LREC-COLING 2024

Games and Natural Language Processing 2024

Workshop Proceedings

Editors
Chris Madge

21st May, 2024
Torino, Italia

Proceedings of the 10th Workshop on Games and Natural Language Processing @LREC-COLING-2024

Copyright ELRA Language Resources Association (ELRA), 2024
These proceedings are licensed under a Creative Commons
Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0)

ISBN 978-2-493814-17-3
ISSN 2951-2093 (COLING); 2522-2686 (LREC)

Jointly organized by the ELRA Language Resources Association
and the International Committee on Computational Linguistics

Message from the General Chair

This volume documents the Proceedings of the Games and Natural Language Processing Workshop, held on (21st May 2024) as part of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)

This workshop examines the use of games and gamification for Natural Language Processing (NLP) tasks, as well as how NLP research can advance player engagement and communication within games. The Games and NLP workshop aims to promote and explore the possibilities for research and practical applications of games and gamification that have a core NLP aspect, either to generate resources and perform language tasks or as a game mechanic itself. This workshop investigates computational and theoretical aspects of natural language research that would be beneficial for designing and building novel game experiences, or for processing texts to conduct formal game studies. NLP would benefit from games in obtaining language resources (e.g., construction of a thesaurus or a parser through a crowdsourcing game), or in learning the linguistic characteristics of game users as compared to those of other domains.

Workshop website: https://gamesandnlp.com

Organizing Committee

- Chris Madge, chair (Queen Mary University of London)
- Jon Chamberlain (University of Essex, UK)
- Karën Fort (Sorbonne Université, France)
- Udo Kruschwitz (University of Regensburg, Germany)
- Stephanie Lukin (U.S. Army Research Laboratory)

Program Committee

- Alice Millour (Sorbonne Université)
- Brent Harrison (University of Kentucky, US)
- Ian Horswill (Northwestern University)
- Jonathan Lessard (Universite Condoria)
- Luisa Coheur (INESC-ID & Instituto Superior Técnico, University of Lisbon)
- Mariët Theune (University of Twente)
- Massimo Poesio (Queen Mary University, UK)
- Mathieu Lafourcade (LIRMM, France)
- Morteza Behrooz (University of California, Santa Cruz, US)
- Pedro Santos (INESC-ID & Instituto Superior Técnico, University of Lisbon)
- Richard Bartle (University of Essex, UK)
- Seth Cooper (Northeastern University, US)
- Valerio Basile (University of Turin, Italy)
- Fatima Althani (Queen Mary University, UK)

Table of Contents

"Actors Challenge": Collecting Data to Study Prosodic Patterns and Their Mappings to Meanings Across Languages
Sia V. Sepanta 1

Empowering Adaptive Digital Game-Based Language Learning for Under-Resourced Languages Through Text Analysis
Elaine Uí Dhonnchadha, Sally Bruen, Liang Xu and Monica Ward 6

Hostomytho: A GWAP for Synthetic Clinical Texts Evaluation and Annotation
Nicolas Hiebel, Bertrand Remy, Bruno Guillaume, Olivier Ferret, Aurélie Névéol and Karen Fort 14

Using In-context Learning to Automate AI Image Generation for a Gamified Text Labelling Task
Fatima Althani, Chris Madge and Massimo Poesio 21

Aspect-based Sentiment Evaluation of Chess Moves (ASSESS): an NLP-based Method for Evaluating Chess Strategies from Textbooks
Haifa Alrdahi and Riza Batista-Navarro 32

Generating Converging Narratives for Games with Large Language Models
Douglas Summers-Stay and Clare R. Voss 43

Leveraging Large Language Models for Spell-Generation in Dungeons & Dragons
Elio Musacchio, Lucia Siciliani, Pierpaolo Basile and Giovanni Semeraro 61

Branching Narratives: Character Decision Points Detection
Alexey Tikhonov 70

Utilizing GPT-4 to Solve TextWorld Commonsense Games Efficiently
Binggang Zhuo and Masaki Murata 76

Linguistic Acceptability and Usability Enhancement: A Case Study of GWAP Evaluation and Redesign
Wateen Abdullah Aliady and Massimo Poesio 85

Riddle Me This: Evaluating Large Language Models in Solving Word-Based Games
Raffaele Manna, Maria Pia di Buono and Johanna Monti 97

LLMs of Catan: Exploring Pragmatic Capabilities of Generative Chatbots Through Prediction and Classification of Dialogue Acts in Boardgames' Multi-party Dialogues
Andrea Martinenghi, Gregor Donabauer, Simona Amenta, Sathya Bursic, Mathyas Giudici, Udo Kruschwitz, Franca Garzotto and Dimitri Ognibene 107

Workshop Program

21st May 2024

14:00–14:05	***Welcome***
14:05–14:25	*"Actors Challenge": Collecting Data to Study Prosodic Patterns and Their Mappings to Meanings Across Languages* Sia V. Sepanta
14:25–14:45	*Empowering Adaptive Digital Game-Based Language Learning for Under-Resourced Languages Through Text Analysis* Elaine Uí Dhonnchadha, Sally Bruen, Liang Xu and Monica Ward
14:45–15:05	*Hostomytho: A GWAP for Synthetic Clinical Texts Evaluation and Annotation* Nicolas Hiebel, Bertrand Remy, Bruno Guillaume, Olivier Ferret, Aurélie Névéol and Karen Fort
15:05–15:25	*Using In-context Learning to Automate AI Image Generation for a Gamified Text Labelling Task* Fatima Althani, Chris Madge and Massimo Poesio
1525–1600	*Keynote* Mathieu Lafourcade
1600–1630	***Poster Session***
16:30–16:50	*Aspect-based Sentiment Evaluation of Chess Moves (ASSESS): an NLP-based Method for Evaluating Chess Strategies from Textbooks* Haifa Alrdahi and Riza Batista-Navarro
16:50–17:10	*Generating Converging Narratives for Games with Large Language Models* Douglas Summers-Stay and Clare R. Voss
17:10–17:30	*Leveraging Large Language Models for Spell-Generation in Dungeons & Dragons* Elio Musacchio, Lucia Siciliani, Pierpaolo Basile and Giovanni Semeraro
17:30–17:50	*Branching Narratives: Character Decision Points Detection* Alexey Tikhonov
17:50–18:00	***Closing***

