Better something than nothing: Analysis of GPT-4 performance in identifying Croatian proverbs

Nikola Bakarić[1]

[1]University of Applied Sciences Velika Gorica, Zagrebačka 5, Velika Gorica, Croatia
E-mail: nbakaric@vvg.hr

Abstract

This paper investigates the performance of OpenAI's GPT-4o model in detecting Croatian proverbs within a real-world corpus. Proverbs, like other idiomatic expressions, present challenges in automatic identification due to their form and cultural specificity. Using a manually curated gold standard of 126 proverb-containing sentences extracted from the Croatian parliamentary corpus (ParlaMeter-hr 1.0), the study evaluates GPT-4o through both the ChatGPT interface and the API using various prompt types. Results show poor accuracy (~54%) for all chat-based approaches, suggesting limitations in prompt interpretation or model access. In contrast, API-based prompts achieved significantly higher accuracy: 79.4% for zero-shot and 93.7% for many-shot prompting. The many-shot method, while more accurate, incurred substantially higher time and token costs. Additional classification tasks confirmed GPT-4o's capability in proverb detection with an F1 score of 0.8681. These findings underscore GPT-4o's potential as a tool for lexicographers and language researchers working with idiomatic expressions, while highlighting the trade-offs between cost, performance, and implementation complexity. The study contributes to the fields of computational linguistics and paremiology by exploring practical applications of large language models in processing fixed expressions in low-resource languages.

Keywords: large language model; GPT-4; proverbs; automatic detection

1. Introduction

The task of automatic detection of idiomatic expressions, including proverbs, is an established problem in natural language processing. The task can be approached in several ways, but this paper aims to analyse the performance of the GPT large language model (GPT-4o) in the task of detecting proverbs and proverb-like expressions. The main motivation is to provide insight for lexicographers, paremiologists and other language researchers into the application of a large language model in detecting proverbs in available textual data, with focus on Croatian language. The paper investigates the use of the GPT chat and API interface while considering the differences in access, cost and duration of processing. It provides an overview of the capabilities and limitations of the GPT-4o model by testing several different prompts and prompting methods. GPT-4 is a transformer-based model made by OpenAI, accessible by a chat and API interface, using free or paying accounts (OpenAI, 2024). GPT-4o is a variation

of the GPT-4 model which accepts input across different media and boasts significant improvement in processing non-English languages (OpenAI, 2024).

There are many different definitions of proverbs from many different perspectives. Croatian folklorists view them as rhetorical genres used to express general truths or traditional wisdom, through metaphors and simple language (Kekez, 1998). Mieder (2004) reviews the many attempts at defining proverbs, but describes them as fixed in form, which is an important aspect of this research, especially in practical terms of proverb detection. As such, proverbs share many characteristics of idiomatic expressions. This paper looks at proverbs mostly from a lexicographer's standpoint, where it is more important to find and record the proverb's manifestation than to discuss its meaning or quality.

The paper is structured as follows: the second chapter discusses related work and previous research, third chapter describes the dataset used in this investigation, and the fourth the experiment setup. The fifth chapter presents the results while the final, sixth chapter contains the conclusion and discusses the directions of future work.

2. Related work

The analysis of multiword expressions in general was recognized as a key issue in NLP long before large language models were available (Sag et al., 2002). Later attempts described proverbs by modelling their syntactic structure and using finite-state automata to detect their appearance in running texts (Rassi et al., 2014). Rule-based systems were further proposed for languages other than English, such as Hindi (Priyanka & Sinha, 2014). More recent research employs contextual embeddings and neural networks to identify idioms (Škvorc et al., 2022) which is a task closely related to proverb detection. Recent research efforts into Croatian (and Norwegian) proverb detection in corpora finds proverb detection a complicated task that requires approaches from semantic and syntactic standpoints (Kljajevic & Šarić, 2025). In a broader sense, understanding of idiomatic expressions is an important contributor to sentiment analysis. Williams et al. (2015) used a collection of 580 English language idioms in an attempt to enhance sentiment classification and concluded that idiom-based features significantly impact the results. Tahayna et al. (2022) propose enhancing a sentiment classification algorithm with a knowledge-based expansion in the form of an annotated idiomatic expression lexicon. Other authors recognise the importance of idiomatic expressions detection and understanding in natural language processing. Himdi (2024) provides insight into the application of deep learning and transformer models combined with word embeddings for detection of idiomatic expression in Arabic. They found that transformer models, namely DistilBERT, a derivative of BERT, works best for a morphologically rich language such as Arabic. The interpretation of abstract language by LLMs can be used as a benchmark of their linguistic competence but with careful consideration when designing prompts (Goren & Strapparava, 2024). We can conclude that the detection of proverbs and similar expressions with more or less fixed

