INTELMO: Enhancing Models' Adoption of Interactive Interfaces

Chunxu Yang[†], **Chien-Sheng Wu**[§], **Lidiya Murakhovs'ka**[§],
Philippe Laban[§], **Xiang 'Anthony' Chen**[†]
[†] UCLA HCI Research, [§] Salesforce Research
{chunxuyang, xac}@ucla.edu
{wu.jason, l.murakhovska, plaban}@salesforce.com

Abstract

This paper presents INTELMO, an easy-to-use toolkit to help model developers adopt user-faced interactive interfaces for their language models. The toolkit provides default style patterns over interaction-based categorization, ensuring that developers can build fully interactive interfaces with minimal and intuitive additional code. Moreover, INTELMO employs a multi-granular hierarchical abstraction to provide developers with flexible control over the generation process. INTELMO is under active development, with document available at https://intelmo.github.io/

1 Introduction

As natural language processing (NLP) and human-computer interaction (HCI) theories advance, the demand for integrating the two has markedly increased. However, a significant challenge persists in bridging the gap between NLP model development and user interaction. This predicament arises due to the divergent skill sets of these two groups: while model developers excel in building advanced algorithms, they may lack expertise in developing complex interactive user interfaces. Conversely, front-end developers might possess proficiency in crafting engaging user interfaces, but they may lack the knowledge of the intricacies involved in NLP model development (Cai and Guo, 2019).

This situation leads to a critical need for model developers to rapidly implement interactive user interfaces. The conventional software development workflow, where model developers pass their projects to front-end engineers to design the user interface, may not be feasible at the model-tuning stage.

In addition, efforts have been made to ensure that NLP models use datasets that closely align with real-world tasks. Chandu et al. (2021) highlights the significance of bridging the gap between common NLP datasets and real-world scenarios, proposing a method to enhance authenticity through dynamic grounding. Additionally, several models, such as TweetNLP (Camacho-collados et al., 2022), Marvista (Chen et al., 2023) and RestGPT (Song et al., 2023), utilized real web data as their training and testing data source.

However, building a web scraper from scratch poses a daunting task for model developers. During the model's tuning and optimization process, developers require real-time feedback on the model's performance and may also need to compare or combine outputs from multiple models. A seamless flow of real-world data can significantly enhance the model's interpretability and enable developers to preview how their model will perform when delivered to end-users.

Considering the current situation, we believe that a toolkit generating interactive interfaces for model developers should have the following characteristics, listed in order of importance:

1. **Interactivity**: The toolkit should allow interaction with the model's interface and provide real-time feedback. It should also include a configurable module to adjust model parameters, enabling the observation of the model's performance under different settings.

2. **Flexibility**: The toolkit should support most common NLP tasks and facilitate interaction with multiple models. Additionally, it should provide fine-grained control over the display of model results.

3. **Usability**: The toolkit should abstract application programming interfaces (APIs) for model developers, ensuring that Python developers without web knowledge can integrate the system into their models with minimal code.

4. **Automation**: The toolkit should automatically crawl information from the real-time

web and stream it into the model with no efforts from developers.

This demo introduces INTELMO, a versatile toolkit that employs multiple layers of abstraction to meet all requirements listed above. (1) INTELMO is built on Flask[1], serving as a foundation to offer HTML templates and a flexible building environment. Through encapsulating default styles, this toolkit can effortlessly construct interfaces based on model developer configurations. (2) The toolkit divides articles into three nested levels: *paragraphs*, *sentences*, and *words*. Each level comes with specific APIs for customization and default styles tailored to different task types. (3) Simplifying the developer's task, INTELMO adopts task categorization. By specifying the task type such as *Modification*, *Generation*, or *Insertion* based on interaction process (detailed classification in Section 3.2.1), developers can initiate a complete web application using within ten lines of non-intrusive code. (4) INTELMO automatically retrieves the Really Simple Syndication (RSS) source from configurable news websites in the background once the application is launched, freeing developers from the burden of constructing scrapers.

We believe that tools like INTELMO can significantly reduce the difficulty of building and deploying interactive interfaces for NLP models, thereby improving model performance, customization, and interpretability through iterative feedback.

