Eval-UA-tion 1.0: Benchmark for Evaluating Ukrainian (Large) Language Models

Serhii Hamotskyi[1], Anna-Izabella Levbarg[2], Christian Hänig[1]
[1] Anhalt University of Applied Sciences
Bernburger Str. 55, 06366 Köthen, Germany
[2] University of Greifswald
Domstraße 11, 17489 Greifswald, Germany
{serhii.hamotskyi, christian.haenig}@hs-anhalt.de, anna-izabella.levbarg@uni-greifswald.de

Abstract

We introduce Eval-UA-tion, a comprehensive suite of novel Ukrainian-language datasets designed for the evaluation of language model performance in the Ukrainian language. The collection encompasses a variety of tasks: UA-CBT (inspired by the Children's Book Test, a fill-in-the-blanks task aimed at assessing comprehension of story narratives), UP-Titles (requiring the association of articles from the online newspaper Ukrainska Pravda with their correct titles from a set of ten similar options), and LMentry-static-UA/LMES (modeled after the LMentry benchmark, featuring tasks that are straightforward for humans yet challenging for language models, such as determining the longer of two words or identifying the Nth word in a sentence). Except for UP-Titles, these tasks are designed to minimize potential contamination, utilizing material unlikely to be found in language models' training datasets. They also include a split specifically for few-shot prompting to further reduce contamination risks. For each task, we provide benchmarks against both human and random performance baselines.

Keywords: LLM Evaluation, Benchmark Dataset, Ukrainian language

1. Introduction

The Ukrainian language has a strong online presence: as of October 2023, estimates of languages used on the internet put Ukrainian at place 19 (Wikipedia contributors, 2023) (between Arabic and Greek); Ukrainian Wikipedia is 15th by number of daily views and number of articles (Meta, 2022). Though an increase of Ukrainian use online can be traced to the Russian attack on Crimea in 2014 (Kulyk, 2018), the full-scale invasion of 2022 accelerated this process, as seen surveys (Group, 2022) and Twitter data (Racek et al., 2024), showing that ≈5% predominantly Russian-tweeting users made a hard switch to Ukrainian in the first months of the invasion. This shows that the need to support the Ukrainian language is stronger than ever.

On a 2020 survey of linguistic diversity in NLP (Joshi et al., 2020), the Ukrainian language was classified as belonging to the "rising stars": languages with a thriving online cultural community that benefits from unsupervised pretraining, but let down by an insufficient amount of *labeled* datasets. A recent review of the performance of LLMs on non-English languages found a very uneven performance based on language used, with ChatGPT performing best in English (Lai et al., 2023)[1]. With the widespread adoption of LLMs these differences become more important, and so is their measurement.

Aiming to increase the availability of labeled Ukrainian datasets and stimulating future and existing efforts on this topic, we present Eval-UA-tion 1.0, a set of benchmark datasets usable for evaluating the performance of LLMs in and on the Ukrainian language.

The issue of data contamination (generally defined as exposure of the model to data similar to the one it would later be tested on) has received much attention in recent years (Roberts et al., 2023). We placed a special emphasis on using sources of data that maximally limit contamination.

Most of the source code and sanitized raw data used to generate the datasets will be publicly available in the Eval-UA-tion Github repository[2].

1.1. Relevant Ukrainian Grammar and Notation

Ukrainian has 3 grammatical genders: female, male, and neutral (in this paper abbreviated as F, M, and N), 7 cases (including nominative/NOM, genitive/GEN, locative/LOC), and 2 numbers (singular/SG and plural/PL). It has a complex morphology with many parts of speech needing agreement, especially by gender and case. Numerals can be ordinals/ORD (*first*), cardinals/CARD (*one*) and adverbial.

The notation used is loosely based on the Leipzig Glossing Rules (Comrie et al., 2008), with the relevant morphemes annotated in the

[1] Ukrainian is an interesting outlier in that study as the only language where English prompts outperformed the language-specific (Ukrainian) ones for Relation Extraction on the SMiLER (Seganti et al., 2021) dataset.

[2] https://github.com/pchr8/eval-UA-tion

The Third Ukrainian Natural Language Processing Workshop (UNLP) @LREC-COLING-2024, pages 109–119

superscripts of words. The English translation of the relevant words will be divided from the morphemes by a dash, and the individual morphemes will be separated from each other by dots: *чоловік*$^{man\text{-}NOM.SG}$ *побачив*$^{saw\text{-}M.SG}$ *собаку*$^{dog\text{-}ACC.SG}$.

2. Related Work

A very thorough overview of the current landscape of benchmarking approaches can be found in (Guo et al., 2023). On LLMs' performance on non-English languages, see Akter et al. (2023) and Lai et al. (2023).

A number of efforts are underway to create Ukrainian-language datasets and benchmarks, a notable one being UA-datasets (Ivanyuk-Skulskiy et al., 2021)[3], with the development of UA-SQuAD and UA News classification in progress as of 04.03.2024 and the Mova Institute POS dataset completed. All three datasets are considerably larger than the ones we are proposing and have been a direct inspiration for us.

Loosely related to our manual correction of LLM-generated stories is the topic of grammaticality in general. UA-GEC (Syvokon and Nahorna, 2022) is a large grammatical error correction corpus separately annotating fluency, grammar, punctuation, and spelling errors.