forms is an important task not only for natural language processing or lexicography, but for folkloristics and philology as well.

3. Dataset

First step included finding an environment where proverbs are regularly used in a contemporary setting. As proverbs are often used in political discourse to underscore messages or augment arguments and points of view (G¡ndara, 2004), the data presented here will be based on the minutes of the Croatian parliament sessions made available by the Croatian parliamentary corpus ParlaMeter-hr 1.0. The corpus consists of the minutes of the Croatian parliament sessions from 2016 to 2018 (Dobranić et al., 2019). In numbers, it consists of 650,390 sentences or over 14 million tokens divided into 89,628 utterances. The corpus is equipped with speaker metadata, the transcripts are lemmatised, tagged using morphosyntactic descriptions, and the named entities are identified and marked. However, the corpus does not contain any information regarding idiomatic expressions such as the use of proverbs.

A list of 107 Croatian proverbs used in contemporary speech and texts was obtained from Varga & Matovac (2016). The list was further expanded by addition of proverbs gathered at the Chair of Croatian Oral Literature at Faculty of Humanities and Social Sciences, University of Zagreb. This resulted in 151 distinct proverbs used in this research.

Next step was based on the fact that proverbs are mostly found in text as idiomatic expressions, with little variation in phrasing and word form, as discussed by Gibbs & Beitel (1995) and later Wu et al. (2023). This was the idea behind the creation of a simple fuzzy search algorithm, which was then applied to the ParlaMeter-hr corpus to extract sentences containing proverbs. The algorithm passes through the list of proverbs one by one and checks for the occurrence of any proverb word token in the ParlaMeter-hr sentence. It then calculates a simple ratio of occurrences against the number of word tokens in a proverb. After several attempts, it was decided to set the threshold to 0.51 which yielded very few false positive results. The extracted list was further manually checked and verified. This simple search technique suggested 135 occurrences of sentences which contained proverbs. After manually confirming true positive results and removing false positives, the collection was reduced to 126 proverb-containing sentences. This small collection was used as the golden standard in the next steps of the experiment.

4. Description of the experiment

The experiment was conducted in two steps. The first step was done using only the ChatGPT chat interface and the second step involved using the API interface through a Python script. Both steps included a list of 126 sentences with confirmed occurrences of Croatian proverbs and a list of 151 proverbs.

4.1 Chat interface

The first step of the experiment used only the chat interface of the ChatGPT service. The interface was used on a paying account (personal account, *Plus* plan, 20 USD/month) and the GPT-4o model was selected.

The first prompt (**Chat prompt 1**) included uploading a list of 126 sentences with confirmed proverb presence as a text file with the following prompt:

The uploaded text file contains a list of sentences in Croatian. For each line, try to determine if it contains a Croatian proverb in any form. Write the results as a table and add a YES/NO/MAYBE column (YES if there is a proverb, NO if there is no proverb detected and MAYBE if you are unsure), and a column with the detected proverb. Do not translate Croatian text.

No examples of proverbs were provided, and it was explicitly forbidden to translate the Croatian text to English, which is a common LLM tactic when dealing with low-resource languages (Nicholas & Bhatia, 2023). This simple prompt resulted in 61 YES and 7 MAYBE out of 126 sentences with proverbs.

The second prompt (**Chat prompt 2**) was identical to the first one, using the same 126 sentences, with the addition of a request to explain the decision rationale by adding the following sentence: *Before deciding if the line contains a proverb, explain your rationale in no more than 10 words and write it to another column in the results*. It was an attempt to add a step-back dimension to the prompt. However, the results were identical to the results of the first prompt (61 YES and 7 MAYBE).