2 Related Work

In recent years, as various NLP models have gained extensive attention, researchers have started to take notice of evaluation metrics beyond language model performance. Wang et al. (2021) emphasized the importance of Human-in-the-loop(HITL) NLP framework, while ITG (Faltings et al., 2023), and WEBGPT (Nakano et al., 2022) incorporated human interaction into *text generation*, and *question answering (QA)* tasks within the NLP domain. Gu et al. (2023), Lee et al. (2023) and Carta et al. (2023) have also recognized the significance of models being applicable to real-world data, proposing evaluation criteria based on real-world human-language model interactions. DMS (Jónsson and Loftsson, 2023) and WEBSHOP (Yao et al., 2023) attempted to integrate different NLP tasks to achieve dynamic delivery of multi-task systems.

Regarding relevant tools and platforms, EXPLAINBOARD (Liu et al., 2021) and ADAPTERHUB (Beck et al., 2022) serves as no-code platforms for developing and testing language models. More recently, IFAN (Mosca et al., 2023) utilizes API technology to enable real-time, interpretation-based interactions with models.

3 INTELMO

We created **IN**terface **T**oolkit for **E**xtensible **L**anguage **MO**dels (**INTELMO**) as a solution for generating interactive interfaces for NLP models. As shown in Figure 1, the system consists of the following components:

1. **Model Configuration Parser**: This component is responsible for providing control interfaces to model developers. The specific configuration abstractions will be detailed in Section 3.2.

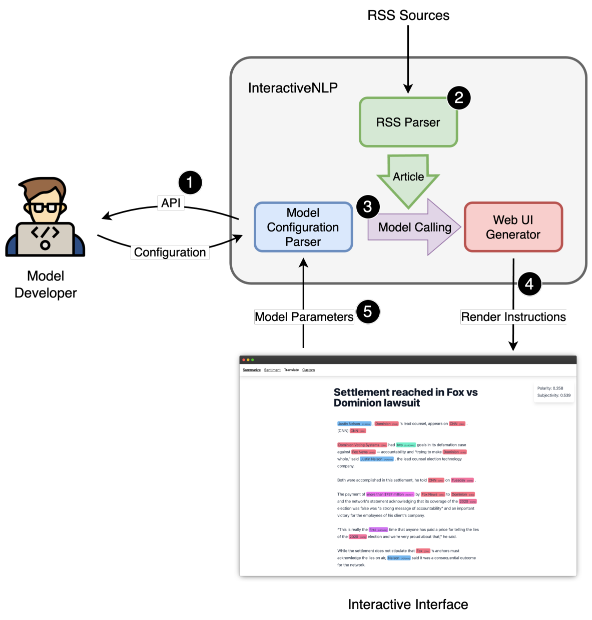
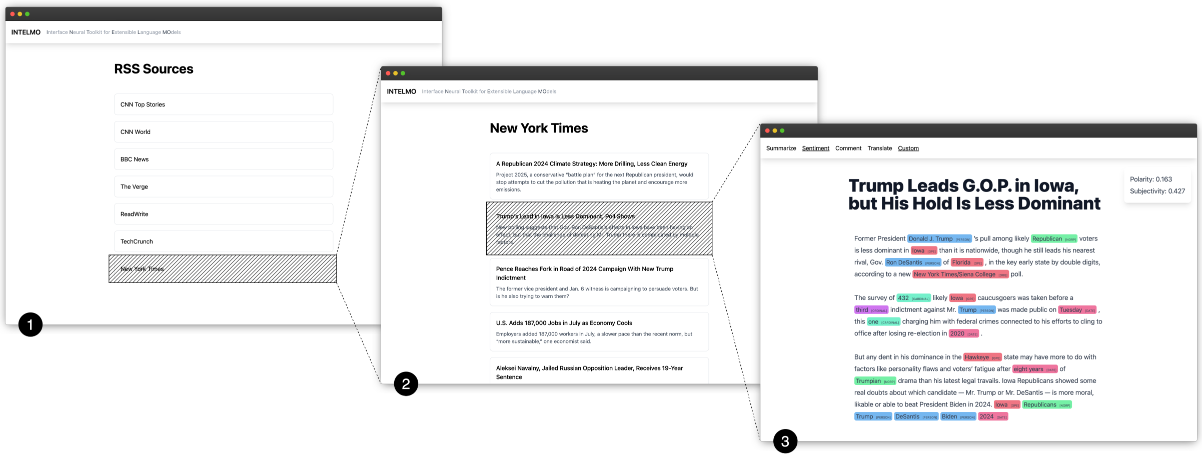


