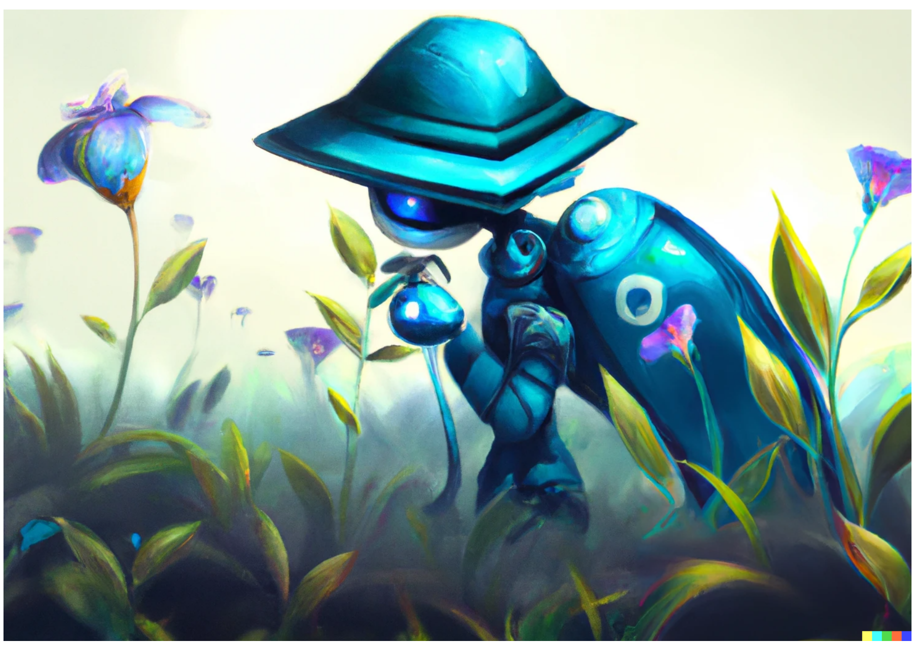
Description-Based Species Identification

Creating a Database for Robust and Interpretable Models for Biodiversity Monitoring

Robert van de Vlasakker

March 20, 2023



Department of Environmental Sciences
Wageningen University & Research
The Netherlands

1 Introduction

At the current rate, more than 50% of all species may become extinct before they are even described [12]. Unfortunately, despite the urgent need to address biodiversity loss, this issue remains understudied. As a result, scalable technologies that can automatically classify species are becoming increasingly necessary [23]. However, the field of species classification is delicate and requires data for all targeted species, particularly for less common ones [4, 22]. One effective approach to achieve this is by building a large dataset that contains species and their descriptions, which can then be used to train a neural network to reason like a taxonomist. This approach not only helps keep the network interpretable but also enables less common species to benefit from shared parts and traits with more common ones.

Although computer vision methods for plant species identification are advancing rapidly, we cannot be certain that they are consistent with current botanical knowledge, which may result in unreliable predictions for data-poor species. To ensure that computer vision methods align with the botanical field, we need to enable them to identify relevant morphological traits in images. This necessitates the construction of a comprehensive dataset of morphological traits, which can be used to train such a system to more closely resemble the process used by botanists. This is where semantic triples come into place.

Semantic triples are the gold standard for relational data. In contrast to well-known SQL databases, semantic triple storage works with predicates. By storing morphological traits in an RDF (Resource Description Framework) triple storage, subject–predicate–object expressions can be used to reason about the data. However, in practice, it is often difficult to populate an RDF triple storage. Complex NLP (Natural Language Processing) pipelines (e.g., entity extraction, entity linking, and co-reference resolution) are often needed to extract the necessary information [21]. The extraction pipelines often require a lot of supervision, making them very vulnerable to errors [17].

Recent progress in pre-training language models on large text corpora has shown remarkable improvements for downstream NLP tasks. These models use a transformer-based architecture [24]. The parameters of the model are optimized by masking a word anywhere in a sequence of words and making a prediction, storing vast amounts of linguistic knowledge [16]. Examples of already proven transformer models are BERT [7] (Bidirectional Encoder Representations from Transformers), BART [13] (Bidirectional and Auto-Regressive Transformers), and GPT-3 [3] (Generative Pre-trained Transformer 3). NLP models could help populate RDF triple storage with morphological traits.

