Wiki-VEL: Visual Entity Linking for Structured Data on Wikimedia Commons

Philipp Bielefeld[*1], Jasmin Geppert[*1], Necdet Güven[*1], Melna Treesa John[*1], Adrian Ziupka[*1], Lucie-Aimée Kaffee[2], Russa Biswas[3], Gerard de Melo [1]

[1]Hasso Plattner Institut / University of Potsdam, Potsdam, Germany,
[2]Hugging Face, [3]Aalborg University, Copenhagen, Denmark
lucie.kaffee@huggingface.co, rubi@cs.aau.dk, gdm@demelo.org

Abstract

Describing images using structured data enables a wide range of automation tasks, such as search and organization, as well as downstream tasks, such as labeling images or training machine learning models. However, there is currently a lack of structured data labels for large image repositories such as Wikimedia Commons. To close this gap, we propose the task of *Visual Entity Linking (VEL) for Wikimedia Commons*, which involves predicting labels for Wikimedia Commons images based on Wikidata items as the label inventory. We create a novel dataset leveraging community-created structured data on Wikimedia Commons. Additionally, we fine-tune pre-trained models based on the CLIP architecture using this dataset. Although the best-performing models show promising results, the study also identifies key challenges of the data and the task.

1 Introduction

Wikimedia Commons is a service that hosts around 100 million community-contributed, openly licensed images and media files, including metadata, multilingual textual descriptions, and categories similar to Wikipedia categories. At the same time, Wikimedia's Knowledge Graph (KG), *Wikidata*, offers detailed structured knowledge descriptions of over 100 million entities. In 2017, the *Commons: Structured Data* project was initiated to organize and search images by better connecting the two efforts. Community members tag relevant Wikidata items in images, adding them to Commons as structured data via new `depict` statements, enabling machine-friendly association of images with universal, language-independent concepts. In Wikimedia Commons, structured data unlocks the full potential of its image repository, providing users with a more enriching and productive experience. Yet, as of November 2023, only 15% of Wikimedia Commons images are accompanied by structured data, suggesting that a considerable portion of this vast resource remains unexplored. This lack of structured data poses a challenge for users seeking to extract meaningful information from the extensive collection. Structured data is crucial for modern information retrieval systems, providing a systematic framework for describing entities and their attributes, and enhancing discoverability and interoperability across platforms and applications.

This gap in the coverage of Commons image annotations can be addressed by automatically suggesting depicted items using ***Visual Entity Linking (VEL)***, a multi-modal task of linking visual items in an image with corresponding entities in a KG.

This paper proposes the ***Wiki-VEL*** framework, applying the task of VEL to Wikimedia Commons using the structured data of Wikidata. This allows users to perform targeted searches and explore images based on specific topics, events, or attributes, enhancing the usability and utility of Wikimedia Commons. Further, integrating VEL on Wikimedia Commons opens opportunities for automation and innovation in content management and analysis. Images annotated with structured data can be used for visual question answering, search algorithms, image classification, object detection, semantic segmentation, and recommender systems (de Melo and Tandon, 2016; Shutova et al., 2015; Li et al., 2017). Our contributions are as follows:

- A novel image dataset[1] for Visual Entity Linking extracted from Wikimedia Commons.
- A framework for Visual Entity Linking (Wiki-VEL) connecting entities in the images of Wikimedia Commons with the KG, Wikidata.
- Human evaluation of Wiki-VEL annotations.

[*]In alphabetical order, as these authors contributed equally to this work.

[1]https://huggingface.co/datasets/aiintelligentsystems/vel_commons_wikidata

Proceedings of the 3rd Workshop on Advances in Language and Vision Research, pages 186–194