Figure 1: The basic structure of INTELMO: (1) The toolkit provides encapsulated API interfaces to model developers. (2) INTELMO utilizes an embedded RSS parser to fetch RSS information and parse it into articles. (3) The system maps model configurations to corresponding functions and passes the articles to the model function. (4) INTELMO renders the interactive interface based the model's results. (5) End-users or model developers can modify the parameters through forms on the webpage. The modified model parameters are incorporated into the model function. (Icon made by @surang from `www.flaticon.com`.)

[1]`https://flask.palletsprojects.com/`


INTELMO Interface Neural Toolkit for Extensible Language MOdels

RSS Sources

CNN Top Stories

CNN World

BBC News

The Verge

ReadWrite

TechCrunch

New York Times

INTELMO Interface Neural Toolkit for Extensible Language MOdels

New York Times

A Republican 2024 Climate Strategy: More Drilling, Less Clean Energy

Project 2025, a conservative "battle plan" for the next Republican president, would stop attempts to cut the pollution that is heating the planet and encourage more emissions.

Trump's Lead in Iowa Is Less Dominant, Poll Shows

New polling suggests that Gov. Ron DeSantis's efforts in Iowa have been having an effect, but that the challenge of defeating Mr. Trump there is complicated by multiple factors.

Pence Reaches Fork in Road of 2024 Campaign With New Trump Indictment

The former vice president and Jan. 6 witness is campaigning to persuade voters. But is he also trying to warn them?

U.S. Adds 187,000 Jobs in July as Economy Cools

Employers added 187,000 workers in July, a slower pace than the recent norm, but "more sustainable," one economist said.

Aleksei Navalny, Jailed Russian Opposition Leader, Receives 19-Year Sentence

Summarize Sentiment Comment Translate Custom

Trump Leads G.O.P. in Iowa, but His Hold Is Less Dominant

Polarity: 0.163
Subjectivity: 0.427

Former President Donald J. Trump [PERSON] 's pull among likely Republican [NORP] voters is less dominant in Iowa [GPE] than it is nationwide, though he still leads his nearest rival, Gov. Ron DeSantis [PERSON] of Florida [GPE] , in the key early state by double digits, according to a new New York Times/Siena College [ORG] poll.

The survey of 432 [CARDINAL] likely Iowa [GPE] caucusgoers was taken before a third [ORDINAL] indictment against Mr. Trump [PERSON] was made public on Tuesday [DATE] , this one [CARDINAL] charging him with federal crimes connected to his efforts to cling to office after losing re-election in 2020 [DATE] .

But any dent in his dominance in the Hawkeye [GPE] state may have more to do with factors like personality flaws and voters' fatigue after eight years [DATE] of Trumpian [NORP] drama than his latest legal travails. Iowa Republicans showed some real doubts about which candidate — Mr. Trump or Mr. DeSantis — is more moral, likable or able to beat President Biden in 2024. Iowa [GPE] Republicans [NORP] Trump [PERSON] DeSantis [PERSON] Biden [PERSON] 2024 [DATE]

Figure 2: The user interface of INTELMO. (1) RSS sources page; (2) Article list page; (3) NLP reader page.

2. **RSS Parser**: This module implements the retrieval of real-world web data. The RSS Parser automatically collects articles from different sources based on configuration settings. After preprocessing, these articles are passed as parameters into the model.

3. **Web UI Generator**: This module generates rendering instructions based on the results returned by the model, creating interactive UI components. The precise control over page elements will be elaborated on in Section 3.1.

These components are embedded within INTELMO. Unless a model developer decides to finely control each step, which the API allows and provides for, they do not need to understand the specific operational details. For a basic workflow, a model developer only needs to pass configuration options to INTELMO which requires usually no more than 10 lines of code per feature, and the generated interactive pages will be produced.

