

# Spatial vs Attribute Language in Multimodal Referring Expression Grounding

## A Study on Phrase Types and Training Data Influence

Jonathan Osuji<sup>1</sup>

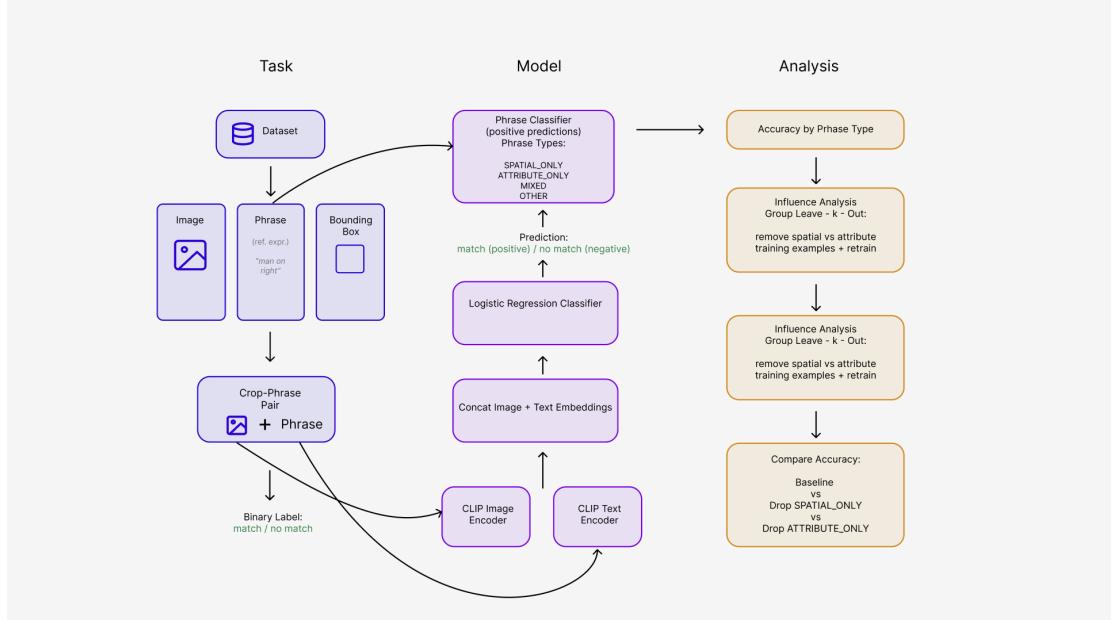
<sup>1</sup>[oosuji@sfu.ca](mailto:oosuji@sfu.ca)

## ABSTRACT

*Referring expressions are expressions in natural language used to localize a target object or person in an image. It is a simple yet very effective aspect of human communication. It allows us to remove ambiguity in describing objects in a scene. I explore the role of referring expressions, and their different phrase types in the influence of training data. Images and phrases are collected from RefCOCO-m, a refined version of RefCOCO, which aims to address poor mask quality and harmful referring expressions from the original dataset. The bounding box data is used to crop and create image-phrase pairs that are then encoded into feature vectors using contrastive language-image pre-training (CLIP). These feature vectors are then concatenated and used for classification. Phrases are categorized into (1) spatial relationships, words/phrases that refer to the ways in which different objects exist in relation to one another, (2) attributes, the quality or characteristic ascribed to something, (3) a mixture of both, and (4) others which do not fit into these categories. Mixed phrases perform best in both the classification task, and the group-based leave k-out experiments suggesting that redundant cues have noticeable influence on accuracy.*

Keywords: Referring expressions, RefCOCO/RefCOCO-m, CLIP, Spatial, Attribute

## VISUAL ABSTRACT



**Figure 1.** Visual abstract of the project. Given an image and a human-written referring expression, I construct crop–phrase pairs and train a simple binary classifier that predicts whether the crop matches the phrase. The model uses CLIP to encode the image crop and the phrase into a shared embedding space, concatenates the image and text embeddings, and feeds them into a logistic regression classifier. I then tag phrases into four types (Spatial-only, Attribute-only, Mixed, Other), compute accuracy by phrase type, and run group leave- $k$ -out experiments that remove subsets of spatial vs. attribute training examples and retrain, in order to study how different regions of the training data influence performance.