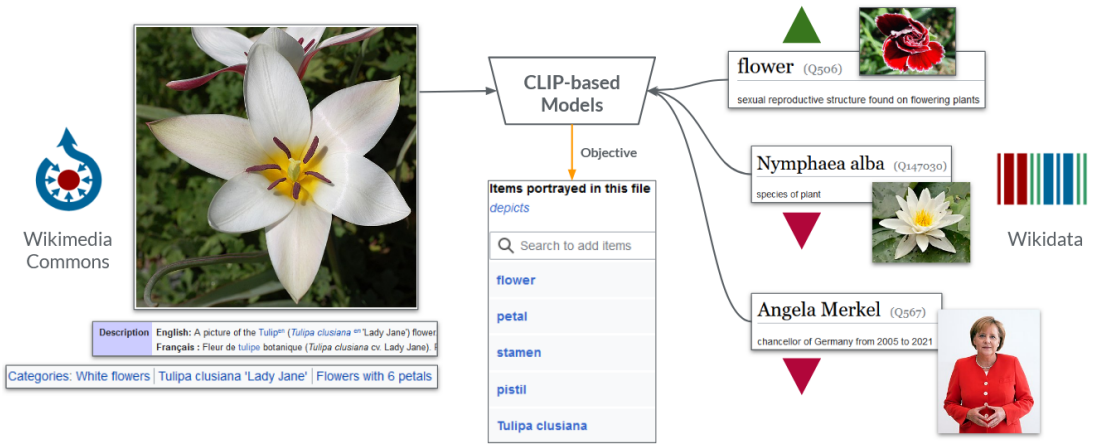


Figure 1: Overview of the Wiki-VEL framework.

2 Related Work

2.1 Visual Entity Linking

Visual Entity Linking (VEL; Weegar et al., 2014; Tilak et al., 2017) is the task of linking entities detected in images to their corresponding entities in a KG. VEL works across different modalities: the images that entities are detected in, the labels of entities in a KG, and the KG entities themselves. Some studies (Müller-Budack et al., 2021; Gan et al., 2021; Dost et al., 2020) focus on coarse-grained entity linking of items in the images to a KG by leveraging the entity mentions in the corresponding textual information. Recent years have also witnessed entity linking models that use visual information to identify entity mentions in social media texts (Moon et al., 2018; Adjali et al., 2020; Zhang et al., 2021; Lu et al., 2018; Biten et al., 2019). Wang et al. (2022) propose a multimodal entity linking dataset based on Wikipedia, emphasizing text input as the primary component, complemented by visual input. However, the entity types are limited to only persons and organisations.

Sun et al. (2022) aim to link the visual mention in the image with the entire image as the context to the corresponding named entity in KGs without textual descriptions. This model focuses mostly on images of persons. For this, they create a human-annotated dataset and then finetune a variety of models that are partly based on CLIP, adding output heads on top of pre-trained models such as CLIP to obtain more task-specific features.

In their *OVEN* task, Hu et al. (2023) aim to link over 6 million open domain images to (English) Wikipedia, given also a natural language question as input. They, too, finetune composed models with CLIP as a backbone along with another much larger multimodal model named PaLI (Chen et al., 2023), and achieve state-of-the-art results on visually-situated text understanding and object localization tasks.

The contributions of this paper close a gap in the aforementioned efforts: Given our goal of applying the VEL process on Wikimedia Commons, we do not provide additional textual queries as input, as we would not have a source for them on Commons. Instead, we aim to predict the depicted items only from the image itself, which gives rise to a multi-label problem, i.e., multiple entities depicted in one image. Additionally, we are not limited to a certain group of items but seek to provide a domain-independent solution. This combination makes it a very difficult problem worth exploring.

2.2 Pre-trained CLIP

Modern deep learning models, such as ResNet-50 (He et al., 2015), excel in computer vision tasks such as image classification, achieving an accuracy of around 80% across the 1,000 different pre-defined candidate classes of the ImageNet dataset. OpenAI introduced Contrastive Language-Image Pre-training (CLIP; Radford et al., 2021) to address the limitation of being confined to predefined classes. CLIP is a multimodal model trained to map images and natural language text to high-dimensional embeddings close to each other based on cosine similarity. It uses two separate encoders for image and text input, allowing for inference comparing image embeddings against text embeddings of freely chosen labels. Different experimental settings for CLIP are investigated in depth by Shen et al. (2021) and Gao et al. (2024).

3 Dataset

On Wikimedia Commons, the community contributes meta-data for the uploaded images. This includes descriptions and licensing information as well as structured data in the form of Wikidata items. In our work, we focus on the structured data describing entities in the images. To express this relation, the property `depicts` is used on Wikimedia Commons.

3.1 Collection

Wikimedia regularly publishes database dumps of its projects[2], including Commons structured data and Wikidata entities. These dumps[3] are used in this project, providing basic information on all Commons images, descriptions and categories, and labels for all current Wikidata items. The advantage of using dumps is that they only need to be downloaded once, and all processing can be done offline afterwards. The following initial data preprocessing is employed to extract relevant information for the dataset:

- For Commons images, we only retain the unique image ID (Commons page ID), description, categories, and list of depicted items. Images without any annotated depicted item are discarded completely at this stage. Also, we only consider images with the (case-insensitive) file name extensions *.jpg*, *.jpeg*, *.png*, and *.svg*.
- For Wikidata, we only keep the unique item ID (known as *QID*), label (short descriptive name), and description. Along with these, the ID of the first linked image from the *image* property (if any) is saved. Items that are never annotated as *depicted* across the entire Commons dump are discarded completely at this stage.