3.1 User Interface

The user interface of INTELMO, as shown in Figure 2, resembles an RSS reader. The interface relies on Flask with Jinja template engine[2] and is styled using TailwindCSS [3]. The initial UI generated by INTELMO serves as the RSS source page. On this page, users can select sources of interest from pre-configured RSS feeds. Upon selection, INTELMO employs a parser to extract the list of articles. The list page displays article titles and brief descriptions.

The reader page of INTELMO is illustrated in Figure 3. After entering this page, users can apply the model functions from the top-left corner. Hovering the cursor over the function name enables the adjustment of available model parameters.

As a core feature for model developers, INTELMO provides relevant configuration options for most page elements. For each element, model developers can set its content or configure child elements at the next level. In the latter case, INTELMO prioritizes rendering the child elements. If model developers are not satisfied with the provided built-in styles, they can specify TailwindCSS class names or even directly include HTML tags within the element's content for complete customization.

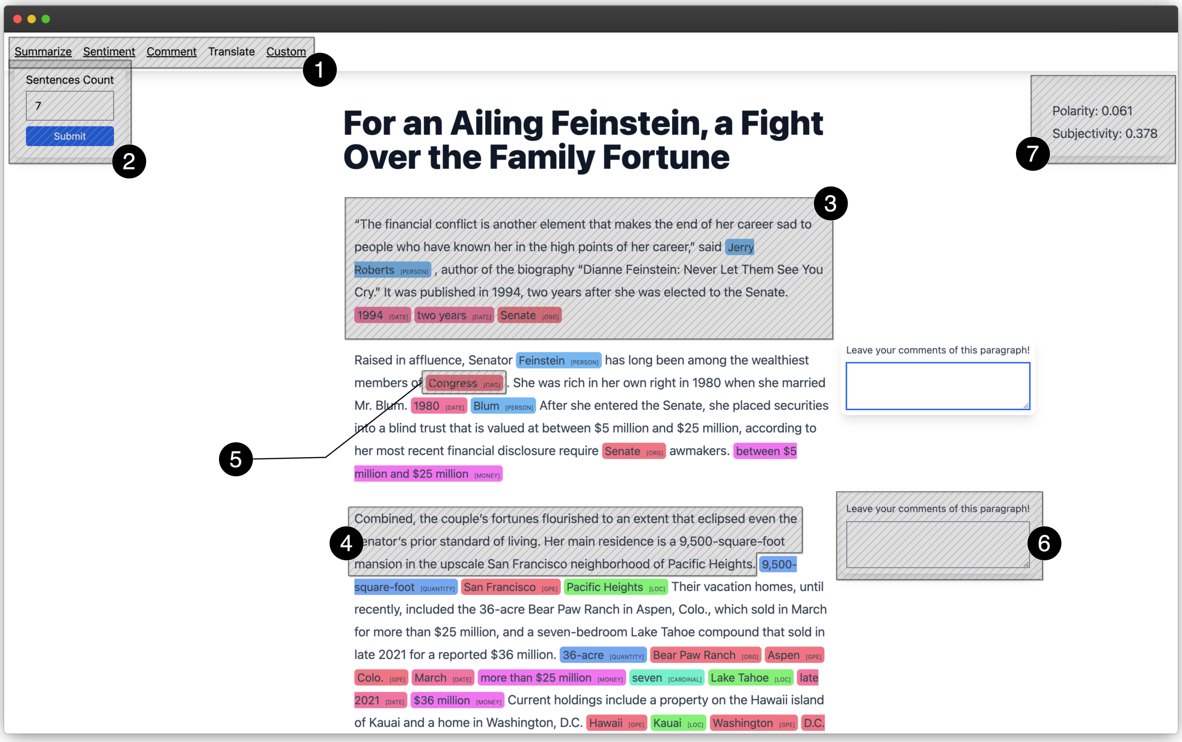
3.2 Configuration

INTELMO offers model developers a multi-layered abstraction to define tasks and build UI systems. Through this approach, the toolkit provides developers with various default styles and task types, while also allowing customization at each level.