For example, a language model could be used to extract information about the flower color of a plant species from a large corpus of unstructured text. Relation information from the text could be used to populate RDF triple storage, by answering ”fill-in-the-blank” cloze statements [1]. The language model could be trained on a dataset of text descriptions of plant species and their associated morphological traits. Using this trained model, the text could be analyzed to identify instances where a particular plant species is described as having a certain flower color. The extracted information could then be converted into structured RDF triples, such as "Plant Species X has Flower Color Y". By using language models to extract information from unstructured text, we can effectively bypass the need for complex NLP pipelines and structured data, which can be time-consuming and costly to produce. Additionally, since language models are trained on vast amounts of textual data, they can capture a wide range of linguistic and semantic information that may not be present in structured data sources.

According to Brown et al. [3], LMs are very efficient few-shot and zero-shot learners. This means that information could potentially be accessed by fine-tuning the models on domain-specific information [8]. Fine-tuning is necessary because the information embedded in the LMs is not directly accessible. To surface the information, the models need to be fine-tuned on domain-specific tasks. The fine-tuning of the model can be seen as creating an interface to retrieve the information. However, there are some limitations: the model cannot retrieve knowledge that is not encoded in it, and the data used during training might be outdated or not learned by the model. The models could be updated with the correct information by re-training the complete model, but this is very expensive and time-consuming, especially with the ever-increasing parameters of the LMs [3].

With the development of Large Language Models (LLMs) such as BLOOM [20] and ChatGPT [5], which have billions of parameters, natural language processing (NLP) models have taken another leap forward. While BERT has 340 million parameters, these recent models have surpassed this number into the billions. LLMs have demonstrated significant improvements in understanding contextual cues and generating appropriate responses, and are currently being extensively investigated in various fields, such as translation [10] and finance research [9]. According to Liu et al. [14], fine-tuning models is no longer necessary, as prompting might retrieve all necessary information.

Prompts can be divided into two large subcategories: Discrete Prompts and Soft Prompts [1]. Discrete Prompts are similar to "fill-in-the-blank" cloze statements, which allow the model to learn specific downstream tasks without fine-tuning [6, 17]. These statements are typically focused on domain-specific knowledge, such as commonsense knowledge [6] or factual information [17]. Leveraging the appropriate Discrete Prompt can result in decent zero-shot performance for language models [19], but the challenging part is creating the right Prompt for a domain-specific task. Soft Prompts, on the other hand, are represented by word vectors, and while they are more difficult to interpret than Discrete Prompts, Qin and Eisner [18] found that Soft Prompts can contain multiple pieces of information giving Soft Prompts a slight edge over Discrete Prompts.

Given the success of large language models and the growing interest in the potential of prompts, the objective of this research is to explore how prompts can be used to populate RDF triple storage with morphological traits. Specifically, our aim is to identify a set of discrete prompts and soft prompts that can effectively extract a list of traits, such as "Flower Colours", "Growth Stage", and "Plant Type", from unstructured text. To accomplish this, we will investigate various methods for constructing prompts, including leveraging domain-specific knowledge and word vectors. Additionally, we will evaluate the effectiveness of our prompts by leveraging hand-annotated datasets. Ultimately, we hope to contribute to the development of more efficient and accurate methods for extracting morphological traits from text, with potential applications in fields such as natural language understanding and information retrieval.

2 Research Objectives

The goal of this project is to enable large-scale biodiversity monitoring by leveraging recent advances in NLP. Specifically, we aim to develop Explainable Machine Learning (XML) methods that reason like taxonomists and enable citizen scientists to collect valuable data from rare or undescribed species. Our XML methods will use NLP techniques to extract relevant morphological and behavioral traits from a large textual database of species descriptions, with the goal of increasing awareness of biodiversity loss and its rapid rate and making a significant impact on the field of biodiversity monitoring and management.