We employ a heuristic filtering strategy to retain only commonly depictable items in the Wikidata dumps, removing other items such as scholarly articles or metadata items. This further ensures that all textual input is in English. Commons categories are assumed to be in English but are filtered to only include categories descriptive of the image. For example, categories merely relating to specific users or upload dates are eliminated via simple pattern matching, using patterns such as *User:* or *Photographs by:*.

[2] `https://dumps.wikimedia.org/commonswiki/`

[3] extracted as of November 7, 2023

Additional information on the `depicts` statements such as the prominence flag or item qualifiers (e.g., *"color: blue"*) is omitted.

A data structure is built while parsing Wikidata to capture the item hierarchy according to Wikidata's *subclass of* and *instance of* properties. This allows for the association of items of differing granularity, as subclassed or instantiated items can also be considered as their respective superclass(es). The data structure is a mapping of an item's QID to all QIDs of its superclasses, for different numbers of hops (for up to three hops).

3.2 Hierarchy-Aware Item Filtering

The distribution of `depicts` annotations across all 2.3 million items is severely skewed, as shown in Figure 3a, with around 50% occurring merely once as ground-truth, and 90% occurring fewer than ten times. This suggests poor model performance on rare items among the large pool of candidates. To mitigate this, we promote the long-tail items to more frequent and generic Wikidata items using Wikidata's class hierarchy and a threshold f. This filtering removes items depicted fewer than f times in the training split generated from the intermediate data. However, item appearances accumulate across three hops in the Wikidata hierarchy, potentially affecting generic items. This accumulation is relevant for more generic Wikidata items such as *human*, for which specific people are often annotated using the `depicts` statement, but rarely annotated as *human*.

To adjust the images' ground-truth, we check if every original ground-truth item fulfils the threshold. If so, it is kept, otherwise, we probe the KG hierarchy for more generic substitute items. Once one or more items fulfil f, they are taken as replacements for the original ground-truth item. To retain as many images as possible and their distribution, an image is only discarded if no replacement item can be found within three hops.

3.3 Experimental Dataset

In the following experiments, we use a dataset consisting of 1 million Commons images. It is created by randomly shuffling the order of the intermediate file to eliminate biases such as by upload date or batch uploads. The dataset is split into 80% training, 10% validation, and 10% testing splits, as shown in Table 1. It also illustrates that the number of rows for train and validation splits is higher than the number of images, as many images have multi-

	f = 0	f = 10
#images train	800,000	800,000
(#rows)	(1,377,684)	(1,498,026)
(#gt_items)	(490,876)	(17,287)
#images validation	100,000	100,000
(#rows)	(195,535)	(212,885)
(#gt_items)	(72,055)	(14,253)
#images test	100,000	100,000
(#rows)	(100,000)	(100,000)
(#gt_items)	(72,271)	(14,351)
#Wikidata items	2,305,611	18,522

Table 1: Statistics of the Experimental Dataset. #rows = no. of labels available for the images, #gt_items = no. of unique Wikidata items as ground-truth labels.

ple ground-truth labels, which are used in the experiments for training and validation mini-batches. Most experiments use an item frequency of $f = 10$. Figure 2 and Table 2 show the item super-category distributions and most frequent items in the entire dataset for $f = 0$ and $f = 10$.

The super-categories are arbitrary selections of generic classes an item can belong to, inferred from the Wikidata dump by certain properties. Figure 2 shows many items that depict humans, animals, plants, or natural objects. Following the skewed distribution as illustrated in Figure 3a, the most frequent items in the train split without applying a threshold are highly overrepresented and fairly generic, as shown in Table 2.

With a threshold of $f = 10$, we have 18,522 items left that are depicted often enough in the train split. Still, only 6,034 images were discarded because of lacking suitable ground-truth items, showing that the KG hierarchy helps in retaining most images. Overall, with one ground-truth item per datapoint, there are about 1.5 million train datapoints and 213,000 validation datapoints, averaging roughly two ground-truth items per image.