21st May 2024 (Poster Session)

Utilizing GPT-4 to Solve TextWorld Commonsense Games Efficiently
Binggang Zhuo and Masaki Murata

Linguistic Acceptability and Usability Enhancement: A Case Study of GWAP Evaluation and Redesign
Wateen Abdullah Aliady and Massimo Poesio

Riddle Me This: Evaluating Large Language Models in Solving Word-Based Games
Raffaele Manna, Maria Pia di Buono and Johanna Monti

LLMs of Catan: Exploring Pragmatic Capabilities of Generative Chatbots Through Prediction and Classification of Dialogue Acts in Boardgames' Multi-party Dialogues
Andrea Martinenghi, Gregor Donabauer, Simona Amenta, Sathya Bursic, Mathyas Giudici, Udo Kruschwitz, Franca Garzotto and Dimitri Ognibene

Actors Challenge : Collecting Data to Study Prosodic Patterns and Their Mappings to Meanings Across Languages

Siavosh Sepanta, Natallia Chaiko, Roberto Zamparelli
University of Trento (CIMeC)
{siavosh.sepanta, natallia.chaiko, roberto.zamparelli }@unitn.it

Abstract

In this paper we describe "Actor's Challenge": a web-based interactive game designed to collect massively multi-speaker, multi-lingual oral data on the connection between prosody and various aspects of meaning. Game participants take on the two roles of auditioners and casting directors. Auditioners are asked to record certain target phrases modulated according to the emotional or attitudinal profiles that correspond to contexts or stage cues given to them. They then switch roles and become Casting Directors. Now they have to listen to other participants' recordings, guess the corresponding context/stage cue that the auditioner tried to convey, and evaluate how good the performance was. By having the players alternate between these two roles we obtain both data creation and data validation from the same set of participants. We expect that the final dataset of labeled recordings will be valuable for a range of applications: training multilingual Speech Emotion Recognition classifiers; discovering correlations and variations in prosodic patterns among unrelated languages; examining correlations between prosodic patterns and emotion recognizability; probing the possibility that some prosodic patterns are universal.

Keywords: Prosody, GWAP, NLP

1. Introduction

Prosody is a fundamental aspect of spoken language, associated to units larger than the segment and related to word/sentence accent, intonation, lexical tone and rhythm/tempo (Cole, 2014). Modification of prosodic patterns is a tool we use either inadvertently or intentionally (e.g. in acting) to convey our attitudes and emotions. By changing nuances in pitch, amplitude or speed, we can remove syntactic or semantic ambiguity, alter or enhance the meaning of words, or change focus.

The study of Affective Prosody, an umbrella term that includes emotional and attitudinal prosody (Mitchell & Ross, 2013) involves mapping prosodic patterns to the information structure encoded in a text. Theoretical researchers working to establish such correspondences could benefit from a labeled set of cross-linguistic data where linguistic contexts are mapped onto utterances produced according to prosodic patterns appropriate for that context. At a computational level, data of this sort would cater to the growing interest in spoken dialogue with AI agents, which will ultimately be expected to detect emotions and attitudes in human speech and answer with suitably nuanced intonations. Our attempt to build such a dataset takes its moves from the success of other linguistic games-with-a-purpose (GWAPs, see Ahn, 2006) like Phrase Detective (Chamberlain et al, 2008) and tries to leverage the power of competitive gaming to enroll a large number of subjects in a game of acting, inspired by the Stanislavski's method (as described in Jakobson, 1960). The byproduct of the game is the collection of highly controlled prosodic data, which are cross-validated by the very same players who provide their voices for data production.

1.1 Background

There exist various data sets that may be compared to ours, mostly designed for Speech Emotion Recognition (SER, Swain, 2018). Some contain manually annotated video recordings from Youtube labeled with the 6 basic emotions in 4 languages (CMU-MOSEAS, Bagher Zadeh, et al 2020). Others draw data from talk-shows (e.g. for German, Vera Am Tag, Grimm et al., 2008) or use a much more limited number of professional actors, instructed to record pre-set English phrases (RAVDESS, Livingstone & Russo, 2018). Without doubting the importance of naturalistic data, we believe that there is a need for a more controlled set of data where the very same linguistic expression is uttered with very different communicative intents, creating a set of expressions that differ *only* in prosody, not in the choice of words. This is difficult to obtain from naturalistic input but allows a fine-grained control of intonation parameters. The datasets above cannot provide quantitative information on the extent to which the communicative intent is met (i.e. how many listeners could recognize a given utterance as expressing e.g. *anger* and not *disgust* or *fear*), cannot be easily extended to other languages and do not address "attitudinal" uses of prosody (association with focus, syntactic or lexical disambiguation, irony, etc.). Some of these issues have been studied by phoneticians with lab experiments (for Romance languages see Origlia, et al 2014, Bocci 2013, Gili Fivela, et al. 2015) but with a limited number of speakers.

There is at least one multilingual, massively multisubject database of spoken language (commonvoice.mozilla.org), but communicative intents are not labeled and may only be inferred from the text. All things considered, the research community is still missing a comprehensive multilingual dataset containing labeled recordings from a large number of speakers, suitable for

Games and NLP @LREC-COLING-2024, pages 1–5
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

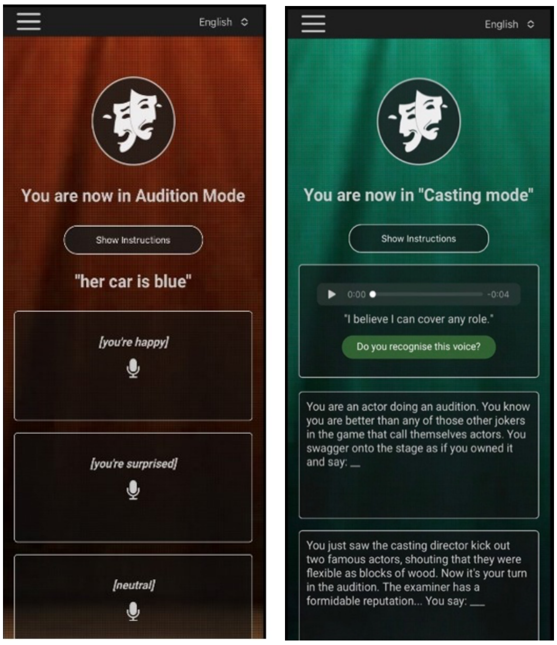
studying emotional as well as attitudinal prosodic patterns in a comparative manner across languages.

