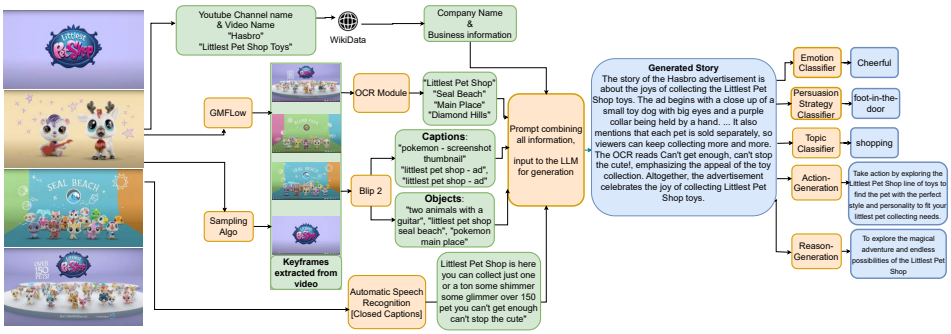


Figure 2.13: The overview of our framework to generate a story from a video and perform downstream video-understanding tasks. First, we sample keyframes from the video which are verbalized using BLIP-2. We also extract OCR from all the frames. Next, using the channel name and ID, we query Wikidata to get company and product information. Next, we obtain automatically generated captions from Youtube videos using the Youtube API. All of these are concatenated as a single prompt and given as input to an LLM and ask it to generate the story of the advertisement. Using the generated story, we then perform the downstream tasks of emotion and topic classification and persuasion strategy identification. This video can be watched at `https://youtu.be/ZBLkTALi1CI`.

present on the frames. Secondly, we analyze the scene depicted in each frame to gain a deeper understanding. In the upcoming sections, we will elaborate on both of these aspects.

Method	Frame Extraction	METEOR	CIDEr	Rougle-l	BLEU-1	BLEU-2	BLEU-3	BLEU-4
GPT-3.5	Uniform Sampling	24.8	102.4	24.3	63.8	56.4	47.2	38.6
GPT-3.5	Pyscenedetect	24.17	67.8	21.17	54.59	49.05	41.54	33.88

Table 2.5: Comparison of Pyscenedetect [161] with uniform sampling of choosing video frames. Based on downstream performance, we can see that uniform sampling works better than Pyscenedetect

a. Visual and Scenic Elements in Frames: For videos with a duration shorter than 120 seconds, we employ an optical flow-based heuristic using the GMFlow model [162] to extract keyframes. In shorter advertisement videos, scene changes often indicate transitions in the story, resulting in keyframes with higher optical flow values. The GMFlow model effectively captures these story transitions. We select frames with an optical flow greater than 50 and prioritize frames with maximum pixel velocity. However, for longer videos, this approach yields a large number of frames that are difficult to accommodate within a limited context. To address this, we sample frames at a uniform rate based on the native frames-per-second (fps) of the video (see Table 2.5 for a comparison between uniform sampling and Pyscenedetect).

Transcript : well hey Zuma that's the wrong pinata .... how old are you .... my birthday cake is its zoo .... paw patrol you .... got as a present ... Thank You Marshall.... are you ready to blow out the candle.... Marshall let's push chase and the pool ... Wow all the surprises we got.... Go first okay put on the blindfold ready ..... for more fun click the red subscribe button

Generated Story : A family had the perfect idea for a Paw Patrol-themed birthday party. The kids had a blast blowing bubbles in the backyard and playing in the pool with a mascot in a Paw Patrol costume. The birthday boy got to play with a pinata in the backyard and a hot air balloon flew over the house. The kids played with a toy bike and a Paw Patrol playhouse, and the birthday girl hugged a mascot. They also set up a LEGO train set and a DC Comics mascot in the pool. The birthday party ended with a game of catch and a pin the badge on the mascot game. Everyone had a great time, and the birthday cake was a Paw Patrol-themed delight. This vlog shows that family fun can be had at any birthday party with a little imagination.

Topic : Celebrations	**Emotion** : Cheerful	**Persuasion Strategy**: Social Identity, Social Proof
Action : Consider using your imagination to create a fun and memorable birthday party for your family or friends.		**Reason** : The reason is to create a uniquely fun-filled and memorable birthday party for your family or friends

Figure 2.14: An example of a story generated by the proposed pipeline along with the predicted outputs of the video-understanding tasks on the generated story. The generated story captures information across scenes, characters, event sequences, dialogues, emotions, and the environment. This helps the downstream models to get adequate information about the video to reason about it correctly. The original video can be watched at `https://youtu.be/_amwPjAcoC8`.

Additionally, we discard frames that are completely dark or white, as they may have high optical flow but lack informative content.

Using either of these methods, we obtain a set of frames that represent the events in the video. These frames are then processed by a pretrained BLIP-2 model [163]. The BLIP model facilitates scene understanding and verbalizes the scene by capturing its most salient aspects. We utilize two different prompts to extract salient information from the frames. The first prompt, "*Caption this image*", is used to generate a caption that describes what is happening in the image, providing an understanding of the scene. The second prompt, "*Can you tell the objects that are present in the image?*", helps identify and gather information about the objects depicted in each frame.

b. Textual elements in frames: We also extract the textual information present in the frames, as text often reinforces the message present in a scene and can also inform viewers on what to expect next [164]. For the OCR module, we sample every 10th frame extracted at the native frames-per-second of the video, and these frames are sent to PP-OCR [160]. We filter the OCR text and use only the unique words for further processing.

3. Text Representation of Audio: The next modality we utilize from

the video is the audio content extracted from it. We employ an Automatic Speech Recognition (ASR) module to extract transcripts from the audio. Since the datasets we worked with involved YouTube videos, we utilized the YouTube API to extract the closed caption transcripts associated with those videos.

4. Prompting: We employ the aforementioned modules to extract textual representations of various modalities present in a video. This ensures that we capture the audio, visual, text, and outside knowledge aspects of the video. Once the raw text is collected and processed, we utilize it to prompt a generative language model in order to generate a coherent story that represents the video. To optimize the prompting process and enable the generation of more detailed stories, we remove similar frame captions and optical character recognition (OCR) outputs, thereby reducing the overall prompt size.

The prompt template is given in Section 2.4.2.2. Through experimentation, we discovered that using concise, succinct instructions and appending the text input signals (such as frame captions, OCR, and automatic speech recognition) at the end significantly enhances the quality of video story generation. For shorter videos (up to 120 seconds), we utilize all available information to prompt the LLM for story generation. However, for longer videos, we limit the prompts to closed captions and sampled frame captions. The entire prompting pipeline is zero-shot and relies on pre-trained LLMs. In our story generation experiments, we employ GPT-3.5 [63], Flan-t5 [159], and Vicuna [165]. A temperature of 0.75 is used for LLM generation. The average length of the generated stories is 231.67 words. Subsequently, these generated stories are utilized for performing video understanding tasks.

2.4.2.2 Prompt format

For verbalization, a template prompt format has been used, including all the data components as objects, captions, asr, ocr, meta-data.

"Please write a coherent story based on the following video advertisement. Use only the information provided and make sure the story feels like a continuous narrative and at the end include one sentence about what product the advertisement was about. Do not include any details not mentioned in the prompt.Use the elements given below to create a coherent narrative,but don't use them as it is.The advertisement for the company {company_name} The video is titled {title}, with captions that include {caption}, voice-over : {transcripts}, and object recognition descriptions : {ocr}. The following objects are present in the advertisement and should be used to help create the story: {objects} Please exclude any empty or stop words from the final text."

For downstream tasks, a template prompt format with an instruction about the specific task,the previous generated verbalization and vocabulary

for the downstream task is prompted to the LLM. Here is the example for the topic detection task, for other tasks context and vocab were changed accordingly.

"Given {topics} identify the most relevant topic from the dictionary keys from topic_vocab related to the story of the video advertisement given below.Consider the definitions given with topics in the topic_vocab dictionary, to identify which topic is most relevant, don't add any extra topics that are not given in dictionary keys and answer with just the most relevant topic. Story : {verbalization}"

Training	**Model**	**Topic**	**Emotion**		**Persuasion**	**Action**	**Reason**
			All labels	**Clubbed**			
Random	Random	2.63	3.37	14.3	8.37	3.34	3.34
Finetuned	VideoMAE [166]	24.72	29.72	85.55	11.17	-	-
	Hussain *et al.* [82]	35.1	32.8	-	-	-	48.45
	Intern-Video [167]	57.47	36.08	86.59	5.47	6.8	7.1
Zero-shot	VideoChat [168]	9.07	3.09	5.1	10.28	-	-
Our Framework	GPT-3.5 Generated Story + GPT-3.5 Classifier	51.6	11.68	79.69	35.02	66.27	59.59
Zero-shot	GPT-3.5 Generated Story + Flan-t5-xxl Classifier	60.5	10.8	79.10	33.41	79.22	81.72
	GPT-3.5 Generated Story + Vicuna Classifier	22.92	10.8	67.35	29.6	21.39	20.89
	Vicuna Generated Story + GPT-3.5 Classifier	46.7	5.9	80.33	27.54	61.88	55.44
	Vicuna Generated Story + Flan-t5-xxl Classifier	57.38	9.8	76.60	30.11	77.38	80.66
	Vicuna Generated Story + Vicuna Classifier	11.75	10.5	68.13	26.59	20.72	21.00
Finetuned	Generated Story + Roberta Classifier	71.3	33.02	84.20	64.67	42.96*	39.09*

Table 2.6: Comparison of all the models across topic, emotion, and persuasion strategy detection tasks. We see that our framework, despite being zero-shot, outperforms finetuned video-based models on the topic classification, persuasion strategy detection and action and reason classification tasks and comes close on the emotion classification task. Further, the Roberta classifier trained on generated stories outperforms both finetuned and zero-shot models on most tasks. Best models are denoted in green and runner-ups in blue.

2.4.2.3 Results

Video Storytelling: The performance comparison between our pipeline and existing methods is presented in Table 2.7. We evaluate multiple generative and retrieval-based approaches and find that our pipeline achieves state-of-the-art results. It is important to note that as our method is entirely generative, the ROUGE-L score is lower compared to retrieval-based methods due to less overlap with ground truth reference video stories. However, overall metrics indicate that our generated stories exhibit a higher level of

	Method	Model Type	METEOR	CIDEr	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Random	Random	Retrieval	13.1	30.2	21.4	43.1	23.1	10.0	4.8
Finetuned	Narrator [155]	Retrieval	19.6	98.4	29.5	69.1	43.0	25.3	15.0
	EMB [155]	Retrieval	19.1	88.8	28.9	64.5	39.3	22.7	13.4
	BRNN [155]	Retrieval	18.1	81.0	28.3	61.4	36.6	20.3	11.3
	ResBRNN [155]	Retrieval	19.6	94.3	29.7	66.0	41.7	24.3	14.7
	Pseudo-GT+ ResBRNN-kNN [155]	Retrieval	20.1	103.6	29.9	69.1	43.5	26.1	15.6
	GVMF [169]	Retrieval	20.7	107.7	30.8	70.5	44.3	26.9	15.9
Zero-shot	VideoChat [168]	Generative	15.49	42.9	17.88	50.00	43.30	34.76	27.21
Zero-shot	GPT-3.5	Generative	24.8	102.4	24.3	63.8	56.4	47.2	38.6
Our Framework	Vicuna	Generative	17.4	73.9	20.9	70.49	60.0	48.25	38.20
	Flant-t5-xxl	Generative	4.8	34.6	10.58	7.9	6.8	5.4	4.3
	Uniformly Sampled BLIP-2 Captions	Generative	21.7	108.9	24.04	55.19	48.5	40.7	33.76

Table 2.7: Comparison on story generation task on the video-story dataset. We see that our framework despite being zero-shot outperforms all the fine-tuned generative prior art on all metrics. Further, it also outperforms fine-tuned retrieval models, which choose from a fixed set of frame descriptions on most metrics. Best models are denoted in green and runner-ups in blue.

Task	Model	METEOR	CIDEr	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Action	GPT-3.5	20.46	41.7	9.5	18.7	14.8	11.8	9.4
Action	Flan-t5-xxl	15.75	61.5	13.6	50.0	34.8	26.9	21.8
Action	Vicuna	21.20	42.6	7.6	16.8	13.08	10.08	7.7
Reason	GPT-3.5	13.34	16.7	7.8	27.1	20.8	14.7	10.4
Reason	Flan-t5-xxl	8.35	24.9	5.9	39.4	24.7	16.7	12.0
Reason	Vicuna	15.82	27.9	7.75	24.6	19.3	14.1	10.3
Reason given action	GPT-3.5	13.77	29.4	8.7	33.5	24.9	17.9	13.2
Reason given action	Flan-t5-xxl	4.29	19.0	7.6	23.2	15.0	10.2	7.5
Reason given action	Vicuna	13.62	24.4	7.61	22.6	17.7	12.8	9.2

Table 2.8: Comparison of the different zero-shot models on the action and reason generation tasks. Note that there are no fine-tuned generative models in the literature for this task and the number of annotated videos is too small to train a generative model. Best models are denoted in green.

Training	**Model**	**relationship**	**way_speaking**	**scene**	**like_ratio**	**view_count**	**director**	**genre**	**writer**	**year**
Trained	R101-slowfast+NL [157]	52.4	35.8	54.7	0.386	3.77	44.9	53.0	36.3	52.5
Trained	VideoBert [170]	52.8	37.9	54.9	0.320	4.46	47.3	51.9	38.5	36.1
Trained	Xiao *et al.* [171]	50.95	34.07	44.19	0.353	4.886	40.19	48.11	31.43	29.65
Trained	Qian *et al.* [172]	50.95	32.86	32.56	0.444	4.600	37.76	48.17	27.26	25.31
Trained	Object Transformers [157]	53.1	39.4	56.9	0.230	3.55	51.2	54.6	34.5	39.1
Zero-shot (Ours)	GPT-3.5 generated story + Flan-t5-xxl	64.1	39.07	60.2	0.061	12.84	69.9	58.1	52.4	75.6
Zero-shot (Ours)	GPT-3.5 generated story + GPT-3.5 classifier	68.42	32.95	54.54	0.031	12.69	75.26	50.84	32.16	75.96
Trained (Ours)	GPT-3.5 generated story + Roberta	62.16	38.41	68.65	0.054	11.84	45.34	39.27	35.93	7.826

Table 2.9: Comparison of various models on the LVU benchmark. We see that our framework, despite being zero-shot, outperforms fine-tuned video-based models on 8/9 tasks. Best models are denoted in green and runner-ups in blue.

Training	Model	Scene	Object	Action	Event	Attribute	Concept	Overall
Trained	3D-Resnet	50.6	28.6	48.2	35.9	29	22.5	35.8
Trained	3D-STCNet	51.9	30.1	50.3	35.8	29.9	22.7	36.7
Trained	HATNet	55.8	34.2	51.8	38.5	33.6	26.1	40
Trained	3D-Resnet (Multitask)	51.7	29.6	48.9	36.6	31.1	24.1	37
Trained	HATNet (Multitask)	57.2	35.1	53.5	39.8	34.9	27.3	41.3
Zero-shot (Ours)	GPT-3.5 generated story + Flan-t5-xxl classifier	59.66	98.89	98.96	38.42	67.76	86.99	75.12
Zero-shot (Ours)	GPT-3.5 generated story + GPT-3.5 classifier	60.2	99.16	98.72	40.79	67.17	88.6	75.77

Table 2.10: Comparison of various models on the HVU benchmark [158]. The models scores are as reported in [158]. We see that our framework, despite being zero-shot, outperforms fine-tuned video-based models on all the tasks. Best models are denoted in green and runner-ups in blue.

similarity to the reference stories and effectively capture the meaning of the source video.

Video Understanding: The performance comparison between our pipeline and other existing methods across six tasks (topic, emotion, and persuasion strategy classification, as well as action and reason retrieval and generation) is presented in Tables 2.6 and 2.8. Notably, our zero-shot model outperforms finetuned video-based baselines in all tasks except emotion classification. Further, our text-based finetuned model outperforms all other baselines on most of the tasks.

Unlike the story generation task, there are limited baselines available for video understanding tasks. Moreover, insufficient samples hinder training models from scratch. To address this, we utilize state-of-the-art video understanding models, VideoMAE and InternVideo. InternVideo shows strong performance on many downstream tasks. Analyzing the results, we observe that while GPT-3.5 and Vicuna perform similarly for story generation (Table 2.7), GPT-3.5 and Flan-t5 excel in downstream tasks (Table 2.6). Interestingly, although GPT-3.5 and Vicuna-generated stories yield comparable results, GPT-3.5 exhibits higher performance across most tasks. Vicuna-generated stories closely follow GPT-3.5 in terms of downstream task performance.

Next, we compare the best models (as in Table 2.6) on the LVU and HVU benchmarks with respect to the state-of-the-art models reported in the literature. Tables 2.9 and 2.10 report the results for the comparisons. As can be noted, the zero-shot models outperform most other baselines. For LVU, the zero-shot models work better than the trained Roberta-based classifier model. For HVU, we convert the classification task to a retrieval task, where in a zero-shot way, we input the verbalization of a video along with 30 randomly chosen tags containing an equal number of tags for each category (scene, object, action, event, attribute, and concept). The model is then prompted to pick the top 5 tags that seem most relevant to the video.

Model	Topic	Emotion		Persuasion	Action	Reason
		All labels	Clubbed			
BLIP-2 Captions + Flant-t5-xxl	32.2	7.4	43.11	32.1	52.98	76.26
BLIP-2 Captions + GPT-3.5	32.7	7.9	76.69	30.1	49.91	58.71
Audio Transcription + Flant-t5-xxl	49.37	10.1	63.56	21.9	66.17	79.68
Audio Transcription + GPT-3.5	32.88	6.4	75.97	32.25	64.98	61.78

Table 2.11: Ablation study of using only visual (caption) or audio (transcripts) and LLMs for downstream tasks. It can be noted that the overall model does not perform as well (compared to Table 2.6) when using only audio or scene description without generating story.

These tags are mapped back to the main category tags, which are treated as the predicted labels.

Furthermore, as a comparative and ablation study of our approach, we evaluate the performance using only the BLIP-2 captions and audio transcriptions (Table 2.11). Our findings highlight that generated stories leveraging both audio and visual signals outperform those using vision or audio inputs alone. This emphasizes the significance of verbalizing a video in enhancing video understanding.

2.4.2.4 Ablation

Among the different components of information input present in the prompt, the LLM utilizes them differently while constructing the verbalization for the videos.For this experiement we use a subset of [82] dataset, considering videos that have spoken audio present.

We use ROUGE-l to get the longest common subsequence (LCS) between the generated verbalization and the individual components,which captures the overlapping content, providing an indication of their semantic similarity.

As generated verbalizations are abstractive as compared to extractive, we also use cosine similarity between the Roberta embeddings of the generated verbalization and the individual components.

We find that despite the order of the components in the prompt, the LLMs tend to utilize the audio components in the videos, in an extractive way.

2.4.2.5 A few examples of the stories generated using our method

1. "Cathy 30 Master" by Grand Canyon University (`https://www.youtube.com/watch?v=lPdD8NvVfw0`): Kathy Ames had always wanted to pursue a doctoral degree but was unsure about the time commitment. When she discovered Grand Canyon University, she knew she had found the perfect fit. Grand Canyon University offered a flexible schedule that would allow her to balance her personal and family life with her studies. She - along

Model	Top-5 Accuracy	mAP
VideoMAE	25.57	24.79
InternVideo	7.477	15.62
GPT-3.5 Generated Story + GPT-3.5	34.2	27.53
Vicuna Generated Story + GPT-3.5	31.54	27.24
GPT-3.5 Generated Story + Flant5	37	27.96
Vicuna Generated Story + Flant5	31.13	27.32

Table 2.12: Top-5 accuracy, and mAP for persuasion strategy detection task

with other students - gathered in the classroom, excitedly listening to their coach, Scott Saunders, explain the program. Afterward, Kathy made her way to the library and settled into a chair with her laptop.

She studied diligently, surrounded by her peers and classmates. In the evenings, she met with her peers around the table to discuss the topics of the day. Everyone was always eager to help and support each other. After a long day, Kathy made her way back to her living room where she relaxed on the couch with a glass of water and a lamp providing a soothing light.

Kathy was grateful for the opportunity to pursue her dream at Grand Canyon University. She was able to learn from experienced faculty and gain real-world experience that would prepare her for success after graduation.

The advertisement for Grand Canyon University was about offering a private, Christian education at an affordable price.
2. "Today is the Day for Doing | The Home Depot" by Home Depot (`https://www.youtube.com/watch?v=f_6QQ6IVa6E`): The woman holding the book stepped onto the patio and looked up to the sky. She was ready to take on the day. Taking out her phone, she opened the furniture catalog app, scrolling through the various designs. She quickly decided on the perfect pieces to brighten up her home. Next, she headed to The Home Depot for the supplies she needed. As she entered the store, the woman was delighted to find all the tools and materials she needed, from the Ryobi Cordless Vacuum to the Leaf Blower. She was even more excited when she spotted the Splatter an object recognition tool that allowed her to easily find the perfect paint color for her project. With her shopping done, the woman made her way to the checkout line with a cup of coffee in hand. She couldn't wait to get to work and make her home more beautiful. She knew that with the help of The Home Depot, Today was the Day for Doing. This advertisement was for The Home Depot - the one-stop-shop for all your home improvement needs.