3. Eval-UA-tion 1.0 Benchmark Datasets

3.1. UA-CBT

3.1.1. Introduction

The UA-CBT[4] dataset builds upon the idea introduced in the English-language Children's Book Test (CBT) benchmark dataset (Hill et al., 2015).

The core idea is the following: a word in a story gets masked (replaced by "_____", hereafter referred to as 'gap') and six options are offered as potential replacements, only one being correct.

3.1.2. Differences from the Original CBT Task

UA-CBT differs from the CBT benchmark in multiple aspects (and through the challenges introduced by the rich morphology of the Ukrainian language).

In the original CBT implementation, the story context was 20 sentences long, with a word in the 21st sentence masked. In UA-CBT, to increase the number of tasks per story, the split is 65% context segment and 35% challenge segment. The number of possible options is reduced from 10 to 6. We additionally omitted prepositions from the question categories, keeping named entities, common nouns, and verbs.

The (2015) CBT task is built from stories from books freely available on Project Gutenberg[5] and the authors explicitly state that they wanted to incentivize models to apply background knowledge and information when solving the tasks — we attempted to avoid that by using original stories and limit the background knowledge usable to story cliches that aren't always applicable[6]. Lastly, the task instances were manually filtered to ensure the dataset contains only unambiguous solvable questions.

3.1.3. Description

The dataset contains **1,061** task instances built on **72** different stories. There are three types of tasks/gaps: NAMED_ENTITY for the characters ('Butterfly'), COMMON_NOUN for inanimate items ('valley', 'water') and VERB for verbs ('fly', 'eat'). Each instance is a multiple-choice question with 6 options.

Distractors For each gap, six different options are provided, five of them are *distractors* (wrong answers). Three to five distractors come from the story itself. To make them plausible, only the lemmas[7] most frequently found in the text are used. All are filtered and inflected to match the morphology of the original word in the gap. For example, in the task shown in Fig. 1, the replacements for *Мисливця*$^{hunter\text{-}M.GEN}$ are all grammatically male and GEN case as well (with the exception of *Змії*$^{snake\text{-}F.GEN}$, described later); all use the same capitalization as the original word (in the story, 'The Hunter' is used in the role of a proper name and is, therefore, capitalized).

If the story doesn't have enough entities usable as distractors (e.g. only one grammatically female character for NAMED_ENTITY), they are sourced in the following order: 1) If the story's most frequently mentioned entity has a different gender than the gap, it's added as a most-frequent-any-gender distractor, marked as a red "F" in Fig. 1; 2)

[3] https://fido-ai.github.io/ua-datasets/

[4] https://hf.co/datasets/shamotskyi/ua_cbt

[5] https://www.gutenberg.org/

[6] The stories, being generated by LLMs and corrected only for logic but not for plausibility, contain atypical elements such as a turtle eating the remains of a zebra: may raise a human's eyebrows, but may be even more confusing to an LM that expects animals to fit archetypal folk tale roles.

[7] different inflections of the same word counted as one (e.g. *кіт, кота, котами*)

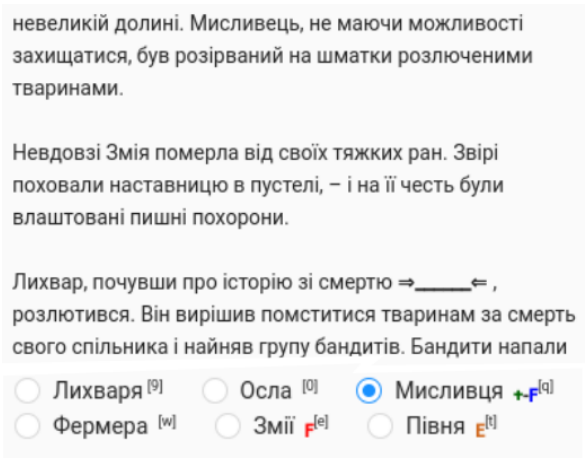

Figure 1: A (partial) sample UA-CBT task. The markings near the options are the ones shown to the annotators during the task filtering process: "E" means the option was taken from an external list of (in this case) male entities, a blue "F" denotes the most frequent relevant word in the text, a *red* "F" is the most frequent word in the text regardless of its gender (and here *змія*$^{snake\text{-}F}$ is the only grammatically female word), and "+" is the correct option.

An external list of words is used, from which the remaining distractors are randomly chosen. The options are then shuffled and deduplicated.

Gaps Only frequent lemmas become gaps. Masking rare words would have increased the chances of a gap being placed on a one-off entity that's not part of a coherent narrative. Lemma frequency for gaps was calculated only up to the gap itself. For verbs and named entities, at least **two** occurrences were needed, for common nouns **four**. The higher minimum occurrences limit for common nouns was needed because many of the stories contained generic endings that resulted in uninteresting tasks, solvable by completing cliches instead of understanding the story narrative ("...and the animals learned that the real treasure is `[friendship|food|fear|...]`, and they `[lived|ate|traveled|...]` together happily ever after"). The three kinds of gaps in more detail:

NAMED_ENTITY animate nouns and proper nouns; usually the main characters in the story ('Butterfly'/*Метелик*)

COMMON_NOUN inanimate nouns; usually objects like 'water' or 'desert', but overlaps heavily with NAMED_ENTITY (because animals weren't always detected as animate by the spacy model we used)

VERBS finite and infinitive

3.1.4. Dataset structure

The dataset is published on the Huggingface Hub[8] with five predefined subsets: NAMED_ENTITY (615 instances), COMMON_NOUN (281), VERBS (165), 'all' with the complete dataset (1,061), and a few-shot split (7 instances based on a separate story). The latter's purpose is avoiding contamination during few-shot prompting (randomly selecting instances for this purpose might lead to the few-shot examples using the same story as the test instance).