This also causes the super-category distribution in the dataset to change, with *human* becoming the most frequent item and *painting* or *taxon* being assigned to specific paintings or species. As shown in Figure 3b, every remaining item occurs ten times or more (across three hops) in the training dataset. This implies that a few items are highly overrepresented among candidates (see Table 2). Instead of balancing the frequencies in the dataset, this work intends to produce a dataset that is a reasonably representative sample of all Commons images. This

f=0		f=10	
Label	Freq.	Label	Freq.
road	34,615	human	119,233
village	16,186	painting	55,213
agriculture	16,117	taxon	44,461
path	15,601	village	37,040
house	14,943	road	36,159

Table 2: Most frequent items in the training split.

approach allows fine-tuned models to work well on the generality of Commons images, rather than ensuring similar performance across all depictable items, many of which are very rare. Therefore, the experiments conducted in this work are on an imbalanced real-world dataset.

3.4 Challenges

While preparing the dataset, we identified the following challenges:

Depicts statements. The guidelines for the `depicts` statement, as many community guidelines, vary across the project, e.g., sometimes suggesting not to add generic items if more specific ones are already marked[4], while with others the recommendation is to add *both* generic and specific items.[5] Therefore, different images with similar content might be annotated differently.

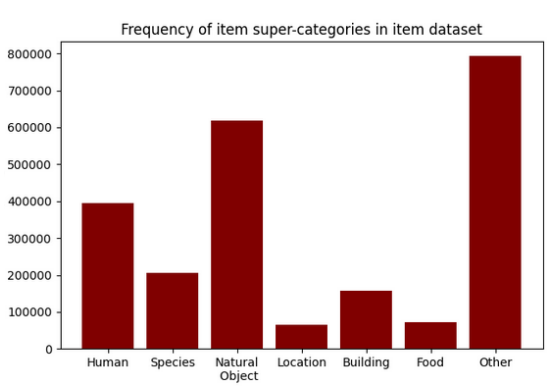
Depicted items. The number of items marked in images on Commons varies considerably, as shown in Figure 4a, due to differing understanding of the guidelines on adding `depicts` statements. Figure 4b contrasts two images that both have *tree* marked, while the red house in the background is very prominent. This inconsistency in ground-truth data can lead to inconsistencies in the diversity of images, making it difficult for models to predict the correct items accurately.

Specific items. Even after filtering with our threshold of 10, there are items that appear overly specific. For example, the item *Flintenweg 8, Orvelte* (Q17447776) is still present in the dataset, as relatively many images are annotated with this item despite it not even having a description on Wikidata.

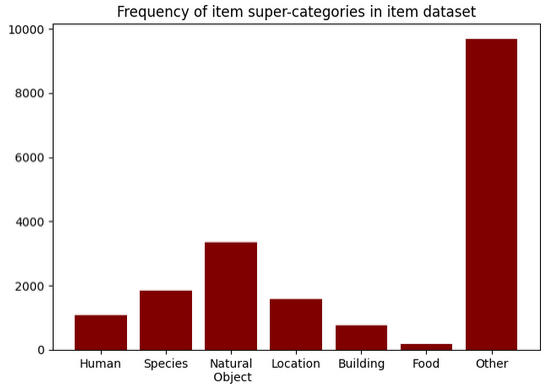
Similar and Dissimilar Items. The KG hierar-

[4] https://commons.wikimedia.org/wiki/Commons:Depicts#What_items_not_to_add

[5] https://commons.wikimedia.org/wiki/Commons:Depiction_guidelines#Depicts_level_of_detail (marked as disputed)

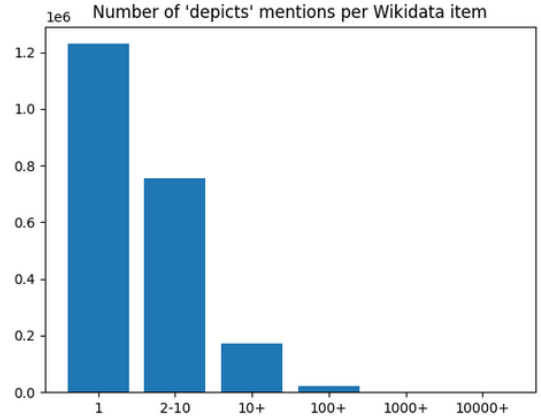


(a) $f = 0$: 2.3 million items

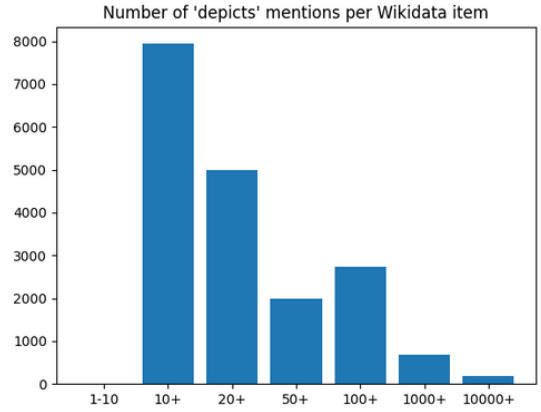


(b) $f = 10$: 18,522 items

Figure 2: Distribution of inferred item super-categories.