The third prompt (**Chat prompt 3**) attempt using the chat interface was expanded by uploading a list of 151 known proverbs described in the previous chapter alongside the 126 sentences. The following was added to the prompt: *The uploaded file Poslovice-list.txt contains a list of Croatian proverbs, one proverb per line. Using the provided list of proverbs, try to determine for each line in the text file if it contains a Croatian proverb in any form.* Again, the results were the same as before, with only one less MAYBE (61 YES, 6 MAYBE) which could mean that the model completely ignored the provided list of proverbs.

It should be noted that the results of all three prompts were in perfect agreement (except the one MAYBE in the third prompt). Further analysis of the results shows that the positive (YES) matches were found only in sentences where there was no variation in proverb form. This leads to the conclusion that the model responded to the prompts by doing a simple search using an entire proverb as a search token. Additionally, the model ignored the uploaded list of proverbs and used its own "knowledge" of Croatian proverbs. When prompted to provide a list of "all known Croatian proverbs", the model generated a list of 57 proverbs. Among actual well-known examples of Croatian

proverbs, there were some hallucinations and attempts at translation. Hallucination examples are available in Table 1.

GPT-4o hallucination	**English translation**
Dobra kobasica ide na kraj sela.	A good sausage goes to the end of the village.
Govori tiho i nosi veliku batinu.	Speak quietly and carry a big stick.
Muškarac vrijedi onoliko koliko riječ drži.	A man (male!) is as good as his word.

Table 1: Examples of hallucinations

4.2 API and Python script

The second step of the experiment relied on API interaction with the GPT-4o model. This method ensures better control of prompt parameters such as temperature. Temperature is a sampling factor, between 0 and 1. Lower values, such as 0.1, instruct the model to create a more deterministic output (Boonstra, 2025). Additionally, using the API within a custom Python script allows for precise input and output control, batch processing and, as the results will show, direct access to the selected LLM model and its reasoning capabilities. It is important to note that the API access is not free, and OpenAI charges processing services separately per number of tokens, according to their pricing list. All three prompts described here generated around 1.3 million tokens, the majority of which were generated in the multi-shot prompt.

The Python script was designed to load the list of sentences and prompt the model to try and detect a Croatian proverb for each sentence. The output should be a simple yes or no accompanied by a limited explanation of rationale. The selected model was GPT-4o, and the temperature was set to 0.1 (very low sampling) which limited the randomness of the explanation. Again, two prompt variations were tested, a zero-shot prompt and a many-shot prompt which was supported by a list of know proverbs.

4.2.1 Zero-shot API prompt

This prompt relied solely on the model's reasoning capabilities without including any examples. The script went through the list of sentences and prompted the model for each sentence. The output was recorded into a text file; an example can be seen in Table 2. This technique detected 100 sentences with proverbs, out of 126 possible. The script run time was comparatively quick, with each sentence taking from 1 to 3 seconds, depending on length.

Sentence	Reply	Reasoning
Neću nastaviti tko laže taj i krade, ali ono što čujemo u zadnjih nekoliko dana užasne optužbe od HDZ-a prema Mostu da su lopovi nesposobni od Mosta prema HDZ-u da su lopovi sposobni. (I won't continue with "he who lies also steals," but what we've been hearing in the last few days are terrible accusations: from HDZ toward Most, that they're incompetent thieves; from Most toward HDZ, that they're competent thieves.)	Da. (Yes.)	*Rečenica parafrazira poznatu hrvatsku poslovicu "Tko laže, taj i krade" koja se koristi za opisivanje nepoštenog ponašanja.* (The sentence paraphrases the well-known Croatian proverb "He who lies also steals," which is used to describe dishonest behavior.)
U ovome se slučaju nažalost zrcale one narodne izreke da ćup ili vrč ide na vodu dok se ne razbije. (In this case, unfortunately, the old saying is reflected: the pot or jug goes to the well until it breaks.)	Da. (Yes.)	*Rečenica parafrazira poznatu hrvatsku poslovicu "Ćup ide na vodu dok se ne razbije", koja znači da se nešto može raditi samo do određene granice prije nego što dođe do problema ili neuspjeha.* (The sentence paraphrases the well-known Croatian proverb "The pot goes to the well until it breaks," which means that something can be done only up to a certain point before it leads to problems or failure.)
Ali kako bi se zadovoljila forma papir trpi sve i zbog toga je ta strategija napisana, zbog toga smo mi danas ovdje i moramo o toj strategiji raspravljati. (But in order to satisfy formal requirements, paper tolerates everything, and that's why that strategy was written, that's why we are here today and have to discuss that strategy.*)*	Da. (Yes.)	*Rečenica parafrazira poznatu hrvatsku poslovicu "Papir trpi sve", koja se koristi kako bi se izrazilo da se na papiru može napisati bilo što, bez obzira na istinitost ili praktičnost.* (The sentence paraphrases the well-known Croatian proverb "Paper tolerates everything," which is used to express that anything can be written on paper, regardless of its truthfulness or practicality.)