To achieve these objectives, we will investigate methods for fine-tuning large language models (LLMs) and using Discrete Prompts to guide the models in extracting relevant traits. Discrete Prompts are particularly well-suited for extracting morphological traits from textual descriptions, and we have hand-annotated datasets that can be used to evaluate their effectiveness. Our research will focus on identifying a set of effective Discrete Prompts that can be used to populate RDF triple storage with morphological traits, such as ”Flower Colors,” ”Growth Stage,” and ”Plant Type.” Ultimately, our goal is to develop XML methods that enable citizen scientists to contribute to large-scale biodiversity monitoring efforts and increase public awareness of biodiversity loss.

3 Methodology

3.1 Data Collection & Pre-Processing

The initial stage of this project involves gathering large volumes of text descriptions about plant species from various internet sources, including scientific publications and community-based websites like Wikipedia and websites with a stronger scientific focus like Plant Database Search. To facilitate this process, we will employ a pre-trained NLP text classifier model that can differentiate between description and non-description text. This model will be deployed in a web crawler, enabling us to efficiently harvest substantial amounts of data in a manner that respects website policies and legal requirements. The web crawler will be designed to navigate and extract data from websites in a systematic and automated manner, and it will be optimized to avoid duplicating data and to handle various formats and structures of text descriptions.

Descriptions in the dataset might be very long or complex. In such cases, breaking them down into shorter manageable segments before feeding them into ChatGPT can help ensure that ChatGPT generates responses relevant to the specific trait of interest. Furthermore, all language models break text into tokens, for example, the sentence ”ChatGPT is great!” is encoded into six tokens: [”Chat”, ”G”, ”PT”, ”is”, ”great”, ”!”]. The model processes a maximum of 4096 tokens at once, so longer conversations may receive incomplete answers.

To ensure that the returned URLs are relevant to the queried plant species, we will use various methods such as analyzing the title, URL, and metadata of the web pages. Additionally, we will utilize natural language processing techniques to identify relevant keywords and phrases within the text itself. This approach helps to minimize the possibility of irrelevant or mismatched results, thus improving the overall quality of the data collected.

We will begin by removing any HTML tags and other non-informative characters to improve the quality of the harvested text. Furthermore, we will break each paragraph into individual sentences and classify them as either descriptive or non-descriptive. By categorizing each sentence, we aim to increase the accuracy of the NLP model, as the resulting text will contain more specific information about a single topic. Non-descriptive sentences will be removed from the dataset, leaving only the descriptive sentences. The remaining descriptive sentences will then be merged to form a new paragraph that provides as much relevant information about the plant species as possible. This approach preserves the coherence of the original paragraph while ensuring that the resulting text focuses on the topic of interest.

We have three hand-annotated datasets available for several taxonomic plant groups. In Table 1 an example of a small part of such a dataset can be found.

- Caribbean plant species (’Source’): 40 species with 25 traits each, including ’Life form’, ’Inflorescence type’, and ’Seed Colour’.
- Palm plant species [11]: The dataset consists of 333[1] palm species, each with a complete trait list comprising of 29 traits, including attributes such as ’Conspicuousness’, ’Fruit Colours’, and ’Seed Size’.
- Mediterranean plant species from Pl@nNet (’Source’): 361 species with 134 traits each, including ’Plant Type’, ’Fruit Shape’, and ’Sexuality’.

It is important to note that the quality and reliability of the hand-annotated datasets used for evaluation can greatly impact the validity of the results. While the Palm dataset used in this study is backed by a published research paper, the other two datasets were created by respected scientists but were not checked by other annotators. As such, there may be some variability in the quality and consistency of the labels in these datasets. While it would be ideal to have multiple annotators for each dataset and calculate inter-annotator agreement metrics, this was not feasible for this study. However, it is important to keep in mind the limitations of the datasets and consider the potential impact on the results.

[1]Note that the original dataset contains over 2,000 palm species, but in this study, we only utilize the 333 species with complete trait information.

The next step involves extracting relevant information from the collected text snippets. To accomplish this, we will leverage the capabilities of LLMs, which have demonstrated the ability to capture relational knowledge inherent in the training data [1, 17]. However, it is unknown which text snippets contain specific information, or whether a given snippet contains any information about a particular species trait. It is possible that the model deployed in the web crawler could store irrelevant text snippets. Nevertheless, we can align our hand-annotated datasets against the text snippets to ensure that the relevant information is present.