3. "October 2018 Color of the Month: Shell White - Sherwin-Williams" by Sherwin-Williams (`https://www.youtube.com/watch?v=PJlHiQJBDMw`): The advertisement for the company Sherwin-Williams opens on a kitchen

table strewn with shells and wicker baskets, with two glasses of iced water beside them. A vase with a blue pattern sits in the foreground, and a person holds up a phone with the Sherwin-Williams logo on the screen. A girl appears from behind a white sheet, peeking out of a white tent as if to signify the timelessness and neutrality of this color. The voice-over begins, as the camera pans to a living room with a staircase, and then to a dining room with a white table, chairs, and a white vase. The words "Color of the Month: Shell White, Sherwin-Williams" appear on the screen, as the camera zooms in on the vase. The words are followed by Öur app makes it a snap,r̈eferring to Color Snap, the company's new way of painting a home. The advertisement ends with the Sherwin-Williams logo, emphasizing the company's commitment to excellence in home painting. This advertisement was promoting the company's color of the month, Shell White.

4. "Side-by-side comparison of SolidWorks w/NVIDIA Quadro FX 1700 vs. Quadro 2000" by NVIDIA (`https://www.youtube.com/watch?v=CDjBIt70fp4`): The story began with a green light glowing in the dark, symbolizing the presence of a powerful technology that can change the way we work. This technology was an advanced graphics card, the NVIDIA Quadro FX 1700. It was compared side-by-side with its successor, the Quadro 2000, and it was clear that the Quadro 2000 was far more powerful. The Quadro FX 1700 had a qt - x700 vs qt - x700 capacity, while the Quadro 2000 had a green light that shone brighter and further than before. As the comparison was being made, the results were clear: the Quadro 2000 was the superior product. This advertisement for the company NVIDIA showcased the power of the Quadro 2000, and the improved performance it could bring to an organization. The advertisement concluded with a green light, signaling that NVIDIA had the answer to improving workflows. The product the advertisement was about was the NVIDIA Quadro 2000."

5. "Training sessions in the nature - Mary Kom | Priyanka Chopra | In Cinemas NOW" by Viacom 18 Studios (`https://www.youtube.com/watch?v=SXTSJC5NrLs`): Mary Kom was training hard to make her mark as a boxer, and she had nothing but her focus and determination to help her succeed. She was a force to be reckoned with in the boxing ring and her hard work was soon paying off.But she knew that she had to take her training to the next level if she wanted to win the boxing championship. That's when she decided to take her own advice and step out of her comfort zone. She headed out to the woods, determined to make the most of her time in nature.The woods were a perfect place for Mary to challenge herself physically. She was soon standing on the rocks of a river, pushing her body to its limits with every movement. She was standing tall, her black top glinting in the sunlight, and her energy was palpable. The sun was setting, and it was the perfect time to relax and unwind.Finally, she ended her day with a yoga routine,

her body twisting and turning in a beautiful display of strength and grace. It was a day of hard work and fun, and it left Mary feeling inspired and ready to take on anything. The video was an advertisement for the movie Mary Kom, released by Viacom18 Studios."

2.4.2.6 Hallucinations Present In the Automatically Generated Stories

The current large language models , do hallucinate or produce imaginary outputs. However, they are able to capture the gist of the videos, helping improve on downstream tasks. Some examples are shown below:

1. "The Fundamentals Of Caring | Clip: "Perverts" | Netflix" by Netflix (`https://youtu.be/_Usyxm9S_Jg`)
   (a) Story: The woman in the advertisement is standing in front of a red background, smiling brightly. She and a man sitting at a table look content, as if they've just come from a conversation. The Netflix logo is visible, indicating that the advertisement is for the company and its products. The voice-over for the advertisement introduces Selena Gomez as the star of the trailer for the popular Twilight Saga. Her popularity has made her a perfect choice to promote Netflix's products. The voice-over then informs viewers that the company's streaming service is not available. The man and woman in the advertisement seem to have a strong connection, and we can infer that they are friends. Perhaps they share an understanding of each other's struggles, since they both seem to be comfortable and happy with each other. The advertisement ends with the Netflix logo, leaving viewers with the knowledge that they can find solace and understanding in the company's products, no matter what they are going through. The advertisement was for Netflix."
   (b) Hallucinations: The story falsely attributes "Selena Gomez" to "Twilight Saga". The LLMs often confuse named entity linkages.
2. "Back-to-School 2016" by Academy Sports + Outdoors (`https://youtu.be/SZwWT5lnHf8`)
   (a) Story: A young girl is walking through her bedroom when she notices a man running up the stairs in his sneakers. She is intrigued and follows him, curious to see where he is going. Arriving at the top of the stairs, she finds a young man standing with his backpack. Across the room, she sees a girl in a dress standing in front of a mirror. The girl smiles and looks content. At that moment, the voice-over of the Academy Sports + Outdoors ad can be heard: "Academy Sports and Outdoors is the place to get all the best brands in thousands of combinations for

back to school. Right stuff, low price, every day." Suddenly, a black cat can be seen flying in the sky and the girl in the tank top is now standing in front of the mirror. The man is wearing blue pants and sneakers, and the shoes, rug, and sneakers of the apartment are visible. The girl smiles and looks content. The scene shifts to the bedroom where the bed, dresser, and mirror are present, along with a sign for Academy Sportswear. This advertisement for Academy Sports + Outdoors was about the company's back-to-school collection of products. It showcased how Academy Sports + Outdoors provides the right stuff, at a low price, every day, for students to get ready for school.

   (b) The scene with a black cat is hallucinated though not mentioned in the prompt

3. "Many Worlds, One Home l Anime 2019 l NX" By Netflix (`https://youtu.be/B63ubKGNqos`)
   (a) Story: The advertisement by Netflix, titled "Many Worlds, One Home", is a stunning vision of many different universes. We see a man standing on a red moon, a desert landscape, and a live performance by Adam Sandler. Purple backgrounds with the word "worlds" convey the idea of many possibilities, while a boy playing with a ball in front of a statue shows a more peaceful and homely side to this. As the night falls, a man is standing on a car, and the Ultraman. There are also glimpses of Tse Seeds, Devilman Crybaby, Saint Seiya: Knights of the Zodiac, Eden Vs. Eden, Carole and Tuesday, Aggretsuko, Seis Manos, Godzilla, Castlevania, B: The Beginning, Baki, Cannon Busters, Rilakkuma and Kaoru, Beastars, and Ultramarine Magmell. We see a teddy bear, knights of the zodiac, a character walking in the water, a man in a snowy area, a pair of shoes, a man running away from the fire, the moon, a man wearing a suit with a green light, eyes being the only thing that can be seen, birds flying in the sky, a red and black background with a red light, bats and castles, the eyes, a man holding a gun, a man flying in the air, a face, a girl with long hair, and two hands. All of this culminates to signify the many different worlds that one can find "home", and the advertisement concludes with the Netflix logo. This advertisement for Netflix captures the idea of many possibilities and of being at home in a single place.

   (b) Hallucination: Blip2 hallucinates entities such as "Adam Sandler" who is not associated with the advertisement. But it causes the generated story to have the same association.

4. "African Cats: Herding Cats - Clip" by Walt Disney Studios (`https://youtu.be/WCIEmrN8ot4`)
   (a) Story: The sun shone brightly in the savanna as the cheetah cub

frolicked in the grass. Suddenly, a cheetah emerged from the shadows. The cheetah and it's cub plan to chase the gazzeles. Despite the presence of death, life in the wild continued on, just as it had for centuries. This story is featured in the advertisement for Walt Disney Studios, which captures the beauty and unpredictable nature of life in the African savanna. This advertisement ultimately serves to promote the film African Cats, which tells the story of a cheetah family as they strive to protect their respective prides against a variety of obstacles in the wild.

(b) Hallucination: The generated caption "a fox cub is playing in the grass cheetah eats in the wild" was insufficient to capture the overall information of the video, however, due to the presence of ASR and world knowledge of the LLM. The LLM is able to generate a coherent story.

5. "Bright & Dreamy: Pastel Tint| Ulta Beauty" by ultabeauty (`https://youtu.be/Lh5n4HcLvCU`)
   (a) Story: The woman in the Ulta Beauty advertisement skillfully applied makeup to her face, creating a beautiful pastel tint. With each brush stroke, her look became brighter and more dreamy. She beamed with confidence and joy, her face glowing with beauty. As the advertisement continued, she highlighted her eyes and cheeks with subtle pastel shades, adding to the stunning effect. With each addition, her face lit up with a vibrant luminosity. When she was finished, her face was a work of art, the perfect combination of bright and dreamy. The Ulta Beauty advertisement showcased the power of a perfect pastel tint, and how it can enhance any look. This advertisement was for Ulta Beauty.
   (b) Hallucination: The video contains only music. However, the generated verbalization can pick the advertisement's gist.
6. "Round Brilliant Diamond Hoop Earrings 14kt White Gold" by Costco `https://youtu.be/VdhRmVnbqQo`
   (a) Story: Jackie had been wanting diamond hoop earrings for what seemed like forever. She had been dreaming of the perfect pair that she could wear on special occasions and that would bring out the best in her. One day Jackie happened to come across an advertisement for Costco, which featured a pair of Round Brilliant Diamond Hoop Earrings in 14kt White Gold. She immediately knew this was the perfect pair for her. The advertisement had a voice-over that said, "These earrings sparkle with glamour and sophistication." Jackie was mesmerized by the sparkle of the diamonds, and she was sure anyone who saw them would also be taken in by their beauty. Jackie quickly decided to purchase the earrings. She was sure they would make the perfect accessory for any special occasion. From date nights to family gatherings,

she knew these earrings would make her look simply stunning. The advertisement for the Costco Round Brilliant Diamond Hoop Earrings in 14kt White Gold proved to be the perfect choice for Jackie. They not only added a touch of glamour and sophistication to her wardrobe, but made her feel extra special when she wore them. The advertisement for the company Costco was for a pair of Round Brilliant Diamond Hoop Earrings in 14kt White Gold.

(b) Hallucination: The video has no audio transcription, and minimal visual information. However the LLM hallucinates a story, maintaining the essence of the advertisement. It also attributes a false voice-over which is not present in the video.

2.5 Conclusion

What does an advertisement say that makes people change their beliefs and actions? With limited works, the computational study of rhetoric of this all-pervasive form of marketing communication is still in its infancy. In this chapter, based on the well-developed social psychology and marketing literature, we develop and release the largest vocabulary of persuasion strategies and labeled dataset. We develop several models for predicting persuasion strategies for video and image based ads. Further, we show the performance of these models on several other advertisement-understanding related tasks, including topic, emotion, and question-answering.

Chapter 3

Large Content and Behavior Models To Predict, Understand, and Generate Content and Human Behavior

In the previous chapter, we dealt with the first culture of social science, explanation, and how to enable it at scale by using machine learning techniques of computer vision and NLP. Marketers make many decisions on a regular basis: what marketing campaign to launch, who to target, what the message should be, which channel it should be sent on, when it should be sent, and how frequently. Extraction of information about advertisements (for example, emotion, persuasion strategy, topic, and question answering) and correlating them with key performance indicators (KPIs) helps decision-makers (in this case, human marketers) to understand and execute campaigns better. Now, in this chapter, we turn to the question of how to encode the complete communication pipeline to enable better and possibly completely automated decision-making.

Thanks to the digitization of various aspects of life, humanity has been collecting a lot of data over the last two decades. For example, let's take the case of email marketing, one of the first marketing tools leveraging Internet technology. Say a Walmart marketer sends a Black Friday offer about a price drop on Apple devices to John, a 27-year-old male grad student living in Buffalo. The email was received at 09:57 AM and opened at 02:00 PM. Upon opening the email, email content consisting of a carousel of four images and three lines is dynamically fetched from the backend. John takes 5 seconds to scan the email quickly, scrolling halfway through, before deciding to click on a photo. During this single macro-transaction, a series of

micro-transactions are recorded and a host of machine learning and software systems are required to function together to make a sequence of decisions.

Amongst all the recorded transactions and algorithms, let's discuss the most prominent ones that are important for our use case. Much before sending the email, depending on business needs, the marketer decides to launch a particular campaign. The business need, for example, in this case, could be precipitated by an upcoming event or festival (Black Friday) or a rising inventory of Apple products. The next step is the creative process, where the marketer designs the email pods consisting of text and images by herself or with a team of creatives. The marketer has to decide the target segments (of which John will be a part). Next, an algorithm has to decide when to send the email and the subject line. Post this, a series of software technologies helps to send the email to the right people on time. When John decides to open the email, an event gets recorded in the backend recording `(customer ID, transaction ID, email ID, time of opening the email, device, email client, [other metadata])`. A personalization system then dynamically selects the email content and sends it to John's device. Those get recorded with the transaction ID. Scrolling on the email also generates transactions recording which images and text were sent to John's device. Further, when John decides to click on one link, another transaction gets recorded of the type `(transaction ID, customer ID, link, time of click, email client, device, [other metadata])`. On an abstract level, all of these transactions can be represented by the seven factors of communication: `(communicator, message, time of message, channel, receiver, time of effect, effect)`.

If this email were sent to a million subscribers, one email message would result in several hundred thousand transactions getting recorded. These transactions capture behavior data of the subscribers in response to a single email sent by the communicator, Walmart. This example illustrates the size and nature of behavioral data that gets captured. Notice that for a message, it is always the case that there is one sender and multiple receivers (an invariance noticed as early as 1950s [173]). Therefore, the scale of behavioral transactions generated is several orders higher than the number of unique pieces of content.

Given the magnitude of behavioral data collected, the natural question is can all that data be used to answer questions related to human behavior prediction, explanation, and optimization. Therefore, the research questions that we investigate in this chapter follow this natural line of inquiry:

1. How can behavior data help? Can behavior data help us to achieve the following goals:
   (a) Behavior Prediction
   (b) Behavior Explanation
   (c) Behavior Optimization?
2. How should we encode behavior data?

3. What kind of behavior can help?
    (a) How can implicit (like eye movements) and explicit (like clicks, likes, and views) behaviors help?
    (b) Can synthetically generated behavior data help?

To solve the behavior problems listed before, we can take inspiration from how the problem of learning natural language is being solved in the domain of large language models (LLMs). Raffel *et al.* [65], in their seminal work on T5, mention that the basic idea underlying large language models is to treat every text processing problem as a "text-to-text" problem, *i.e.*, taking the text as input and producing new text as output. This framework allows for a direct application of the same model, objective, training procedure, and decoding process to every task we consider. Further, this allows us to pre-train a model on a data-rich task like the next-word prediction, which can then be transferred to downstream tasks. Notably, thanks to the Internet, the next-word prediction task has huge amounts of available data. Consider the Common Crawl project (https://commoncrawl.org), one common source of data included in most language models. It produces more than 20TB of text per month sampled from random web pages across the internet.

T5 and other language models like GPT-3, Pythia [174], and Llama [68] can solve a wide variety of tasks, including the ones for which they were not explicitly trained. For instance, language models trained on the next word prediction task showed generalization capabilities across a wide variety of tasks like question-answering, summarization, natural language inference, and translation [63]. Recently, a series of papers have shown that this generalized "world understanding" captured in LLMs can be leveraged to enable them to "see" [22, 72, 163, 168, 175–177]. This is a significant capability enhancement since a model trained in language only settings can be made to reason about images and videos. These papers follow the same transfer learning approach advocated by T5, where they convert visual information to language space to leverage the "text-to-text" framework. They show that it is possible to teach a large language model, the new modality of vision, without needing to pre-train the model from scratch. Rather, using only a few million tokens, it is possible to scale LLMs' abilities to vision as well. Following this chain of thought, it could be possible to solve the effectiveness problem by posing it as a "text-to-text" problem. This is one of the paradigms we explore in this work. We show behavior generalization using several different types of behaviors.

Another possible way to integrate behavior with text is an encoder approach, which we will detail next. While behavior is a downstream effect of content, behavior contains signals about the content sent to the receiver and can help improve content-understanding and natural language processing. For instance, integration of human gaze data into neural network architectures has been explored for a range of computer vision tasks such as image captioning, visual question answering, and tagging [178–181]. In language

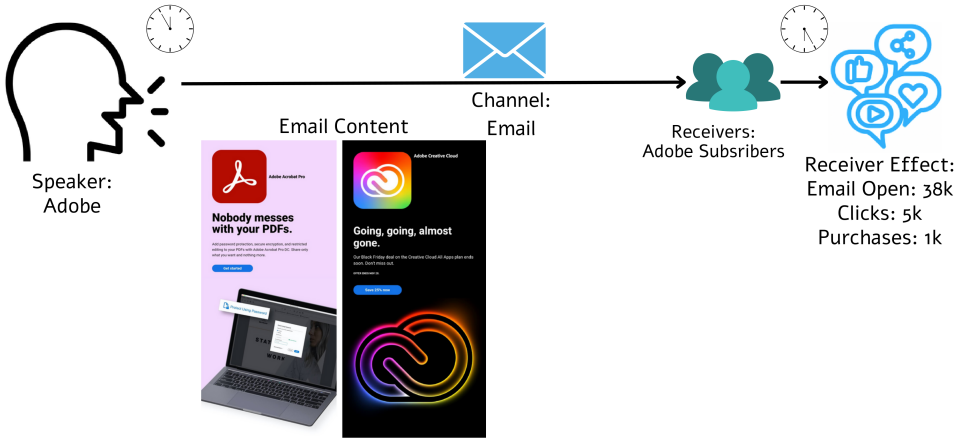


Figure 3.1: Communication process can be defined by seven factors: Communicator, Message, Time of message, Channel, Receiver, Time of effect, and Effect. Any message is created to serve an end goal. For marketers, the end goal is to bring in the desired receiver effect (behavior) (like clicks, purchases, likes, and customer retention). The figure presents the key elements in the communication pipeline - the marketer, message, channel, receivers, and finally, the receiver effect.

processing, tracking a reader's eye movements provides information about the cognitive processes of text comprehension [182, 183]. Hence, recent research has utilized features gleaned from readers' eye movement to improve the performance of complex NLP tasks such as sentiment analysis [184, 185], sarcasm detection [186], part-of-speech tagging [187], NER [188], and text difficulty [189]. While these studies show promise that behavior can be used to extract information about content, these are done in relatively small-scale lab settings needing real-time behavior to infer about content. Given these limitations, these approaches are not possible to scale. Scale helped LLMs to learn language. We therefore explore the paradigm of synthetic behavior generated over content and then scale it over to fine-tune a large language model to understand the possibilities of this paradigm better. We cover both the approaches next.

3.1 Large Content and Behavior Models (LCBM)

In this work, we explore the paradigm of the effectiveness problem as a text-to-text problem. The problem of effect is to know what the receiver does after receiving the message [4]. In general, for a piece of content, other than the content itself, we often have information about *who* consumes the content and what his *action* is on consuming the content. The latter is the effect described in Shannon's three levels of communication. For instance,

an email, as a message from the communicator to the receiver, elicits certain actions from the receiver like link-clicks, replies, and read-time. While LLMs are trained on trillions of tokens of content, the training does not include the receiver effect. For instance, Enron Email [190] is a popular corpus that is included in the training of LLMs like Pythia [174]. It contains 600K email content sourced from the Enron corporation, which LLMs use to learn how to write emails. However, it does not contain data about the receivers' activities, such as whether they opened the email, how long they kept it open (read-time), and what their reply was. Similarly, while major text corpora include a large number of public blogs and user forums to train LLMs like CommonCrawl, they are stripped of receiver behavior on forum messages, such as the number of likes, shares, and replies, before including them in LLM training (for instance, see [70, 71]). To pose the effectiveness problem as a text-to-text problem, we can include these *behavior tokens* in the text along with content tokens and train the LLM to model both of those in the same space. This might help the LLM simulate the receiver effect, optimize for it, and reason about it.

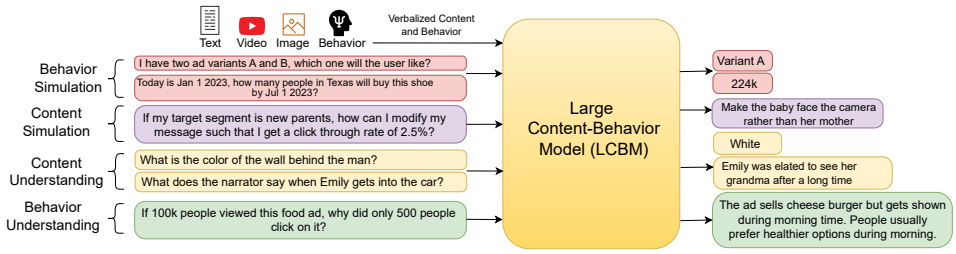


Figure 3.2: Encoding and predicting content (images, videos, and text) and behavior in the language space. Large Content Behavior Models (LCBMs), once trained, can enable a host of different applications, including behavior simulation, content understanding, content-behavior optimization, and content-behavior understanding.