Table 2: Examples of API zero-shot prompt output

4.2.2 Many-shot API prompt

The previous zero-shot prompt was extended using the previously described list of 151 proverbs. The list of proverbs was added to the script and the model was instructed to move through the list and check each sentence for proverbs. If a proverb was detected, the script would move to the next sentence and start from the beginning of the proverb list. Since the data was manually checked prior to the experiment, there was no need to worry about the possibility of more than one proverb per sentence. The input consisted of the list of sentences and the list of proverbs. The output was recorded into a text file. Apart from a yes/no verdict and a brief explanation, the model provided the matched proverb, as seen in Table 3. However, manual examination of the results shows that while the model correctly detected the presence of a proverb, in several instances the exact proverb was not correctly matched. This can be seen in the second example in Table 3 where the model ignores the obvious proverb "the full don't believe the hungry" and interprets the whole sentence as being semantically close to the proverb "You can't dig a well with a needle."

Sentence	**Reply**	**Matched proverb**	**Reasoning**
I to je istina kažem o kojoj ćemo danas razgovarati i uvijek ima dva kraja. (And that is the truth, I say, which we will talk about today — and it always has two sides.)	*Da. (Yes.)*	*Batina ima dva kraja.* (Every stick has two ends.)	*Rečenica parafrazira poslovicu "Batina ima dva kraja" jer prenosi ideju da svaka situacija ili istina ima dvije strane ili perspektive.* (The sentence paraphrases the proverb "Every stick has two ends," as it conveys the idea that every situation or truth has two sides or perspectives.)
Ovaj zakon i ovo povećanje od 131 kunu minimalne plaće je mali korak, ali premali i zato kažem sit gladnom ne vjeruje. (This law and this 131 kuna increase in the minimum wage is a small step, but too small, and that's why I say: the full don't believe the hungry.*)*	*Da. (Yes.)*	*Iglom se bunar ne kopa.* (You can't dig a well with a needle.)	*Rečenica prenosi sličnu poruku kao poslovica, naglašavajući nedostatnost malih napora u rješavanju većih problema.* (The sentence conveys a similar message as the proverb, emphasizing the inadequacy of small efforts in solving bigger problems.*)*

Table 3: Examples of API many-shot prompt output

This prompt, supported by the extensive proverb list, managed to detect proverbs in 118 of the 126 possible sentences. This prompt was the most time consuming, as each sentence was matched with the list of proverbs. In total, the script ran for over 120 minutes. This was also the most expensive prompt due to the large number of tokens processed with each iteration. The duration and cost of the prompt could be reduced by script optimisation.

4.2.3 Zero-shot classification prompt

A small, manually labelled set of 135 sentences was created. Half of the set consisted of sentences with proverbs, and the other half contained sentences without proverbs. The GPT-4o model was then prompted to classify each sentence as containing/not containing a proverb without any examples (zero-shot). This typical task was added to the experiment to further evaluate the model as a classifier using accuracy and F1 measure. Accuracy was at 87.5% and the F1 score at 0.8681.