The columns are described in the README of the dataset. Notable ones are:

context, question the story segments
options, answer the options and correct answer
taskType gap type (COMMON_NOUN, ...)
storyId unique identifier of the story used

A large amount of other metadata is included, such as the source of each distractor, the size of the segments, and metadata from the story generation stage (e.g. which model was used; see Section 3.1.5).

3.1.5. Story Generation and Filtration

Roberts et al. (2023) describe contamination as composed of two distinct phenomena: *contamination* proper, which refers to an LLM's exposure during training to examples similar or identical to the ones the model will later be evaluated on, and *memorization*, the ability to extract (near) verbatim the examples the model has seen during training. When generating stories for this task, the latter facet was at the forefront. Many sources of stories were considered and rejected. The crux of the issue was that stories not widely available online were unusable for intellectual property reasons, while public domain stories were often available online and, therefore, basically guaranteed to be part of the training data of current (and future) LLMs. Our stories were generated using OpenAI *gpt-4-1106-preview*[9] and Google Gemini Pro[10], followed by manual review and correction. The main challenge we faced was that the LLM would recite a memorized story instead of writing a more original one, thereby contaminating the dataset.

We mitigated this issue by using detailed **prompts**. For example, if the prompt asks for a story about a raven and a fox, the names and details would vary but the story will almost always be about the fox tricking the raven into giving it a piece of cheese, as in the well-known Aesop fable. But if

[8] https://hf.co/datasets/shamotskyi/ua_cbt
[9] https://platform.openai.com/docs/models
[10] https://deepmind.google/technologies/gemini/

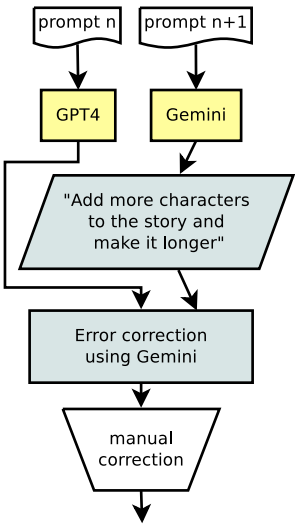

Figure 2: The flow used to create UA-CBT stories.

```
options:
    - not learning anything
    - helping their mentor with {problem_type} problem
    - resolving a dispute involving {dispute_topic}
    - proving that they are a good {profession}
    - rescuing {entity} from {rescue_from}
    - proving their innocence
parts:
    problem_type:
        - an embarassing
        - an unexpected
        - a recurring
        - a financial
        - a communication
        - "a totally predictable"
    dispute_topic:
        - lost food
        - stolen food
    profession:
        - friend
        - tailor
        - hunter
    entity:
        - a relative
        - a lost traveler
    rescue_from:
        - a tornado
        - the cold
        - captivity
```

Figure 3: Part of the template used to generate story generation prompts.

the prompt asks for ”a story about a *greedy* raven *rescuing* a fox from *a tornado*”, there’s a much smaller set of pre-existing stories fitting the criteria to recite verbatim, resulting in more creative stories. A number of such elements in the template were randomized, such as asking it for stories in the style of Ukrainian/Arabic folk tales, changing the number of main/minor characters, etc. Lastly, specifying that the story should have an unhappy ending often increased the originality of the entire story, so half of the prompts required stories with unhappy endings.

Generating these prompts involved sampling a subset out of all possible permutations of values in the templates. Part of the YAML file containing the source data (redacted for brevity) is shown in Fig. 3. The need for logic, consistency, and a coherent structure and *recurring* characters was emphasized, since this was needed to be able to create a story from which a higher number of solvable task instances could be generated. Otherwise, the bulk of the prompt was static and contained criteria for the story. It specified the naming of the characters, the complexity of the story, and instructions aimed at avoiding specific recurring motifs (e.g. prompts involving specific objects, such as bread, often defaulted to a narrative centered on *magic bread* rather than incorporating bread in a conventional role).

Half of the stories were generated using *gpt-4-1106-preview* and half using Gemini Pro. In our experience, the OpenAI model followed instructions (such as number of characters) more reliably, while the Gemini model had dramatically better Ukrainian grammar (which agrees with the literature; compare with Akter et al. (2023)). We leveraged Gemini Pro’s Ukrainian language abilities by piping all the stories through it after generation, instructing it to improve their logic, consistency, and grammar, with good results. We mitigated its tendency to generate shorter and simpler stories than required by using a chat interface, and asking it after its first attempt to ”add more major/minor characters to the story make it longer, while keeping it logically consistent”, with good results.

The flow for both models is shown on Fig. 2.

Manual story correction and filtration was done by human annotators based on the stories produced after the above steps, in a Label Studio[11] environment. For each story, the annotators were given a choice of fixing the errors in the story or marking it as completely unusable. Reasons for the latter included continuity errors that required substantial rewriting to fix, a large number of errors in gender agreement or entities having adjectival names (e.g. a rabbit named Quick), or having too few characters.