Table 1: One-hot encoded hand-annotated datasets for flower colours and leaf shapes. The table shows the presence or absence of specific traits for each species, represented using one-hot encoding, where a "1" indicates the presence of a trait and "0" indicates its absence.

#	Species	Flower Colour			Leaf Shape	
		Yellow	**Blue**	**Orange**	**Simple**	**Bifoliate**
1	Coccoloba uvifera	0	1	1	1	1
2	Clusia rosea	1	0	0	1	0
3	Bourreria succulenta	1	1	1	0	0

3.2 Prompt Crafting

We must be careful when crafting and using Prompts. LLMs are known to make up facts or exhibit strange behavior when prompted for information [2, 25, 15]. This can be especially problematic when using LLMs in real-world applications, where the consequences of incorrect or misleading information can be significant. Which could be problematic in the context of biodiversity monitoring. According to Brown et al. [3], careful crafting and use of prompts can help mitigate the risks of using LLMs. Specifically, they the use of "few-shot learning," where the LLM is trained on a small number of examples or prompts to generalize to new tasks. The prompts used in few-shot learning should be carefully designed to ensure that the LLM does not learn to make up facts or exhibit strange behavior. In this research we will focus on crafting high-quality Discrete Prompts for question answering. We will do this by creating prompts based on hand-annotated datasets that serve as ground-truth data. This allows us to fact-check the model's output against known information and avoid any issues with the model making up false facts or exhibiting strange behavior.

All datasets consist of dataframes that are one-hot encoded. An example for three different species with two different morphological traits can be found in Table 1. In this case, there are three possible flower colors: 'Yellow', 'Blue', and 'Orange', and two possible leaf shapes: 'Simple' or 'Bifoliate'. When we create a Discrete Prompt for the first species, we use descriptive text from various internet sources that could contain any data about the species, but with the combination of the hand-annotated data, we can fact-check the information. We will present both textual descriptions and the ground-truth data in the same Discrete Prompt.

To create a prompt for ChatGPT, we will combine all the traits and their corresponding values into a list of possibilities. The model will not know which values are ground-truth values. The prompt will also contain a list of traits without values that need to be inferred from the text. For example, the descriptive text of the first species could be: "Sea grape is an evergreen shrub, or sometimes a tree, varying in height and habit according to its environment. In more exposed conditions, it can be a spreading shrub just 1 meter tall; it usually flowers with blue or orange flowers." The final prompt for ChatGPT could look something like this:

> Text:
> "Sea grape is an evergreen shrub, or sometimes a tree, varying in height and habit according to its environment. In more exposed conditions, it can be a spreading shrub just 1 meter tall; it usually flowers with blue or orange flowers.
>
> Dictionary of traits and values:
> {'Flower Colour': ['Yellow', 'Blue', 'Orange'], 'Leaf Shape': ['Simple', 'Bifoliate']}
>
> List of traits:
> ['Flower Colour:', 'Leaf Shape':]
>
> Can you fill in the list of traits, based on the choices you have on the dictionary of traits and values with the presented text? I case you cannot find a value, fill in 'BLANK'.

In the example prompt, we can treat the prompt as a set consisting of the descriptive text τ, the list of traits λ, and the answer to the question, which is the intersection between both: $\lambda \cap \tau$. In this example, we expect ChatGPT to return both "Flower Colour: Blue" and "Flower Colour: Orange" as it is able to infer these values from the presented text. However, for 'Leaf Shape', we expect ChatGPT to fill in 'BLANK' as instructed, as these trait values cannot be inferred from the text.

We will start by prompting the ChatGPT ('gpt-3.5-turbo') model from OpenAI (https://openai.com/). This model is available through their API and is an improved version of GPT-3 [3]. ChatGPT is a powerful natural language processing model that can be used to generate human-like responses to text-based prompts, making it ideal for tasks which require high-level language understanding and reasoning skills. Despite having 100 times fewer parameters than its predecessor (175 billion vs. 1.3 billion), it outperforms the previous model with prompt distributions based on human evaluations [15]. The ChatGPt model was specifically designed to excel at text-based tasks that require high-level language understanding and reasoning skills.