## 1 INTRODUCTION

Referring expressions are expressions in natural language used to localize a target object or person in an image. They are simple and effective for communication. It allows us to remove ambiguity in describing objects. Humans say things such as ‘the woman on the left in a gray shirt’ when they want to pick out objects, this requires the use of either spatial relations (“on the left”) or visual attributes (“woman”, “gray”, “shirt”) for comprehension of the expression. Higher prevalence of these expressions allow for easier object identification in a scene. Since the models also need to understand both spatial relations and visual attributes, understanding their differences could pose a serious challenge. For multimodal systems such as visual assistants, robots, or image search interfaces, it is important that the interpretations of these expressions are processed correctly to allow user interaction to be seamless. However, it is not obvious that all kinds of phrases are equally easy for current models, or that all types of training examples contribute to performance in the same way. I use the RefCOCO-m Moondream AI (2025) dataset, a refined version of RefCOCO which is a referring expressions dataset Yu et al. (2016). RefCOCO-m is a significantly smaller dataset than RefCOCO, this is due to its attempts to address two major issues of the original: poor mask quality and harmful expressions. The data cleaning involves replacing original instance masks with pixel-accurate masks and also removing samples from the original considered harmful. This is an important point to note as the removal of a significant portion of data could potentially render the training data unable to support some phrase types more than others.

In this project, I study referring expression grounding as a binary classification problem: given an image crop and a natural language phrase, a model determines if they match. My focus is mainly on the difference between spatial language (e.g., left/right, top/bottom) and attribute language (e.g., color, size, texture). Spatial phrases require understanding the configuration of the objects and their relative positions, while attribute phrases depend more on local visual properties like color or size. Mixed expressions like “the man on the left in a red shirt”, combine both kinds of information. Understanding model behavior on these phrase types is important to know the likelihood of it helping or failing in real world cases.

## 2 METHODS AND MATERIALS

### 2.1 Dataset and Binary Classification Setup

I use the RefCOCO-m dataset, a cleaned version of the RefCOCO UNC validation split Yu et al. (2016) with improved masks while maintaining the human-written referring expressions. The dataset was also created by removing many previous referring expressions that were deemed to be harmful, this caused a drop in the total number of sample images from 1500 to 1190. I then created a 80/20 train/val split with each example consisting of an image, referring expression, and a ground-truth bounding box that indicates the target region. To create positive examples, I pair each referring expression with its ground-truth region, while the same image paired with random region constitutes a negative example. I then use the training split to learn the classifier and the validation split for analysis and influence experiments.

### 2.2 Multimodal Features with CLIP

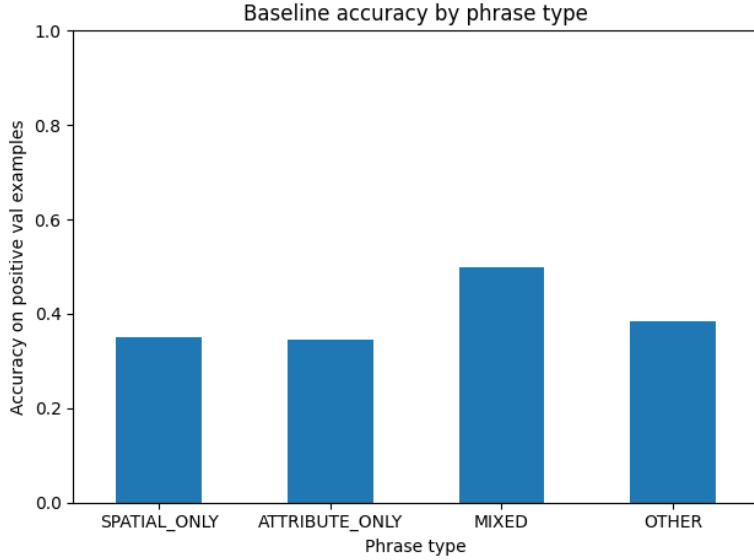
I use contrastive language-image pre-training (CLIP) Radford et al. (2021) from Hugging Face ([openai/clip-vit-base-patch32](#)) as the multimodal encoder OpenAI (2021). I first crop the image using its associated bounding box data and pass that through the image encoder, then I pass the matching referring expression into CLIP’s text encoder. This produces a crop image, phrase pair which I then concatenate into a single feature vector and use as input for a downstream classifier. I do not perform finetuning on the model, instead, I compare the performance of different classifiers on the dataset. This was mainly to keep the experiment simple and make the analysis easily interpretable.