Out of the 117 generated stories, 72 (62%) were considered usable and subsequently manually corrected. A typology of errors found during this process is out of scope of this paper, but the main language issues found were noun agreement (with nouns that have a different gender in Ukrainian and Russian using the Russian gender), the use of Russian words and phrases, and strange and often funny fluency errors. Issues in the logic involved illogical actions by the characters (such as money being returned to the wrong character) and continuity issues (e.g., a character giving advice despite having died two paragraphs ago).

The before-and-after stories dataset[12] is available on request.

[11] https://labelstud.io/

[12] https://hf.co/datasets/shamotskyi/ua_cbt_stories

3.1.6. Human Filtration of Task Instances

Departing from the approach taken by the original CBT task, we manually filtered all generated task instances to remove unsuitable ones. Of the 1,418 manually processed instances only 1,063 (75%) were deemed suitable.

Here a more extensive taxonomy was created, with two main classes of errors:

1. Logic/continuity errors
   (a) Answer unknown: the story doesn't contain the information that allows the answer to be inferred. Example: "The Cat and the Turtle go to `[Cat|Turtle|Lion]`'s house to sew the coat, and later deliver it to the Lion's house".
   (b) Multiple options are correct: it's clear which entity/action is involved, but it can be described in different ways. Example: "The Lion liked the Cat and Turtle's `[coat|work]`."This accounts for approx. 24% of unusable tasks and was the largest category.
   (c) Duplicate options: multiple almost identical options referring to the same thing, e.g. bird/birdie. Caused by incorrect lemmatization.
2. Language errors
   (a) Ungrammatical option: one of the options is a non-existing word. Caused by failures in the parsing-normalization-inflection pipeline. Examples from the dataset include **друзь*[13] and **комаревом*.
   (b) Incorrectly inflected option: an option is an existing grammatical word, but is a different inflection than needed. Usually caused by an incorrectly detected morphology of the masked word.

Both error classes are roughly equally distributed. We see this taxonomy and breakdown as a stepping stone towards fully automated filtering of task instances, eventually leading to larger datasets of this type.

3.1.7. Baselines

The human baseline accuracy result for this task was **94%**: 6 wrong out of a total of 99 test instances. This score is based on answers by 8 different annotators inside a Telegram[14] bot. The random baseline for this task is **16.7%** (6 possible options). The most-frequent baseline of this task (choosing the option most frequently seen in the story) is **57%** (in other words, in 57% of the tasks the correct answer is simply the most frequently mentioned lemma). This is visualized in Fig. 4.

3.2. LMentry-static-UA (LMES)

3.2.1. Description

LMentry-static-UA (LMES) is a set of 6 loosely related datasets inspired by the (English-language) LMentry (Efrat et al., 2022) benchmark. It focuses on tasks considered trivial for humans but harder for LMs.

The six included tasks are:

1. N-in-M–type tasks:
   (a) LOW[15] (letters of word): "What is the first/Nth/last letter in the word ..."
   (b) WIS[16] (words in sentence): "What is the first/Nth/last word in this sentence:..."
2. Tasks involving categories:
   (a) CATS-MC[17] (multiple choice): "Which of these words is different from the rest?"
   (b) CATS-BIN[18] (binary): "Do all of these words belong to the category 'emotions'?"
3. Comparing-two-things-type tasks:
   (a) WordAlpha[19]: "Which of these words is first in alphabetical order?"
   (b) WordLength[20]: "Which of these words is longer?"

3.2.2. Differences from LMentry

LMentry represents a comprehensive framework that includes evaluation code[21], assesses the models' accuracy and robustness to perturbations, and extends beyond the scope of our (static) dataset in many ways. The two commonalities lie in the tasks themselves and in a focus on investigating the robustness of LMs to changes in the templates.

LMES focuses on tasks that can be evaluated as a dataset (as opposed to regular expressions in the original benchmark), hence 'static'. This necessitated dropping some tasks, such as "write a sentence/word that contains/(starts/ends with) the word/letter X." A number of other tasks were also dropped.

[13] Following linguistic conventions, ungrammatical words will be denoted by a leading asterisk.

[14] https://telegram.org/

[15] https://hf.co/datasets/shamotskyi/lmes_LOW

[16] https://hf.co/datasets/shamotskyi/lmes_WIS

[17] https://hf.co/datasets/shamotskyi/lmes_catsmc

[18] https://hf.co/datasets/shamotskyi/lmes_catsbin

[19] https://hf.co/datasets/shamotskyi/lmes_wordalpha

[20] https://hf.co/datasets/shamotskyi/lmes_wordlength

[21] https://github.com/aviaefrat/lmentry

The remaining tasks were regrouped, merged together, and expanded. For example, the original benchmark considered ”what’s the first/last ...” separate tasks. We merged them into one and expanded by adding questions about specific numbers (”What’s the fifth ...”).

3.2.3. Datasets Structure

The datasets have been uploaded on HuggingFace Hub as individual datasets, each with a separate few-show split that uses different sentences/words/categories than the train split to reduce contamination.

As a variation of what the LMentry benchmark terms *robustness*, our LMES tasks place a heavy emphasis on the use of different templates with the same input. For example, ”Which word is longer: ’dog’ or ’cat’?” would also ask which word is *shorter*, would ask the same question reversing the order of the words, ask which word has more letters, etc. The specific changes to the template are contained in each task instance metadata to simplify analysis. The tasks involving words also include extensive metadata about the words, such as which part of speech they are, their frequency, their length, etc.