3.2.1 Categorization of NLP Tasks Based on Interaction

NLP tasks are often divided based on the internal processes of the tasks (Dudhabaware and Madankar, 2014), which provides a clear structure for constructing evaluation metrics within specific categories. However, in the context of interactive interfaces, this approach becomes overly specific and less conducive to abstraction. For example, *Sentiment Analysis* and *Question Generation* over entire articles might belong to completely differ-

[2]https://jinja.palletsprojects.com/
[3]https://tailwindcss.com/


Summarize
Sentiment
Comment
Translate
Custom
1
Sentences Count
7
Submit
2
3
4
5
6
7
Leave your comments of this paragraph!
Leave your comments of this paragraph!

For an Ailing Feinstein, a Fight Over the Family Fortune

Polarity: 0.061
Subjectivity: 0.378

"The financial conflict is another element that makes the end of her career sad to people who have known her in the high points of her career," said Jerry Roberts [PERSON] , author of the biography "Dianne Feinstein: Never Let Them See You Cry." It was published in 1994, two years after she was elected to the Senate. 1994 [DATE] two years [DATE] Senate [ORG]

Raised in affluence, Senator Feinstein [PERSON] has long been among the wealthiest members of Congress [ORG]. She was rich in her own right in 1980 when she married Mr. Blum. 1980 [DATE] Blum [PERSON] After she entered the Senate, she placed securities into a blind trust that is valued at between $5 million and $25 million, according to her most recent financial disclosure require Senate [ORG] awmakers. between $5 million and $25 million [MONEY]

Combined, the couple's fortunes flourished to an extent that eclipsed even the enator's prior standard of living. Her main residence is a 9,500-square-foot mansion in the upscale San Francisco neighborhood of Pacific Heights. 9,500-square-foot [QUANTITY] San Francisco [GPE] Pacific Heights [LOC] Their vacation homes, until recently, included the 36-acre Bear Paw Ranch in Aspen, Colo., which sold in March for more than $25 million, and a seven-bedroom Lake Tahoe compound that sold in late 2021 for a reported $36 million. 36-acre [QUANTITY] Bear Paw Ranch [ORG] Aspen [GPE] Colo. [GPE] March [DATE] more than $25 million [MONEY] seven [CARDINAL] Lake Tahoe [LOC] late 2021 [DATE] $36 million [MONEY] Current holdings include a property on the Hawaii island of Kauai and a home in Washington, D.C. Hawaii [GPE] Kauai [LOC] Washington [GPE] D.C.

Figure 3: The UI of INTELMO has the following controllable elements: (1) Model function list; (2) Model parameters form; (3) Paragraph-level elements; (4) Sentence-level elements; (5) Word-level elements; (6) Extra elements of paragraphs; (7) Global elements over the entire article.

ent task categories under traditional classification. Nevertheless, the interactive operations required by these models are quite similar, which is generating one or more new page elements based on potential parameters and displaying them on the page. From an interactive perspective, these two tasks should be grouped together and abstracted using a unified approach.

By adopting this categorization, it's expected that model developers would generalize models from the perspective of human-model interaction. This approach can reduce the complexity of creating interactive interfaces, simplifying the workflow and APIs.

Based on the interaction behavior of models, we proposed the following categories for NLP tasks, as shown in Figure 4:

1. **Modification**: Making changes to the existing elements rather than adding new ones to the article. An example of this is *Information Filtering*, which means highlighting certain content within articles while fading other parts.

2. **Insertion**: Adding new paragraphs, sentences, or words to the article. This classification results in a change in the structure of the article's content. An example is *Machine Translation*.

3. **Generation**: Generating new information based on parts or the entirety of the article. This new information is displayed outside the article. One example of this is *Sentiment Analysis*.

4. **Cross-document Tasks**: Tasks of this nature may require access to or interaction with other articles, including *Named Entity Linking* and *Information Aggregation*. INTELMO provides specialized APIs for such tasks.

The NLP tasks depicted in Figure 4 are not exhaustive. Based on the characteristics of different categories, new tasks can be easily classified. Additionally, INTELMOoffers custom task types. By specifying this type in the configuration, models can access the entire RSS article information and precisely control the rendering of results.