In this work, we show initial experiments to integrate behavior as a new modality to increase the scope of multimodal LLMs from only content to both content and behavior. We call this new type of model a Large Content Behavior Model (LCBM). This class of models shows promise in enabling the LLMs to not only reason about content but also reason about and predict human behavior over that content. Further, LCBMs have the potential for behavior domain adaptation where models trained on one type of behavior can generalize on another behavior type (Fig. 3.3). Behavior simulation can enable many real-world applications, such as content recommendation, customer journey optimization, and A/B testing. To build LCBM, we introduce behavior instruction tuning (§3.1.5), an attempt to extend the instruction tuning paradigm to behavior space, bringing all seven communication fac-

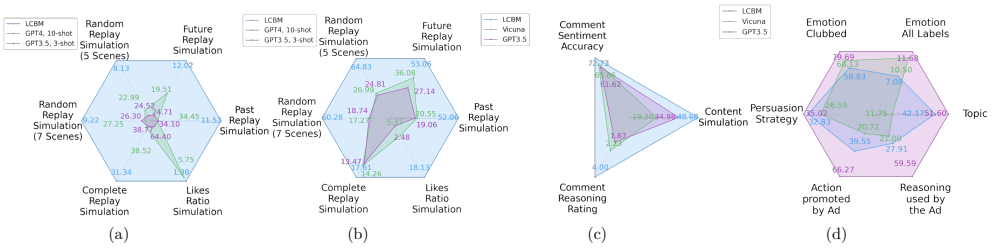


Figure 3.3: Comparison of GPT-3.5, GPT-4, Vicuna-13B, and LCBM-13B on ((a),(b)) Behavior Simulation on two types of behaviors: replay value prediction and likes/views prediction. The task is, given the video content and channel information, to predict replay values corresponding to each scene and the ratio of likes to views. (a) is the negative RMSE scores, and (b) is the accuracy. (c) Content simulation and behavior understanding tasks. The task for content simulation is, given the channel information and scene-level behavior, to predict the scene content. Given information on the video platform and the video content, the task of behavior understanding is to predict and explain the sentiments of the viewers and the commenters. Six evaluators scored the models' explanations between 0-5 to get the predicted sentiment and explanation scores by comparing the ratings and reasons with the user comments. The annotators did not know which model gave the reasoning. (d) Content understanding tasks. We evaluate four tasks: emotion, topic, and persuasion strategy prediction, and action-and-reason understanding. It can be noted that on the behavior simulation, content simulation, and behavior understanding tasks, LCBM performs better than 3-shot GPT-3.5 and 10-shot GPT-4 (covering a larger area. On the content understanding tasks, while LCBM outperforms similar-sized Vicuna models, GPT-3.5 performs better. However, we also note that GPT-3.5 and GPT-4 are at least 12 times larger than LCBM-13B. Further, we show the behavior domain adaptation results in Table 3.6.

tors (communicator, message, channel, receiver, send time, receive time, and effect) into the same space (Fig. 3.1). Similar to [63, 65, 72, 176], we do not design best-in-class predictors for any of the downstream tasks. Rather, we show a model which shows generalization capabilities across a wide variety of content- and behavior-related tasks. To summarize, we make the following two contributions:

- **Large Content Behavior Model (LCBM).** We develop a large multimodal model that shows capabilities of behavior simulation (given content), content simulation (given behavior), content understanding, and behavior understanding (Fig. 3.2). Following the text-to-text framework, we connect the Vicuna LLM [68, 165] with an open-set visual encoder of EVA-CLIP [191] and instruction fine-tune it end-to-end on behavior

instruction data. EVA-CLIP and QFormer [163] help the model to understand visual content in the language space, making it a Vision Language Model (VLM). During behavior instruction tuning, we teach the model to predict behavior given content and content given behavior using various instruction tasks (§3.1.5). This helps us teach behavior modality to the VLM while grounding it in the natural language space. We use three datasets to show the performance of LCBM: a dataset consisting of YouTube videos as the content and the corresponding retention graph, likes, the number of views, and comment sentiment as receiver behavior; a dataset consisting of Twitter posts (text, images, and videos) and corresponding human behavior (like counts) extracted from 168 million tweets across 10135 enterprise Twitter accounts from 2007 to 2023; and an internal dataset of in-house Marketing Emails[‖] (content) and the click-through rate corresponding to each segment they were sent to (behavior). We observe that teaching the LCBM behavior and content simulation improves its capabilities on them (expected), but the model also shows signs of domain-adaptation in behavior modality (few-shot capability, *unexpected*) (Tables 3.6, 3.7, 3.8) and improvements in behavior understanding (Figs. 3.7, 3.8, §3.1.6) (zero-shot capability, *unexpected*) [63]. See Fig. 3.3 for a radar plot of all the capabilities and comparisons of performances across LCBM and state-of-the-art LLMs: GPT-3.5 and GPT-4.

- **Dataset and Test Benchmark.** To spur research on the topic of large content and behavior models, we release our generated behavior instruction fine-tuning data from over 40,000 public-domain YouTube videos and 168 million Twitter posts. The data contains: 1) YouTube video links, automatically extracted key scenes, scene verbalizations, replay graph data, video views, likes, comments, channel name, and subscriber count at the time of collection, and 2) Twitter extracted account names, tweet text, associated media (image and video) verbalizations (including image captions, keywords, colors, and tones), tweet timestamps, and like counts. We also release a benchmark to test performance on the joint content behavior space (§3.1.4), introducing two types of tasks in this space: predictive and descriptive. In the predictive benchmark, we test the model's ability to predict behavior given the content and predict content given the behavior. In the descriptive benchmark, we validate its explanation of human behavior by comparing it with ground-truth annotations we obtain from human annotators that try to explain human behavior. See Figs. 3.7, 3.8 for a few examples.

[‖]We obtain in-house Marketing Emails dataset by collaborating with the in-house team.

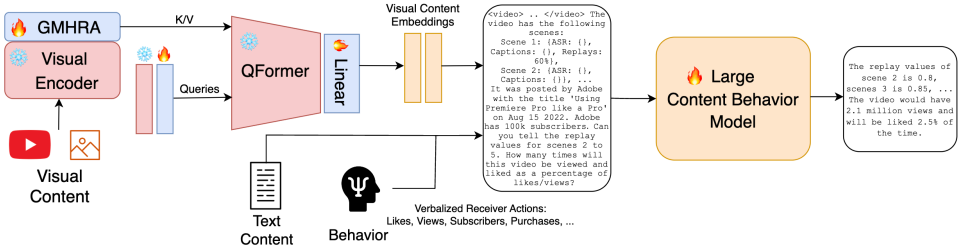


Figure 3.4: Encoding and predicting content (images, videos, and text) and behavior in the language space. Strategy to behavior instruction fine-tune (BFT) LLMs to create LCBMs. We capture visual concepts through the visual encoder (EVA-CLIP), and world knowledge is through an LLM (Llama). To leverage the rich knowledge of LLMs, we use GMHRA and QFormer to convert visual tokens of ViT to language tokens that Llama can understand. Further, we find that verbalizing the visual stimulus helps Llama to gather information more explicitly than what is provided by ViT+QFormer. We fine-tune the combined model end-to-end to predict 1) behavior given content and 2) content given behavior. Snowflake and fire symbols denote the frozen and unfrozen parts of the architecture.

3.1.1 Setup

In this section, we introduce our approach to model content and behavior together as a text-to-text problem. Since most publicly available corpora strip off receiver behavior from content, we first introduce our dataset, "The Content Behavior Corpus (CBC)", a dataset consisting of content and the corresponding receiver behavior. Next, we introduce our methodology to convert the content and behavior into text and our approach to model it using an LLM. Then, we cover the tasks through which we test various capabilities of LCBM (Fig. 3.2): content-understanding, behavior understanding, content simulation, behavior simulation, and behavior domain adaptation.

3.1.2 The Content Behavior Corpus (CBC)

The availability of large-scale unlabeled text data for unsupervised learning has fueled much of the progress of LLMs. In this work, we are interested in modeling content and the corresponding receiver behavior in the same space. While available datasets contain trillions of content tokens (text, images, audio, and videos), they unfortunately do not contain the receiver effect. To address this, we utilize YouTube and Twitter, two large publicly available sources of content-behavior data, consisting of (a) account name, account description, and number of subscribers and followers (*communicator data*), (b) rich content in the form of videos, images, creator-provided captions, titles, and descriptions (*message*), (c) behavior in the form of likes,

views, user comments, and replay graph (*receiver effect*). This covers all the seven factors of communication (Fig. 3.1), with the channel being fixed (as YouTube or Twitter) and receivers being average channel followers and viewers of the communicator. Since content data is multimodal in the form of a combination of images, videos, and text, and behavior data is in the form of numbers, to model it using a text-to-text paradigm, we *verbalize* both of them following the methodology we detail next.

Verbalization: For the video V, YouTube provides us with 100 average viewer retention values r_i for $i \in [0..100)$, corresponding to the entire video. The sampling rate of 100 is constant and independent of video length (T). Replay value r_i corresponds to video frames between the timestamps $(T/100 \times i, T/100 \times (i+1))$, which denotes how often these frames were replayed compared to the most replayed frames. The metric has a value between 0 and 1 that identifies the video's relative retention performance at a given point in the video. To accommodate longer video lengths, we merge replay values until $T/100 \times (i+j) - T/100 \times i > 1$ second with $j \in \{i+1, 100\}$. We choose the replay value for this merged group of scenes as $max(r_i, ..., r_j)$. Using this logic, we get replay values R_i for $i \in [0..m]$, where $m = \lfloor 100/(\lceil 100/T \rceil) \rfloor$. Next, we sample two frames randomly corresponding to each $i \in [0..m]$. We caption the frames using BLIP [163]. We also obtain the automatic speech recognition for the speech for the video between the timestamps corresponding to replay value R_i using Whisper [192]. The ASR and BLIP captions are content for scenes, and replay values are the behavior corresponding to them. We include the scene content and behavior in the video verbalization (Listing 3.1) with the sampling for both scene content and behavior as described above.

We also include video content by encoding video frames through EVA-CLIP [191] (explained in §3.1.3). Other than video embeddings, we include the video title and description as part of the video content. Corresponding to the overall video content, we verbalize overall video behavior metrics like video views and the ratio of likes and views. Finally, we append it with communicator information on the video channel and the subscriber count. The Listing 3.1 presents the overall verbalization for video and frame level content and behavior. The verbalization for Twitter posts is similar and is given in Listing 3.7.

Listing 3.1: Verbalization pattern for inputting content and behavior in the same space

Input: <video> ..[Video Tokens] .. </video>
The video has the following scenes:
Scene 1: {ASR: Welcome to a quick tutorial, OCR: Adobe Premiere Pro, Captions: A desktop interface, Replays: 60},
Scene 2: {ASR: on using Premiere Pro to edit, Captions: A computer interface, with an image of a white horse. Objects − Horse, Grass, Fence., Replays: 53},
...
It was posted by Adobe with the title 'Using Premiere Pro like a Pro' on Aug 15 2022. Adobe has 100k subscribers. This video was viewed by 346 thousand people and liked (as a

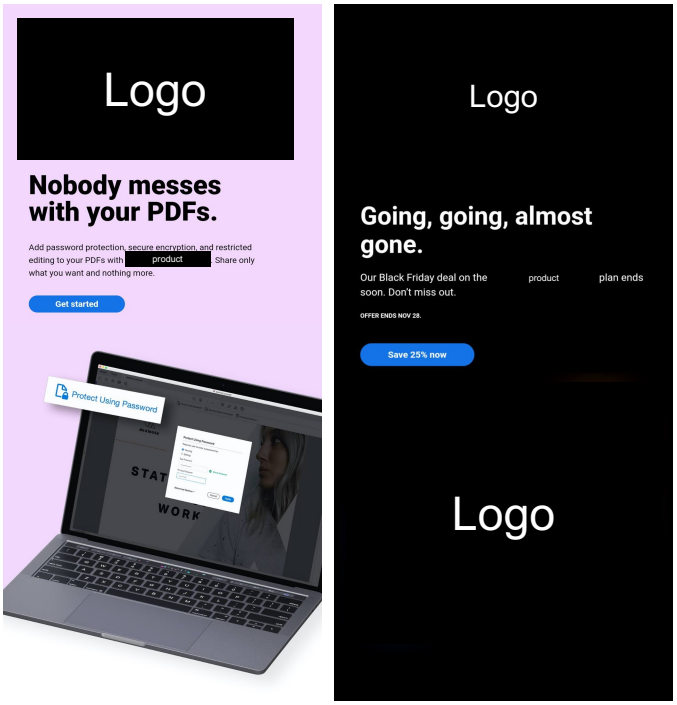


Figure 3.5: The in-house marketing emails used in the Email dataset look similar to the ones shown here.

Date Range	April 1, 2022 to June 12, 2023
Number of Countries	225
Target Products	Top Products used by millions of users
Customers Segmented on the basis of	Type of use, user expertise, frequency of use, and others

Figure 3.6: Details of the in-house Marketing Email dataset used to evaluate behavior generalization capabilities of the LCBM

percentage of likes/views) by 2.3% people.

3.1.3 Model

To understand both visual and textual contents, we follow a similar approach as was taken by recent models like BLIP, Llava, VideoLlama, and others [72, 163, 175, 176], we use visual encoders to encode visual knowledge and an LLM to encode text and world knowledge. Fig. 3.4 shows our architecture to encode visual content into the language space. We include video content by encoding video frames through EVA-CLIP [191] and Global Multi-Head Relation Aggregator (GMHRA) from Uniformer [193]. GMHRA helps aggregate the information better across the time dimension. The combination of ViT and GMHRA gives us a good representation of

Model	#Params	Training	Past		Future		Random Window Size				All Masked	
							5		7			
			RMSE	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE	Accuracy
LCBM	13B	3-BFT	8.12	55.10	15.05	42.42	8.55	61.41	9.91	55.10	-	-
LCBM		5-BFT	11.53	52.06	12.02	53.06	8.13	64.83	9.22	60.26	31.34	17.16
LCBM		7-BFT	16.17	35.61	15.14	44.11	9.02	59.22	10.47	53.84	-	-
LCBM		11-BFT	18.25	30.95	15.05	41.44	10.01	55.15	10.49	52.61	-	-
GPT-4	>100B†	10-shot-ICL	34.45	20.55	19.51	36.08	22.99	26.99	27.25	17.27	38.52	14.26
GPT-4		2-shot-ICL	35.05	19.34	18.07	39.33	17.42	38.10	21.26	28.05	37.60	13.73
GPT-3.5	175B	3-shot-ICL	34.10	19.06	24.71	27.14	24.52	24.81	26.30	18.74	38.77	13.47
GPT-3.5		2-shot-ICL	33.36	18.02	26.44	25.42	23.35	25.35	24.68	21.24	37.16	13.39
Random	-	-	34.10	10.00	34.10	10.00	34.10	10.00	34.10	10.00	34.10	10.00

Table 3.1: **Behavior Simulation.** Mean RMSE and accuracy scores for scene-by-scene predictions of video replay values. Replay values are the normalized replay scores of each scene as provided by YouTube. The normalized scores are considered to 2 decimal places and multiplied by hundred to convert the score to an integer score in the range 0-100. RMSE is calculated for each video in the test set and the mean is calculated for this score and reported. The model is said to classify correctly if the absolute error between the predicted and ground truth value is less than or equal to 5. The scores are calculated in four regimes: past, future, random, and all-masked. In the past (future) regimes, first (last) 5-20% scenes are masked; in the random setting, 5-20% scenes are masked randomly, and in all masked setting, everything is masked. LCBM was behavior-fine-tuned (BFT) with 3,5,7,11 context window masking strategy, while GPT was compared with an in-context learning (ICL) setting. We note that behavior fine-tuned LCBM, while being at least 10x smaller than other models, performs the best. Best models are denoted in green and runner-ups in blue.

the visual content. Next, to effectively leverage the LLM's rich language representations, we use Q-Former from BLIP-2 [163] with an extra linear layer and additional query tokens to convert from visual tokens to language tokens. Further, similar to [22], we find that while encoding visual tokens is powerful, converting visual content to text adds to the downstream performance. Therefore, we include the BLIP caption for each scene along with the scene replay graph.

We use the Llama-based Vicuna-13B LLM [68, 165] as our base LLM. Similar to prior works [72, 163, 175, 176], we follow a two-stage training paradigm where in the first stage, we utilize the WebVid [194], COCO caption [195], Visual Genome [196], CC3M [197], and CC12M [198] datasets to align the visual encoder embeddings with LLM. In the second stage, we train the model with behavior instructions prepared by following the approach described in §3.1.5. In summary, LCBM takes concatenated inputs of visual tokens, scene ASR, caption, scene behavior of replays, channel information, and video title and behavior metrics of views and a ratio of likes to views. Based on the instruction, we test LCBM's abilities on various tasks we cover in the next paragraphs.

3.1.4 Content Behavior Test Benchmark

We test the capabilities of large content-behavior models on predictive and descriptive abilities on content and behavior, as illustrated in Fig: 3.2. We design the following five tasks to test these capabilities: behavior simulation, content simulation, content understanding, behavior understanding, and behavior domain adaptation. We cover each of these tasks next.

1. **Behavior Simulation.** We test simulation capability on four behaviors across two datasets: YouTube replay values, the ratio of YouTube likes to views, Twitter likes, and the number of views of the YouTube video. The common task amongst all of them is to predict the behavior given the content and content attributes like captions, scene-by-scene descriptions for videos, and sender characteristics like account and subscriber count and date of posting. The behavior to be predicted is masked and asked as a question to the LLM. Listings 3.6 and 3.7 lists the verbalization pattern for this task. For replay value prediction, we test the masked behavior in three settings: *Masked Past* (all replay values of the first 5-20% scenes are masked), *Masked Future* (all replay values of last 5-20% scenes are masked), and *Random Masks* (random masking of replay values for 5-20% scenes).
2. **Content Simulation.** Here, the task is to predict content given receiver behavior (Listings 3.5, 3.8). For YouTube, given the video content in terms of scene-by-scene descriptions with the content of one group of five consecutive scenes content being masked, behavior values of all scenes, and channel information, the task is to choose the masked scene speech from a list of 25 options, chosen randomly from the entire test set. For YouTube, we chose to model this task as a discriminative task instead of a generative one since videos are generally long, and there could be multiple possible contents for a given behavior, whereas the ground truth is available only for one specific characterization of the content for a given behavior. For Twitter, we model this task as content generation. The Listing 3.8 presents the format for this task.
3. **Behavior Understanding.** The goal of this task is to check if the model can reason about observed or unobserved receiver behavior. For this task, we could ask the model to explain any behaviors given the content. However, only the YouTube receiver comments have ground truth available with the video. Without ground truth, we found that other behaviors, such as replay values, likes, and views, are difficult to explain by non-experts. Therefore, we ask the model to simulate the sentiment of the receivers' comments and describe its reasoning. To evaluate, we asked six annotators to annotate the reasons provided by the model on a scale of 0-5, with 0 implying the LLMs provided no sentiment or rea-

soning and 5 implying perfect reasoning. The annotators were free to rate the LLMs as they seemed fit. The annotators were asked to review the video content and the comments to help them evaluate the reasons. We average the ratings of three annotators to get an average rating for every video. Similarly, to review the sentiment correctness, we asked the annotators to judge the predicted sentiment rating with respect to user comments.

4. **Content Understanding.** To check if a model trained on both content and behavior tokens does not forget its original content understanding capabilities, we test the content understanding tasks on YouTube videos, following [22]. They use the following tasks for video-understanding: topic, emotion, persuasion, and action-reason classification. For topic, emotion, and action-reason classification tasks, they use the advertisements dataset by [82], which contains 3,477 video advertisements and the corresponding annotations for emotion and topic tags and action-reason statements for each video. There are a total of 38 topics and 30 unique emotion tags per video. Further, we have 5 action-reason statements for each video for the action-reason generation task. For our experiment, we use the subset of 1,785 public videos. Following [22], for the topic and emotion classification task, we evaluate our pipeline using top-1 accuracy as the evaluation metric. Further, we evaluate emotion classification on clubbed emotion labels as well. For action and reason prediction, we evaluate our accuracy on the action and reason retrieval tasks where 29 random options along with 1 ground truth are provided to the model to find which one is the ground truth. In the persuasion strategy classification, we use the 1002 persuasion strategy videos and corresponding labels released by [22]. Given the video, the model has to predict which persuasion strategy the video conveys. Persuasion strategy classification could be an important task for evaluating LCBM since the concept of persuasion in psychology views human communication as the means to change the receiver's beliefs and actions (*i.e.*, to persuade) [20], and understanding the different strategies present in communication may help understand human behavior better. We evaluate the top-1 accuracy of the model on this task.
5. **Behavior Domain Adaptation.** In the past work, we have observed strong generalization capabilities from LLMs [65, 69, 199]. While training on next token prediction, LLMs show generalization across tasks, including question answering, natural language inference, and sentiment analysis. Given this, the natural question is, does LCBM, too, show this kind of generalization, where a model trained on one kind of behavior, can show performance on another behavior? To understand this, we test the model on a different dataset and task than what it was originally trained for. We do this over three datasets, LVU [157], in-house Email Marketing[‖], and generalization between Twitter and YouTube likes.

Model	#Params	Training type	Training	RMSE	R^2	Accuracy
LCBM	13B	BFT	Replay values 3-masked	1.31	0.87	15.89
LCBM		BFT	Replay values 5-masked	1.48	0.82	19.93
LCBM		BFT	Replay values 7-masked	1.71	0.78	15.20
LCBM		BFT	Replay values 11-masked	1.55	0.82	13.94
GPT-4	>100B[†]	ICL	10-shot	3.50	-0.01	7.84
GPT-4		ICL	2-shot	3.58	-0.03	5.39
GPT-3.5	175B	ICL	3-shot	64.40	-256.96	2.48
GPT-3.5		ICL	2-shot	64.88	-375.83	1.27
Random	-	-	-	4.67	0	3.94

Table 3.2: **Behavior Simulation.** RMSE, R^2, and accuracy scores for like/view ratio prediction task. To calculate accuracy, the model is said to classify correctly if the absolute error between the predicted and ground truth likes/views is less than or equal to 10%. BFT denotes behavior fine-tuning, and ICL stands for in-context learning. Replay values k-masked means a model which is trained by masking k consecutive values of the replay graph while doing BFT. We note that LCBM while being at least 10x smaller than the other models, performs the best. The best results over four runs are reported for all models. Best models are denoted in green and runner-ups in blue.

- **LVU Benchmark.** Wu *et al.* [157] released a benchmark for long video understanding with over 1000 hours of video. In the benchmark, they have two behavior related tasks: ratio of likes to likes+dislikes and view prediction. YouTube has discontinued the dislike count, therefore, our corpus does not contain the dislike count. We use the LVU test benchmark to check if a model trained on other available behaviors (views, likes, and replay graphs) is able to predict the like ratio.
- **in-house Email Marketing.** In this task, we ask the model to predict the click-through rate for a given target segment of an email, given the email content, subject, and verbalized descriptions of the images in the email. We use the emails sent by in-house marketing team to its subscribers. The emails were sent from April 1, 2022 to June 12, 2023 and covered many of the premiere products The emails were sent to many customer segments (as defined by the marketing team) across 225 countries (Fig. 3.6). Listing 3.3 lists the verbalization format to verbalize emails to input to the LCBM.

