ID	Method	Training Mode	Accuracy (∼ Rank@1)	Rank@5	Example-F1	F1-Score
			Location Reasoning			
1	ResNet-50 (He et al., 2016)	Supervised	3.18%	9.82%	22.19%	2.27%
2	Swin-T (Liu et al., 2021)	Supervised	6.70%	17.07%	33.56%	5.02%
3	CLIP (Radford et al., 2021)	Zero-Shot	11.11%	27.85%	44.96%	9.74%
4	CLIP† (Fu et al., 2022)	Fine-tune	15.72%	37.13%	49.74%	13.82%
5	CLIP+Seg (Fu et al., 2022)	Fine-tune	16.46%	37.48%	50.52%	14.63%
6	**QR-CLIP** (Ours)	Fine-tune	**19.31%**	**38.78%**	**50.96%**	**17.70%**
Improvements (AVG: 10.66%)			*+17.31%*	*+3.47%*	*+0.87%*	*+20.98%*
			Time Reasoning			
7	ResNet-50 (He et al., 2016)	Supervised	0.84%	5.14%	39.99%	0.46%
8	Swin-T (Liu et al., 2021)	Supervised	0.97%	5.53%	43.95%	0.72%
9	CLIP (Radford et al., 2021)	Zero-Shot	0.46%	2.42%	39.90%	0.25%
10	CLIP† (Fu et al., 2022)	Fine-tune	1.00%	3.07%	43.09%	0.54%
11	CLIP+Seg (Fu et al., 2022)	Fine-tune	0.92%	3.15%	42.89%	0.71%
12	**QR-CLIP** (Ours)	Fine-tune	**3.53%**	**10.90%**	**47.89%**	**2.01%**
Improvements (AVG: 134.38%)			*+253%*	*+97.11%*	*+8.23%*	*+179.17%*

Table 1. Summary of the performance for different baselines on the image location and time prediction. † means fine-tune the original CLIP (Radford et al., 2021). '*AVG*': average relative lift.

W_i^{owk} and W_i^v are the weights of the $\texttt{CLS}]_i^{owk}$ and $[\texttt{CLS}]_i^v$; q in this place is the addition of weight vision-language features $W_i^{owk} \times [\texttt{CLS}]_i^{owk} + W_i^v \times [\texttt{CLS}]_i^v$; q is the ground-truth features generated by $F^{GT} = \text{Enc}_t(GT)$.

Location and Time Reasoning. We use the fused features $F^{fused} = \sum_1^6 (W_i^{owk} \times [\texttt{CLS}]_i^{owk} + W_i^v \times [\texttt{CLS}]_i^v)$ as our final features to predict the location and time. The prediction is completed by calculating the similarity between F^{fused} and the candidate location/time embeddings.

We believe that by using the CLIP pre-trained 400M open-world corpus and then fine-tuning it by adding additional `[CLS]` with location-and-time-specific data, it can basically reason about meta information. QR-CLIP will then improve its performance by retrieving valuable open-world knowledge and using it as auxiliary cues. Finally, the model balances vision and language embeddings, and by incorporating them into prediction, the model achieves its peak performance. The process is related to Horn's QR rule (Horn, 1984). Also, it mimics a procedure of information spreading (Wang et al., 2011): diverse individuals have diverse perspectives and attitudes regarding the same thing (sec 3.2), but combining them effectively fosters a more profound comprehension (sec 3.3).

4. Experiments

4.1. Training Settings

Dataset. We used two datasets: TARA dataset (Fu et al., 2022) and our collected OWK dataset. **TARA dataset** includes 15,429 samples. Each sample contains a news picture and the corresponding location, time description. Following the original setup, we train QR-CLIP on a train set containing 12,306 instances and evaluate our method using a test set containing 1,644 instances. The **OWK dataset** is derived from the WIT dataset (Srinivasan et al., 2021). Considering the limited computation resource, we only use 122,408 texts from the 37.5 million entity-rich image-text examples in English Wikipedia that correspond to the countries and years as our open-world knowledge.

Evaluation Metrics. For a fair comparison, we first follow the same evaluation metrics on the TARA benchmark (Fu et al., 2022): Accuracy and Example-F1. Accuracy is calculated by comparing the predicted results with the entire labels. Example-F1 is calculated by comparing predictions with hierarchical labels:

$$\text{Example-F1} = \frac{1}{N} \sum_{i=1}^{N} \frac{2|\text{GT}_i \cap \text{Pred}_i|}{|\text{GT}_i| + |\text{Pred}_i|}, \tag{8}$$

where GT_i represents the hierarchical label, and Pred_i represents the hierarchical prediction. If the entire label is {*'Zurich, Switzerland, Europe'*}, the progressive hierarchical labels are the three combinations of true label as {*'Zurich, Switzerland, Europe'*}, {*'Switzerland, Europe'5948*} and {*'Europe'*}. In addition, Rank@5 and F1-Score are utilized to evaluate the performance of the proposed method.

Implementation Details. QR-CLIP is based on CLIP+VIT-B/32 model with an input size of 224×224, and it is implemented on the PyTorch 1.10.1 platform with the Adam optimizer to update the neural network's weights and biases. The training batch size is 32, and the initial learning rate is $1e-6$. Our model utilizes a pre-trained model and takes hours in the fine-tune process on an NVIDIA RTX 3090 GPU running CUDA 11.7.1.