2. Our Project

This need prompted us to implement a web-based game, *[Project-Name]* (AC), designed to collect large amounts of recordings expressing prosody-meaning mappings in multiple languages. These recordings are generated and validated by the same players.

2.1 Game Setup

The game works as follows: to address emotional prosody, we prepare a series of linguistic expressions (**targets**) that could be uttered in various contexts and are as neutral as possible in their affective value, such as “It’s a cappuccino”. Similarly, for the attitudinal prosody, target phrases are chosen that can lend themselves to various topics of study, e.g., focus: “Kevin isn't drinking because he is unhappy.” Next, we prepare a set of discriminating contexts in which the target phrase could be found. These contexts give the background to understand how the target should be uttered, evoking certain emotions (e.g., sad, angry, happy) or giving cues that resolve the target’s ambiguity.



"I believe I can cover any role."

You are an actor doing an audition. You know you are better than any of those other jokers in the game that call themselves actors. You swagger onto the stage as if you owned it and say: __

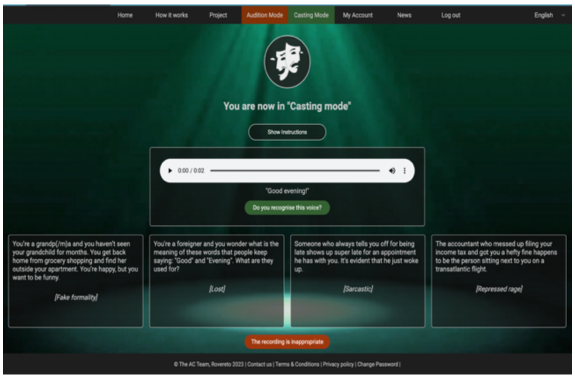
You just saw the casting director kick out two famous actors, shouting that they were flexible as blocks of wood. Now it's your turn in the audition. The examiner has a formidable reputation... You say: __

Figure 1: The two modes of the AC Game, in its mobile version. Left: audition; right: casting

Example contexts for the target phrases might be: “She had asked you twice ‘Did you say coffee or cappuccino?’ and you patiently told her ‘Coffee’. Now she is handing you your cup and you almost yell ‘_____’!”; “Kevin drinks a lot. No matter if he is happy, or sad. He is a classic alcoholic.” We are also experimenting with contexts that are simply stage directions (“[you are angry]”), to test the difference between linguistic and metalinguistic cues.

When they sign up, new participants are asked to provide basic information such as age, gender, region of provenance, native language and language they want to play in. They are also asked to sign an informed consent declaration, which clearly states that their anonymized data will be made publicly available under a Creative Commons Attribution Sharealike 4.0 license. Entering the game, players alternate between two roles: actors who are doing an audition (“Auditioners”) and “casting directors”, who have to evaluate actors’ performances.

- *Auditioners:* In this role participants are asked to read a randomly assigned context or stage cue and *act out* the target phrase in a way that would be best suited for the situation described in that particular context. They can record their voice multiple times, listen to their performance, submit the recording once they are satisfied and turn to the next context for the same target (there are between 2 and 4 contexts per target; see Fig.1, left). After recording the targets in all the contexts proposed, the players move on to another audition session. After a few auditions, however, they are automatically sent to the *Casting* mode.
- *Casting Directors:* in this role the player is prompted to listen to other actors’ utterances and evaluate them. Specifically, the player hears the recording of a target phrase uttered by another actor and sees the set of contexts which were presented to the actor (Fig.1, right). The task is now to assign the performance to the context for which it was intended. After matching the recording, the castind director rates the performance of the actor on a 1-5 Likert scale (see Fig.2).


Home
How it works
Project
Audition Mode
Casting Mode
My Account
News
Log out
English
You are now in "Casting mode"
Show Instructions
0:00 / 0:02
Do you recognise this voice?
[Fake formality]
[Lost]
[Sarcastic]
[Repressed rage]
The recording is inappropriate
© The AC Team, Rovereto 2023 | Contact us | Terms & Conditions | Privacy policy | Change Password |

"Good evening!"

You're a grandp(/m)a and you haven't seen your grandchild for months. You get back home from grocery shopping and find her outside your apartment. You're happy, but you want to be funny.

You're a foreigner and you wonder what is the meaning of these words that people keep saying: "Good" and "Evening". What are they used for?

Someone who always tells you off for being late shows up super late for an appointment he has with you. It's evident that he just woke up.

The accountant who messed up filing your income tax and got you a hefty fine happens to be the person sitting next to you on a transatlantic flight.

Figure 2: Rating the performance in Casting mode.

In the Casting mode players are also requested to filter out audio clips that have a poor sound quality, do not match the target phrase written on the page, contain inappropriate contents or somehow include tips that help the casting director in identifying the corresponding context. Such reports will be relayed to the players, who will be warned or, in more serious cases, removed from the game. An additional button allows casting directors to identify voices already heard. This can provide training data for voice recognition tasks (see AL-Shakarchy, et al., 2022).

2.2 Scoring System

The game is designed to appeal to people who would like to test their skill level at acting or recognizing the communicative intent conveyed by human voice. Each player receives scores based on how well they performed as actors, calculated as the number of times their performance was correctly matched by the casting directors, multiplied by the number of alternatives the target had (from 2 to 4). Players also receive casting points, calculated as a function of the number of times a recording meant for a given context is attributed to that context. The two scores are combined to give an overall score and a place in a public Scoreboard. Players receive score-related titles (from “Grinding stone” to “Acting god”).