3.1.5 Behavior Instruction Fine-Tuning (BFT)

To teach an LLM the behavior modality over multimodal content, we convert both the visual tokens and behavior modality in the text format and

[†]The exact size of GPT-4 is unknown.

instruction fine-tune the LLM end to end. This follows a two-stage approach: first, we teach the LLM the visual modality (§3.1.3), and next, we teach the LLM the behavior modality. We call the latter "Behavior Instruction Fine-Tuning (BFT)" inspired by instruction fine-tuning (IFT) and its variants like visual instruction tuning [72].

We prepare the content-behavior instruction datasets as explained next.

Teaching behavior in the forward direction (predict behavior given content): In this instruction tuning task, we teach the model to predict behavior given the message sent by the communicator. Essentially, this teaches the model to predict behavior in the forward direction (as in Fig. 3.1). Concretely, we include the following information as part of verbalization - image and video embedding converted to the text space (using EvaCLiP [191]), scene-by-scene verbalization covering automatic speech recognition, scene captions, video/post caption and description, receiver behavior covering replay rates, views, and likes, and communicator information covering account name and follower count. The verbalisation pattern for this task is the same as given in the Listing 3.6.

Teaching behavior in the reverse direction (predict content given behavior): This task teaches the model to learn about behavior in the reverse direction (Fig. 3.1). Here, the model learns to simulate content given behavior. The instruction for this task is given in Listing 3.4.

Using the prepared content and behavior instruction datasets consisting of pairs of content and behavior tokens, we treat the content tokens ($\mathbf{X}_C$) as input and behavior tokens ($\mathbf{X}_B, x_i \in \mathbf{X}_B$) as output of the language model. We then perform instruction-tuning of the LLM on the prediction tokens, using its original auto-regressive training objective. Specifically, for a sequence of length L, we compute the probability of generating target answers ($\mathbf{X}_B$) by:

$$p(\mathbf{X}_B|\mathbf{X}_C) = \prod_{i=1}^{L} p_\theta(x_i|\mathbf{X}_C, \mathbf{X}_{B,<i}) \tag{3.1}$$

For the behavior instruction tuning, we keep the visual encoder weights frozen and continue to update the pre-trained weights of the LLM in LCBM.

3.1.6 Results and Discussion

Here, we discuss the results for the five tasks we discuss in Section 3.1.4, namely, behavior simulation, content simulation, behavior understanding, content understanding, and behavior domain adaptation. We compare the behavior fine-tuned model discussed in §3.1.5 with state-of-the-art content-only models like GPT-3.5, GPT-4, and Vicuna-13B. This allows us to compare how much including behavior tokens in the training of an LLM helps

Model	#Params	Accuracy
Vicuna	13B	19.30%
LCBM	13B	48.68%
GPT-3.5	175B	34.98%
Random	-	4%

Table 3.3: **Content Simulation.** In this task, the models have to choose the speech segment from a list of 25 options given the video description, non-masked scenes. and replay behavior. We see that despite being similar to masked language modeling (which is a content-only task), LCBM performs better than both Vicuna and GPT-3.5. Best models are denoted in green and runner-ups in blue.

Model	#Params	Sentiment Accuracy	Reasoning Score
Vicuna	13B	65.66%	2.23
LCBM	13B	72.73%	4.00
GPT-3.5	175B	61.62%	1.67

Table 3.4: **Behavior Understanding.** In this task, the models have to simulate the sentiment of comments that a video would get by looking at only the video. Further, they also have to explain the reason for such sentiment. The responses were annotated by humans on a scale of 0-5 for the reason, with 0 being no response provided and 5 being the response matches exactly with the (ground truth) comments received on the video. Best models are denoted in green and runner-ups in blue.

Training	Model	#Params	Topic	Emotion		Persuasion	Action	Reason
			All labels	Clubbed				
Random	**Random**	-	2.63	3.37	14.3	8.37	3.34	3.34
Zero-shot	**GPT-3.5**	175B	51.6	11.68	79.69	35.02	66.27	59.59
	Vicuna	13B	11.75	10.5	68.13	26.59	20.72	21.00
	VideoChat [168]	13B	9.07	3.09	5.1	10.28	-	-
	LCBM	13B	42.17	7.08	58.83	32.83	39.55	27.91

Table 3.5: **Content Understanding.** Comparison of several models, including behavior instruction tuned models before and after BFT. We compare the models across topic, emotion, and persuasion strategy detection tasks as per the framework given by [22]. We see that our model outperforms similarly sized models (Vicuna, VideoChat) in most tasks. Best models are denoted in green and runner-ups in blue.

in improving the LLM's understanding of behavior and joint content and behavior spaces while retaining its understanding of the content space.

The results for the five tasks are presented in Tables 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, and 3.8. We note a few general trends. LCBM, while being 10x smaller than GPT-3.5 and 4, performs better than them on all behavior-related tasks. Further, we see that there is no significant difference between 10-shot and 2-shot GPT-4 or between GPT-3.5 and GPT-4, indicating that unlike other tasks, it is harder to achieve good performance through in-context-learning on the behavior modality. It can be observed that often GPT-3.5 and 4 achieve performance comparable to (or worse than) random baselines. Interestingly, the performance of GPTs on the content simulation task is also substantially behind LCBM. The way we formulate the content simulation task (Listing 3.5), it can be seen that a substantial performance could be achieved by strong content knowledge, and behavior brings in little variance. We still see a substantial performance gap between the two models. All of this indicates that large models like GPT-3.5 and 4 are not trained on behavior tokens.

For the content understanding tasks (Table 3.5), predictably GPT-3.5, being the largest model, achieves the best results. However, we see that BFT helps the LLM to learn most content understanding tasks better than the base LLM. LCBM gets better results than both Vicuna and VideoChat. This indicates that behavior modality might carry additional information about the content, which might help an LLM understand content better [8, 36, 200]. Next, we see that LCBM also shows signs of domain adaptation in the behavior modality. We see that on five tasks: comment sentiment prediction, comment sentiment reasoning (Table 3.4), email behavior simulation (Table 3.6), and Twitter behavior (Table 3.7) and content simulation (Table 3.8). We note that if the LCBM is trained on only email behavior simulation samples, it underperforms the model trained on both YouTube data and a few samples to make the model learn email format. Similarly, LCBM trained on both Twitter and YouTube performs better than the one just trained on Twitter, showing performance improvement by domain adaptation. Finally, Figs. 3.7,3.8 show a few samples where we query LCBM to explain replay and comment behavior and compare it with human explanations. We see that LCBM while verbose, can explain behavior well.

[‡]Brand Separated means that the train and test set don't have any overlap in terms of brands, Time Separated means that the test set starts after the last tweet in the train set. BFT denotes behavior fine-tuning, and ICL stands for in-context learning. The best results over four runs are reported for all models. Best models are denoted in green and runner-ups in blue.

in-house Email Marketing							
LCBM Type	**Fine-tuned on YouTube?**	**Trained On**			**Tested On**	**RMSE**	**R^2**
		Unique Emails	**Unique Segments**	**Email-Segment Pairs**			
Domain-Adapted	Yes	100	10	1k	Different Segment (emails could be same)	14.47	0.64
In-Domain	No	600	560k	350k		25.28	0.55
Domain-Adapted	Yes	100	10	1k	Different Segments & Different Emails	27.28	0.54
In-Domain	No	600	560k	350k		29.28	0.5

LVU Benchmark			
Training	**Model**	**Testing**	**MSE**
Trained	R101-slowfast+NL [157]	Test set	0.386
Trained	VideoBERT [170]	Test set	0.32
Trained	[172]	Test set	0.353
Trained	[171]	Test set	0.444
Trained	Object Transformers [157]	Test set	0.23
Zero-shot	LCBM (Ours)	Test set	0.14
Zero-shot	GPT-3.5	Test set	0.03
Zero-shot	Vicuna	Complete dataset	0.44
Zero-shot	LCBM (Ours)	Complete dataset	0.30
Zero-shot	GPT-3.5	Complete dataset	0.02

Table 3.6: **Behavior Domain Adaptation.** We test the generalization capability of LCBM on two tasks: (1) Behavior simulation on in-house Email Marketing Data, (2) Behavior simulation on the LVU benchmark. For (1), we train two versions of LCBM with the in-house Email Marketing data: one was trained on YouTube videos and further BFT on a few email samples (*domain-adapted*), and the other was BFT on a larger set of emails, but not including YouTube data (*in-domain*)[§]. We report the RMSE and R^2 scores for this task. For (2), we compare LCBM with other state-of-the-art results and GPT-3. In (1), we note that the domain-adapted LCBM performs better than the in-domain LCBM in both settings. We posit that YouTube data helps LCBM understand how a company's viewers like to hear from it, giving LCBM an edge over a model trained on a small amount of the same data (600 unique emails). In (2), LCBM performs better than the existing state-of-the-art. Surprisingly, GPT-3.5 does better than LCBM on this task. From both (1) and (2), we gather that a model trained on certain YouTube behaviors performs better on other behaviors, thus showing promise of domain-adaptation in the behavior modality. Best models are denoted in green and runner-ups in blue.

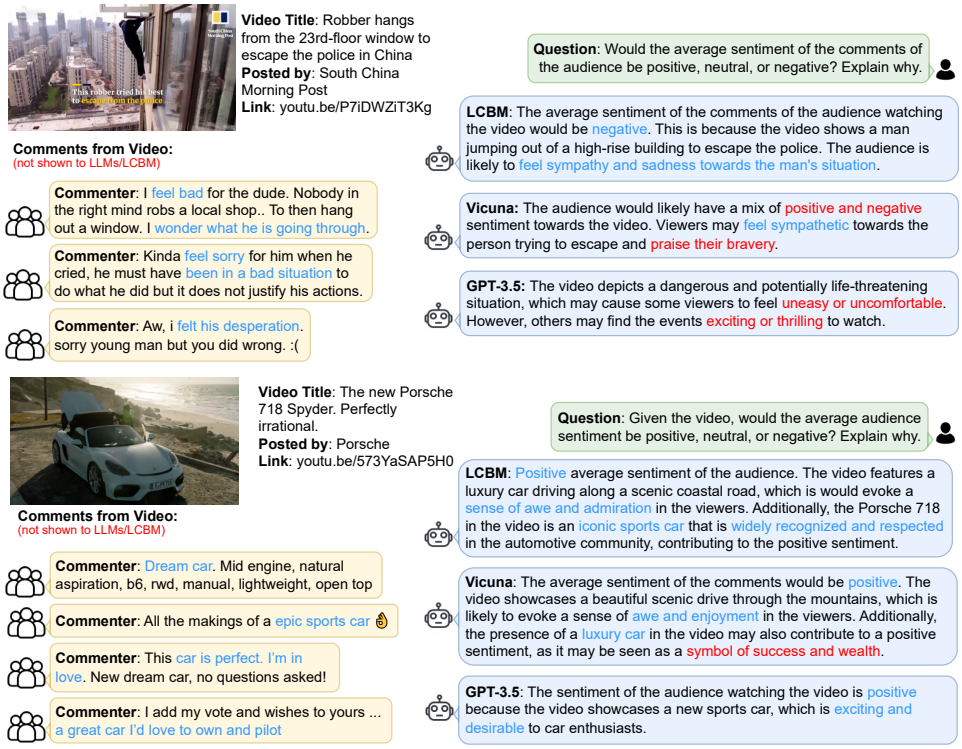


Figure 3.7: A few examples showing LCBM's ability to understand and explain human behavior of audience sentiment. We also compare it against other models like Vicuna and GPT-3.5.

3.1.7 Related Work

Models of Human Communication: Communication is the situation in which a source transmits a message to a receiver with conscious intent to affect the latter's behaviors [201, 202]. Thus, in the most general terms, communication implies a sender, a channel, a message, a receiver, a relationship between sender and receiver, an effect, a context in which communication occurs and a range of things to which 'messages' refer [47, 203]. As per this, all of the content produced by humanity is essentially communication from a sender to a receiver over some channel and with some effect. Despite much research on communication in social sciences since the 1900s, there has been little adoption of it in machine learning modeling. A prime artefact of this is that the biggest models in machine learning (LLMs) are trained only on content (messages) and ignore other factors in communication (the intended receiver, channel, and behavior) even when they are available.

Prior Efforts To Model Behavior: While there has been much research in ML to model human behavior, it has been disconnected from language and, sometimes, real-world data. For instance, Agent-based modeling

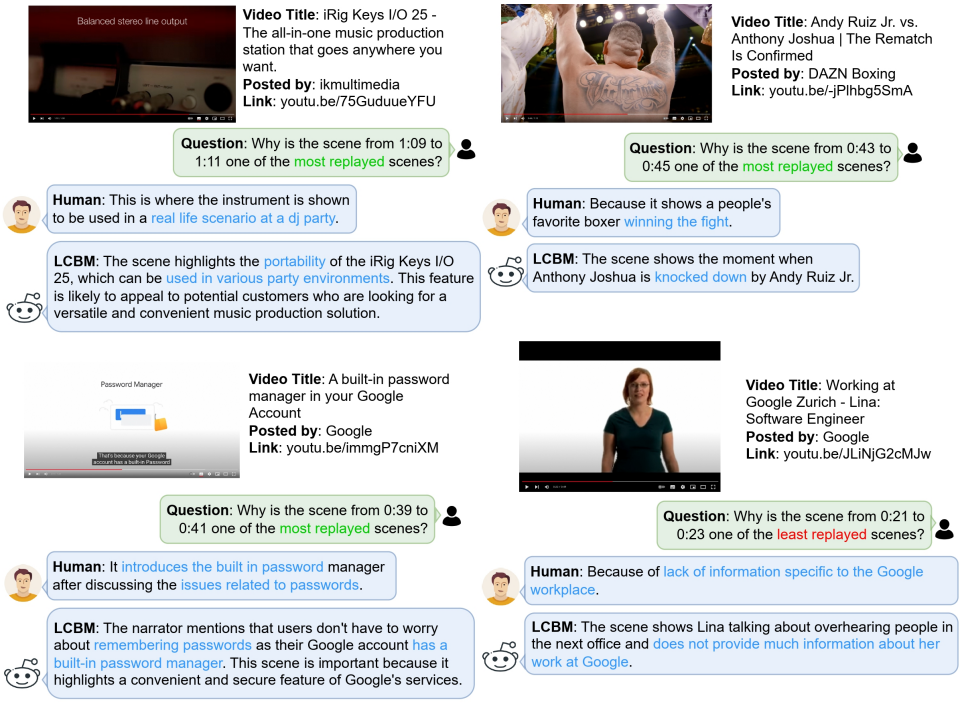


Figure 3.8: A few examples showing LCBM's ability to understand and explain human behavior of scene replayability. We compare it against human-provided explanations of the same.

(ABMs), a popular paradigm in Reinforcement Learning, has been employed to model behavior [204–206]. Nevertheless, ABMs tend to view humans as rational economic agents who communicate primarily through their actions, neglecting the significance of content in communication. In ABMs, agents strive to maximize their rewards, whereas communication does not always aim to optimize specific, well-defined reward signals. Moreover, the scarcity of large repositories containing extensive records of human actions poses a challenge when training ABMs to learn human behavior. Consequently, existing large models trained on human behavior, such as the ABMs and decision transformer and its variants, often rely on simulated data, such as game environments, rather than real human behavior [207]. This reliance on artificially generated data introduces biases inherent to the creators of the training data, making it difficult to capture authentic human behavior. However, recent advancements have demonstrated the potential of large models trained on real-world tokens encompassing various modalities, like images, videos, audio, and text, as the basis for diverse tasks [163, 176]. Notably, LLMs, as exemplars of foundation models, have exhibited impressive performance across a range of tasks, including those they were not explicitly trained for, such as emotion recognition, named entity recognition, and

Model	#Params	Training type	Training	Time Separated	Brand Separated
GPT-3.5	175B	ICL	Few-shot	58.84	64.19
LCBM	13B	BFT	Twitter	74.3	97.69
LCBM	13B	BFT	Twitter and YouTube data	76.87	92.19

Table 3.7: **Behavior Simulation and Behavior Domain Adaptation**[‡]. Two-way classification accuracies for like prediction on Twitter. Given content, channel, and time, predict behavior (High, Low). We note that LCBM trained on Twitter and YouTube performs better than the one trained only on Twitter, showing signs of performance improvement by domain adaptation.

Model	Training	Test	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-l
GPT-3.5	ICL	Brand Separated	53.95	42.36	31.84	24.28	15.24
		Time Separated	57.69	45.11	33.67	25.52	15.27
LCBM	BFT on Twitter	Brand Separated	62.29	46.59	33.98	25.64	14.44
		Time Separated	70	54.4	41.43	32.48	17.38
LCBM	BFT on Twitter + Youtube	Brand Separated	64.28	48.1	35.17	26.63	14.83
		Time Separated	70.23	54.54	41.52	32.54	17.45

Table 3.8: **Content Simulation and Behavior Domain Adaptation**[‡]. Given behavior, channel, time, tweet media caption as prompt, predict content (tweet text). We note that LCBM trained on Twitter and YouTube performs better than the one trained only on Twitter, showing signs of performance improvement by domain adaptation.

complex tasks like table understanding [22, 208].

Further, there has also been much work in modeling behavior using conventional modeling techniques, such as regression, bagging and boosting [209, 210], neural networks [211–213], and transformers [157, 171]. While these models can certainly model behavior, LLMs show generalization powers which extend to capabilities much beyond just behavior simulation. For instance, once trained on behavior tokens, other than behavior simulation, LLMs can now generate behavior optimized content (Table 3.3), explain behavior (Table 3.4), and domain-adapt to other behaviors (Table 3.6), none of which are shown by other models. The other concurrent works which model behavior using LLMs [214] model just behavior (for example, by CTR prediction) by attaching classification or regression heads to LLMs and thereby lose out on the text-to-text paradigm where LLMs show their best performance and generalization capabilities. In addition, similar to non LLM paradigm, this method loses out on other capabilities like generating behavior optimized content and explaining behavior.

3.1.8 Verbalization Patterns

Listing 3.2: Verbalization pattern of videos for the behavior understanding task:

Input: <video> .. </video>
The video has the following scenes:
Scene 1: {ASR: Welcome to a quick tutorial, OCR: Adobe Premiere Pro, Captions: A desktop interface, Replays: 60},
Scene 2: {ASR: on using Premiere Pro to edit, Captions: A computer interface, with an image of a white horse. Objects — Horse, Grass, Fence., Replays: 53},
...
It was posted by Adobe with the title 'Using Premiere Pro like a Pro' on Aug 15 2022. Adobe has 100k subscribers. This video was viewed by 346 thousand people and liked (as a percentage of likes/views) by 2.3% people. Why is the scene 23 one of the most replayed scenes?

Output: The scene shows the transformation of the image after the changes.

Listing 3.3: Verbalization pattern of emails for the behavior domain adaptation task:

Input: Email with Subject: Lock it down before you send it out.
Header: Nobody messes with your PDFs.
Body text: Add password protection, secure encryption, and restricted editing to your PDFs with Adobe Acrobat Pro DC. Share only what you want and nothing more. A button that says 'Get started'. An image of a laptop, with window open on it. Image text: "Protect using password".
Foreground colors: grey, blue. Background colors: lavender, white. Image Emotions: security, serious. Image keywords: laptop, protect, password, lock. Aesthetic value: low. Clutter level : medium. The email is created by a Creative Professional, for the product Adobe Acrobat Pro. It is sent to users in the United States, in the commercial market. Specifically, it is sent to Power users with the intent of Active Use.
The email was sent 109 times between 25 August, 2022 and 26 August, 2022, and had a click through rate of [MASK]%.

Output: 0.037%.

Listing 3.4: Verbalization pattern to teach behavior in the reverse direction (predicting content given behavior):

Input: <video> .. </video> The video has the following scenes: Scene 1: {ASR: [MASK], Replays: 60%}, Scene 2: {ASR: with Premiere, Captions: Woman looking at screen, Replays: 34%},
...
Scene 5: {ASR: has never been, Captions: Colour Pallete, Replays: 47%},
Scene 6: {ASR: been easier, Captions: Colour Pallete, Replays: 54%},
...
It was posted by Adobe with the title 'Using Premiere Pro like a Pro' on Aug 15 2022. It is viewed 203k times and liked 1.2%. Adobe has 100k subscribers. Predict the masked ASR value for the masked scenes.

Output: Scene 1:{ASR: Welcome to a quick tutorial.}

Listing 3.5: Verbalization pattern of videos for the content simulation task:

Input: <video> .. </video> The video has the following scenes: Scene 1: {ASR: [MASK], Replays: 60%}, Scene 2: {ASR: with Premiere, Captions: Woman looking at screen, Replays: 34%},
...
Scene 5: {ASR: has never been, Captions: Colour Pallete, Replays: 47%},
Scene 6: {ASR: been easier, Captions: Colour Pallete, Replays: 54%},
...

It was posted by Adobe with the title 'Using Premiere Pro like a Pro' on Aug 15 2022. It is viewed 203k times and liked 1.2%. Adobe has 100k subscribers. Predict the masked ASR value for scene 1. Choose from the given options.
Option−1: Welcome to a quick tutorial,
Option−2: Samsung Galaxy A20 smartphone,
...
Option−25: regulations. We haven't had.