### 2.3 Classifier Comparison: What I Tried

On top of the CLIP features, I train three standard classifiers: logistic regression, random forest classification, and HistGradientBoostingClassifier using scikit-learn Pedregosa et al. (2011). For Logistic regression, setting a max iteration of 1000, I was able to achieve training and validation accuracies of 0.767 and 0.747 respectively. The random forest classifier with 200 decision trees and a depth of 20 achieved train and validation accuracies of 1.0 and 0.665 respectively. The gradient boost classifier had a maximum iteration of 200, depth of 12, and a learning rate of 0.1, this produced a training and validation accuracy of 1.0 and 0.667 respectively. The logistic regression classifier performed the best which indicates that its generalization was optimal. While the random forest and gradient boost both had higher training accuracies, with gradient boost having a marginally better validation accuracy, they performed worse on the validation set which indicates that they were strongly overfitting for the training set. Based on these results, I select the logistic regression as the main model for all further analyses.

### 2.4 Phrase-Type Tagging: Spatial, Attribute, Mixed, Other

To connect model behavior to linguistic structure, I implement a simple rule-based phrase-type tagger. The phrase tagger iterates over each referring expression and places the detected phrase(s) into one of four categories: Spatial-only, which consists of phrases used to describe object positions in relation to other objects (e.g., “left”, “right”, “top”, “bottom”, etc.,); Attribute-only, which describes objects based on their explicit characteristics (e.g., “red”, “tall”, “striped”, etc.,); Mixed, a combination of both categories; and Other, which constitutes of any phrase not belonging to the prior categories. The initial implementation of the script seemed to be too brittle as it caused approximately 91 percent of phrases to be tagged as other. Spatial-only phrases were the most affected, suggesting that the initial rules were probably too narrow and literal, causing many spatial phrases to be mistagged. The initial script attempted to match exact multi-word strings, (e.g., “on the left”, “in front of”) but the dataset also contained phrases such as “man on right”, or “woman in middle right” which are clearly spatial but do not contain the exact substring in the spatial category. I decided to implement single-word special tokens, and multi-word special expressions as subcategories. The single-word special tokens allowed for expressions with just one spatial phrase to still be tagged, while the multi-word spatial expressions allowed the script to be more robust at catching common phrases. This change significantly improved the spread of the data across categories. One other issue I encountered involved the creation of the categories. Since there is not definitive list indicating all spatial words and phrases, I had to improvise by writing them down from memory, this leaves the results open to scrutiny as performance could potentially improve with better refined spatial and attribute categories.



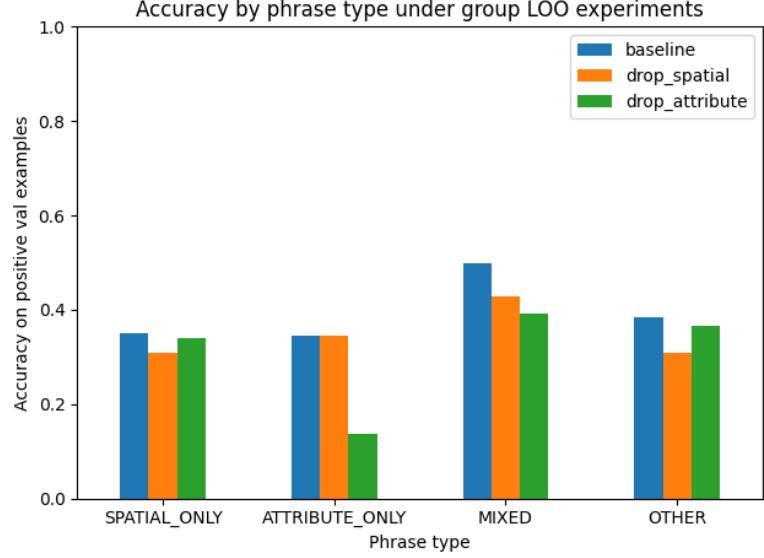
**Figure 2.** Baseline Accuracy by Phrase Type

## 2.5 Group Leave-k-Out Influence Experiments

To determine the training data influence, I establish three training conditions, (1) a baseline model on the full training set, (2) drop spatial only phrases, (3) drop attribute only phrases. For all training conditions, I first retrain the dataset using logistic regression and then applying a fit to the result. Then I use the predict() function to predict the labels of the data values for positive examples in the validation split. For the baseline model, I observed an accuracy of 0.38 on all positive validation examples, accuracy by phrase type was also observed with 0.34, 0.5, 0.38, and 0.35 for attribute, mixed, other, and spatial phrases respectively. For the spatial drop, I randomly remove 50 of the 471 spatial phrases and observe an accuracy of 0.33, and phrase type accuracies of 0.34, 0.43, 0.31, and 0.31 in similar order to the baseline. 50 examples were also removed from the 88 attribute phrases at random for an accuracy of 0.33 and phrase type accuracies of 0.14, 0.39, 0.37, and 0.34 in similar order to the baseline as well. While an accuracy drop in spatial-only and mixed phrases was expected for the spatial drop training, I also observed that phrases in the 'other' category had a more significant drop in accuracy (0.38 to 0.31) which supports the previous suggestion that performance is bottle-necked by the lack of a more refined spatial category. For the attribute drop training condition, there was a significant drop in accuracy for attribute-only phrases, which is most likely a result of dropping 50 of the 88 attribute phrases from the training set. Outside of the expected drop in accuracy for attribute-only and mixed phrases, there was only a slight drop in accuracy for other phrases, also indicating a need for a more refined attribute category.