An analysis of the impact is outside the scope of this paper, but we hope it will stimulate research in this direction.

3.2.4. Dataset Construction

Since contamination is not an issue for the tasks involved (e.g. a sentence being in the training set of a LLM doesn’t increase the odds of it knowing what’s the third word in it), we used the UP-Titles (see subsection 3.3) dataset and the example sentences in spacy as sources for the sentences.

The words were taken from the David Klinger Ukrainian dictionary[22], which in turn uses DBnary (Sérasset, 2015) and WikiDictionary. We removed words containing apostrophes or dashes (to ensure clarity if counting letters is needed, e.g. the sixth letter in the word *пліч-о-пліч* depends on what is considered a letter). We left only nouns, verbs, adjectives, and adverbs; then we binned word frequency into high, mid, and low frequency. Then for each POS+frequency pair we sampled 60 words (or the number words available if it’s less than 60), leading to a diverse choice of words.

3.2.5. Ukrainian Morphology in the Templates

The templates used in the LOW/WIS tasks involved converting integers (4) into natural-language words, which were represented by numerals of different types (ordinal and cardinal) and involved agreement in gender and case. For instance, asking for the first word in a sentence could be formulated as:

1. ***Перша***$^{\textit{first-F.ORD.NOM}}$ *літера*$^{\textit{letter-F.NOM}}$
2. *Літера*$^{\textit{letter-F.NOM}}$ *номер* ***один***$^{\textit{one-CARD.NOM}}$
3. *На* ***першому***$^{\textit{first-N.ORD.LOC}}$ *місці*$^{\textit{place-N.LOC}}$

We found no library that supported such arbitrary conversions. An additional challenge was keeping track of the numeral type and morphology required by each template.

We solved the latter problem by capitalizing the numeral directly in the template string: *На ПЕРШОМУ місці знаходиться...* When using the template to generate task instances, the target morphology and numeral type are parsed from the capitalized numeral in the template, and the needed number is inflected correspondingly and put in the place of the capitalized numeral.

We release the code for the number-to-numeral conversion as a library, *ukr_numbers*[23], currently in beta. It uses pymorphy2[24] and num2words[25].

To the best of our knowledge, using natural language inside templates instead of requiring the user to manually specify the required inflection is a novel idea.

3.2.6. Baselines

The human and random baselines are shown on Table 1 and on Fig. 4.

3.3. UP-Titles

3.3.1. Description

UP-Titles is a multiple-choice dataset with 5,000 instances, where each article needs to be matched to the correct title, out of 10 similar titles. It’s built from the ukr_pravda_2y[26] dataset, which contains articles from the Ukrainska Pravda[27] (UP) newspaper, published in the years 2022-2023. It’s provided in a masked[28] and an unmasked[29] version (see below).

For each article text, its title and the titles of 9 most similar articles are given as choices. Article similarity is estimated through a simple cosine distance over article tag binary vectors: articles with the same tags will have a similarity of 1, and ones with no tags in common will have a similarity of 0.

[22] https://github.com/dmklinger/ukrainian

[23] https://github.com/pchr8/ukr_numbers

[24] https://github.com/pymorphy2/pymorphy2

[25] https://github.com/savoirfairelinux/num2words

[26] https://hf.co/datasets/shamotskyi/ukr_pravda_2y

[27] https://pravda.com.ua

[28] https://hf.co/datasets/shamotskyi/up_titles_masked

[29] https://hf.co/datasets/anilev6/up_titles_unmasked

	num_total	num_wrong	bl_random	bl_human
UP-Titles (unmasked)	99	12	10.00	87.88
UP-Titles (masked)	98	16	10.00	83.67
LMES-wordalpha	98	8	50.00	91.84
LMES-wordlength	100	6	50.00	94.00
LMES-cats_bin	99	3	50.00	96.97
LMES-cats_mc	100	2	20.00	98.00
LMES-LOW	100	3	9.43	97.00
LMES-WIS	100	6	4.69	94.00
UA-CBT	99	6	16.67	93.94

Table 1: Random and human baselines for the datasets part of this benchmark. num_total refers to the total size of the human-evaluated subset of the dataset, num_wrong is the number of instances where the human answer differs from ground truth; bl_random and bl_human are the random and the human baselines respectively. bl_random can be interpreted as the probability of randomly guessing the correct answer: bl_random = $\frac{1}{\text{num\_total}} \sum_{i=1}^{\text{num\_total}} \frac{1}{M_i}$, where M_i is a number of answer options in the i-th task instance. The random baselines were calculated on the complete datasets.

Most instances would be trivial to solve by matching by the numbers mentioned in the title and the article text — e.g. if an article text contains the number 232 (prisoners of war, dead russians, millions of dollars...) it's a very safe bet that whichever title contains that same number is the correct one. To mitigate that, we replace all integers in the article text and article titles with "X" (leading to titles such as "Bucha Mayor: XXX civilians killed by Russian troops identified").

The solution doesn't remove all potential clues: among others, numerals written as text ('twenty-three'), months, names of individuals stay unchanged. Nevertheless, this simple masking approach complicates the task by a surprising amount, in some rare cases rendering it unsolvable (see discussion below about human baselines), and we believe a more thorough masking would bring diminishing returns while increasing the number of unsolvable instances even further.