3. Current Data Prompts

The game can currently be played in English, Italian, German and French, soon to be followed by Russian, Farsi and Arabic. The current version contains 35 target phrases, and 185 unique contexts. The current set of targets and contexts has been primarily designed to study the linguistic expression of the basic emotions (*fear, joy, surprise, disgust, sadness, anger*, plus *neutral,* see Ekman, 1999). We chose targets of two sizes: long and short (monosyllabic); among the short ones we tested a number of nonlinguistic vocables (“Oh”, “Ah”, “Ha” for English and equivalent for the other languages). Targets were chosen to be similar and sometimes identical in the various languages. Emotions can be prompted by textual contexts (see an example in Sec. 2.1), stage directions or both, to test which cue is more effective.

Apart from emotions, current data include target/contexts to test syntactic ambiguity (Prepositional Phrase attachment), broader attitudes (e.g. embarrassment, concern, pessimism, dignity, fake cordiality, perplexity, correction, sarcasm, grieving, boredom, pleasant and unpleasant surprise), association of negation with focus, normal vs. rhetorical questions, long distance Wh-extractions (“When did you say that John left?”). Other examples are specific for different languages (e.g. the definite vs. kind-denoting reading of the definite article in Italian; the universal vs. existential reading of bare plurals in English). Note that for some of these phenomena the system could simply provide evidence that prosody cannot disambiguate them.

Yet additional materials are fillers, designed only to make the game more fun and engaging. Some of these prompts are next to impossible and might not be part of the data released. Note however that adding new data is quite straightforward. We are presently in contact with phoneticians at the university of Padua, Siena and Bolzano and we welcome new collaborations.

4. Potential Applications

The AC data can be used to investigate a range of topics across theoretical and computational linguistics.

- Speech Emotion Recognition (SER) with a broader range of speakers from different regions and age groups. Artificial Emotional Speech Production can also be tested by injecting artificially produced samples in the Casting phase.

- Examining the effect of combining multiple intonational patterns (e.g. *question+surprise, question+emotion, multiple emotions*). The compositionality of emotions is currently mostly focused on bodily/facial features (Cavicchio, et al., 2018) but the combination of emotions in speech could benefit from a data set such as the one we are creating with AC.

- Examining how the intonation patterns vary from speaker to speaker and from language to language. Inter-speaker variation is actively studied in labs (Niebuhr, et al., 2011; Myberg, 2013; Feldhausen, 2016) but not with the large volume of data that a web game could gather. Interlinguistic variation has been discussed (see Rabanus, 2003, Gili Fivela et al. 2015), but not systematically evaluated for very different languages.

- Discovering ambiguous intonational patterns (i.e. targets consistently assigned to multiple contexts) and ordering semantic/emotional contexts w.r.t. how hard it is to consistently translate them into unambiguous prosody.

- Discovering the individual extent to which passive prosodic competence differs from active one (can one be good casting director without being a good actor, or vice versa?) and whether the recognition ability is affected by regional, gender or age differences.

5. Universality of Prosody

There seems to be a consensus that having emotions is universal among humans (Ekman et al, 1972). But is the same notion true when it comes to expressing our emotions, particularly in speech? In other words, are there any prosodic patterns that correspond to the same emotion in more than one language?

Examining this question could entail the following two approaches:

5.1 Non-Computational

One way to test whether prosodic patterns could be linked to certain emotions independent of language would be having native speakers of Lang1, listen to and label recordings done in Lang2, which is unfamiliar to them. Shakuf et al (2022) following this method demonstrated that native speakers of German and Hebrew could for the most part correctly identify emotions in the language they were not familiar with.

In AC, we have a set of stimuli where participants have to play in an invented language.

- Target phrase “Sotaki”
  Context: The woman came from a distant tribe and when she spoke, we couldn't understand her language. She had no idea what gunpowder was: when we lit up some firecrackers, she went pale in her face and said: "___"

Successful identification of emotions in this task could be interpreted as the notion that prosodic patterns can be linked to emotions independent of the language. At the moment we do not have enough data to arrive at conclusive results.

5.2 Computational

Emotional Speech Classifiers are typically trained on large datasets of recordings in a language and then tested on that same language. There have been studies in which the classifier is trained on one language and tested on another. Wish et al (2021) tested a classifier trained on Urdu on several European languages. Their results show that the model was able to correctly categorize the emotions in languages it was not familiar with.

6. Preliminary Results

To begin investigating the inter-speaker variation in recognition ability (3rd question above) we looked at the rate at which casters recognized the intentions of actors with the same or the opposite gender (see Table 1).

Casting Directors ↓	Actors	
	Male	Female
Male	42,1	57,8
Female	54,8	45,1

Table 1: Correct recognition %, by gender

These preliminary results hint at female emotions being better recognized than male, and at an advantage in recognizing emotions in the opposite gender.

To test the effectiveness of the data collected for SER, we trained a classifier, (Hubert, Hsu, 2021) on the English utterances recorded by AC players and tested the model on the Italian dataset. Our results, again very preliminary, show that even without cleaning the files available so far the English model reaches 43% accuracy on Italian (chance = 25%), showing that prosodic patterns do transfer from English to Italian.

7. Conclusion

The website was officially launched in the late summer of 2023, and we are at the moment promoting it via direct contacts and Italian social media. The site has at the moment 102 registered users, of which 44 are English, 43 Italian, 10 German, and 5 French. We have a total of 778 recordings. We are now working to add two new languages: Russian and Farsi. The feedback we are receiving from players is mostly positive. They find it quite entertaining, with some preferring one role to another. It seems playing the game is not becoming part of their daily routine and after a while they stop playing. We are considering changing the contexts and/or target phrases to make the game more fun to play.

8. Bibliographical References

Ahn, L. v. (2006). Games with a purpose. *Computer*, 92-94.

AL-Shakarchy, N. D., Obayes, H. K., & Abdullah, Z. N. (2022). Person identification based on voice biometric using deep neural network. *International Journal of Information Technology*, 789–795.