(a) $f = 0$: 2.3 million items (no hops)



(b) $f = 10$: 18,522 items (over three hops)

Figure 3: Number of `depicts` mentions across items.

QID	Label
Q466066	BMW Series 3
Q608824	BMW Series 3
Q730915	BMW Series 3
Q756792	BMW Series 3 (E46)
Q838837	BMW Series 3

Table 3: Excerpt of highly similar items.

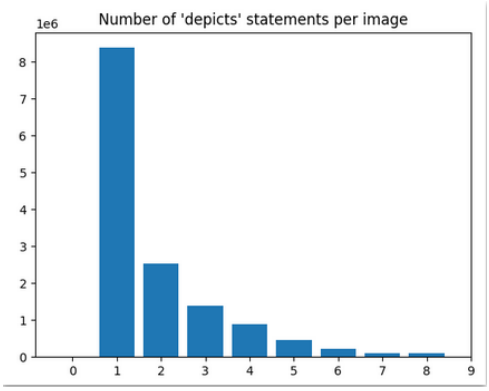
chy captures candidate items of varying granularity, while multiple items with QIDs and statements share labels and descriptions. For example, Table 3 lists an excerpt of Wikidata items related to the same car model series. However, there are many near-identical items, describing similar concepts with different labels.

4 Experimental Setup

In the following, we describe the CLIP variants used in the proposed Wiki-VEL framework to link the WikiCommons images to Wikidata entities.

4.1 Naive CLIP

Our Naive CLIP model (see Figure 5) is a multimodal approach to the VEL task, leveraging the CLIP's image encoder for the Wikimedia Commons images and each item's label concatenated with its description is passed through CLIP's text encoder. The resulting image and text embeddings are then normalized and compared by their cosine similarity to determine a relevance score. Additional multi-layer perceptron (MLP) heads are added to the image and text encoders to adjust the semantically rich CLIP features to the task. Each MLP head consists of a linear layer of double the input dimensionality, followed by a ReLU activation function, a dropout layer with probability 0.5, and a final linear layer mapping back to the input dimensionality. A residual connection is added to the CLIP embeddings to facilitate training. This model is named Naive CLIP because it does not utilize all



(a) Greatly varying number of `depicts` statements.

(b) Contrary ways on how to add *depicts* statements.

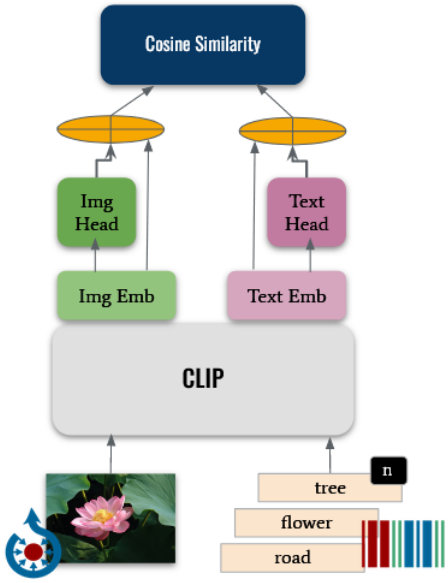


Figure 5: Architecture of the Naive CLIP model.

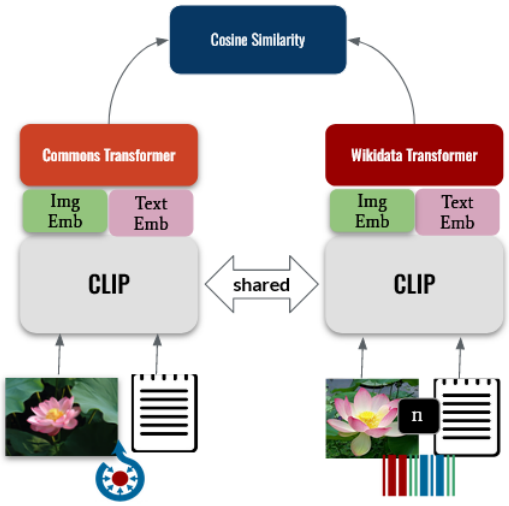


Figure 6: Architecture of the CLIP Fusion model.

available information, such as image descriptions and categories.