The dataset is provided in two versions: with masked and unmasked numbers. We evaluated the masked and unmasked versions of the dataset separately, and the masked option was harder for both human annotators and LLMs.

It's released under the CC BY-NC 4.0[30] license, reflecting Ukrainska Pravda's terms[31] forbidding the use of its articles for commercial purposes.

3.3.2. Baselines

The random baseline for this task is **10%**. The human baseline was **84%** for the masked and **88%** for the unmasked version.

The low human baseline may be explained through different means, with the most likely ones being: 1. The title doesn't contain the information needed for disambiguation ("Another XXX Russians killed in Ukraine" would fit many articles written in the last two years); 2. Human error, inability to correct a wrong answer due to bot interface limitations.

[30] https://creativecommons.org/licenses/by-nc/4.0/

[31] https://www.pravda.com.ua/rules/

4. Experiments

4.1. Evaluation Process

The datasets have been evaluated on five different models aiming to provide a baseline for the tasks. Baselines were calculated using the EleutherAI evaluation harness[32] (lm-eval).

All of the tasks in our benchmark can be seen as multiple-choice ones, and there are multiple approaches to leveraging LLMs for solving such tasks (Robinson et al., 2023). In cloze prompting, a question is passed to the LLM and the probabilities it gives to the different answers are compared, and the option given the highest probability by the model is used as prediction. We used multiple choice prompting (MCP), where the question and if applicable the possible answers are provided to the model in the prompt, structuring it in such a way that the model predicts a single token. For the UA-CBT and UP-Titles tasks this involved converting the list of possible answers into an enumerated list, e.g. "A: cat; B: dog; C: uncle". For the UP-Titles datasets, parentheses were used to avoid conflicts with article titles containing semicolons. Additionally, all newlines in the stories and UP articles were replaced by spaces. For the LMentry tasks, no letters were used, with the prompt expecting the correct word/letter [33] or *так/ні* (yes/no)

[32] https://github.com/EleutherAI/lm-evaluation-harness/

[33] The LOW/WIS random baselines were calculated as if they were a multiple-choice question with the op-

for the LMES-cats_bin task.

The prompts used were all in Ukrainian and all tasks were evaluated in a 3-shot setting. Due to time and budgeting constraints, the OpenAI models evaluated only 200 instances of the UA-CBT and UP-Titles tasks and 500 instances of all LMES tasks; the other models were evaluated on the entire dataset.

One known limitation of the lm-eval harness is the lack of support for models' instruction formats to leverage instruction finetuning. Practically speaking, in our experiments all models used the same 3-shot prompting without any model-specific prompt finetuning. Even small changes to prompt templates can drastically change model scores, and our goal is to provide a baseline instead of maximizing accuracy by finetuning individual models' instruction prompt.

The lm-eval YAML task implementations (including the exact prompts and modifications) are posted in the Eval-UA-tion GitHub repository to ensure reproducibility.

4.2. Evaluation Results

The models tested were *gpt-3.5-turbo*, *gpt-4-1106-preview*, *mistralai/Mistral-7B-Instruct-v0.2*, *Radu1999/Mistral-Instruct-Ukrainian-slerp*, and *SherlockAssistant/Mistral-7B-Instruct-Ukrainian* (Boros et al., 2024) (the winner of the UNLP-2024 shared task), all from the Huggingface Hub. The results are shown on Fig. 4.

The *SherlockAssistant/Mistral-7B-Instruct-Ukrainian* model outperformed the other non-OpenAI models for all tasks and outperformed GPT3 for both UP-Titles tasks. Notably, that model was not finetuned on Ukrainian news datasets.

The effect of masking/unmasking numbers in the UP-Titles dataset can clearly be seen: masking decreased the scores of the models.

GPT4 outperformed or roughly equaled models on all tasks, most dramatically for the UA-CBT task; it also beat the human baselines for both versions of the UP task and UA-CBT. This may point either towards inattention being the source of the human errors on it, or the presence of UP articles in its training dataset. Splitting the UA-CBT instances by story generation model, the scores were practically identical for both subsets, at 0.97 (SD 0.17/0.18 for GPT4/Gemini). So instances from stories generated by Gemini and improved by Gemini weren't harder for GPT4 than the instances based on stories that it generated.

tions being the letters/tokens of the word/sentence, but the actual evaluation involved simply comparing the predicted output with the exact expected ground truth value.