Compared to GPT-3, ChatGPT shows better performance in generating text in response to discrete prompts. While GPT-3 is also capable of performing the same tasks, it requires more careful prompting and is more error-prone [15]. Although both ChatGPT and GPT-3 can generate high-quality text, ChatGPT is generally considered to be better at generating text in response to specific prompts. So while both models are capable of performing similar tasks, ChatGPT may excel in certain areas where it can be more specifically prompted. As ChatGPT cannot be fine-tuned at this time, we will focus only on prompting the model and not on any fine-tuning.

3.3 Evaluation

To evaluate the performance of the ChatGPT model in responding to handcrafted discrete prompts, we can use the standard evaluation metrics of precision, recall, and F1 score. Precision measures the proportion of correctly identified instances among all identified instances, while recall measures the proportion of correctly identified instances among all instances that should have been identified. The F1 score is the harmonic mean of precision and recall, providing an overall measure of the model's performance.

To compute these metrics, we can compare the model's predicted traits against the ground-truth traits in the annotated datasets. Specifically, we can compute precision, recall, and F1 score for each trait in the dataset by treating it as a binary classification problem, where the positive class indicates that the trait is present in the text and the negative class indicates that it is not. We can then compute the macro-averaged precision, recall, and F1 score across all traits to obtain an overall evaluation of the model's performance. This evaluation protocol will allow us to assess the model's ability to identify the correct traits from a list of options in response to a given prompt.

3.4 Knowledge Graph Creation

To enable the use of ChatGPT for knowledge extraction, we need to store the extracted information in a structured format that can be easily queried and used by other applications. In this work, we use RDF triples to represent the extracted information as a knowledge graph. RDF (Resource Description Framework) is a standard format for representing data on the web, and RDF triples are a basic building block of RDF data. Each RDF triple consists of a subject, a predicate, and an object, where the subject is a resource, the predicate is a property or relationship, and the object is a value or another resource.

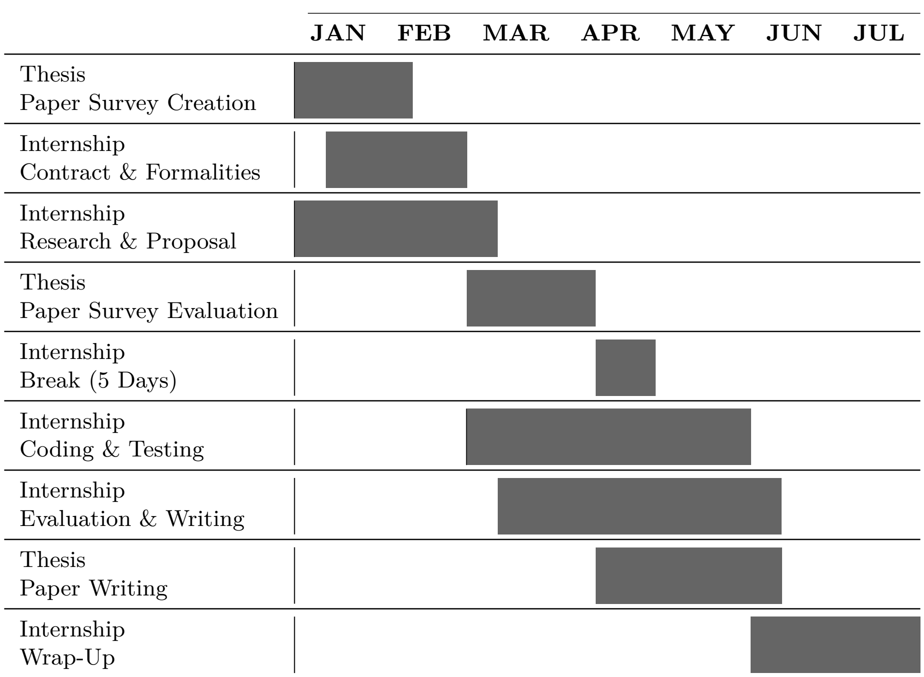
Our approach involves using ChatGPT to extract information from descriptive text snippets and converting the extracted information into RDF triples. We then store the RDF triples in a triplestore, a specialized database designed to store and query RDF data. This allows us to build a knowledge graph that can be queried using SPARQL, a query language for RDF data. In the following sections, we describe the details of our knowledge graph creation process.