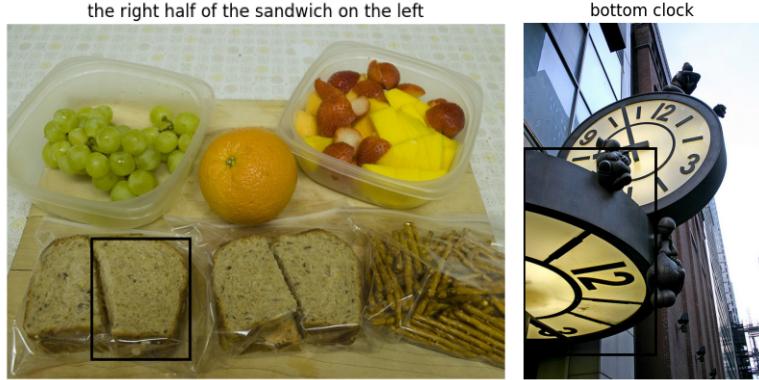
## 2.6 Qualitative Examples

After observing the results from the influence experiment, I extracted various examples from the dataset to further analyze any possible reasons for the accuracy scores. I collected one image example each of positive and negative examples for both spatial-only and attribute-only phrase types. What I found was that from visually examining the images, there were examples that were clearly positive matches between the phrase and image but were incorrectly flagged as negative matches by CLIP. When looking at the correctly flagged spatial example, the model correctly identifies the right half of the sandwich, which requires combining two spatial relations ("right half" and "on the left"). So this shows that, at least in some cases, the classifier can handle layered spatial descriptions when the layout is like fairly clear. For the incorrectly flagged spatial phrase, you can see that the bounding box actually matches the description ("bottom clock") but the model predicted a non-match. This is likely due to the classifier struggling to process simple vertical relations like "top" vs "bottom". On the attribute side, the bounding box shows a small dish containing some form of brown spice which somewhat matches the phrase ("the little dish with brown"), but the model predicts a non-match. This indicates that more subtle attribute descriptions (small size and a less salient color) are harder for the classifier, even when the match is obvious to a human.



**Figure 3.** Group Leave-k-Out Accuracy by Phrase Type

observer. For the positive attribute phrase, the model correctly matches the phrase "red tie" to a man in a suit with a red tie, this color attribute is more clear, indicating that CLIP's visual features, combined with a simple classifier are capable of capturing straightforward attributes when they are clear in the image crop.



**Figure 4.** Referring expressions from the RefCOCO dataset Yu et al. (2016) based on images from MS COCO Lin et al. (2014) showing Spatial Examples of Positive match (left) and Negative match (right)

## 3 DISCUSSION

### 3.1 Summary of Findings

The experiments show that the performance of a simple CLIP-based classifier results in unevenness across phrase types. Mixed phrases that consist of both spatial and attribute cues showed the strongest performance. Purely spatial and Purely attribute phrases performed the weakest with only, even more than phrases in the other category. This showed only a moderate positive-only phrase accuracy. The group leave-k-out experiments showed that removing spatial-only phrases negatively affected mixed, other, and spatial-only phrases, but left the accuracy of attribute-only phrases unchanged. Removing a similar sized subset of attribute-only phrases showed a more significant negative effect on attribute-only accuracy, but had a similar but slightly weaker effect on accuracy for mixed, spatial-only, and other phrase types. Further analysis of the image-phrase examples by manually inspecting some samples also showed the volatility in CLIP's classification accuracy.