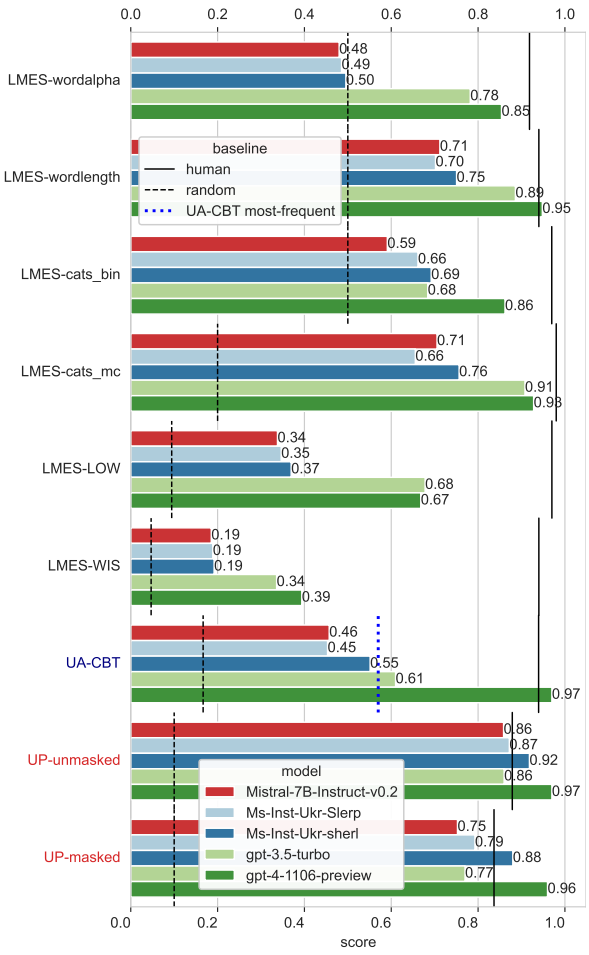

Figure 4: Evaluation scores of selected models.

5. Approaches to Human Data Annotation and Baseline Creation

For the presented datasets, volunteering contributors were found amongst family, friends, and through Telegram channels. This was coordinated in a group chat where instructions were given and annotators' questions answered. Initially, we employed Label Studio for tasks such as correcting LLM-generated UA-CBT stories and manual filtering. However, recognizing the need for a more streamlined and accessible method, we subsequently introduced a Telegram bot to simplify the process. A poll among our contributors regarding their preferred method of data annotation revealed a unanimous preference for the Telegram bot. To increase engagement, we incorporated simple gamification elements in the bot - transforming any button presses into animated emojis, which proved to be an effective strategy to maintain user interest and participation (Raftopoulos, 2015). Remarkably, this approach enabled a more rapid collection of data (compared to the same bot without gamification). This underscores the potential of this method as a valuable strategy for

data annotation. Ultimately, the choice of platform should not be restricted to what we used; it heavily depends on the demographics.

6. Limitations

6.1. UP-Titles

Since the UP-Titles dataset was built from articles of a well-known online newspaper (eighth most cited source in Wikipedia in 2017 (Lewoniewski et al., 2017)), the already discussed issues of contamination/memorization apply to it: it's very likely that the articles are and/or will be part of the training data of LLMs. Most of the articles from the dataset involve the Russian-Ukrainian war, with predictable effects on the language used (both topic-wise and through the changes in the vocabulary (Synchak, 2023) in that context).

6.2. UA-CBT

Half of the stories were generated using GPT4 and half using Gemini, then all were piped through Gemini to improve grammar and consistency. This raises the question of encapsulation: testing a model on tasks generated (even partially) with its output would lead to inflated scores. GPT4's very high scores on this task would seem to confirm this, but its performance on pure-Gemini stories was just as high. Nevertheless, the fact that all of the stories were 'touched' by Gemini and half by both Gemini and GPT4 is context crucial for the interpretation of scores of either of these models on the dataset.

Due to the limited number of annotators, multiple questions based on the same story could have been shown to the same annotators, who could have memorized the token in the gap from a previous task instance. This could have contributed to a higher human baseline. The Telegram bot did not allow going back to an already answered question, so the inability to fix errors could have had the opposite effect. We don't believe either to have been significant.

7. Discussion

We acknowledge the potential risks associated with the datasets introduced, particularly their utility in enhancing AI-driven bots for malicious political influence on social media (Radivojevic et al., 2024) (Eady et al., 2023) (Stukal et al., 2017), especially during the ongoing war. We advocate for an open proactive approach to exploring various classifiers and AI methods for the detection of malicious instances. During the generation and human filtration of task instances (see Section 3.1.6), we found clear patterns in the errors. We think some of the errors found were specific to Ukrainian, and that leveraging them could be a promising avenue of future research parallel and complementary towards existing research focusing on language-independent bot detection. The influence of a native tongue on a second language, known as language interference, is established in the literature. If these patterns are different in humans (e.g. most bilingual speakers in Ukraine) and LLMs (trained on multilingual data containing a significant amount of Russian), this could become basis of a classifier.

We evaluated two models that were fine-tuned on Ukrainian datasets and/or instructions. Among these models, the Sherlock model demonstrated superior performance when compared to the vanilla Mistral-7B model. We believe a more thorough analysis using more models and different evaluation approaches would be beneficial and would confirm the finding that fine-tuning on Ukrainian data improves performance on Ukrainian tasks.

An additional avenue for future research would be to systematically evaluate models tuned on Russian language, and quantify the impact on the scores. Evaluating instruction-finetuned models in a way that takes advantage of it by using proper templates would allow deeper insights into this.

8. Acknowledgments

We are grateful to Daria Kravets, Mariia Tkachenko, Oleksii K., Lina Mykhailenko, @arturius453, and Viacheslav Kravchenko for their contribution to the datasets and human baselines creation.

9. Conclusion

This paper presents a significant stride towards enhancing language model performance in Ukrainian through Eval-UA-tion. By introducing novel datasets, we provide a comprehensive evaluation framework that assesses models' abilities. Our work highlights the essential need for linguistic diversity in AI, with a focus on Ukrainian as a case study. Despite acknowledging our approach's limitations, such as potential memorization and contamination risks, we suggest directions for future research to refine and broaden our methodologies. Our contributions aim to advance more inclusive and representative language technologies.