In order to create an ontology to support the knowledge graph creation process, we will use the hand-annotated datasets as a starting point. These datasets contain a set of pre-defined traits for each plant species that we want to represent in the knowledge graph. We will use these traits as the basis for the ontology classes and properties, and map them to existing ontologies when possible.

To create the RDF triples, we will use the ontology classes and properties to represent the traits of the plant species extracted from the text snippets using ChatGPT. The RDF triples will follow the subject-predicate-object format, where the subject is the plant species, the predicate is the trait, and the object is the value of the trait. By representing the extracted information as RDF triples, we will be able to store and query the data in a more structured and efficient way, and link it to other datasets and knowledge graphs.

4 Schedule

Table 2: The Internship Timetable. The internship will focus on two aspect: producing a paper from the thesis and producing a paper as described in this research proposal.



References

[1] Badr AlKhamissi et al. *A Review on Language Models as Knowledge Bases*. Apr. 12, 2022. arXiv: 2204.06031[cs]. URL: http://arxiv.org/abs/2204.06031 (visited on 12/12/2022).

[2] Emily M. Bender et al. "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜" in: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '21. New York, NY, USA: Association for Computing Machinery, Mar. 1, 2021, pp. 610–623. ISBN: 978-1-4503-8309-7. DOI: 10.1145/3442188.3445922. URL: https://doi.org/10.1145/3442188.3445922 (visited on 02/20/2023).

[3] Tom B. Brown et al. "Language Models are Few-Shot Learners". In: *arXiv:2005.14165 [cs]* (July 22, 2020). arXiv: 2005.14165. URL: http://arxiv.org/abs/2005.14165 (visited on 07/11/2021).

[4] Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. "Machine Learning Interpretability: A Survey on Methods and Metrics". In: *Electronics* 8.8 (Aug. 2019). Number: 8 Publisher: Multidisciplinary Digital Publishing Institute, p. 832. DOI: 10.3390/electronics8080832. URL: https://www.mdpi.com/2079-9292/8/8/832 (visited on 07/09/2021).

[5] *ChatGPT: Optimizing Language Models for Dialogue*. OpenAI. Nov. 30, 2022. URL: https://openai.com/blog/chatgpt/ (visited on 12/22/2022).

[6] Joe Davison, Joshua Feldman, and Alexander Rush. "Commonsense Knowledge Mining from Pretrained Models". 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)*. EMNLP-IJCNLP 2019. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 1173–1178. DOI: 10.18653/v1/D19-1109. URL: https://aclanthology.org/D19-1109 (visited on 02/19/2023).

[7] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *arXiv:1810.04805 [cs]* (May 24, 2019). arXiv: 1810.04805. URL: http://arxiv.org/abs/1810.04805 (visited on 07/04/2021).

[8] Li Dong et al. *Unified Language Model Pre-training for Natural Language Understanding and Generation*. Oct. 15, 2019. DOI: 10.48550/arXiv.1905.03197. arXiv: 1905.03197[cs]. URL: http://arxiv.org/abs/1905.03197 (visited on 02/18/2023).

[9] Michael Dowling and Brian Lucey. "ChatGPT for (Finance) research: The Bananarama Conjecture". In: *Finance Research Letters* (Jan. 25, 2023), p. 103662. ISSN: 1544-6123. DOI: 10.1016/j.frl.2023.103662. URL: https://www.sciencedirect.com/science/article/pii/S1544612323000363 (visited on 02/15/2023).

[10] Wenxiang Jiao et al. *Is ChatGPT A Good Translator? A Preliminary Study*. Jan. 31, 2023. DOI: 10.48550/arXiv.2301.08745. arXiv: 2301.08745[cs]. URL: http://arxiv.org/abs/2301.08745 (visited on 02/15/2023).