Listing 3.6: Verbalization pattern of videos for the behavior simulation task:

Input: <video> .. </video> The video has the following scenes:
Scene 1: {ASR: Welcome to a quick tutorial, OCR: Adobe Premiere Pro, Captions: A desktop interface, Replays: [MASK]},
Scene 2: {ASR: on using Premiere Pro to edit, Captions: A computer interface, with an image of a white horse. Objects − Horse, Grass, Fence., Replays: [MASK] }, ...
It was posted by Adobe with the title 'Using Premiere Pro like a Pro' on Aug 15 2022. Adobe has 100k subscribers. Can you tell the replay values for scenes 2 to 5. How many times will this video be viewed and liked as a percentage of likes/views?

Output: Scene 1: {Replay: 60%}, Scene 2: {Replay: 85%}, ..., Views: 2.1 Million, Likes−per−View: 2.5%

Listing 3.7: Verbalization pattern of Twitter posts for the behavior simulation task:

Input: Given a tweet of pfizer posted by the account PfizerMed on 2023−01−12. Tweet : Announcing a new ASGCT−Pfizer grant to support independent medical education initiatives on genetic medicines. For details, click Request for Proposals. <hyperlink>. Apply by January 30, 2022 #raredisease #ASGCT #GeneTherapy <hyperlink>. Verbalisation of media content: \"caption\": \"A close−up of a DNA double helix, showcasing its structure and blue color\",\"keywords\": \"DNA, double helix, structure, blue, close−up, molecular biology, genetics, biology, scientific illustration\"}. Predict whether it will recieve high or low likes?",

Output: This tweet has low likes.

Listing 3.8: Verbalization pattern of Twitter posts for the content simulation task:

Input: Generate a tweet given the media verbalization and the likes it got. Tweet is for pfizer to be posted by the account PfizerMed on 2023−01−12. Verbalisation of media content: \" caption\": \"A close−up of a DNA double helix, showcasing its structure and blue color \",\"keywords\": \"DNA, double helix, structure, blue, close−up, molecular biology, genetics, biology, scientific illustration\"}. This tweet has low likes."

Output: "Tweet : Announcing a new ASGCT−Pfizer grant to support independent medical education initiatives on genetic medicines. For details, click Request for Proposals. < hyperlink>. Apply by January 30, 2022 #raredisease #ASGCT #GeneTherapy < hyperlink>"}

3.1.9 Conclusion

In this work, we make initial strides towards solving the effectiveness problem proposed by Shannon in his seminal paper on communication. The effectiveness problem deals with predicting and optimizing communication to get the desired receiver behavior. This can be seen as consisting of a string of capabilities: behavior simulation, content simulation, and behavior domain adaptation. We show that while large language models have great generalization capabilities, are unable to perform well on the effective-

ness problem. We posit that the reason for this could be a lack of "behavior tokens" in their training corpora. Next, we train LLMs on behavior tokens to show that other than content understanding tasks, the trained models are now able to have good performance across all the behavior-related tasks as well. We also introduce a new Content Behavior Corpus (CBC) to spur research on these large content and behavior models (LCBMs).

3.2 Encoding Behavior To Improve Content Understanding

In the last section, we discussed training a single model which learns about both content and behavior. We saw that a model trained on content and behavior together shows capabilities of behavior and content simulation, behavior domain adaptation, and improvements in behavior and content understanding. Behavior, as an artifact of communication, is generated by a receiver in response to content (Fig. 3.1) sent by a communicator. Therefore, it comes later than content in the time axis. Hence, behavior contains signals about content, which can help in *understanding* content. However, since it comes after content, the signals are available post-hoc. Therefore, in this section, we talk about how behavior can be used to improve content understanding in a post-hoc manner. Next, we solve the problem of behavior being available post-hoc by generating synthetic behavior for a content, and showing that using the synthetic behavior also improves understanding content. We show this using the cognitive behavior of scanpaths. The choice of scanpaths as the target behavior is motivated by prior literature [178–181, 184, 186, 215, 216] where they show that eye movements of the receiver can help determine linguistic and perceptual factors in text and images.

3.2.1 Introduction

Integrating human signals with deep learning models has been beginning to catch up in the last few years. Digital traces of human cognitive processing can provide valuable signals for Natural Language Processing [37, 217]. Various approaches for integrating human signals have been explored. For example, human feedback for better decisioning [218], NLP tasks [39, 219], and most recently language modeling using reinforcement learning with human feedback (RLHF) based reward [199, 220]. RLHF involves explicit human feedback and is expensive and hard to scale. On the other hand, previous studies have also tried to use implicit human feedback in the form of eyetracking signals. It has proven to be a useful signal for inferring human cognitive processing [188, 221, 222]. NLP researchers have focused on

[§]Note that we cannot compare this model with GPT-3 due to the private nature of data.

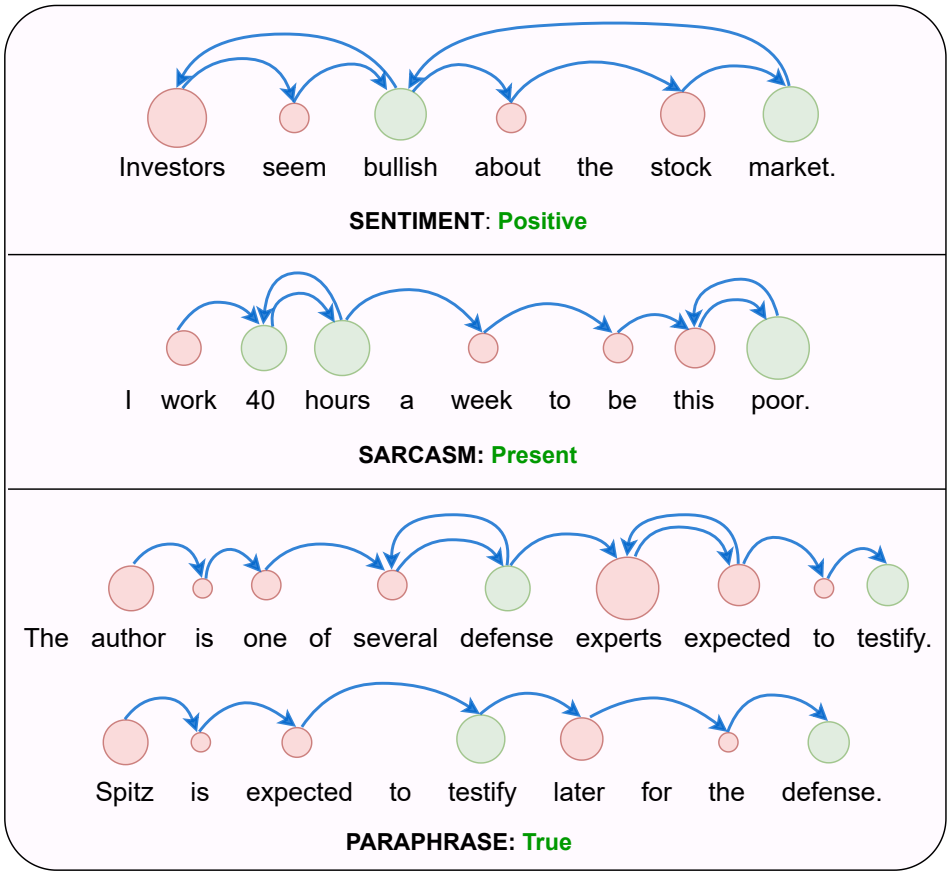


Figure 3.9: Generated scanpaths over text samples taken from various natural language processing (NLP) tasks. The green circles denote the important words characteristic of that task. The circles' size denotes the fixation duration, and the arrows depict the saccadic movements. As can be seen, linguistically important words often have a higher fixation duration and revisit. Regressions (word revisits) also appear in the examples.

assessing the value of gaze information extracted from large, mostly disjointly labeled gaze datasets in recurrent neural network models [223–225]. The proposed approaches under this paradigm include gaze as an auxiliary task in multi-task learning [200, 226], as additional signals [186], as word embeddings [227], as type dictionaries [188, 228], and as attention [225].

Previous studies demonstrate that human scanpaths (temporal sequences of eye fixations, see Fig. 3.9) gleaned from eye tracking data improve the performance of NLP models. However, the real-world application of these methods remains limited primarily due to the cost of precise eye-tracking equipment, users' privacy concerns, and manual labor associated with such a setup. Therefore, generating scanpaths from existing eyetracking corpora would add great value to NLP research. To the best of our knowledge, this

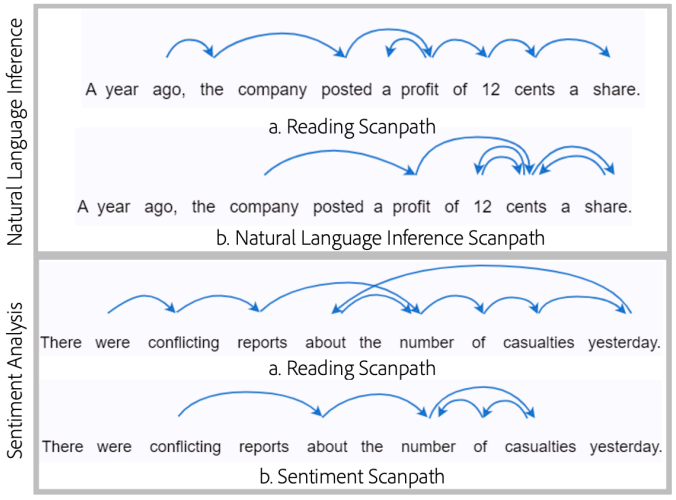


Figure 3.10: (Intent-aware) Scanpath samples generated by conditioning scanpath generation on different downstream natural language tasks. Note that the conditioned scanpaths are heavily biased to words important for that downstream task.

is the first work to propose a model that generates scanpaths for a given read text with good accuracy. We call the model, ScanTextGAN.

We demonstrate the scanpath generation capability of ScanTextGAN over three eye-tracking datasets using multiple evaluation metrics. Further, we evaluate the utility of *generated* scanpaths for improvements in the performance of multiple NLP tasks (see Figs. 3.9,3.10) including the ones in the GLUE benchmark [229]. The generated scanpaths achieve similar performance gains as the models trained with real scanpaths for classic NLP tasks like sentiment classification, paraphrase detection, entailment, and sarcasm detection.

Our contributions are threefold:

1. We propose ScanTextGAN, the first scanpath generator over text.

2. We compare ScanTextGAN with multiple baselines and conduct ablation experiments with varying models and configurations. The model performs well on the test sets and cross-domain generalization on two additional eye-tracking datasets belonging to different text domains.

3. We tested the usefulness of generated scanpaths in downstream NLP tasks such as sentiment analysis, paraphrase detection, and sarcasm detection on six different datasets. The results show that the downstream NLP tasks benefited significantly from cognitive signals inherent in generated scanpaths. Further, we show how scanpaths change when finetuning with downstream natural language tasks (Figs.3.10,3.14) and that they lead to further improvements in downstream task performance (§3.2.4.3) showing how they can act as additional controls beyond the task architecture.

3.2.2 Related Work

When reading a text, humans do not focus on every word and often do not read sequentially [183]. A series of studies in psycho-linguistics have shown that the number of fixations and the fixation duration on a word depend on several linguistic factors. The linguistic factors can also be determined given the cognitive features [215, 216]. Though advances in ML architecture have helped bring machine comprehension closer to human performance, humans are still superior for most NLP tasks [230, 231].

It has been shown in the literature that integrating explicit [199, 220] and implicit (cognitive processing) human feedback signals in traditional ML models is expected to improve their performance [183]. However, the cost of explicit feedback (e.g., using MTurk) and implicit feedback (e.g., eye tracking) at scale is excessively high. Similarly, privacy-invasive eye-tracking processes limit the scope of this idea. One way to address this problem is to use generated eye movements to unfold the full potential of eye-tracking research. Hence, the idea is to architect ScanTextGAN, a scanpath generator for text reading, and test its usefulness in downstream NLP tasks.

More precisely, this work builds upon previous works on 1) human attention modeling and 2) gaze integration in neural network architectures, which are described as follows:

Human Attention Modeling: Predicting what people visually attend to in images (saliency prediction) is a long-standing challenge in neuroscience and computer vision, the fields have seen many data-based models [232]. In contrast to images, most attention models for eye movement behaviors during reading are cognitive process models, *i.e.*, models that do not involve machine learning but implement cognitive theories [231, 233]. Key challenges for such models are a limited number of parameters and hand-crafted rules. Thus, it is difficult to adapt them to different tasks and domains and use them as part of end-to-end trained ML architectures [234]. In contrast, learning-based attention models for text remain under-explored. Within that, all eye tracking models are saliency prediction models with non-existent work in predicting scanpaths. On the other hand, visual scanpaths generation for image-based eye tracking data has been recently explored for both traditional [235] and 360° images [236].

Matthies *et al.* [237] presented the first fixation prediction work for text. They built a person-independent model using a linear Conditional Random Fields (CRF) model. Hahn and Keller [238] designed the Neural Attention Trade-off (NEAT) language model, which was trained with hard attention and assigned a cost to each fixation. Other approaches include sentence representation learning using surprisal and part of speech tags as proxies to human attention [239].

Our work differs from previous studies as we combine cognitive theory and data-driven approaches to predict scanpaths and further show its ap-

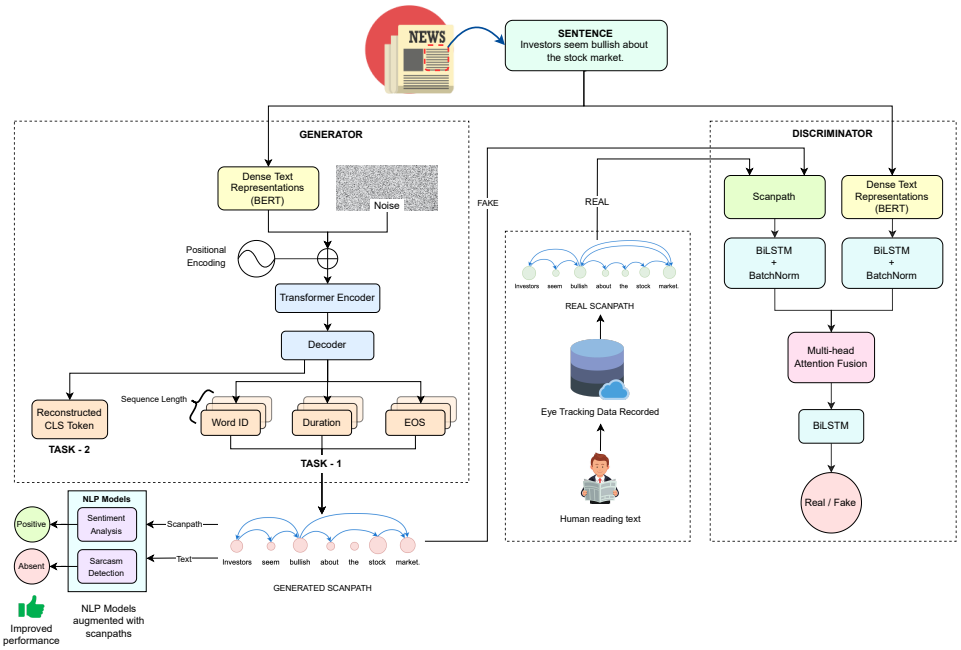


Figure 3.11: The architecture of the proposed **ScanTextGAN** model. The model consists of a conditional generator and a discriminator playing a zero-sum game. The generator is trained by two cognitively inspired losses: text content reconstruction and scanpath content reconstruction.

plication in downstream NLP tasks [240, 241].

Integrating Gaze in Network Architecture: Integration of human gaze data into neural network architectures has been explored for a range of computer vision tasks such as image captioning, visual question answering, and tagging [178–181]. Hence, recent research has utilized features gleaned from readers' eye movement to improve the performance of complex NLP tasks such as sentiment analysis [184, 185], sarcasm detection [186], part-of-speech tagging [187], NER [188], and text difficulty [189].

While in recent years, eye tracking data has been used to improve and evaluate NLP models, the scope of related studies remains limited due to the requirement of real-time gaze data at inference time. Mathias *et al.* [222] reported that there exists no automated way of generating scanpaths yet in the literature. With high-quality artificially generated scanpaths, the potential of leveraging eyetracking data for NLP can be unfolded. Additionally, generating scanpaths that mimic human reading behavior will help advance our understanding of the cognitive processes behind language understanding. Hence, we propose ScanTextGAN; researchers can use that to generate scanpaths over any text without worrying about collecting them from real users.

3.2.3 Proposed Model

In this section, we define the scanpath generation task, describe the ScanTextGAN model architecture, and provide details on loss functions and model training.

Task Definition: The task of scanpath generation is to generate a sequence $\mathcal{S}(\mathcal{T})$ representing a scanpath over the text $\mathcal{T} = \{w_1, w_2, ..., w_n\}$ composed of a sequence of words, can be defined as follows:

$$\mathcal{S}(\mathcal{T}) = \{.., (w_a^i, t^i), ..., (w_b^j, t^j), ..., (w_c^k, t^k)\} \tag{3.2}$$

where t^i represents the fixation duration over the word w_a occurring at the position i. Note that it is not necessary to have $a < b$ (words being read in linear order) or that $k = n$ (the number of fixations being equal to the number of words). Due to regressions, *i.e.*, backward saccades to previous words, words are also revisited. Hence, the same word could appear multiple times in the sequence.

3.2.3.1 ScanTextGAN Model Architecture

Fig. 3.11 illustrates the proposed conditional GAN architecture of the model. The ScanTextGAN model is composed of two competing agents. First, a conditional generator that generates scanpaths given text prompts. The second is a discriminator network, which distinguishes real human scanpaths from the generated ones. The ScanTextGAN model is trained by combining text content loss, scanpath content loss, and adversarial loss (Eq. 3.7). The scanpath content loss measures the difference between the predicted scanpath and the corresponding ground truth scanpath. The text content loss reconstructs the input text, and the adversarial loss depends on the real/synthetic prediction of the discriminator over the generated scanpath. We describe the losses along with the generator and discriminator architectures next.

Generator: The ScanTextGAN generator constitutes a transformer-based encoder-decoder framework. The encoder is conditioned on BERT-based text embeddings [62], which are concatenated with noise to make the generator's output non-deterministic. The output of the transformer encoder is supplied to the decoder, which consists of task-specific feed-forward networks. One branch generates the scanpath (*Task 1*), while the other reconstructs the 768 dimensional CLS token embedding of the sentence (*Task 2*). The scanpath is output as a temporal sequence of word ID (fixation points) w_a^i, fixation duration t^i, and end-of-sequence probability EOS^i. At inference time, the length $L(G)$ of generated scanpath G is determined as follows:

$$L(G) = \begin{cases} \min_{1 \leq k \leq M}(k) & \text{if } EOS^k > \tau \\ M & \text{otherwise} \end{cases} \tag{3.3}$$

where M is the maximum scanpath length as described in section §3.2.3.2 and $\tau \in (0, 1)$ is a probability threshold. We use $\tau = 0.5$. The loss functions of the two branches are described below.

Scanpath Content Loss tries to minimize the deviation of generated scanpaths $\mathcal{G}(\mathcal{T}, \mathcal{N})$ from the ground-truth scanpaths $\mathcal{R}(\mathcal{T}, h))$ over text $\mathcal{T}$ where ground-truth scanpaths are recorded from the human h and $\mathcal{N}$ stands for Gaussian noise $\mathcal{N}(0, 1)$. The loss function $\mathbb{L}_s$ is given as:

$$\begin{aligned}\mathbb{L}_s(\mathcal{G}(\mathcal{T},\mathcal{N}), \mathcal{R}(\mathcal{T}, h)) = \frac{1}{k}\Sigma_{i=0}^{k}(\alpha(id_g^i - id_r^i)^2 + \\ \beta(t_g^i - t_r^i)^2 + \gamma(E_g^i - E_r^i)^2)\end{aligned} \tag{3.4}$$

which is a weighted sum of three terms. The first term measures the error between real and predicted *fixation points* given by the mean squared difference between generated and real word-ids $(id_g^i - id_r^i)$. It penalizes permutations of word ids and trains the model to approximate the real sequence of fixation points closely.

The second term measures the difference in *fixation durations* given by the mean squared difference between generated and real duration $(t_g^i - t_r^i)$. Fixation durations simulate human attention over words in the input text. Thus, a word with a larger fixation duration is typically synonymous with greater importance than other words in the input text. This error term supplements the generator's ability to learn human attention patterns over the input text.

Finally, the third term measures the mean squared error between the prediction of end-of-sequence probability by real and generated distributions $(E_g^i - E_r^i)$. These are weighted by the hyperparameters α, β, and γ. Preliminary experiments showed that optimizing the mean squared error leads to better performance over the cross-entropy loss for optimizing the EOS probability output.

Text Content Loss: Scanpaths depend heavily on the linguistic properties of the input text. Therefore, to guide the generator towards near the probable real data manifolds, we adopt reconstruction of the CLS token embedding of the input text (*Task 2*) by the generator as an auxiliary task since the CLS token embedding encodes a global representation of the input text. This text content reconstruction loss $\mathbb{L}_r$ is given as:

$$\begin{aligned}\mathbb{L}_r(\mathcal{G}(\mathcal{T},\mathcal{N}), \mathcal{R}(\mathcal{T}, h)) = (BERT(w_i^g, w_j^g, ..., w_k^g \\ - BERT(w_a^r, w_b^r, ...w_n^r))^2\end{aligned} \tag{3.5}$$

where $BERT(w_a^r, w_b^r, ...w_n^r)$ and $BERT(w_i^g, w_j^g, ...w_k^g)$ stand for the *CLS* vector representations of real and generated text respectively.

Discriminator: The goal of the discriminator is to distinguish between the real and synthetic scanpaths supplied to it. Similar to the generator, it requires text representations to distinguish between real and generated

scanpaths. Specifically, the discriminator comprises two blocks of BiLSTMs that perform sequential modeling over the scanpaths and BERT embeddings. The outputs of the two branches are combined and passed to an attention fusion module with four heads, followed by another network of BiLSTMs. The hidden states of the last BiLSTM layer from both forward and backward directions are concatenated and supplied to a feed-forward network. A Sigmoid function activates the output of the feed-forward network. In this manner, the discriminator classifies the input scanpaths as either *real* or *fake*.