5. Results

Table 4 gives an overview of the accuracy, duration and cost of each tested prompt method. The accuracy for the chat interface prompts is very poor, barely over 50%. This leads to the conclusion that either the chat interface is not suitable for complex tasks such as proverb detection, or the prompt phrasing was inadequate. Prompting through the API interface returns much better results. However, a distinction should be made between the two API prompts described here, namely from the duration and cost perspective. The API many-shot prompt is by far the most resource intensive prompt, being approximately 7 times more expensive and 24 times more time consuming than the API zero-shot prompt. The 14% gap in accuracy between the two is substantial, but improvements could be made by better crafting the prompt and upgrading it with examples (few-shot). The cost and duration of the many-shot prompt could also be lowered by batch processing and optimisation of the script and/or preprocessing of the input data.

In conclusion, the many-shot API prompt produced the best results at 93.65% accuracy, while being the most expensive and time-consuming approach. In contrast, the zero-prompt API prompt offers a more optimised and accessible approach which could be further improved. It is further validated with the additional zero-shot prompt classification experiment which managed to return very good results at 87.5% accuracy and 0.8681 F1 score in a small balanced test set.

Prompt type	Chat prompt 1	Chat prompt 2	Chat prompt 3	API zero-shot	API many-shot	API zero-shot class.
Accuracy	53.97%	53.97%	53.17%	79.37%	93.65%	87.5%
Approx. duration	< 1minute	< 1minute	< 1minute	< 5 minutes	> 120 minutes	< 5 minutes
Approx. cost	Free	Free	Free	~ 1.00 USD	~ 7.00 USD	~ 1.00 USD

Table 4: Proverb detection accuracy, duration, and cost across different prompt types

6. Conclusion and future work

While a simple fuzzy search algorithm can be utilized to find occurrences of proverbs in a text, it is limited by the extent of its knowledge base. It will not be able to detect proverbs with too much variation in delivery or those not included in its knowledge base. However, it can still serve as a useful starting point in automatic proverb detection from which more capable models can build on.

This paper demonstrates that GPT-4o can be a valuable tool for detecting Croatian proverbs, but its effectiveness depends heavily on the method of interaction and prompt design. The ChatGPT interface, despite being accessible and free, yielded suboptimal results with limited ability to recognize proverb variations or utilize supplementary data. It is probable that the chat interface does not parse through the uploaded proverb list one at a time, as the API prompt does, which further reduces prompt clarity. In contrast, API-based prompting—especially the many-shot approach supported by an external proverb list—achieved significantly higher accuracy, confirming that GPT-4o performs better when given structured guidance and context. However, this comes at the cost of increased processing time and financial expense. The results emphasize the importance of careful prompt engineering and highlight the potential of large language models in supporting lexicographic and paremiological research, particularly in low-resource languages like Croatian. Additionally, the successful classification experiment suggests GPT-4o's robustness in handling idiomatic expressions when framed as a classification task. It should be noted that the model does suffer from occasional hallucinations. It produced a few new proverbs and misinterpreted existing ones. Some of the hallucinations, such as the erroneous mapping of the proverb *The full don't believe the hungry* (seen in Table 3) indicates that the model prefers semantic similarity over lexical form as a matching tactic. Overall, GPT-4o shows promise in automating the identification of fixed expressions in natural language, offering practical benefits across NLP, linguistics, and digital humanities. As a lexicographer's tool, LLMs can provide insight into the use and structure of existing proverbs (or other idiomatic expressions) and the detection of new ones at a large scale and across languages.

Future work should explore several avenues: a) optimizing cost-performance balance through script and prompt refinement, b) exploring multilingual proverb detection, and c) investigating and comparing the capabilities of other LLMs in the task of automated proverb detection.

7. References

Boonstra, L. (2025, February). *Prompt engineering*. Google. https://www.kaggle.com/whitepaper-prompt-engineering

Dobranić, F., Ljubešić, N., & Erjavec, T. (2019). *Croatian parliamentary corpus ParlaMeter-hr 1.0* (Corpus, Text No. http://hdl.handle.net/11356/1209; Version 1.0). Slovenian language resource repository CLARIN.SI. https://www.clarin.si/repository/xmlui/handle/11356/1209

Gibbs, R. W., & Beitel, D. (1995). *What proverb understanding reveals about how people think*. Psychological Bulletin, 118(1), 133–154. https://doi.org/10.1037/0033-2909.118.1.133

Goren, G., & Strapparava, C. (2024). Context Matters: Enhancing Metaphor Recognition in Proverbs. *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 3825–3830. Torino, Italia. ELRA and ICCL.