4.2 CLIP Fusion

The CLIP Fusion architecture (Hu et al., 2023) uses two separate encoders for the query and the entity, each relying on a CLIP backbone for image and text embeddings. A transformer-model head outputs a single embedding per encoder, which can be matched against each other. We adopt this architecture, with the CLIP backbone shared by both encoders, referred to as Commons encoder and Wikidata encoder, as shown in Figure 6. For the Commons encoder, the Wikimedia Commons image description and categories are concatenated to form the textual input. In the Wikidata encoder, in addition to the Wikidata labels, we use their item images. Since Wikidata item images also come from Commons, there is a risk that item images could be part of the test dataset. To avoid leaking test data, we removed these images from the test dataset, which was the case for 74 items in the $f = 10$ dataset.

4.3 Loss Targets

The in-batch contrastive loss function of CLIP's pre-training assumes all matching pairs (the loss targets) of images and texts to lie on the diagonal of the input matrix. The composed loss function is: $0.5 \times$ (image_to_text_loss + text_to_image_loss), where both the individual loss functions are the cross-entropy loss. We aim to relax the diagonal requirement by allowing all combinations of images to be set as loss targets. This means that the same Wikidata item can be depicted in multiple images and potentially multiple ground-truth items. This means that each batch must determine the corresponding matches before setting them as equally weighted loss targets. Our method also allows the loss targets to be dependent on the number of hops

between a ground-truth item and another item in the batch, using the Wikidata item hierarchy. This is to force the model to move embeddings of related specific and generic items closer together.

4.4 Experimental Details

The models described in Section 4 are trained with two validation loops per epoch and early stopping before evaluation on the test split. The optimal hyperparameters for our model include a learning rate of 0.001, a batch size of 1,024, and AdamW optimization. We also rescale item gradients by inverse batch frequency and set one loss target hops. The inverse batch frequency accounts for the fact that Wikidata items like *human* occur frequently. The train split contains 800,000 small images, which creates a massive IO overhead during finetuning. However, at the cost of increased memory usage, latency is reduced, speeding up finetuning. Due to resource limitations, the experiments use *ViT-B/32* as CLIP's image encoder, limiting the batch size to 256, despite a larger batch size being preferable in contrastive learning (Chen et al., 2022) for finetuning experiments. The study focuses on testing common learning rates and optimizer values without sophisticated hyperparameter tuning, retaining those that initially yielded good results. We use the following evaluation metrics to analyse the models. **Recall@k** measures the proportion of relevant items retrieved within the *top-k* results. **Diversity Recall@k** measures the percentage of the relevant items matched by the *top-k* predictions. **Mean Average Precision@k (mAP)** measures the percentage of predictions matching any relevant item for every rank up to *k*, considering their order.

5 Results

5.1 Empirical Evaluation

Zero-shot model & baseline algorithms. The zero-shot CLIP model, without output head, performs poorly on the test split, but achieves a recall score of over 15 at the tenth rank. In a qualitative analysis, we find that the model predicts more specific items, e.g., people in an image often get predicted with their specific names. We believe this results from CLIP's pre-training, where the ground-truth texts were more specific to the image compared to our dataset's labels. The random baseline algorithm randomly picks items from the candidate pool with a probability equal to their frequency in the train split, but results are comparably poor compared to the zero-shot model. The top-k baseline algorithm predicts the same ten items for every image, namely the most frequent ones in the train split, which performs well based on metrics. However, no rare item is predicted correctly, which is the main shortcoming of this baseline.

MLP Naive CLIP model. The Naive CLIP model with both CLIP encoders frozen and a simple MLP head performs well with a recall score of over 50 at rank ten. It suggests a correct item on every second image, making it the best Naive CLIP model. However, the precision is lower at the top rank. The recall scores at ranks 20, 50, and 100 increased, with rank 100 still being among the first 0.5% of all candidate items. The actual prediction scores are close to each other, with an average of 0.29 at rank one and 0.25 at rank 100. The model achieves a good balance between more specific and generic items, considering image content instead of outputting specific persons' names. This makes it a good choice for predicting diverse kinds of items. For example, it accurately predicts *presenters*, *microphones*, *awards*, and *human*[6] instead of suggesting specific names of people.

CLIP Fusion model. The CLIP Fusion model outperforms all tested models, with double precision and recall and a recall value of 92.4 at rank 100. We found that this is due to the Commons category input often revealing the correct answer, especially for infrequent items. The corresponding image category in some cases may be named almost or even exactly the same as the name of the item, such as "*London Victoria station*"[7].