**Figure 5.** Referring exressions from the RefCOCO dataset Yu et al. (2016) based on images from MS COCO Lin et al. (2014) showing Attribute Examples of Positive match (left) and Negative match (right)

### 3.2 What Worked and What Did Not

On the modeling side, keeping CLIP frozen and using a simple logistic regression head worked well: it provided strong baseline features and a classifier that generalized better than more complex tree-based models. While the rule-based phrasal tagging is unrefined, it provided meaningful differences across spatial, attribute, mixed, and other expressions. The group leave-k-out design also worked as a practical way to determine influence values without having to resort to much more computationally expensive methods. On the other hand, random forests and gradient boosting, which might have seemed attractive due to their flexibility, mostly overfit the CLIP features and offered no gains in validation accuracy, although adjusting the parameters could potentially improve their accuracies. Some qualitative failures were also hard to interpret; in several cases the crop clearly matched the phrase to a human observer, yet the model predicted non-match, making it difficult to pinpoint whether the issue lay in the visual embedding, the text embedding, or the classifier. I hypothesized whether the quality of the images impacts performance, for example, in the clock image, the clock in the bounding box is facing the floor but the classifier might not be detecting it as a clock since the orientation is atypical for a clock. More research would need to be conducted to form a conclusive opinion.

### 3.3 Limitations

This study has several limitations. First, it operates on a single split of RefCOCO-m and a relatively small number of CLIP-based training examples, this limits the amount of conclusions that can be drawn from the dataset. Second, the CLIP model is used as a fixed backbone; I do not explore fine-tuning or more expressive cross-attention architectures that might better capture spatial relationships and attributes. Third, the phrase-type tagger is purely rule-based and relies on simple keyword lists, so mislabeling of complex phrases is inevitable. Fourth, the leave-k-out experiment uses a fixed K=50 for both spatial and attribute positives, this is problematic since the dataset does not have an equal spread of phrase types (attribute contains only 88 phrases).

### 3.4 Future Work

Future work could address some of these limitations in several ways. Regarding the dataset, RefCOCO-g could be used in place of RefCOCO-m, this is because it provides a much larger dataset with a train/val/test split of over 20,000 images and 40,000 referring expressions Mao et al. (2016). RefCOCO-g also has significantly higher words per expression which allows for more rich expressions of objects. Regarding rule-based phrasal tagging, curating user-generated data could help refine the categories and also improve the robustness of the system. For CLIP-based classification, a more customized model such as RefCLIP, a universal weakly supervised teacher which is specifically tasked with referring expression comprehension Jin et al. (2023), could potentially reduce volatility in multimodal classification tasks. If these limitations are addressed, an area of future implementation I would like to see this work be directed to is visually-

assisted navigation specifically targeted at those with visual impairments. This would contribute greatly towards furthering the development of Human-Centered AI.

## 4 CONNECTION TO COURSE THEMES

This project connects directly to themes of data-centric and human-centered AI. Rather than treating the dataset as a fixed, neutral resource, I explicitly examine how different slices of the training data, such as spatial versus attribute referring expressions shape model behavior. The group leave-k-out experiments allow me to perform data attribution and also show that the model's performance on attribute-only phrases is extremely sensitive to the lack of attribute training examples, while the performance of spatial accuracy depends more on spatial examples and mixed phrases. This kind of analysis aligns with data-centric AI's emphasis on understanding and improving the data that models learn from. The use of group leave-k-out experiments also provides hand-on experience with learning about the the complexities of retraining-based data values.

This project also allowed me learn more directly the importance for reliable, safe, and trustworthy (RST) systems. When researching RefCOCO, I learned that it contained some harmful expressions that were directly targeting people in the images, and that RefCOCO-m was created to address that issue. The process of being able to audit the dataset and make improvements to it is evidence of the importance of open source software (OSS) in AI. Influence analysis on the dataset showed that not all phrase types are equally represented. The model is more reliable for some referring expressions than others, this is a data-centric issue. It showed that expressions assumed to be most common in a practical setting are also some of the most volatile. This suggests that auditing and potentially adapting the human-controlled systems employed in collecting the data is needed. Overall, the project showed how relatively simple experiments such as tagging human-written phrases and selectively removing training examples can help to determine where a multimodal system could fail its users and how making changes to the training data could change the behavior.

## REFERENCES

- Jin, L., Luo, G., Zhou, Y., Sun, X., Jiang, G., Shu, A., and Ji, R. (2023). Refclip: A universal teacher for weakly supervised referring expression comprehension. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2681–2690.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *ECCV*.
- Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A., and Murphy, K. (2016). Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11–20.
- Moondream AI (2025). Refcoco-m: Cleaned refcoco (unc) validation split. <https://huggingface.co/datasets/moondream/refcoco-m>. Hugging Face dataset.
- OpenAI (2021). Clip vit-b/32. <https://huggingface.co/openai/clip-vit-base-patch32>. Hugging Face model.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*.
- Yu, L., Poirson, P., Yang, S., Berg, A. C., and Berg, T. L. (2016). Modeling context in referring expressions. In *European Conference on Computer Vision (ECCV)*.