3.2.2 Task Composition

If a model developer needs to showcase multiple tasks on a single page, INTELMO offers the following types of compositions:

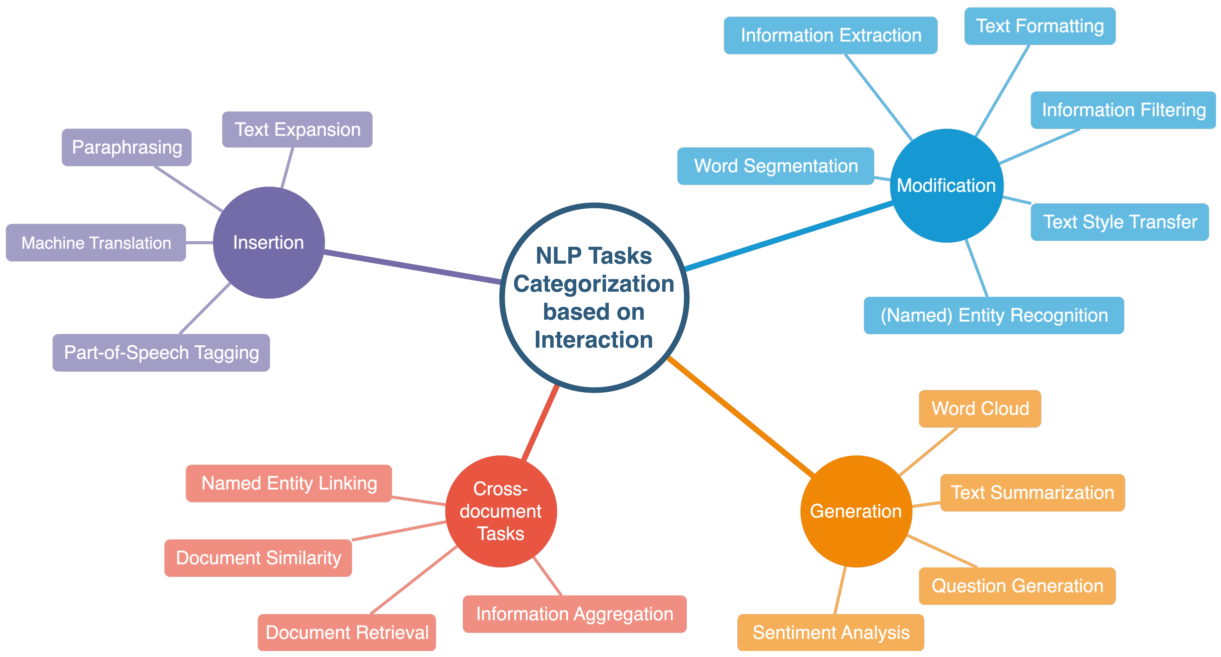


Figure 4: The classification of NLP Tasks based on interaction produces 4 basic categories: (1) Modification; (2) Insertion; (3) Generation; (4) Cross-document Tasks.

1. **Exclusive**: This mode disallows simultaneous execution of different tasks. When one task is activated, others are disabled.
2. **Compatible**: Multiple tasks can run concurrently. Each task gets the same original article. INTELMO takes care of rendering the combined results on the page.
3. **Pipelined**: Different tasks are connected sequentially, where each task function receives the output of the previous task function as its input. This mode is effective in systems with a complete workflow like MARVISTA (Chen et al., 2023).

These compositions could be applied recursively, allowing developers to achieve complex control flows by specifying nested compositions.

4 Discussion & Future Work

Traditional model interaction testing and deployment often require developers to possess web development skills or collaborate with front-end engineers. This decreases the efficiency of model development and testing, limiting the applicability of many models. INTELMO aims to bridge this gap by enhancing developers' development experience and efficiency through real-world datasets and a comprehensive UI framework. However, INTELMO is not currently ready to assist in building interactive pages for large-scale, long-latency complex models. Given the need to accommodate the requirements of all models, reducing user wait times will be a key focus for INTELMO's future optimization efforts.

Furthermore, we also recognize that obstacles faced by model developers include deploying models to existing web services such as Vercel[4] and Google Cloud Platform[5] and setting up CI/CD pipelines. We aspire to continue iterating on INTELMO and provide streamlined model deployment solutions.

References

Tilman Beck, Bela Bohlender, Christina Viehmann, Vincent Hane, Yanik Adamson, Jaber Khuri, Jonas Brossmann, Jonas Pfeiffer, and Iryna Gurevych. 2022. AdapterHub Playground: Simple and Flexible Few-Shot Learning with Adapters. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 61–75, Dublin, Ireland. Association for Computational Linguistics.