Adversarial Loss: The generator and discriminator networks are trained in a two-player zero-sum game fashion. The loss is given by:

$$\begin{aligned}\mathbb{L}_a = \min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x|\mathcal{T}, h)] + \\ \mathbb{E}_{z\sim p_z(z)}[1 - \log D(G(z|\mathcal{T}, \mathcal{N}))]\end{aligned} \tag{3.6}$$

Therefore, the net generator loss becomes:

$$\mathbb{L}_g = \mathbb{L}_s + \mathbb{L}_r + \mathbb{E}_{z\sim p_z(z)}[1 - \log D(G(z|\mathcal{T}, \mathcal{N}))] \tag{3.7}$$

3.2.3.2 Dataset

For training the ScanTextGAN model, we use the CELER dataset [242]. It contains eyetracking data of 365 participants for nearly 28.5 thousand newswire sentences, sourced from the Wall Street Journal Penn Treebank [243]. Each participant in CELER reads 156 newswire sentences. Half of the sentences are shared across participants, and the rest is unique to each participant. The maximum sentence length was set to 100 characters. Participant eyetracking data were recorded using Eyelink 1000 tracker in a desktop mount configuration with a sampling rate of 1000 Hz. The ScanTextGAN model is trained to approximate the average eye movements of all the participants who read given sentences. The CELER dataset was envisioned to enable research on language processing and acquisition and to facilitate interactions between psycholinguistics and natural language processing. Furthering the goal, we use it to train our conditional GAN model through which we show human scanpath approximation capabilities (§3.2.4.2). Also, we use it to show improvements in the performance of NLP tasks (§3.2.4.3).

The data consist of tuples of participant ID, sentence ID, and word ID corresponding to fixation point and fixation duration. We compute the 99th percentile of fixation durations and treat it as the largest value. Fixations of durations longer than this are treated as outliers and hence dropped from the dataset. To apply the scanpath reconstruction loss (Eq. 3.4), we scale all fixation durations by the maximum value and then normalize them to [0,1]. Similarly, word IDs in each sentence are normalized to [0, 1] after scaling them by the length of that sentence. For the last fixation point in every scanpath, the binary EOS token is set to 1. The maximum scanpath length is set

to 80 fixation points (99th percentile of the lengths). Thus shorter scanpaths are padded while longer scanpaths are trimmed. We use BERT to encode the sentences and obtain their 768-dimensional embeddings, keeping the max length parameter as 80, thus resulting in an 80×768 dimensional tensor.

3.2.3.3 Parameter Settings

Sinusoidal positional encoding is applied over the input embeddings fed to the generator. We use a 3-layer transformer encoder with four head attention and a hidden dimension size of 776 in the generator. In the discriminator, we use bidirectional LSTMs over sentence embeddings and generated scanpaths with a hidden size of 64 and a dropout ratio of 0.3, followed by batch normalization for faster convergence. An attention module with four attention heads is applied after concatenating the outputs. We employ the Adam and RMSProp optimizer to minimize generator and discriminator losses. The batch size is set to 128, the initial learning rate of the generator to 0.0001, and that of the discriminator to 0.00001. The model is trained for 300 epochs. Our implementation uses PyTorch, a popular deep-learning framework in Python. All experiments are run on an Intel Xeon CPU with Nvidia A100-SXM GPUs.

3.2.4 Performance Evaluation

We quantify the performance of ScanTextGAN in two regimes[*]; first, scanpath generation with three datasets, and second, NLP tasks with six datasets. Similar to prior computer vision studies [244–247], we evaluate the ScanTextGAN model over the scanpath generation task. For this, we use the test split of the CELER dataset, Mishra *et al.* (2016) [248], and Mishra *et al.* (2017) [249]. In addition, unlike the computer vision studies, we also evaluate the ScanTextGAN model for improvement in NLP tasks. The hypothesis is that the human eyes (and consequently the brain) process many language comprehension tasks unconsciously and without visible effort. The next logical step is to capture (or, in our case, generate) this mental representation of language understanding and use it to improve our machine-learning systems. For evaluation, we use four tasks from the GLUE benchmark and two from the tasks proposed by [248]. While the ScanTextGAN model is trained over news text from the CELER dataset, with the help of the other datasets, we expand our testing to other domains, including reviews, quotes, tweets, and Wikipedia text.

[*]All results are calculated with five random seeds and reported as the mean of those five runs

Generator Model	MultiMatch ↑				Levenshtein Distance ↓
	Vector↑	Length↑	Position↑	Duration↑	
Inter-subject score[§]	0.973	0.958	0.830	0.698	0.691
LSTM Encoder-Decoder trained with scanpath content loss	0.975	0.956	0.765	0.344	0.865
ScanTextGAN – Text Reconstruction – GAN Loss	0.968	0.947	0.728	0.703	0.779
ScanTextGAN	**0.983**	**0.972**	**0.787**	0.733	**0.769**
ScanTextGAN – Text Reconstruction	0.974	0.957	0.773	0.703	0.798
ScanTextGAN – GAN Loss	0.973	0.955	0.750	**0.761**	0.786
ScanTextGAN + addition of noise	0.971	0.952	0.756	0.736	0.791
ScanTextGAN – Text (CLS) Reconstruction + sentence reconstruction	0.978	0.963	0.724	0.721	0.805

Table 3.9: In-domain Evaluation of Scanpath Generation on the CELER dataset [242].

3.2.4.1 Evaluation Datasets

Mishra *et al.* (2017) [249] comprises eye movements and reading difficulty data recorded for 32 paragraphs on 16 different topics, *viz.* history, science, literature, *etc.* For each topic, comparable paragraphs were extracted from Wikipedia[†] and simple Wikipedia[‡]. The participant's eye movements are tracked using an SR-Research Eyelink-1000 Plus eye tracker. Using the ground truth scanpaths over the text corpora, we evaluate the quality of generated scanpaths.

Mishra *et al.* (2016) [248] contains eye fixation sequences of seven participants for 994 text snippets annotated for sentiment and sarcasm. These were taken from Amazon Movie Corpus , Twitter, and sarcastic quote websites. The task assigned to the participants was to read one sentence at a time and annotate it with binary sentiment polarity labels (*i.e.*, positive/negative). The same datasets were used in several studies [185, 186, 250] to show improvements in sarcasm and sentiment analysis. We use the datasets to evaluate both the generation quality and potential improvements in NLP tasks.

Furthermore, we explore the potential of including cognitive signals contained in scanpaths in NLP models for a range of GLUE tasks which include Sentiment Analysis using Stanford Sentiment Treebank (SST), Paraphrase Detection using Microsoft Research Paraphrase Corpus (MRPC) and Quora Question Pairs (QQP), Natural Language Inference using Recognizing Textual Entailment (RTE) dataset.

Next, we cover the results of scanpath generation and its application in NLP tasks.

[†]`https://en.wikipedia.org/`

[‡]`https://simple.wikipedia.org/`

[§]In the CELER dataset, there are only 78 shared sentences amongst all the participants. Therefore, inter-subject scanpath evaluation is done only for these sentences. In contrast, the ScanTextGAN results are reported for the entire test set (including these 78 sentences).

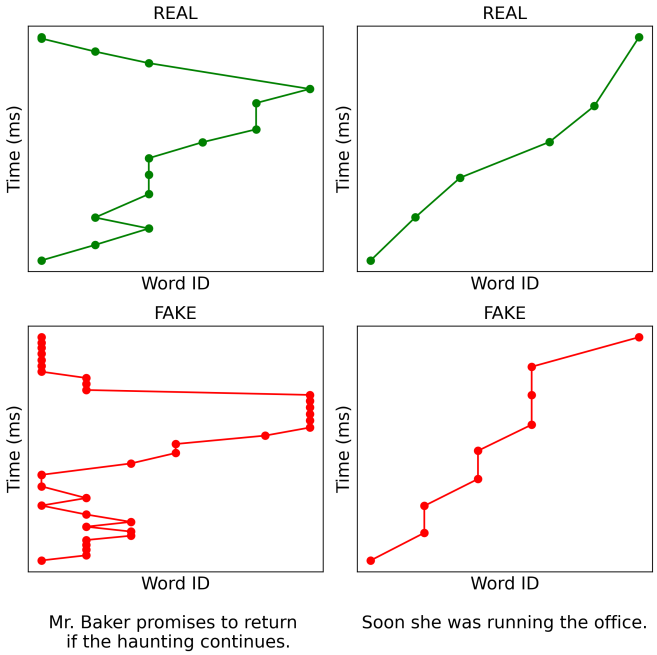


Figure 3.12: Comparison of *real* and *synthesized* scanpaths corresponding to a few text samples. The proposed ScanTextGAN model generates the latter.

3.2.4.2 Evaluation of Scanpath Generation

We evaluate the scanpath generation model on two most commonly used metrics in image scanpath generation studies [244, 245, 251, 252]: **MultiMatch** [253] and **Levenshtein Distance** [254]. Multimatch is a geometrical measure that compares scanpaths across a comprehensive set of dimensions composed of shape, lengths, position, and fixation duration. Levenshtein Distance between a pair of sequences measures the least number of edits (inserts, deletes, substitution) to transform one into the other.

3.2.4.2.1 Scanpath Evaluation Metrics **MultiMatch** is a geometrical measure that models scanpaths as vectors in 2-D space, wherein the vectors represent saccadic eye movements. Starting and ending coordinates of these saccades constitute the fixation positions. It compares scanpaths across multiple dimensions, *viz.* shape, length, position, direction, and fixation duration. Shape measures the vector difference between aligned saccade pairs, which is then normalized by twice the diagonal screen size. Length measures the normalized difference between the endpoints of real and generated saccade vectors. Direction is the angular distance between the two vectors. The position is the Euclidean difference in position between aligned vectors, and duration measures the difference in fixation durations normalized against the maximum duration. Since our work deals with scanpaths over text, we use 1-D space to represent the saccade vectors where word IDs denote the fixation positions. Thus, it is easy to see that computing scanpath direction similarity is redundant here (it is subsumed within position); hence we drop it from our analysis.

Levenshtein Distance between a pair of sequences measures the least number of character edits, i.e., insertion, deletion, and substitution needed to transform one sequence into the other. Specifically, we use it to gauge the degree of dissimilarity between a pair of real R and generated G scanpaths. To account for the fixation durations of each word, R and G are temporally binned using a 50 ms bin size, similar to the computation of ScanMatch metric [255]. The resulting sequences of word IDs, R_W and G_W are transformed into character strings, $R_S = \{r_1, r_2, ..., r_n\}$ and $G_S = \{g_1, g_2, ..., g_m\}$, where R_S and G_S are strings over the ASCII alphabet and $n = |R_S|$ and $m = |G_S|$. Thus, a lower NLD score is indicative of greater scanpath similarity.

Further, as a top-line comparison, we use **inter-subject scanpath similarity** [244]. It measures the degree of variation among real human scanpaths corresponding to each text input. To compute this, we first calculate each subject's performance by treating the scanpaths of other subjects as the ground truth. Then, the average value of all subjects is used as inter-subject performance.

Baselines: Since ScanTextGAN is the first text-based scanpath generation model, we conduct an ablation study to compare ScanTextGAN with

Generator Model	MultiMatch ↑				Levenshtein Distance ↓
	Vector↑	Length↑	Position↑	Duration↑	
Inter-subject score	0.977	0.963	0.839	0.715	0.723
LSTM Encoder-Decoder trained with scanpath content loss	**0.984**	**0.973**	0.714	0.379	0.918
ScanTextGAN – Text Reconstruction – GAN Loss	0.977	0.960	0.780	0.769	0.847
ScanTextGAN	0.966	0.945	**0.791**	**0.771**	**0.836**
ScanTextGAN – Text Reconstruction	0.976	0.961	0.763	0.757	0.845
ScanTextGAN – GAN Loss	0.976	0.959	0.774	0.768	0.839
ScanTextGAN + addition of noise	0.968	0.947	0.737	0.743	0.838
ScanTextGAN – Text (CLS) Reconstruction + sentence reconstruction	0.964	0.934	0.747	0.733	0.869

Table 3.10: Cross-domain Evaluation of Scanpath Generation on the Dataset by [248].

Generator Model	MultiMatch ↑				Levenshtein Distance ↓
	Vector↑	Length↑	Position↑	Duration↑	
Inter-subject score	0.994	0.991	0.834	0.620	0.845
LSTM Encoder-Decoder trained with scanpath content loss	**0.992**	**0.987**	0.596	0.329	0.969
ScanTextGAN – Text Reconstruction – GAN Loss	0.990	0.984	0.729	0.705	0.951
ScanTextGAN	0.984	0.977	**0.759**	0.693	**0.931**
ScanTextGAN – Text Reconstruction	0.986	0.981	0.756	**0.706**	0.939
ScanTextGAN – GAN Loss	0.990	0.984	0.739	**0.706**	0.945
ScanTextGAN + addition of noise	0.984	0.976	0.759	0.703	0.943
ScanTextGAN – Text (CLS) Reconstruction + sentence reconstruction	0.983	0.974	0.667	0.674	0.958

Table 3.11: Cross-domain Evaluation of Scanpath Generation on the Dataset by [249].

its other variants. Specifically, we compare ScanTextGAN with the following six configurations: (1) An LSTM-based network trained with scanpath content loss. Sentence embeddings obtained through BERT are concatenated with noise in this model. The resultant is fed to an attention module with four heads, then passed to a network of LSTMs and Batch Normalization layers applied in tandem. (2) ScanTextGAN model trained with only the scanpath content loss. (3) ScanTextGAN model without the text reconstruction loss (Task-2). (4) ScanTextGAN model with BERT-based sentence embeddings reconstruction instead of CLS token reconstruction. (5) ScanTextGAN model with the addition of noise instead of concatenation. (6) ScanTextGAN model trained without GAN loss.

Results: Table 3.9 presents the results of our scanpath prediction model on the CELER dataset. Further, we also compare ScanTextGAN with baselines on two other contemporary datasets of movie reviews, tweets, and sarcastic quotes [248], Wikipedia and simple Wikipedia paragraphs [249]. Tables 3.10 and 3.11 present the results of our model on those datasets. For obtaining results on these corpora, we use the model trained on the CELER dataset, thus helping us evaluate the cross-domain performance of the model.

As can be seen in Table 3.9, Table 3.10 and Table 3.11, ScanTextGAN outperforms other models for scanpath prediction on most metrics. The performance of ScanTextGAN even surpasses inter-subject reference on Duration and comes very close to Vector, Length, and Position.

We observe that adopting the reconstruction of the CLS token as an

auxiliary task (Task - 2) boosts the model performance. Reconstructing the full sentence embeddings rather than the CLS tokens only as an auxiliary task does not always improve the results, despite adding a larger computational overhead. The results also reveal that concatenating noise with text embeddings is more rewarding than adding it.

Further, to compare the skipping behavior of ScanTextGAN with humans, we calculate the weighted F1 score of the words skipped and attended by both model types. We find the weighted F1 to be 64.6 between them. Fig. 3.12 presents a visual comparison between real scanpaths from the available eyetracking data and scanpaths generated by ScanTextGAN, corresponding to some randomly chosen text samples. We can observe that the generated scanpaths resemble the real ones to a great extent. Thus, the quantitative and qualitative results on in-domain and cross-domain settings lead us to believe that our proposed scanpath generation model can be deemed a good approximator of the human scanpaths.

3.2.4.3 Application to NLP Tasks

We use them to augment various NLP models and measure their performance to demonstrate the usefulness of cognitive signals hidden in the *generated* scanpaths.

Sentiment Classification and Sarcasm Detection: For these tasks, we use a model consisting of a network of two branches of BiLSTMs and Batch Normalization layers that perform sequential modeling over text representations obtained through BERT and scanpaths fed as input to the model. The outputs of both branches are combined and passed to another layer of BiLSTMs, followed by a feed-forward network that predicts binary sentiment/sarcasm labels corresponding to the input after activating with the Sigmoid function. We follow a 10-fold cross-validation regime.

We compare the models with generated scanpaths, real scanpaths, and without scanpaths. Further, to investigate whether performance gains observed by adding scanpaths are due to scanpaths and not the increase in the number of parameters, we train a *Random-Random* variant in which we send Random noise as scanpaths to the model with an increased number of parameters. We also simulate the real-world case where both real and generated scanpaths are available during train time, but only generated ones are available during test time, for example, during user deployment.

Table 3.12 records the results of sentiment analysis and sarcasm detection tasks [248]. We note that generated scanpaths training and testing lead to similar gains for sentiment analysis and sarcasm detection as real scanpaths. The model with an increased number of parameters fed random noise in place of scanpaths performs similarly to the model trained without any scanpaths. Interestingly, the best results are obtained when model training uses both real and generated scanpaths. We believe this is due to ScanTextGAN

Model Configuration		F1 score	
Train	**Test**	**Sentiment**	**Sarcasm**
w/o	w/o	0.7839	0.9438
Random	Random	0.7990	0.9397
Random	Generated	0.7773	0.9313
Real	Generated	0.8319	0.9378
Real	Real	0.8334	0.9501
Generated	Real	0.8402	0.9452
Generated	Generated	0.8332	0.9506
Real + Generated	Generated	**0.8404**	**0.9512**
Intent-Aware	Intent-Aware	**0.8477**	**0.9528**

Table 3.12: Sentiment analysis and sarcasm detection results on the dataset by [248]. Model configuration refers to the type of scanpath included in train and test data.

bringing additional cognitive information from the news-reading CELER corpus, which is not present in the real scanpaths in [248]. In addition to the intrinsic evaluation presented in §3.2.4.2, this downstream evaluation demonstrates the high quality of the synthesized scanpaths, showing that they contain valuable cognitive processing signals for NLP tasks.

GLUE Tasks: To validate further, we augment classification models (based on sequential modeling using LSTMs) with generated scanpaths to show performance improvement in downstream NLP tasks on four GLUE benchmark datasets – SST, MRPC, RTE, QQP as described in §3.2.4.1. Table 3.13 reports the accuracy and weighted-F1 scores of the models trained with and without scanpaths for these tasks. We observe that in all four tasks, the model trained with generated scanpaths outperforms the one without scanpaths.

Intent-Aware Scanpaths: Finally, we try to condition scanpaths generation on the downstream natural language task. We back-propagate gradients from the downstream NLP task to the conditional generator. In this fashion, the model learns to generate *intent-aware* scanpaths. The hypothesis is that finetuning scanpath generation based on feedback from the natural language task will bias the generator towards words more pertinent to that task and thus could help further improve performance on the downstream task. The architecture is shown in Fig 3.13. The results in Tables 3.12 and 3.13 validate the hypothesis that we observe consistent improvements in all downstream tasks. Fig 3.10 and Fig 3.14 show a few examples of scanpaths and saliency generated for three downstream natural language tasks.

Together these results corroborate the hypothesis that leveraging the cognitive signals approximated by synthetic scanpaths in NLP models leads

Dataset	Model	Acc	F1 score
SST	w/o scanpaths	0.8090	0.8089
	w/ random scanpaths	0.8059	0.8061
	w/ generated scanpaths	**0.8138**	**0.8138**
	w/ intent-aware scanpaths	**0.8269**	**0.8272**
MRPC	w/o scanpaths	0.6902	0.6656
	w/ random scanpaths	0.6623	0.6680
	w/ generated scanpaths	**0.6969**	**0.6828**
	w/ intent-aware scanpaths	**0.7009**	**0.6911**
RTE	w/o scanpaths	0.6162	0.6080
	w/ random scanpaths	0.5802	0.5794
	w/ generated scanpaths	**0.6211**	**0.6205**
	w/ intent-aware scanpaths	**0.6293**	**0.6278**
QQP	w/o scanpaths	0.8499	0.8513
	w/ random scanpaths	0.8491	0.8503
	w/ generated scanpaths	**0.8578**	**0.8596**
	w/ intent-aware scanpaths	**0.8648**	**0.8658**

Table 3.13: Results of training NLP models with and without scanpaths on the GLUE benchmark tasks. Including scanpaths leads to consistent improvements across all the NLP tasks.

to performance gains.

3.2.5 Intent-Aware Scanpaths

As described in section §3.2.4.3, the generator conditioned on the downstream natural language task yields *intent-aware* scanpaths. Augmenting NLP models with these scanpaths leads to higher performance gains. Here, we provide more details on *intent-aware* scanpath generation. Please refer to figures 3.13 and 3.14 on the following page. Saliency corresponding to intent-aware scanpaths are shown in Fig. 3.14.