Bagher Zadeh, A., Cao, Y., Liang, P., Poria, S., & Morency, L.-P. (2020). CMU-MOSEAS: A Multimodal Language Dataset for Spanish, Portuguese, German and French. *EMNLP*, (pp. 1801-1812).

Bocci, G. (2013). The Syntax Prosody Interface: A cartographic perspective with evidence from Italian. Amsterdam/Philiadephia: John Benjamins Publishing Company.

Cavicchio, F., Dachkovsky, S., Leemor, L., Shamay Tsoori, S., & Sandler, W. (2018). Compositionality in the language of emotion. *PloS one*.

Chamberlain, J., Poesio, M., & U., K. (2008). Phrase Detectives: A web-based collaborative annotation. *International Conference on Semantic Systems.*

Cole, J. (2014). Prosody in context: a review. *Language, Cognition and Neuroscience*, 1-31.

Ekman, P. (1999). *Handbook of cognition and emotion.*

Ekman P., Friesen W. V., Ellsworth P. (1972). *Emotion in the human face: Guidelines for research and an integration of findings*. New York, NY: Pergamon Press.

Feldhausen, I. (2016). Inter-speaker variation, optimality theory, and the prosody of clitic left dislocations in Spanish . *Probus*.

Gili Fivela, B., Avesani, C., Barone, M., Bocci, G., Crocco, C., D'Imperio, M., & Sorianello, P. (2015).
Intonational phonology of the regional varieties of Italian. Dans P. P. Sónia Frota (ed.), *Intonation in Romance* (pp. 140–197). Oxford University Press.

Grimm, M. K. (2008). The Vera Am Mittag German Audio-Visual Emotional Speech Database . *IEEE international conference on multimedia and expo*, (pp. 865-868).

Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., & Mohamed, A. (2021). HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *Computer Science*.

Jakobson, R. (1960). Linguistics and Poetics. Dans T. Sebeok, *Style in Language* (pp. 350-377). Cambridge: Massachusetts Institute of Technology Press.

Mitchell, R. L., & Ross, E. D. (2013). Attitudinal prosody: What we know and directions for future study. *Neuroscience & Biobehavioral Reviews*, 471-479.

Myberg, S. (2013). Sisterhood in Prosodic Branching *Phonology*.

Niebuhr, O. D., Fivela, B. G., & Gangemi, F. (2011). Are there "shapers" and "aligners"? individual differences in signalling pitch accent category. *17th ICPhS*, 120–123.

Origlia, A., Cutugno, F., & Galatà, V. (2014). Continuous emotion recognition with phonetic syllables. *Speech Communication*, 155-169.

Rabanus, S. (2003). A Cross-Linguistic Study of German and Italian. *Zeitschrift für Sprachwissenschaft*.

Shakuf, V., Ben-David, B., Wegner, T. G., Wesseling, P. B., Mentzel, M., Defren, S., ... & Lachmann, T. (2022). Processing emotional prosody in a foreign language: the case of German and Hebrew. Journal of Cultural Cognitive Science, 6(3), 251-268.

Steven R. Livingstone, F. A. (2018). The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PLoS ONE*.

Swain, M. R. (2018). Databases, features and classifiers for speech emotion recognition: a review. . *Int J Speech Technol* , 93–120.

Wisha, Z., Javed, A., Khan, J., & Ghadekallu, T. (2021). Cross corpus multi-lingual speech emotion recognition using ensemble learning. *Complex & Intelligent Systems*, 1-10.

Empowering Adaptive Digital Game-Based Language Learning for Under-Resourced Languages Through Text Analysis

Elaine Uí Dhonnchadha[1], Sally Bruen[1], Liang Xu[2], Monica Ward[2]
[1] Trinity College Dublin, [2] Dublin City University
[1] Centre for Language and Communication Studies, Trinity College Dublin, Dublin, Ireland
[2] School of Computing, Dublin City University, Dublin, Ireland
{uidhonne, sbruen}@tcd.ie, liang.xu6@mail.dcu.ie, monica.ward@dcu.ie

Abstract

This study explores Cipher, an adaptive language learning game tailored for the under-resourced Irish language, aimed mainly at primary school students. By integrating text analysis techniques, Cipher dynamically adjusts its difficulty based on the player's language proficiency, offering a customised learning experience. The game's narrative involves decoding spells to access Irish myths and stories, combining language learning with cultural elements. Development involved collaboration with educators to align the game content with curriculum standards and incorporate culturally relevant materials. This paper outlines the game's development process, emphasising the use of text analysis for difficulty adjustment and the importance of engaging, educational gameplay. Preliminary results indicate that adaptive games like Cipher can enhance language learning by providing immersive, personalised experiences that maintain player motivation and engagement.

Keywords: adaptivity, text analysis, digital game-based language learning

1. Introduction

This paper focuses on the use of Natural Language Processing (NLP) to enhance player engagement in the *Cipher* language learning game. We do this by adapting the level of challenge in the game in response to the player's progress and proficiency. The scenario in the Cipher game is that an evil creature is putting stories and myths under a spell and making them unreadable, and the player's mission is to discover what spells were used and thwart the evil creature. In order to present stories and myths of an appropriate level we need to be able to rank and grade the stories. As this language learning game is being developed for an under-resourced language, in this case Irish, there are no tools for automatically ranking and grading Irish language texts, to our knowledge.

The layout of the paper is as follows. Section 2 gives background to learning Irish in Ireland, and introduces the benefits of educational games for language learning and the factors that influence their effectiveness. Section 3 gives an overview of *Cipher – Faoi Gheasa,* an adaptive educational game. Section 4 highlights our co-creation methodology for game development. Section 5 focuses on the development of text analysis tools for Irish which allow us to rank the game texts and thereby ensure that the level of challenge is adaptive to the individual player's proficiency.