10. Bibliographical References

Syeda Nahida Akter, Zichun Yu, Aashiq Muhamed, Tianyue Ou, Alex Bäuerle, Ángel Alexander Cabrera, Krish Dholakia, Chenyan Xiong, and Graham Neubig. 2023. An In-depth Look at Gemini's Language Abilities. ArXiv:2312.11444 [cs].

Tiberiu Boros, Radu Chivereanu, Stefan Dumitrescu, and Octavian Purcaru. 2024. Fine-tuning and retrieval augmented generation for question answering using affordable large language models. In *Proceedings of the third ukrainian natural language processing workshop*, Torino, Italy. European Language Resources Association.

Bernard Comrie, Martin Haspelmath, and Balthasar Bickel. 2008. The Leipzig Glossing Rules: Conventions for interlinear morpheme-by-morpheme glosses. *Department of Linguistics of the Max Planck Institute for Evolutionary Anthropology & the Department of Linguistics of the University of Leipzig. Retrieved January*, 28:2010.

Gregory Eady, Tom Paskhalis, Jan Zilinsky, Richard Bonneau, Jonathan Nagler, and Joshua A. Tucker. 2023. Exposure to the Russian Internet Research Agency foreign influence campaign on Twitter in the 2016 US election and its relationship to attitudes and voting behavior. *Nature Communications*, 14(1):62.

Avia Efrat, Or Honovich, and Omer Levy. 2022. LMentry: A language model benchmark of elementary language tasks. Tex.copyright: Creative Commons Attribution 4.0 International.

Rating Group. 2022. The sixth national poll: The language issue in Ukraine (March 19th, 2022) — ratinggroup.ua.

Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Supryadi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, and Deyi Xiong. 2023. Evaluating Large Language Models: A Comprehensive Survey. ArXiv:2310.19736 [cs].

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. Tex.copyright: arXiv.org perpetual, non-exclusive license.

Bogdan Ivanyuk-Skulskiy, Anton Zaliznyi, Oleksand Reshetar, Oleksiy Protsyk, Bohdan Romanchuk, and Vladyslav Shpihanovych. 2021. ua$_d$atasets: a collection of Ukrainian language datasets.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. *CoRR*, abs/2004.09095. ArXiv: 2004.09095 tex.bibsource: dblp computer science bibliography, https://dblp.org tex.biburl: https://dblp.org/rec/journals/corr/abs-2004-09095.bib tex.timestamp: Wed, 22 Apr 2020 12:57:53 +0200.

Volodymyr Kulyk. 2018. Shedding Russianness, recasting Ukrainianness: The post-Euromaidan dynamics of ethnonational identifications in Ukraine. *Post-Soviet Affairs*, 34(2-3):119–138. Publisher: Taylor & Francis.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. ChatGPT Beyond English: Towards a Comprehensive Evaluation of Large Language Models in Multilingual Learning. ArXiv:2304.05613 [cs].

Włodzimierz Lewoniewski, Krzysztof Węcel, and Witold Abramowicz. 2017. Analysis of references across wikipedia languages. pages 561–573.

Meta. 2022. List of Wikipedias/Table2 — Meta, discussion about wikimedia projects.

Daniel Racek, Brittany I. Davidson, Paul W. Thurner, Xiao Xiang Zhu, and Göran Kauermann. 2024. The Russian war in Ukraine increased Ukrainian language use on social media. *Communications Psychology*, 2(1):1.

Kristina Radivojevic, Nicholas Clark, and Paul Brenner. 2024. LLMs Among Us: Generative AI Participating in Digital Discourse. ArXiv:2402.07940 [cs].

Marigo Raftopoulos. 2015. How enterprises play: Towards a taxonomy for enterprise gamification.

Manley Roberts, Himanshu Thakur, Christine Herlihy, Colin White, and Samuel Dooley. 2023. Data Contamination Through the Lens of Time. ArXiv:2310.10628 [cs].

Joshua Robinson, Christopher Michael Rytting, and David Wingate. 2023. Leveraging Large Language Models for Multiple Choice Question Answering. ArXiv:2210.12353 [cs].

Alessandro Seganti, Klaudia Firląg, Helena Skowronska, Michał Satława, and Piotr Andruszkiewicz. 2021. Multilingual entity and relation extraction dataset and model. In

Proceedings of the 16th conference of the european chapter of the association for computational linguistics: Main volume, pages 1946–1955, Online. Association for Computational Linguistics.

Denis Stukal, Sergey Sanovich, Richard Bonneau, and Joshua A. Tucker. 2017. Detecting Bots on Russian Political Twitter. *Big Data*, 5(4):310–324.

Vasyl Starkoand Olena Synchak. 2023. Feminine personal nouns in ukrainian: Dynamics in a corpus.

Oleksiy Syvokon and Olena Nahorna. 2022. UA-GEC: Grammatical Error Correction and Fluency Corpus for the Ukrainian Language. ArXiv:2103.16997 [cs].

Gilles Sérasset. 2015. DBnary: Wiktionary as a Lemon-based multilingual lexical resource in RDF. *Semantic Web*, 6(4):355–361.

Wikipedia contributors. 2023. Languages used on the internet — Wikipedia, the free encyclopedia.