[11] W. Daniel Kissling et al. "PalmTraits 1.0, a species-level functional trait database of palms worldwide". In: *Scientific Data* 6.1 (Dec. 2019), p. 178. ISSN: 2052-4463. DOI: 10.1038/s41597-019-0189-0. URL: http://www.nature.com/articles/s41597-019-0189-0 (visited on 03/07/2022).

[12] Alexander C. Lees and Stuart L. Pimm. "Species, extinct before we know them?" In: *Current Biology* 25.5 (Mar. 2, 2015), R177–R180. ISSN: 0960-9822. DOI: 10.1016/j.cub.2014.12.017. URL: https://www.sciencedirect.com/science/article/pii/S0960982214016182 (visited on 09/22/2021).

[13] Mike Lewis et al. *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension*. Oct. 29, 2019. DOI: 10.48550/arXiv.1910.13461. arXiv: 1910.13461[cs,stat]. URL: http://arxiv.org/abs/1910.13461 (visited on 02/18/2023).

[14] Nan Liu et al. "Learning to Compose Visual Relations". In: *arXiv:2111.09297 [cs, stat]* (Nov. 17, 2021). arXiv: 2111.09297. URL: http://arxiv.org/abs/2111.09297 (visited on 12/06/2021).

[15] Long Ouyang et al. *Training language models to follow instructions with human feedback*. Mar. 4, 2022. DOI: 10.48550/arXiv.2203.02155. arXiv: 2203.02155[cs]. URL: http://arxiv.org/abs/2203.02155 (visited on 02/20/2023).

[16] Matthew E. Peters et al. *Dissecting Contextual Word Embeddings: Architecture and Representation*. Sept. 27, 2018. DOI: 10.48550/arXiv.1808.08949. arXiv: 1808.08949[cs]. URL: http://arxiv.org/abs/1808.08949 (visited on 02/15/2023).

[17] Fabio Petroni et al. *Language Models as Knowledge Bases?* Sept. 4, 2019. arXiv: 1909.01066[cs]. URL: http://arxiv.org/abs/1909.01066 (visited on 12/12/2022).

[18] Guanghui Qin and Jason Eisner. "Learning How to Ask: Querying LMs with Mixtures of Soft Prompts". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. NAACL-HLT 2021. Online: Association for Computational Linguistics, June 2021, pp. 5203–5212. DOI: 10.18653/v1/2021.naacl-main.410. URL: https://aclanthology.org/2021.naacl-main.410 (visited on 02/18/2023).

[19] Alec Radford et al. "Language Models are Unsupervised Multitask Learners". In: (2018).

[20] Teven Le Scao et al. *BLOOM: A 176B-Parameter Open-Access Multilingual Language Model*. Nov. 9, 2022. arXiv: 2211.05100[cs]. URL: http://arxiv.org/abs/2211.05100 (visited on 12/12/2022).

[21] Mihai Surdeanu and Heng Ji. "Overview of the English Slot Filling Track at the TAC2014 Knowledge Base Population Evaluation". In: *TAC2014* Text Analysis Conference (2014).

[22] Grant Van Horn and Pietro Perona. "The Devil is in the Tails: Fine-grained Classification in the Wild". In: *arXiv:1709.01450 [cs]* (Sept. 5, 2017). arXiv: 1709.01450. URL: http://arxiv.org/abs/1709.01450 (visited on 10/03/2021).

[23] Grant Van Horn et al. *The iNaturalist Species Classification and Detection Dataset*. arXiv:1707.06642. type: article. arXiv, Apr. 10, 2018. DOI: 10.48550/arXiv.1707.06642. arXiv: 1707.06642[cs]. URL: http://arxiv.org/abs/1707.06642 (visited on 06/01/2022).

[24] Ashish Vaswani et al. *Attention Is All You Need*. Dec. 5, 2017. DOI: 10.48550/arXiv.1706.03762. arXiv: 1706.03762[cs]. URL: http://arxiv.org/abs/1706.03762 (visited on 03/04/2023).

[25] Laura Weidinger et al. *Ethical and social risks of harm from Language Models*. Dec. 8, 2021. DOI: 10.48550/arXiv.2112.04359. arXiv: 2112.04359[cs]. URL: http://arxiv.org/abs/2112.04359 (visited on 02/20/2023).