2. Background

2.1 Irish language: background and challenges

The key features of successful language learning are engagement, motivation and early success. These factors are particularly important for first-time language learners who do not know what to expect when they start their language learning journey. This is the case for learners of Irish at primary school level in Ireland. Those who have prior experience of learning a language know that it can be challenging at times and requires a degree of resilience and that it is important to persevere when difficulties are encountered. For most L1 (first language) English primary school students in Ireland, their first exposure to another language is when they start learning Irish in primary school. This critical first experience can have an impact on future language learning. For example, if a student has a negative experience learning Irish, this could impact on their enthusiasm or self-confidence in learning another language. For this reason, it is very important that students have a positive experience of learning Irish. However, this is not always the case. In the majority of schools, teachers are not L1 speakers and lack confidence in their Irish ability. There are limited digital resources available for teaching and learning the language. Also, Irish plays a complex socio-cultural role in Ireland, where there is a tension between cultural value and economic value.

2.2 Game-Based Language Learning

Adaptive educational games blend the engagement of gaming with personalised learning, creating environments that motivate and immerse learners. Research (Peirce et al, 2008) argues for the intrinsic motivation these games offer, emphasising their capacity to adapt without sacrificing engagement. The innovation in non-invasively adapting games shows a balance between providing educational benefit and maintaining a game's immersive appeal.

According to Acquah and Katz's (2020) systematic review of empirical evidence for the effectiveness of digital games on second language learning, the main game features that influence outcomes are ease-of-use, challenge (i.e. being in the zone of proximal development), rewards and feedback, control or autonomy, goal-orientation, and interactivity. In this

Games and NLP @LREC-COLING-2024, pages 6–13
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

paper we focus on the first two features: ease-of-use and zone of proximal development (ZPD) (Vygotsky, 1978) as a means of making the game more engaging and adaptive to the users.

In striving to achieve the ZPD, since learning happens when the game is challenging but not too challenging, it is important that adapting educational content happens in a way that does not disrupt the gaming experience. This involves innovative approaches to personalising learning challenges and providing meta-cognitive support, ensuring that adaptations are non-invasive and augmentative rather than interruptive. Kickmeier-Rust et al. (2008) further highlight the role of micro-adaptivity in assessing competencies within games seamlessly, contributing to an improved learning and gaming experience compared to non-adaptive counterparts.

Adaptive educational games offer a promising avenue for enhancing learning through personalised, engaging experiences. The careful integration of adaptivity, aimed at bolstering motivation without compromising immersion, is pivotal. This approach not only enriches the educational potential of games but also preserves their intrinsic appeal as engaging and immersive entertainment mediums.

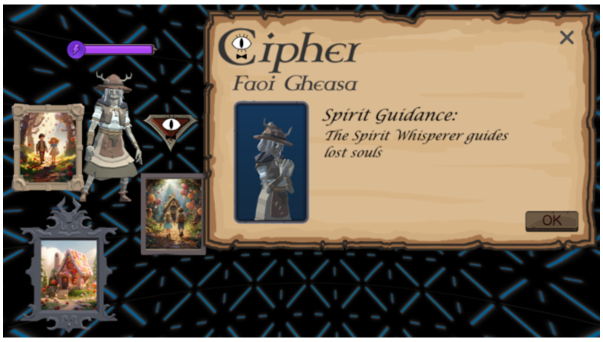
In Cipher, adaptivity is implemented by varying the number of Ciphers (spells) on each page, as well as adjusting the page's text length and the difficulty level of the text.

3. Cipher – an adaptive game with a purpose

Cipher, an adaptive language learning game, epitomizes the fusion of digital innovation and pedagogical strategy to enhance engagement and efficacy in learning under-resourced languages, notably Irish (Xu, 2023). The game focuses on engaging learners with tasks that promote incidental language learning through gameplay, centred around correcting language errors in the guise of magic spells. This approach not only makes learning interactive but also culturally resonant by embedding Irish mythology into the gameplay.

Acquah and Katz (2020) remarked on the lack of focus on cultural competence in the studies they reviewed, and recommended more research into understanding how digital learning games can be used to improve sociocultural outcomes. Attention to the 'spirit of the language' is a core feature of the Cipher language learning game, with a focus on folk tales and mythology which are culturally specific.

Content creation is carried out in collaboration with teachers ensuring that the game's educational material is aligned with curriculum standards, further enhancing its pedagogical value.



Spirit Guidance:
The Spirit Whisperer guides lost souls

Figure 1: A screenshot of Cipher

Cipher's adaptability, which is also underscored by its modular design, showcases its potential for broader application across different linguistic contexts. Its development journey illustrates the power of combining engaging game mechanics with robust educational content, making it a model for future digital language learning tools.

4. Co-creation and Ease-of-use

The development of Cipher underscores the importance of a co-creation approach to educational game development, as it harnesses the collective expertise of a multidisciplinary team in crafting a digital game-based language learning resource. This collaboration is vital, as aligning the game with educational curricula is a critical factor for its acceptance and use in schools as highlighted by Ward (2007).

Given the challenges of assembling a comprehensive team for under-resourced languages, Cipher's development adopted a pragmatic strategy, utilising a core team consisting of a game designer, an NLP researcher specialising in Irish, a Computer Assisted Language Learning (CALL) researcher, and a primary school teacher. This team effort was essential in ensuring the game's relevance and suitability to learners' needs by incorporating curriculum-aligned content and simplifying language ensuring enhanced accessibility (Ward, 2023). An important advantage of this co-creation and curriculum-aligned approach is that the game is more suited to the needs of learners highlighting the necessity of diverse expertise in developing such resources.

Some enhancements to the Cipher game include the selection of texts at the correct level for the learners (see section 5) and the use of simpler and shorter sentences in the texts. While these may seem like obvious elements, it was essential to have confirmation from the teacher that these were necessary changes to the game. This co-creation approach, combined with a focus on ease-of-use and relevance for students enabled the development of an enhanced version of Cipher that would otherwise not have been possible.

5. Text analysis for adaptive learning

Xia et al (2016) note that while most studies of text readability are carried out by native speakers, L2 learners can perceive text comprehensibility very differently. Therefore a system such as Cipher that adapts to the L2 learner's perception of text difficulty based on their performance in the game will better facilitate language learning.