Carrie J. Cai and Philip J. Guo. 2019. Software developers learning machine learning: Motivations, hurdles, and desires. In *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 25–34.

[4]https://vercel.com/
[5]https://cloud.google.com/

Jose Camacho-collados, Kiamehr Rezaee, Talayeh Riahi, Asahi Ushio, Daniel Loureiro, Dimosthenis Antypas, Joanne Boisson, Luis Espinosa Anke, Fangyu Liu, and Eugenio Martínez Cámara. 2022. TweetNLP: Cutting-edge natural language processing for social media. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–49, Abu Dhabi, UAE. Association for Computational Linguistics.

Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. 2023. Grounding Large Language Models in Interactive Environments with Online Reinforcement Learning. ArXiv:2302.02662 [cs].

Khyathi Raghavi Chandu, Yonatan Bisk, and Alan W Black. 2021. Grounding 'Grounding' in NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4283–4305, Online. Association for Computational Linguistics.

Xiang 'Anthony' Chen, Chien-Sheng Wu, Lidiya Murakhovs'ka, Philippe Laban, Tong Niu, Wenhao Liu, and Caiming Xiong. 2023. Marvista: Exploring the Design of a Human-AI Collaborative News Reading Tool. ArXiv:2207.08401 [cs].

Rahul S. Dudhabaware and Mangala S. Madankar. 2014. Review on natural language processing tasks for text documents. In *2014 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–5.

Felix Faltings, Michel Galley, Baolin Peng, Kianté Brantley, Weixin Cai, Yizhe Zhang, Jianfeng Gao, and Bill Dolan. 2023. Interactive Text Generation. ArXiv:2303.00908 [cs].

Yu Gu, Xiang Deng, and Yu Su. 2023. Don't Generate, Discriminate: A Proposal for Grounding Language Models to Real-World Environments. ArXiv:2212.09736 [cs].

Haukur Páll Jónsson and Hrafn Loftsson. 2023. DMS: A System for Delivering Dynamic Multitask NLP Tools. In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence*, pages 504–510.

Mina Lee, Megha Srivastava, Amelia Hardy, John Thickstun, Esin Durmus, Ashwin Paranjape, Ines Gerard-Ursin, Xiang Lisa Li, Faisal Ladhak, Frieda Rong, Rose E. Wang, Minae Kwon, Joon Sung Park, Hancheng Cao, Tony Lee, Rishi Bommasani, Michael Bernstein, and Percy Liang. 2023. Evaluating Human-Language Model Interaction. ArXiv:2212.09746 [cs].

Pengfei Liu, Jinlan Fu, Yang Xiao, Weizhe Yuan, Shuaicheng Chang, Junqi Dai, Yixin Liu, Zihuiwen Ye, Zi-Yi Dou, and Graham Neubig. 2021. ExplainaBoard: An Explainable Leaderboard for NLP. ArXiv:2104.06387 [cs].

Edoardo Mosca, Daryna Dementieva, Tohid Ebrahim Ajdari, Maximilian Kummeth, Kirill Gringauz, and Georg Groh. 2023. IFAN: An Explainability-Focused Interaction Framework for Humans and NLP Models. ArXiv:2303.03124 [cs].

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2022. WebGPT: Browser-assisted question-answering with human feedback. ArXiv:2112.09332 [cs].

Yifan Song, Weimin Xiong, Dawei Zhu, Cheng Li, Ke Wang, Ye Tian, and Sujian Li. 2023. RestGPT: Connecting Large Language Models with Real-World Applications via RESTful APIs. ArXiv:2306.06624 [cs].

Zijie J. Wang, Dongjin Choi, Shenyu Xu, and Diyi Yang. 2021. Putting Humans in the Natural Language Processing Loop: A Survey. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 47–52, Online. Association for Computational Linguistics.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. 2023. WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents. ArXiv:2207.01206 [cs].

A Supplementary Files

The code is open-sourced under MIT license on Github[6]. A supplementary video demo is hosted at Vimeo[7].

[6]`https://github.com/INTELMO/intelmo/`
[7]`https://vimeo.com/852034145/`