3.2.6 Conclusion

In this work, we make two novel contributions toward integrating cognitive and natural language processing. (1) We introduce the first scanpath generation model over text, integrating a cognitive reading model with a data-driven approach to address the scarcity of human gaze data on text. (2) We propose generated scanpaths that can be flexibly adapted to differ-

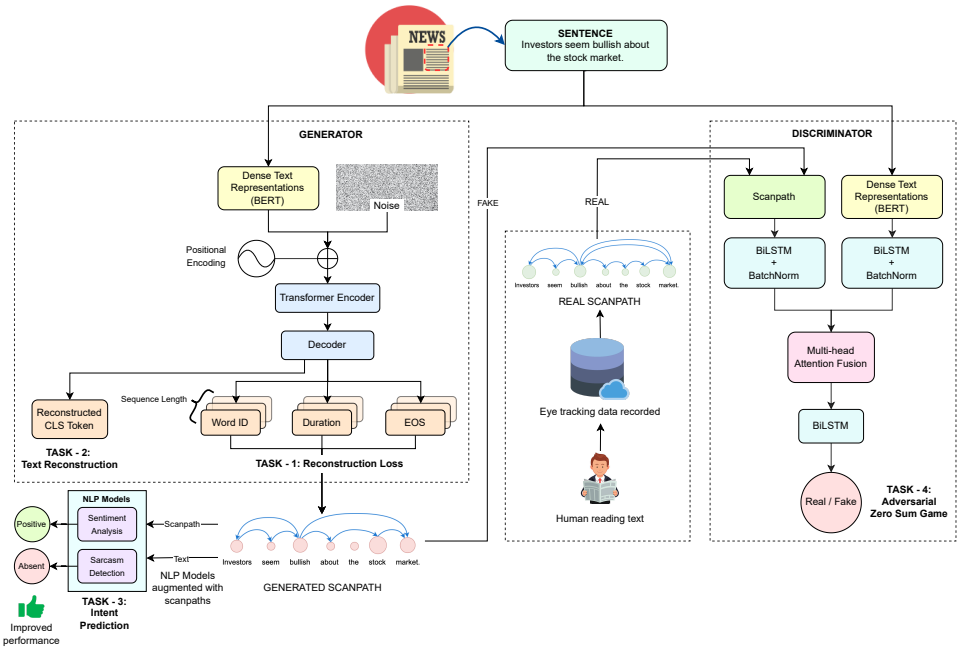


Figure 3.13: The architecture of the proposed Intent-Aware **ScanTextGAN** model. The model consists of a conditional generator and a discriminator playing a zero-sum game. Two cognitively inspired losses train the generator: scanpath (Task-1) and text (Task-2) reconstruction, a loss from the downstream intent of the natural language task (Task-3), and finally, the loss from the adversarial zero-sum game (Task-4). Variations of scanpaths are generated based on the downstream natural language task.

ent NLP tasks without needing task-specific ground truth human gaze data. We show that both advances significantly improve performance across six NLP datasets over various baselines. Our findings demonstrate the feasibility and significant potential of combining cognitive and data-driven models for NLP tasks. Without the need for real-time gaze recordings, the potential research avenues for augmenting and understanding NLP models through the cognitive processing information encoded in synthesized scanpaths are multiplied.

3.2.7 Limitations

In this work, we demonstrated artificial scanpath generation over multiple eye-tracking datasets. Further, our experiments build a link between cognitive and natural language processing and show how one can inform the other. However, the proposed method has a few limitations, which we aim to address in the future. The field needs work on bigger and more diverse eye-tracking datasets, which can enable scanpath generation over

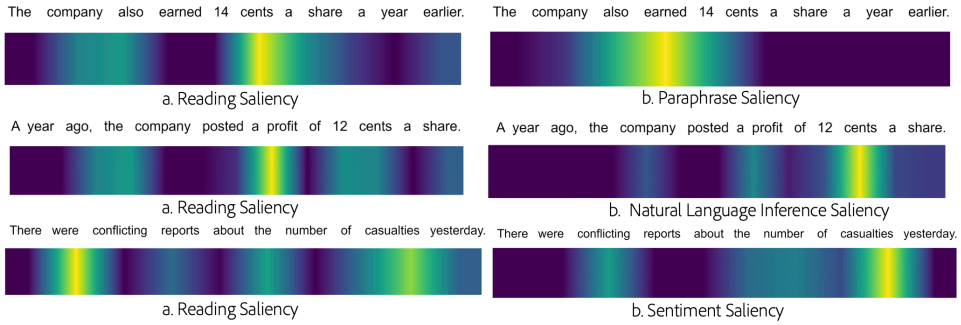


Figure 3.14: Saliency samples generated by conditioning scanpath generation on different downstream natural language tasks. It can be observed that the conditioned saliency pays much more attention to words important for that downstream task.

longer text sequences and can model generating scanpaths conditioned on previously read context. Besides, a better understanding of the entire scanpath generation process can help model the intra and inter-sentence scanpath generation process. The understanding would enable the integration of scanpaths to generative modeling tasks, which we intend to take up in future work. Another parallel direction is to include both explicit (like using RLHF) and implicit signals (like using cognitive signals) to better NLP tasks like language modeling.

Chapter 4

Generating Content to Optimize Behavior

In the last chapter, we discussed large content and behavior models (LCBMs) with the ability to generate content conditioned on behavior, generate (simulate) behavior conditioned on content, and understanding of content and behavior. In this chapter, we focus specifically on the capability of generating content conditioned on behavior. LCBMs were built following the instruction fine-tuning paradigm. Using LCBMs, we showed that including behavior data as receiver tokens along with content data (communicator tokens) helps complete the entire communication flow and train the LLM to teach it both the receiver side and the communicator side of the flow.

In this chapter, we take a deeper look into the common use case of generating content which can help get the behavior the communicator wants. For instance, a marketer wants to write emails or compose tweets that will bring her the maximum number of link clicks and likes. We propose several solutions to solve this problem and compare several paradigms to achieve this. We show this over both short-term key performance indicators (downloads and likes), and long-term indicators (brand and content memorability).

4.1

Publications

1. Khurana, V., Kumar, Y., Hollenstein, N., Kumar, R., & Krishnamurthy, B. (2023). Synthesizing Human Gaze Feedback for Improved NLP Performance. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (pp. 1895-1908).
2. Kumar, Y., Jha, R., Gupta, A., Aggarwal, M., Garg, A., Malyan, T., Bhardwaj, A., Ratn Shah, R., Krishnamurthy, B., & Chen, C. (2023). Persuasion Strategies in Advertisements. Proceedings of the AAAI Conference on Artificial Intelligence, 37(1), 57-66. `https://doi.org/10.1609/aaai.v37i1.25076`
3. Bhattacharya, A., Singla, Y. K., Krishnamurthy, B., Shah, R. R., & Chen, C. (2023). A Video Is Worth 4096 Tokens: Verbalize Videos To Understand Them In Zero Shot. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 9822–9839, Singapore. Association for Computational Linguistics. (**Nominated for the best paper award!**)
4. Khandelwal, A., Agrawal, A., Bhattacharyya, A., Singla, Y.K., Singh, S., Bhattacharya, U., Dasgupta, I., Petrangeli, S., Shah, R.R., Chen, C. and Krishnamurthy, B., 2024. Large Content And Behavior Models To Understand, Simulate, And Optimize Content And Behavior. International Conference on Learning Representations. (**Spotlight and nominated for award!**)
5. S I, H., Singh, S., K Singla, Y., Krishnamurthy, B., Chen, C., Baths V., & Ratn Shah, R. (2024). Long-Term Ad Memorability: Understanding and Generating Memorable Ads. arxiv preprint (Under review).
6. Khurana, V., Singla, Y.K., Subramanian, J., Shah, R.R., Chen, C., Xu, Z. and Krishnamurthy, B., 2023. Behavior Optimized Image Generation. arXiv preprint arXiv:2311.10995. (Under review)

Bibliography

[1] Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Foundations and recent trends in multimodal machine learning: Principles, challenges, and open questions. *arXiv preprint arXiv:2209.03430*, 2022.

[2] Patrizia Grifoni. *Multimodal human computer interaction and pervasive services*. IGI Global, 2009.

[3] Jean-Claude Martin, Sarah Grimard, and Katerina Alexandri. On the annotation of the multimodal behavior and computation of cooperation between modalities. In *Proceedings of the Workshop on Multimodal Communicatiom and Context in Embodied Agents, Fifth International Conference on Autonomous Agents*, pages 1–7, 2001.

[4] Claude E. Shannon and Warren Weaver. *The mathematical theory of communication.* The mathematical theory of communication. University of Illinois Press, Champaign, IL, US, 1949.

[5] Ashmit Khandelwal, Aditya Agrawal, Aanisha Bhattacharyya, Yaman K Singla, Somesh Singh, Uttaran Bhattacharya, Ishita Dasgupta, Stefano Petrangeli, Rajiv Ratn Shah, Changyou Chen, et al. Large content and behavior models to understand, simulate, and optimize content and behavior. *arXiv preprint arXiv:2309.00359*, 2023.

[6] Varun Khurana, Yaman K Singla, Jayakumar Subramanian, Rajiv Ratn Shah, Changyou Chen, Zhiqiang Xu, and Balaji Krishnamurthy. Behavior optimized image generation. *arXiv preprint arXiv:2311.10995*, 2023.

[7] Harini SI, Somesh Singh, Yaman K Singla, Aanisha Bhattacharyya, Veeky Baths, Changyou Chen, Rajiv Ratn Shah, and Balaji Krishnamurthy. Long-term ad memorability: Understanding and generating memorable ads. *arXiv preprint arXiv:2309.00378*, 2023.

[8] Varun Khurana, Yaman Kumar, Nora Hollenstein, Rajesh Kumar, and Balaji Krishnamurthy. Synthesizing human gaze feedback for improved NLP performance. In *Proceedings of the 17th Conference of*

the European Chapter of the Association for Computational Linguistics, pages 1895–1908, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[9] Leo Breiman. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231, 2001.

[10] Jake M Hofman, Amit Sharma, and Duncan J Watts. Prediction and explanation in social systems. *Science*, 355(6324):486–488, 2017.

[11] Galit Shmueli. To explain or to predict? *Statistical Science*, 2010.

[12] Stanley Milgram and Christian Gudehus. Obedience to authority, 1978.

[13] Solomon E Asch. The doctrine of suggestion, prestige and imitation in social psychology. *Psychological review*, 55(5):250, 1948.

[14] Open Science Collaboration. Estimating the reproducibility of psychological science. *Science*, 349(6251):aac4716, 2015.

[15] Philip E Tetlock. Expert political judgment. In *Expert Political Judgment*. Princeton University Press, 2017.

[16] The Forecasting Collaborative. Insights into the accuracy of social scientists' forecasts of societal change. *Nature human behaviour*, 7(4):484–501, 2023.

[17] Chenhao Tan, Lillian Lee, and Bo Pang. The effect of wording on message propagation: Topic-and author-controlled natural experiments on twitter. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 175–185, 2014.

[18] Phillip Isola, Jianxiong Xiao, Devi Parikh, Antonio Torralba, and Aude Oliva. What makes a photograph memorable? *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1469–1482, 2013.

[19] Ivan Habernal and Iryna Gurevych. What makes a convincing argument? empirical analysis and detecting attributes of convincingness in web argumentation. In *EMNLP*, pages 1214–1223, 2016.

[20] Yaman Kumar, Rajat Jha, Arunim Gupta, Milan Aggarwal, Aditya Garg, Tushar Malyan, Ayush Bhardwaj, Rajiv Ratn Shah, Balaji Krishnamurthy, and Changyou Chen. Persuasion strategies in advertisements. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 57–66, 2023.

[21] Kelvin Luu, Chenhao Tan, and Noah A Smith. Measuring online debaters' persuasive skill from text over time. *Transactions of the Association for Computational Linguistics*, 7:537–550, 2019.

[22] Aanisha Bhattacharyya, Yaman K Singla, Balaji Krishnamurthy, Rajiv Ratn Shah, and Changyou Chen. A video is worth 4096 tokens: Verbalize videos to understand them in zero shot. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9822–9839, Singapore, December 2023. Association for Computational Linguistics.

[23] Justin Cheng, Lada Adamic, P Alex Dow, Jon Michael Kleinberg, and Jure Leskovec. Can cascades be predicted? In *Proceedings of the 23rd international conference on World wide web*, pages 925–936, 2014.

[24] Travis Martin, Jake M Hofman, Amit Sharma, Ashton Anderson, and Duncan J Watts. Exploring limits to prediction in complex social systems. In *Proceedings of the 25th international conference on world wide web*, pages 683–694, 2016.

[25] Hyunyoung Choi and Hal Varian. Predicting the present with google trends. *Economic record*, 88:2–9, 2012.

[26] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[27] Matthew J Salganik. *Bit by bit: Social research in the digital age*. Princeton University Press, 2019.

[28] Yaman Kumar Singla, Jui Shah, Changyou Chen, and Rajiv Ratn Shah. What do audio transformers hear? probing their representations for language delivery & structure. In *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 910–925. IEEE, 2022.

[29] Shobhana Chandra, Sanjeev Verma, Weng Marc Lim, Satish Kumar, and Naveen Donthu. Personalization in personalized marketing: Trends and ways forward. *Psychology & Marketing*, 39(8):1529–1562, 2022.

[30] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

[31] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov,

Daniel Golovin, et al. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1222–1230, 2013.

[32] Reid Pryzant, Youngjoo Chung, and Dan Jurafsky. Predicting sales from the language of product descriptions. *eCOM@ SIGIR*, 2311, 2017.

[33] Phillip Isola, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. What makes an image memorable? In *CVPR 2011*, pages 145–152. IEEE, 2011.

[34] Aditya Khosla, Akhil S Raju, Antonio Torralba, and Aude Oliva. Understanding and predicting image memorability at a large scale. In *Proceedings of the IEEE international conference on computer vision*, pages 2390–2398, 2015.

[35] Kate Newstead and Jenni Romaniuk. Cost per second: The relative effectiveness of 15-and 30-second television advertisements. *Journal of Advertising Research*, 50(1):68–76, 2010.

[36] Barbara Plank. Keystroke dynamics as signal for shallow syntactic parsing. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 609–619, 2016.

[37] Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. Improving sentence compression by learning to predict gaze. In *NAACL: Human Language Technologies*, San Diego, California, June 2016. Association for Computational Linguistics.

[38] Julia Kreutzer, Shahram Khadivi, Evgeny Matusov, and Stefan Riezler. Can neural machine translation be improved with user feedback? *arXiv preprint arXiv:1804.05958*, 2018.

[39] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

[40] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

[41] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju,

William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

[42] Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023.

[43] John David Pressman, Katherine Crowson, and Simulacra Captions Contributors. Simulacra aesthetic captions, 2023.

[44] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Better aligning text-to-image models with human preference. *arXiv preprint arXiv:2303.14420*, 2023.

[45] Denis McQuail. *Mass communication theory: An introduction*. Sage Publications, Inc, 1987.

[46] Klaus Krippendorff. *Content analysis: An introduction to its methodology*. Sage publications, 2018.

[47] Harold D Lasswell. The structure and function of communication in society. *The communication of ideas*, 37(1):136–139, 1948.

[48] Harold D Lasswell. *Propaganda technique in world war I*. MIT press, 1971.

[49] Richard E Petty, John T Cacioppo, and Martin Heesacker. Effects of rhetorical questions on persuasion: A cognitive response analysis. *Journal of personality and social psychology*, 40(3):432, 1981.

[50] Shelly Chaiken. Heuristic versus systematic information processing and the use of source versus message cues in persuasion. *Journal of personality and social psychology*, 39(5), 1980.

[51] Andranik Tumasjan, Timm Sprenger, Philipp Sandner, and Isabell Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the international AAAI conference on web and social media*, volume 4, pages 178–185, 2010.

[52] Sitaram Asur and Bernardo A Huberman. Predicting the future with social media. In *2010 IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology*, volume 1, pages 492–499. IEEE, 2010.

[53] Seungbae Kim, Jyun-Yu Jiang, Masaki Nakada, Jinyoung Han, and Wei Wang. Multimodal post attentive profiling for influencer marketing. In *Proceedings of The Web Conference 2020*, pages 2878–2884, 2020.

[54] Yan Carrière-Swallow and Felipe Labbé. Nowcasting with google trends in an emerging market. *Journal of Forecasting*, 32(4):289–298, 2013.

[55] Nicholas Generous, Geoffrey Fairchild, Alina Deshpande, Sara Y Del Valle, and Reid Priedhorsky. Global disease monitoring and forecasting with wikipedia. *PLoS computational biology*, 10(11):e1003892, 2014.

[56] Giovanni De Toni, Cristian Consonni, and Alberto Montresor. A general method for estimating the prevalence of influenza-like-symptoms with wikipedia data. *Plos one*, 16(8):e0256858, 2021.

[57] Márton Mestyán, Taha Yasseri, and János Kertész. Early prediction of movie box office success based on wikipedia activity big data. *PloS one*, 8(8):e71226, 2013.

[58] Coco Krumme, Alejandro Llorente, Manuel Cebrian, Alex Pentland, and Esteban Moro. The predictability of consumer visitation patterns. *Scientific reports*, 3(1):1645, 2013.

[59] Yves-Alexandre De Montjoye, Laura Radaelli, Vivek Kumar Singh, and Alex "Sandy" Pentland. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science*, 347(6221):536–539, 2015.

[60] Dirk Brockmann, Lars Hufnagel, and Theo Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–465, 2006.

[61] Giovanna Miritello, Rubén Lara, Manuel Cebrian, and Esteban Moro. Limited communication capacity unveils strategies for human interaction. *Scientific reports*, 3(1):1950, 2013.

[62] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL: Human Language Technologies, Volume 1 (Long and Short Papers)*. ACL, 2019.

[63] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

[64] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[65] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

[66] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[67] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[68] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[69] OpenAI. Gpt-4 technical report, 2023.

[70] Stella Biderman, Kieran Bicheno, and Leo Gao. Datasheet for the pile. *arXiv preprint arXiv:2201.07311*, 2022.

[71] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023.

[72] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023.

[73] Christof Rapp. Aristotle's rhetoric. 2002.

[74] Joan Meyers-Levy and Prashant Malaviya. Consumers' processing of persuasive advertisements: An integrative framework of persuasion theories. *Journal of marketing*, 63(4_suppl1):45–60, 1999.

[75] Punam Anand Keller, Isaac M Lipkus, and Barbara K Rimer. Affect, framing, and persuasion. *Journal of Marketing Research*, 40(1):54–64, 2003.

[76] Carl Iver Hovland, Irving Lester Janis, and Harold H Kelley. Communication and persuasion. 1953.

[77] Richard E Petty and John T Cacioppo. The elaboration likelihood model of persuasion. In *Communication and persuasion*, pages 1–24. Springer, 1986.

[78] Chongyang Bai, Haipeng Chen, Srijan Kumar, Jure Leskovec, and VS Subrahmanian. M2p2: Multimodal persuasion prediction using adaptive fusion. *IEEE Transactions on Multimedia*, 2021.

[79] Jungseock Joo, Weixin Li, Francis F Steen, and Song-Chun Zhu. Visual persuasion: Inferring communicative intents of images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 216–223, 2014.

[80] Jiaao Chen and Diyi Yang. Weakly-supervised hierarchical models for predicting persuasive strategies in good-faith textual requests. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12648–12656, May 2021.

[81] Paul AM Van Lange, E Tory Higgins, and Arie W Kruglanski. Handbook of theories of social psychology. *Handbook of Theories of Social Psychology*, pages 1–568, 2011.

[82] Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. Automatic understanding of image and video advertisements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1705–1715, 2017.

[83] Pranav Anand, Joseph King, Jordan Boyd-Graber, Earl Wagner, Craig Martell, Doug Oard, and Philip Resnik. Believe me—we can do this! annotating persuasive acts in blog text. In *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[84] Demetrios Vakratsas and Tim Ambler. How advertising works: what do we really know? *Journal of marketing*, 63(1):26–43, 1999.

[85] Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th international conference on world wide web*, pages 613–624, 2016.

[86] Jiaao Chen and Diyi Yang. Weakly-supervised hierarchical models for predicting persuasive strategies in good-faith textual requests. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12648–12656, 2021.

[87] Omar Shaikh, Jiaao Chen, Jon Saad-Falcon, Polo Chau, and Diyi Yang. Examining the ordering of rhetorical strategies in persuasive requests. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1299–1306, Online, November 2020. Association for Computational Linguistics.

[88] Jialu Li, Esin Durmus, and Claire Cardie. Exploring the role of argument structure in online debate persuasion. In *EMNLP*, pages 8905–8912, Online, November 2020. Association for Computational Linguistics.

[89] Stephanie Lukin, Pranav Anand, Marilyn Walker, and Steve Whittaker. Argument strength is in the eye of the beholder: Audience effects in persuasion. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 742–753, Valencia, Spain, April 2017. Association for Computational Linguistics.

[90] Roxanne El Baff, Henning Wachsmuth, Khalid Al Khatib, and Benno Stein. Analyzing the Persuasive Effect of Style in News Editorial Argumentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3154–3160, Online, July 2020. Association for Computational Linguistics.

[91] Tim Althoff, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. How to ask for a favor: A case study on the success of altruistic requests. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 8, pages 12–21, 2014.

[92] Zhongyu Wei, Yang Liu, and Yi Li. Is this post persuasive? ranking argumentative comments in online forum. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 195–200, 2016.

[93] Ivan Donadello, Mauro Dragoni, and Claudio Eccher. Explaining reasoning algorithms with persuasiveness: a case study for a behavioural change system. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pages 646–653, 2020.

[94] Emmanuel Hadoux, Anthony Hunter, and Sylwia Polberg. Strategic argumentation dialogues for persuasion: Framework and experiments based on modelling the beliefs and concerns of the persuadee. *arXiv preprint arXiv:2101.11870*, 2021.

[95] Ivan Donadello, Anthony Hunter, Stefano Teso, and Mauro Dragoni. Machine learning for utility prediction in argument-based computa-

tional persuasion. In *proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 5592–5599, 2022.

[96] Jungseock Joo, Francis F Steen, and Song-Chun Zhu. Automated facial trait judgment and election outcome prediction: Social dimensions of face. In *Proceedings of the IEEE international conference on computer vision*, pages 3712–3720, 2015.

[97] Behjat Siddiquie, Dave Chisholm, and Ajay Divakaran. Exploiting multimodal affect and semantics to identify politically persuasive web videos. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 203–210, 2015.

[98] Elliot Aronson, Judith A Turner, and J Merrill Carlsmith. Communicator credibility and communication discrepancy as determinants of opinion change. *The Journal of Abnormal and Social Psychology*, 67(1):31, 1963.

[99] Robert B Cialdini and Robert B Cialdini. *Influence: The psychology of persuasion*, volume 55. Collins New York, 2007.