In order to provide texts of an appropriate level for adaptive learning it is necessary to be able to classify texts according to their relative complexity. In relation to second language acquisition, Michel (2017) notes that language complexity can be defined in a number of ways. It can be thought of as a) developmental complexity, i.e. the order in which linguistic structures are typically acquired, b) cognitive complexity, the subjective difficulty of a language feature from the perspective of the learner and c) linguistic complexity, i.e. objective complexity of forms, meanings and form-meaning mappings (Michel, 2017). In this section we focus on linguistic complexity and the features that can be used to determine the complexity of a text. In the literature, a wide variety of measures have been used to calculate linguistic complexity using a combination of lexical, grammatical and discourse features.

Regarding lexical measures, Lu (2012) describes lexical richness as a combination of lexical density (ratio of content words to total words), lexical variation or diversity (ratio of different words to total words) and lexical sophistication (proportion of advanced or sophisticated words in a text, using frequency lists or specialised word lists). Lexical familiarity/unfamiliarity (Collins-Thompson, 2014) is a similar concept to lexical sophistication.

Grammatical complexity can be characterised in terms of syntactic variation and sophistication, and most measures are based on mean length of sentence/utterance or on structural analysis (Lu, 2010; Vajjala, and Meurers, 2012).

Discourse measures of text quality include text cohesion, discourse relations and entity chaining (Pitler & Nenkova, 2008; Feng et al, 2010). Measures of discourse complexity that are associated with comprehension difficulty include propositional idea density (Kintsch & Keenan,1973; Brown et al, 2008). A text with greater propositional density is more difficult to comprehend than a text with lower propositional density. In this measure, predicates (e.g. verbs) modifiers (adjectives, adverbs, qualifiers) and conjunctions are considered to be indicative of propositions.

5.1 Text Analysis for Irish

Much of the research to date on text analysis and complexity measures has focussed on languages such as English (Vajjala & Meurers, 2012) while under-resourced languages such as Irish have not been researched to the same extent. In this section we calculate lexical, grammatical and discourse measures and investigate their usefulness as predictors for ranking and grading Irish texts. Given the small amount of pre-graded material available for an under-resourced language such as Irish, we choose to do a detailed linguistic analysis of the available material. This will satisfy our immediate need for grading texts for Cipher as well and help to generate graded material for machine learning methods.

We calculate lexical density as the ratio of content words to total words, lexical diversity includes corrected type-token ratio, and for grammatical complexity we use average sentence length, following findings for Irish in (Ó Meachair, 2019; Uí Dhonnchadha et al, 2022).

As noted in (Pitler and Nenkova, 2008) the vocabulary used in a text largely determines its readability, therefore we determine lexical sophistication (or familiarity) as the percentage of words that are among the N most frequent in a range of frequency bands using frequency word lists.

As regards discourse, since we are interested in reading comprehension difficulty rather than assessing text quality we investigate propositional idea density as described in (Brown et al, 2008).

5.2 Data and Methodology

We carry out a detailed analysis of pre-graded reading materials to develop predictive measures for ungraded material, i.e. to grade new stories for use in the Cipher game.

5.2.1 Data selection and preparation

Two sets of graded materials are used: *Taisce Tuisceana*[1] (TT), a collection of Irish reading comprehension texts for primary school students and *Séideán Sí*[2] (SS) a series of textbooks for *Gaeltacht* and Gaelscoileanna – Irish-medium primary schools in Ireland.

For initial exploration, we use ten samples of fiction from Taisce Tuisceana (TT). These texts are graded3 from A to E with A-C for Key Stage 1 (1st & 2nd class) and D-E for Key Stage 2 (3rd and 4th class). For increased robustness, we asked 5 Irish speakers to rank the texts in order of increasing difficulty based on their intuition, and we took the average of these ratings. While there was some variation within categories, there was very little variation between the categories, as shown in Table 1,which provided added confidence in the ranking.

---

[1] https://ccea.org.uk/learning-resources/taisce-tuisceana

[2] https://seideansi.ie/

[3] Graded A for 1st & 2nd class, B&C for 3rd and 4th class and D&E for 5th and 6th class on www.tairseachcogg.ie/

Taisce Tuisceana	Cat.	Class	R1	R2	R3	R4	R5	Ave Rank	StDev
A06	A	1st, 2nd	1	1	1	1	1	**1.00**	0.00
B22	B	1st, 2nd	3	2	2	2	2	**2.20**	0.45
B20	B	1st, 2nd	2	5	4	6	3	**4.00**	**1.58**
C23	C	1st, 2nd	5	3	3	3	6	**4.00**	**1.41**
C08	C	1st, 2nd	4	4	5	5	5	**4.60**	0.55
D05	D	3rd, 4th	6	7	6	4	4	**5.40**	**1.34**
D08	D	3rd, 4th	8	6	7	7	7	**7.00**	0.71
D21	D	3rd, 4th	7	8	9	9	8	**8.20**	0.84
D22	D	3rd, 4th	9	9	8	10	9	**9.00**	0.71
E02	E	3rd, 4th	10	10	10	8	10	**9.60**	0.89

Table 1: *Taise Tuisceana* texts and grading

The second dataset *Séideán Sí* (SS), is larger and more comprehensive. It consists of textbooks which are pre-graded for primary school students from first class (ages 6-7) to sixth class (ages 11-12). From this collection we selected 86 samples of fiction, 10% of which (8 files) were set aside for testing and the rest were used to develop grading measures.

Stage	Class	Age	Files	Words
Keystage 1 (6-8 years)	1st Class	7-8	30	5721
	2nd Class	8-9	24	15679
Keystage 2 (8-11 years)	3rd Class	9-10	6	15539
	4th Class	10-11	10	23585
	5th Class	11-12	9	28894
	6th Class	12-13	7	40793
TOTAL			86	130211

Table 2: *Séideán Sí* texts

The PDF files which were converted to text and cleaned, were then part-of-speech (POS) tagged and lemmatized using existing tools for Irish (Uí Dhonnchadha and Van Genabith, 2006). This rule-based POS tagger achieved an average accuracy of 97% on a random selection of eight *Séideán Sí* texts covering all levels. Along with these stories, a 10K frequency wordlist was extracted from the fiction part (6 million words approx.) of the New Corpus of Ireland[4] (NCI).