However, the effectiveness of the model drops when no descriptive text input is available for the existing Wikicommons images or when a new image is uploaded. Combining categories and a threshold dataset can make tasks harder when specific categories are provided but mapped to generic items with little in common in textual representation. While fitting the model on the full pool of candidate items might be promising, it does not address the issue of input dependency.

5.2 Human Evaluation

We evaluated the model performance of the Naive CLIP model with a human evaluation study. The simplicity of the Naive CLIP model, and its reduced reliance on large amounts of training data,

[6] https://commons.wikimedia.org/?curid=28127864
[7] https://commons.wikimedia.org/?curid=12289864

Model	Recall			Div. Recall			mAP		
	@1	@5	@10	@1	@5	@10	@1	@5	@10
Zero-shot	4.7	11.5	15.9	4.7	7.5	10.3	4.7	4.7	5.1
Random baseline	2.1	9.6	17.2	2.1	6.5	11.5	2.1	3.1	3.7
Top-k baseline	12.4	29.8	40.8	12.4	20.5	29.8	12.4	14.3	15.9
MLP Naive CLIP	16.2	**40.5**	**51.8**	16.2	**27.5**	**37.2**	16.2	17.1	18.7
TE Naive CLIP BS 256	**20.6**	37.5	45.0	**20.6**	26.0	31.8	**20.6**	**19.0**	**20.0**
MLP Naive CLIP BS 256	14.2	38.8	49.8	14.2	26.3	35.6	14.2	15.7	17.3
CLIP Fusion	36.4	62.4	71.8	36.4	45.5	56.3	36.4	32.3	34.3

Table 4: Comparison of the performance of various model setups on our test split (zero hops in the metrics). Default batch size is 1,024. "MLP" = CLIP encoders frozen, "TE" = finetuned text encoder, "BS" = batch size.

	CC	TG	OR	CI	IDK
k=1	**43.1**	5.2	20.7	24.8	6.2
k=5	**34.3**	6.6	28.0	24.4	6.7
k=10	**29.8**	6.8	28.7	26.4	8.3

Table 5: Human Evaluation Results in %. CC = completely correct, TG = too general, OR = only related, CI = completely incorrect, IDK = I don't know.

make it more realistic for this model to be deployed on Wikimedia Commons.

With this study, we aim to understand to what extent a model genuinely predicts reasonable items. Given the large variety in data, and the data challenges enumerated in Section 3, we believe the actual model output may be more useful than is evident from the metrics relying on the ground-truth data. To this end, we set up a website using a subset of test split images, their ground-truth items, and the top 10 model predictions. For each prediction, participants choose between four qualitative ratings (*"completely correct"*, *"too generic"*, *"only related"*, or *"completely incorrect"*), as well as an alternative *"I don't know"* option. In our human evaluation study, 100 random images from the test split were annotated, each image by three people.

Our study focuses on quantifying the inter-annotator agreement in image evaluations using Fleiss' Kappa measure (Fleiss, 1971). The average agreement across all images for rank one is 0.54. To estimate model performance, the chosen options are aggregated over all users and images. We calculate distributions across ranks $k = 1$, $k = 5$, and $k = 10$ to compare previous metric-based evaluation results. The results illustrated in Table 5 show a value of 43.1 for the top prediction being completely correct, which is over 2.5 times the precision/recall value of 16.2 (cf. Table 4) with MLP Naive CLIP, indicating better model performance than the metric-based evaluation results.

The value for *"completely correct"* decreases for later ranks, as only a few completely correct answers per image are predicted for later ranks. The option with the highest percentage is *"only related"*, as it is the model's best next guess. *"Too general"* predictions occur in certain model setups, and completely incorrect and obscure predictions are observed at rank ten.

6 Conclusion

In this paper, we propose the Wiki-VEL framework, linking the items portrayed in images with structured knowledge stemming from Wikidata. We create a dataset from community-contributed, open-licensed Wikimedia Commons images labeled with the depicted entities in the form of Wikidata entities. In our VEL experiments, we show that the Naive CLIP model shows promising performance by outperforming the zero-shot model and simple baselines. The performance of the CLIP Fusion model also improved with more input data. However, all setups reached a plateau in learning due to the noisy real-world data. In our human evaluation, we show that the data quality also affects the metrics to evaluate model performance – humans perceive the model to be correct more than the automated metrics.

Looking towards the future, our results are promising for automatically providing structured labels for Wikimedia Commons images. To realise this vision, the Wikimedia community could participate in a large-scale human evaluation to assess the integration of the model into Commons to support contributors on image uploads and achieve the desired benefits from the structured data project. Further, the dataset can easily be extended to a multilingual dataset by extracting the image description and item names in different languages from structured data.