[100] Stanley Milgram. Behavioral study of obedience. *The Journal of abnormal and social psychology*, 67(4):371, 1963.

[101] Elliott McGinnies and Charles D Ward. Better liked than right: Trustworthiness and expertise as factors in credibility. *Personality and Social Psychology Bulletin*, 6(3):467–472, 1980.

[102] Kim Giffin. The contribution of studies of source credibility to a theory of interpersonal trust in the communication process. *Psychological bulletin*, 68(2):104, 1967.

[103] Rahul Radhakrishnan Iyer and Katia Sycara. An unsupervised domain-independent framework for automated detection of persuasion tactics in text. *arXiv preprint arXiv:1912.06745*, 2019.

[104] Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Tim Alberdingk Thijm, Graeme Hirst, and Benno Stein. Computational argumentation quality assessment in natural language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 176–187, 2017.

[105] Esin Durmus and Claire Cardie. Exploring the role of prior beliefs for argument persuasion. In *NAACL: Human Language Technologies, Volume 1 (Long Papers)*, pages 1035–1045, New Orleans, Louisiana, June 2018. ACL.

[106] Morton Deutsch and Harold B Gerard. A study of normative and informational social influences upon individual judgment. *The journal of abnormal and social psychology*, 51(3):629, 1955.

[107] Richard E Petty, Duane T Wegener, and Leandre R Fabrigar. Attitudes and attitude change. *Annual review of psychology*, 48(1):609–647, 1997.

[108] Wendy Wood. Attitude change: Persuasion and social influence. *Annual review of psychology*, 51(1):539–570, 2000.

[109] Robert B Cialdini and Noah J Goldstein. Social influence: Compliance and conformity. *Annual review of psychology*, 55(1):591–621, 2004.

[110] Nataly Levesque and Frank Pons. The human brand: A systematic literature review and research agenda. *Journal of Customer Behaviour*, 19(2):143–174, 2020.

[111] Sara Rosenthal and Kathleen Mckeown. Detecting influencers in multiple online genres. *ACM Transactions on Internet Technology (TOIT)*, 17(2):1–22, 2017.

[112] Diyi Yang, Jiaao Chen, Zichao Yang, Dan Jurafsky, and Eduard Hovy. Let's make your request more persuasive: Modeling persuasive strategies via semi-supervised neural nets on crowdfunding platforms. In *NAACL: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3620–3630, 2019.

[113] Fan Zhang, Diane Litman, and Kate Forbes-Riley. Inferring discourse relations from pdtb-style discourse labels for argumentative revision classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2615–2624, 2016.

[114] Christian Stab and Iryna Gurevych. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659, 2017.

[115] Christopher Hidey, Elena Musi, Alyssa Hwang, Smaranda Muresan, and Kathleen McKeown. Analyzing the semantic types of claims and premises in an online persuasive forum. In *Proceedings of the 4th Workshop on Argument Mining*, pages 11–21, 2017.

[116] Dennis T Regan. Effects of a favor and liking on compliance. *Journal of experimental social psychology*, 7(6):627–639, 1971.

[117] Margaret S Clark. Record keeping in two types of relationships. *Journal of personality and social psychology*, 47(3), 1984.

[118] Margaret S Clark and Judson Mills. Interpersonal attraction in exchange and communal relationships. *Journal of personality and social psychology*, 37(1), 1979.

[119] Margaret S Clark, Judson Mills, and Martha C Powell. Keeping track of needs in communal and exchange relationships. *Journal of personality and social psychology*, 51(2):333, 1986.

[120] Jonathan L Freedman and Scott C Fraser. Compliance without pressure: the foot-in-the-door technique. *Journal of personality and social psychology*, 4(2):195, 1966.

[121] Jerry M Burger. The foot-in-the-door compliance procedure: A multiple-process analysis and review. *Personality and social psychology review*, 3(4):303–325, 1999.

[122] Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. Persuasion for good: Towards a personalized persuasive dialogue system for social good. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5635–5649, Florence, Italy, July 2019. Association for Computational Linguistics.

[123] John Paul Vargheese, Matthew Collinson, and Judith Masthoff. Exploring susceptibility measures to persuasion. In *Persuasive Technology. Designing for Future Change: 15th International Conference on Persuasive Technology, PERSUASIVE 2020, Aalborg, Denmark, April 20–23, 2020, Proceedings 15*, pages 16–29. Springer, 2020.

[124] William J McGuire and Demetrios Papageorgis. The relative efficacy of various types of prior belief-defense in producing immunity against persuasion. *Journal of Abnormal Psychology*, 62(2), 1961.

[125] Eric S Knowles and Jay A Linn. *Resistance and persuasion*. Psychology Press, 2004.

[126] William J McGuire. Inducing resistance to persuasion. some contemporary approaches. *CC Haaland and WO Kaelber (Eds.), Self and Society. An Anthology of Readings, Lexington, Mass.(Ginn Custom Publishing) 1981, pp. 192-230.*, 1964.

[127] Angela Y Lee, Punam Anand Keller, and Brian Sternthal. Value from regulatory construal fit: The persuasive impact of fit between consumer goals and message concreteness. *Journal of Consumer Research*, 36(5):735–747, 2010.

[128] Adrian Furnham and Hua Chu Boo. A literature review of the anchoring effect. *The journal of socio-economics*, 40(1):35–42, 2011.

[129] Duane T Wegener, Richard E Petty, Brian T Detweiler-Bedell, and W Blair G Jarvis. Implications of attitude change theories for numerical anchoring: Anchor plausibility and the limits of anchor effectiveness. *Journal of Experimental Social Psychology*, 37(1):62–69, 2001.

[130] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *science*, 185(4157):1124–1131, 1974.

[131] Fritz Strack and Thomas Mussweiler. Explaining the enigmatic anchoring effect: Mechanisms of selective accessibility. *Journal of personality and social psychology*, 73(3):437, 1997.

[132] Chitrabhan B Bhattacharya and Sankar Sen. Consumer–company identification: A framework for understanding consumers' relationships with companies. *Journal of marketing*, 67(2):76–88, 2003.

[133] Amy Zhang, Bryan Culbertson, and Praveen Paritosh. Characterizing online discussion using coarse discourse sequences. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 357–366, 2017.

[134] Liane Longpre, Esin Durmus, and Claire Cardie. Persuasion of the undecided: Language vs. the listener. In *Proceedings of the 6th Workshop on Argument Mining*, 2019.

[135] Jack W Brehm. A theory of psychological reactance. 1966.

[136] Michael Lynn. Scarcity effects on value: A quantitative review of the commodity theory literature. *Psychology & Marketing*, 8(1):43–57, 1991.

[137] Alexander J Rothman, Steven C Martino, Brian T Bedell, Jerusha B Detweiler, and Peter Salovey. The systematic influence of gain-and loss-framed messages on interest in and use of different types of health behavior. *Personality and Social Psychology Bulletin*, 25(11):1355–1369, 1999.

[138] Amos Tversky and Daniel Kahneman. The framing of decisions and the psychology of choice. In *Behavioral decision making*, pages 25–41. Springer, 1985.

[139] Susan Fournier. Consumers and their brands: Developing relationship theory in consumer research. *Journal of consumer research*, 24(4):343–373, 1998.

[140] Nicholas Epley, Adam Waytz, and John T Cacioppo. On seeing human: a three-factor theory of anthropomorphism. *Psychological review*, 114(4):864, 2007.

[141] Sally Hibbert, Andrew Smith, Andrea Davies, and Fiona Ireland. Guilt appeals: Persuasion knowledge and charitable giving. *Psychology & Marketing*, 24(8):723–742, 2007.

[142] Richard E Petty, John T Cacioppo, and David Schumann. Central and peripheral routes to advertising effectiveness: The moderating role of involvement. *Journal of consumer research*, 10(2):135–146, 1983.

[143] He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. Decoupling strategy and generation in negotiation dialogues. *arXiv preprint arXiv:1808.09637*, 2018.

[144] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[145] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[146] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37. Springer, 2016.

[147] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.

[148] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[149] Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. Dense captioning with joint inference and visual context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2193–2202, 2017.

[150] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[151] Stella Frank, Emanuele Bugliarello, and Desmond Elliott. Vision-and-language or vision-for-language? on cross-modal influence in multimodal transformers. *arXiv preprint arXiv:2109.04448*, 2021.

[152] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[153] Ran Gilad-Bachrach, Amir Navot, and Naftali Tishby. Query by committee made real. *Advances in neural information processing systems*, 18, 2005.

[154] David D Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, pages 148–156. Elsevier, 1994.

[155] Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S. Kankanhalli. Video storytelling: Textual summaries for events. *IEEE Transactions on Multimedia*, 22(2):554–565, feb 2020.

[156] ROBERT PLUTCHIK. Chapter 1 - a general psychoevolutionary theory of emotion. In Robert Plutchik and Henry Kellerman, editors, *Theories of Emotion*, pages 3–33. Academic Press, 1980.

[157] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021.

[158] Ali Diba, Mohsen Fayyaz, Vivek Sharma, Manohar Paluri, Jürgen Gall, Rainer Stiefelhagen, and Luc Van Gool. Large scale holistic video understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 593–610. Springer, 2020.

[159] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022.

[160] Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, sep 2014.

[161] Breakthrough. Pyscenedetect: Video scene cut detection and analysis tool, 2023.

[162] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching, 2022.

[163] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023.

[164] Jing Wang, Jinhui Tang, Mingkun Yang, Xiang Bai, and Jiebo Luo. Improving ocr-based image captioning by incorporating geometrical relationship. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1306–1315, 2021.

[165] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[166] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training, 2022.

[167] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Junting Pan, Jiashuo Yu, Yali Wang, Limin Wang, and Yu Qiao. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022.

[168] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023.

[169] Youwei Lu and Xiaoyu Wu. Video storytelling based on gated video memorability filtering. *Electronics Letters*, 58(15):576–578, 2022.

[170] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7464–7473, 2019.

[171] Fanyi Xiao, Kaustav Kundu, Joseph Tighe, and Davide Modolo. Hierarchical self-supervised representation learning for movie understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9727–9736, 2022.

[172] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6964–6974, 2021.

[173] Richard L Meier. The measurement of social change. In *Papers presented at the the March 3-5, 1959, western joint computer conference*, pages 327–331, 1959.

[174] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.

[175] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

[176] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. *arXiv preprint arXiv:2307.08041*, 2023.

[177] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023.

[178] Nour Karessli, Zeynep Akata, Bernt Schiele, and Andreas Bulling. Gaze embeddings for zero-shot image classification. In *IEEE CVPR*, 2017.

[179] Youngjae Yu, Jongwook Choi, Yeonhwa Kim, Kyung Yoo, Sang-Hun Lee, and Gunhee Kim. Supervising neural attention models for video captioning by human gaze data. In *IEEE CVPR*, 2017.

[180] Sen He, Hamed R Tavakoli, Ali Borji, and Nicolas Pugeault. Human attention in image captioning: Dataset and analysis. In *ICCV*, 2019.

[181] Aidan Boyd, Kevin W Bowyer, and Adam Czajka. Human-aided saliency maps improve generalization of deep learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2735–2744, 2022.

[182] Keith Rayner, Kathryn Chace, Timothy Slattery, and Jane Ashby. Eye movements as reflections of comprehension processes in reading. *Scientific Studies of Reading - SCI STUD READ*, 07 2006.

[183] Marcel Just and Patricia A. Carpenter. A theory of reading: From eye fixations to comprehension. 1 1980.

[184] Yunfei Long, Qin Lu, Rong Xiang, Minglei Li, and Chu-Ren Huang. A cognition based attention model for sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 462–471, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[185] Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. Leveraging cognitive features for sentiment analysis. In *SIGNLL*. ACL, August 2016.

[186] Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. Harnessing cognitive features for sarcasm detection. In *ACL (Volume 1: Long Papers)*. ACL, August 2016.

[187] Maria Barrett, Frank Keller, and Anders Søgaard. Cross-lingual transfer of correlations between parts of speech and gaze features. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1330–1339, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[188] Nora Hollenstein and Ce Zhang. Entity recognition at first sight: Improving NER with eye movement information. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1–10, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[189] David Robert Reich, Paul Prasse, Chiara Tschirner, Patrick Haller, Frank Goldhammer, and Lena A. Jäger. Inferring native and non-native human reading comprehension and subjective text difficulty from scanpaths in reading. In *ETRA*. Association for Computing Machinery, 2022.

[190] Bryan Klimt and Yiming Yang. The enron corpus: A new dataset for email classification research. In *European conference on machine learning*, pages 217–226. Springer, 2004.

[191] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[192] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023.

[193] Kunchang Li, Yali Wang, Gao Peng, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. Uniformer: Unified transformer for efficient spatial-temporal representation learning. In *International Conference on Learning Representations*, 2021.

[194] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021.

[195] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

[196] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715, 2017.

[197] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018.

[198] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021.

[199] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

[200] Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. Improving sentence compression by learning to predict gaze. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1528–1533, San Diego, California, June 2016. Association for Computational Linguistics.

[201] Charles Egerton Osgood, George J Suci, and Percy H Tannenbaum. *The measurement of meaning*. Number 47. University of Illinois press, 1957.

[202] Gerald R Miller. On defining communication: Another stab. *Journal of Communication*, 1966.

[203] Denis McQuail and Sven Windahl. *Communication models for the study of mass communications*. Routledge, 2015.

[204] Steven C Bankes. Agent-based modeling: A revolution? *Proceedings of the National Academy of Sciences*, 99(suppl_3):7199–7200, 2002.

[205] Elena Romero, Manuel Chica, Sergio Damas, and William Rand. Two decades of agent-based modeling in marketing: a bibliometric analysis. *Progress in Artificial Intelligence*, pages 1–17, 2023.

[206] Joon Sung Park, Joseph C O'Brien, Carrie J Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023.

[207] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

[208] Yunhu Ye, Binyuan Hui, Min Yang, Binhua Li, Fei Huang, and Yongbin Li. Large language models are versatile decomposers: Decompose evidence and questions for table-based reasoning. *arXiv preprint arXiv:2301.13808*, 2023.

[209] Masoud Mazloom, Robert Rietveld, Stevan Rudinac, Marcel Worring, and Willemijn Van Dolen. Multimodal popularity prediction of brand-related social media posts. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 197–201, 2016.

[210] Francisco Villarroel Ordenes, Dhruv Grewal, Stephan Ludwig, Ko De Ruyter, Dominik Mahr, and Martin Wetzels. Cutting through content clutter: How speech and image acts drive consumer sharing of social media brand messages. *Journal of Consumer Research*, 45(5):988–1012, 2019.

[211] Keyan Ding, Ronggang Wang, and Shiqi Wang. Social media popularity prediction: A multiple feature fusion approach with deep neural networks. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2682–2686, 2019.

[212] Ke Wang, Mohit Bansal, and Jan-Michael Frahm. Retweet wars: Tweet popularity prediction via dynamic multimodal regression. In

2018 IEEE winter conference on applications of computer vision (WACV), pages 1842–1851. IEEE, 2018.

[213] Aditya Khosla, Atish Das Sarma, and Raffay Hamid. What makes an image popular? In *Proceedings of the 23rd international conference on World wide web*, pages 867–876, 2014.

[214] Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed Chi, and Derek Zhiyuan Cheng. Do llms understand user preferences? evaluating llms on user rating prediction. *arXiv preprint arXiv:2305.06474*, 2023.

[215] Charles Clifton Jr, Adrian Staub, and Keith Rayner. Eye movements in reading words and sentences. *Eye movements*, 2007.

[216] Vera Demberg and Frank Keller. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 2008.

[217] Barbara Plank. Keystroke dynamics as signal for shallow syntactic parsing. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[218] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[219] Jeff Wu, Long Ouyang, Daniel M Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021.

[220] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[221] Ekta Sood, Simon Tannert, Philipp Müller, and Andreas Bulling. Improving natural language processing tasks with human gaze-guided neural attention. *NeurIPS*, 2020.

[222] Sandeep Mathias, Diptesh Kanojia, Abhijit Mishra, and Pushpak Bhattacharya. A survey on using gaze behaviour for natural language processing. In Christian Bessiere, editor, *IJCAI*, 2020. Survey track.

[223] Yuqi Ren and Deyi Xiong. CogAlign: Learning to align textual neural representations to cognitive language processing signals. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3758–3769, Online, August 2021. Association for Computational Linguistics.

[224] Michalina Strzyz, David Vilares, and Carlos Gómez-Rodríguez. Towards making a dependency parser see. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1500–1506, Hong Kong, China, November 2019. Association for Computational Linguistics.

[225] Maria Barrett, Joachim Bingel, Nora Hollenstein, Marek Rei, and Anders Søgaard. Sequence classification with human attention. In *CNLL*. Association for Computational Linguistics, October 2018.

[226] Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, and Ce Zhang. Advancing nlp with cognitive language processing signals. *arXiv preprint arXiv:1904.02682*, 2019.

[227] Maria Barrett, Ana Valeria González-Garduño, Lea Frermann, and Anders Søgaard. Unsupervised induction of linguistic categories with records of reading, speaking, and writing. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2028–2038, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[228] Maria Barrett, Joachim Bingel, Frank Keller, and Anders Søgaard. Weakly supervised part-of-speech tagging using eye-tracking data. In *ACL (Volume 2: Short Papers)*. ACL, August 2016.

[229] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[230] Matthias Blohm, Glorianna Jagfeld, Ekta Sood, Xiang Yu, and Ngoc Thang Vu. Comparing attention-based convolutional and recurrent neural networks: Success and limitations in machine reading comprehension. In *CNLL*. ACL, October 2018.

[231] Menglin Xia, Ekaterina Kochmar, and Ted Briscoe. Automatic learner summary assessment for reading comprehension. In *NAACL: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, June 2019.

[232] Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, Haibin Ling, and Ruigang Yang. Salient object detection in the deep learning era: An in-depth survey. *IEEE T-PAMI*, 2021.

[233] Ralf Engbert, Antje Nuthmann, Eike M Richter, and Reinhold Kliegl. Swift: a dynamical model of saccade generation during reading. *Psychological review*, 2005.

[234] Iuliia Kotseruba and John K Tsotsos. 40 years of cognitive architectures: core cognitive abilities and practical applications. *Artificial Intelligence Review*, 2020.

[235] Marc Assens, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E. O'Connor. Pathgan: Visual scanpath prediction with generative adversarial networks. In Laura Leal-Taixé and Stefan Roth, editors, *Computer Vision – ECCV 2018 Workshops*. Springer International Publishing, 2019.

[236] Daniel Martin, Ana Serrano, Alexander W. Bergman, Gordon Wetzstein, and Belen Masia. Scangan360: A generative model of realistic scanpaths for 360° images. *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[237] Franz Matthies and Anders Søgaard. With blinkers on: Robust prediction of eye movements across readers. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 803–807, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

[238] Michael Hahn and Frank Keller. Modeling human reading with neural attention. In *EMNLP*, Austin, Texas, 2016. Association for Computational Linguistics.

[239] Shaonan Wang, Jiajun Zhang, and Chengqing Zong. Learning sentence representation with guidance of human attention. In *IJCAI*. AAAI Press, 2017.

[240] Nora Hollenstein, Federico Pirovano, Ce Zhang, Lena Jäger, and Lisa Beinborn. Multilingual language models predict human reading behavior. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 106–123, Online, June 2021. Association for Computational Linguistics.

[241] Nora Hollenstein, Emmanuele Chersoni, Cassandra L. Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus. CMCL 2021 shared task on eye-tracking prediction. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 72–78, Online, June 2021. Association for Computational Linguistics.

[242] Yevgeni Berzak, Chie Nakamura, Amelia Smith, Emily Weng, Boris Katz, Suzanne Flynn, and Roger Levy. Celer: A 365-participant corpus of eye movements in l1 and l2 english reading. *Open Mind*, 2022.

[243] Mary Ann Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Using Large Corpora*, 1994.

[244] Wanjie Sun, Zhenzhong Chen, and Feng Wu. Visual scanpath prediction using ior-roi recurrent mixture density network. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):2101–2118, 2019.

[245] Ryan Anthony Jalova de Belen, Tomasz Bednarz, and Arcot Sowmya. Scanpathnet: A recurrent mixture density network for scanpath prediction. In *IEEE CVPR*, 2022.

[246] Matthias Kümmerer and Matthias Bethge. State-of-the-art in human scanpath prediction. *arXiv preprint arXiv:2102.12239*, 2021.

[247] Ming Jiang, Xavier Boix, Gemma Roig, Juan Xu, Luc Van Gool, and Qi Zhao. Learning to predict sequences of human visual fixations. *IEEE transactions on neural networks and learning systems*, 2016.

[248] Abhijit Mishra, Diptesh Kanojia, and Pushpak Bhattacharyya. Predicting readers' sarcasm understandability by modeling gaze behavior. In *AAAI*, 2016.

[249] Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. Scanpath complexity: Modeling reading effort using gaze information. *AAAI*, Feb. 2017.

[250] Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 757–762, Beijing, China, July 2015. Association for Computational Linguistics.

[251] Zhenzhong Chen and Wanjie Sun. Scanpath prediction for visual attention using ior-roi lstm. In *IJCAI*, 2018.

[252] Matthias Kümmerer, Matthias Bethge, and Thomas SA Wallis. Deepgaze iii: Modeling free-viewing human scanpaths with deep learning. *Journal of Vision*, 2022.

[253] Halszka Jarodzka, Kenneth Holmqvist, and Marcus Nyström. A vector-based, multidimensional scanpath similarity measure. In *Proceedings of the 2010 symposium on eye-tracking research & applications*, 2010.

[254] V Levenshtein. Leveinshtein distance, 1965.

[255] Filipe Cristino, Sebastiaan Mathôt, Jan Theeuwes, and Iain D Gilchrist. Scanmatch: A novel method for comparing fixation sequences. *Behavior research methods*, 2010.