G¡ndara, L. (2004). *'They That Sow the Wind...': Proverbs and Sayings in Argumentation*. Discourse & Society, 15(2–3), 345–359. https://doi.org/10.1177/0957926504041023

Himdi, H. (2024). Arabic Idioms Detection by Utilizing Deep Learning and Transformer-based Models. *Procedia Computer Science*, 244, 37–48. https://doi.org/10.1016/j.procs.2024.10.176

Kekez, J. (1998). *Uvod u književnost: Teorija, metodologija* (Peto izdanje). Nakladni zavod Globus.

Kljajevic, V., & Šarić, L. (2025). *Corpus-Based Investigation of Proverbs: Challenges and New Directions*. Corpus Pragmatics. https://doi.org/10.1007/s41701-024-00181-2

Mieder, W. (2004). *Proverbs: A handbook*. Greenwood Press.

Nicholas, G., & Bhatia, A. (2023). Lost in Translation: Large Language Models in Non-English Content Analysis (No. arXiv:2306.07377). arXiv. https://doi.org/10.48550/arXiv.2306.07377

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., … Zoph, B. (2024). *GPT-4 Technical Report* (No. arXiv:2303.08774). arXiv. https://doi.org/10.48550/arXiv.2303.08774

OpenAI, Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A. J., Welihinda, A., Hayes, A., Radford, A., Mądry, A., Baker-Whitcomb, A., Beutel, A., Borzunov, A., Carney, A., Chow, A., Kirillov, A.,

Nichol, A., … Malkov, Y. (2024). *GPT-4o System Card* (No. arXiv:2410.21276). arXiv. https://doi.org/10.48550/arXiv.2410.21276

Priyanka, & Sinha, R. M. K. (2014). A system for identification of idioms in Hindi. *2014 Seventh International Conference on Contemporary Computing (IC3)*, 467–472. https://doi.org/10.1109/ic3.2014.6897218

Rassi, A. P., Baptista, J., & Vale, O. (2014*). Automatic Detection of Proverbs and their Variants* [Application/pdf]. OASIcs, Volume 38, SLATE 2014, 38, 235–249. https://doi.org/10.4230/OASICS.SLATE.2014.235

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., & Flickinger, D. (2002). Multiword Expressions: A Pain in the Neck for NLP. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing* (Vol. 2276, pp. 1–15). Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-45715-1_1

Škvorc, T., Gantar, P., & Robnik-Šikonja, M. (2022). MICE: Mining Idioms with Contextual Embeddings. *Knowledge-Based Systems*, 235, 107606. https://doi.org/10.1016/j.knosys.2021.107606

Tahayna, B. M. A., Ayyasamy, R. K., & Akbar, R. (2022). *Automatic Sentiment Annotation of Idiomatic Expressions for Sentiment Analysis Task*. IEEE Access, 10, 122234–122242. https://doi.org/10.1109/access.2022.3222233

Varga, M. A., & Matovac, D. (2016). *KROATISCHE SPRICHWÖRTER IM TEST*. Proverbium : Yearbook of International Proverb Scholarship, 33(1). https://hrcak.srce.hr/278302

Williams, L., Bannister, C., Arribas-Ayllon, M., Preece, A., & Spasić, I. (2015). The role of idioms in sentiment analysis. *Expert Systems with Applications*, 42(21), 7375–7385. https://doi.org/10.1016/j.eswa.2015.05.039

Wu, J., Zhou, W., & Shao, B. (2023). On English proverb variation from the perspective of linguistic creativity. *Frontiers in Psychology*, 14. https://doi.org/10.3389/fpsyg.2023.1213649

This work is licensed under the Creative Commons Attribution ShareAlike 4.0 International License.

http://creativecommons.org/licenses/by-sa/4.0/