References

Omar Adjali, Romaric Besançon, Olivier Ferret, Hervé Le Borgne, and Brigitte Grau. 2020. Multimodal entity linking for tweets. In *European Conference on Information Retrieval*, pages 463–478. Springer.

Ali Furkan Biten, Lluis Gomez, Marçal Rusinol, and Dimosthenis Karatzas. 2019. Good news, everyone! context driven entity-aware captioning for news images. In *Proceedings of CVPR 2019*.

Changyou Chen, Jianyi Zhang, Yi Xu, Liqun Chen, Jiali Duan, Yiran Chen, Son Dinh Tran, Belinda Zeng, and Trishul Chilimbi. 2022. Why do we need large batchsizes in contrastive learning? a gradient-bias perspective. In *Advances in Neural Information Processing Systems*.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish Thapliyal, James Bradbury, Weicheng Kuo, Mojtaba Seyedhosseini, Chao Jia, Burcu Karagol Ayan, Carlos Riquelme, Andreas Steiner, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. 2023. PaLI: A jointly-scaled multilingual language-image model. *Preprint*, arXiv:2209.06794.

Gerard de Melo and Niket Tandon. 2016. Seeing is believing: The quest for multimodal knowledge. *ACM SIGWEB Newsletter*, (Spring 2016):4:1–4:9.

Shahi Dost, Luciano Serafini, Marco Rospocher, Lamberto Ballan, and Alessandro Sperduti. 2020. VTLINKER: visual-textual-knowledge entity linker. In *ECAI 2020*, pages 2897–2898. IOS Press.

Joseph Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–.

Jingru Gan, Jinchang Luo, Haiwei Wang, Shuhui Wang, Wei He, and Qingming Huang. 2021. Multimodal entity linking: a new dataset and a baseline. In *Proceedings of ACM Multimedia 2021*.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. 2024. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. *Preprint*, arXiv:1512.03385.

Hexiang Hu, Yi Luan, Yang Chen, Urvashi Khandelwal, Mandar Joshi, Kenton Lee, Kristina Toutanova, and Ming-Wei Chang. 2023. Open-domain visual entity recognition: Towards recognizing millions of Wikipedia entities. *Preprint*, arXiv:2302.11154.

Huadong Li, Yafang Wang, Gerard de Melo, Changhe Tu, and Baoquan Chen. 2017. Multimodal question answering over structured data with ambiguous entities. In *Proceedings of WWW 2017*. ACM.

Di Lu, Leonardo Neves, Vitor Carvalho, Ning Zhang, and Heng Ji. 2018. Visual attention model for name tagging in multimodal social media. In *Proceedings of ACL 2018*.

Seungwhan Moon, Leonardo Neves, and Vitor Carvalho. 2018. Multimodal named entity disambiguation for noisy social media posts. In *Proceedings of ACL 2018*.

Eric Müller-Budack, Jonas Theiner, Sebastian Diering, Maximilian Idahl, Sherzod Hakimov, and Ralph Ewerth. 2021. Multimodal news analytics using measures of cross-modal entity and context consistency. *International Journal of Multimedia Information Retrieval*, 10(2):111–125.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. 2021. How much can CLIP benefit vision-and-language tasks? *arXiv preprint arXiv:2107.06383*.

Ekaterina Shutova, Niket Tandon, and Gerard de Melo. 2015. Perceptually grounded selectional preferences. In *Proceedings of ACL 2015*, pages 950–960.

Wen Sun, Yixing Fan, Jiafeng Guo, Ruqing Zhang, and Xueqi Cheng. 2022. Visual named entity linking: A new dataset and a baseline. In *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Neha Tilak, Sunil Gandhi, and Tim Oates. 2017. Visual entity linking. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE.

Xuwu Wang, Junfeng Tian, Min Gui, Zhixu Li, Rui Wang, Ming Yan, Lihan Chen, and Yanghua Xiao. 2022. WikiDiverse: a multimodal entity linking dataset with diversified contextual topics and entity types. *arXiv preprint arXiv:2204.06347*.

Rebecka Weegar, Linus Hammarlund, Agnes Tegen, Magnus Oskarsson, Kalle Åström, and Pierre Nugues. 2014. Visual entity linking: A preliminary study. In *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Li Zhang, Zhixu Li, and Qiang Yang. 2021. Attention-based multimodal entity linking with high-quality images. In *Database Systems for Advanced Applications: 26th International Conference, DASFAA 2021, Taipei, Taiwan, April 11–14, 2021, Proceedings, Part II 26*, pages 533–548. Springer.