5.2.2 Calculation of Text Statistics

Text statistics were obtained for each story using Python programs which process each POS-tagged file to get the required lexical, grammatical and frequency measures.

For the lexical measures, TTR (type token ratio), WTR (word type ratio) and CTTR (corrected type token ratio) were calculated by using formulae in (Vajjala & Meurers, 2012; Lu, 2012). For the grammatical measure, WDSEN (average number of words per sentence) was calculated along with various other text statistics as shown in Table 3. For word familiarity (sophistication) measures, we compared the word types in each text with the frequency wordlist and calculated the number of words that were within frequency ranges 0-100, 100-300, 300-500, 500-1000, 1000-2000, 2000-3000, 3000-4000, 4000-5000, 10000+. Each story was labelled based on its year group e.g. first class stories were labelled '01_' followed by the story name while sixth class stories were labelled '06_' followed by the story name.

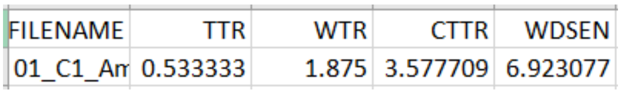
Finally, propositional idea density was calculated using POS tags for verbs, copula, adjectives, adverbs and conjunctions.

Feature Counter

The first tool calculates text statistics such as number of sentences, tokens, types, words, lemmas, longest sentence, and POS totals (using the first two characters of the PAROLE[5] POS tag e.g. Np, Vm, Pp, Nv, etc.) and nouns with genitive case. Table 3 gives an example of the output from this tool.

Lexical and Grammatical Calculator

The second tool calculates lexical and grammatical measures for the files. These measures were TTR, WTR, CTTR and WDSEN. Table 4 gives an example of the output from this tool.



FILENAME	TTR	WTR	CTTR	WDSEN
01_C1_Am	0.533333	1.875	3.577709	6.923077

Table 4: Output from the Lexical and Grammatical calculator tool

Frequency Range Tool

The third tool calculates the frequency measures for each file by using a wordlist of the 10,000 most frequently used words in Irish fiction. For every word in each file, if the word was within the frequency list, the appropriate frequency range was incremented, otherwise the 10K+ range was incremented. Table 5 gives an example of the output from this tool.

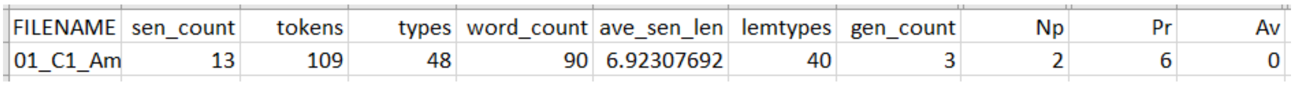
5.3 Results

Graphs were used to visualise the results. CTTR and average sentence length prove to be the most predictive measures, showing increasing values for first to sixth class texts. Figure 2 shows the range of CTTR values for *Séideán Sí* (SS) files (labels 01 – 06), and *Taisce Tuisceana* (TT) files (labels A – E). Figure 3 shows the average sentence length for SS and TT files.

---

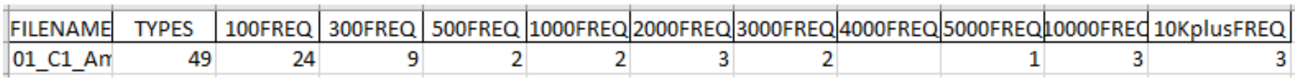
[4] https://corpas.focloir.ie/

[5] https://www.scss.tcd.ie/~uidhonne/parole.htm



FILENAME	sen_count	tokens	types	word_count	ave_sen_len	lemtypes	gen_count	Np	Pr	Av
01_C1_Am	13	109	48	90	6.92307692	40	3	2	6	0

Table 3: Output from the Feature counter tool



FILENAME	TYPES	100FREQ	300FREQ	500FREQ	1000FREQ	2000FREQ	3000FREQ	4000FREQ	5000FREQ	10000FREC	10KplusFREQ
01_C1_An	49	24	9	2	2	3	2		1	3	3

Table 5: Output from Frequency tool

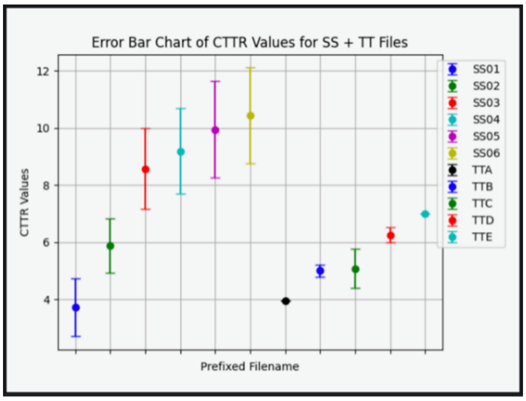


Figure 2: CTTR values for SS, TT texts

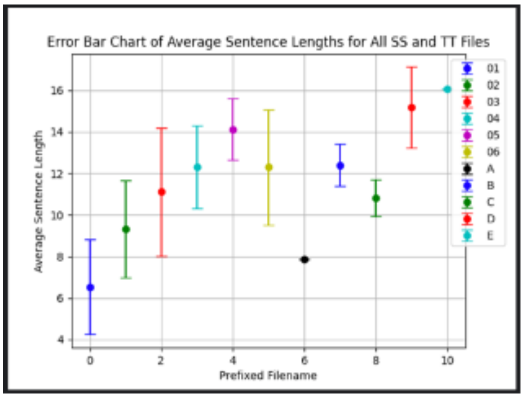


Figure 3. Average sentence length values for SS (01-06) and TT(A-E) files

Figures 4 and 5 show each SS and TT text groups and their percent of types in each frequency range from 0-100 to 10K+. In this Zipf-like curve the proportion of words in the 1-100 frequency range appears to be the most useful frequency range for predicting text level.

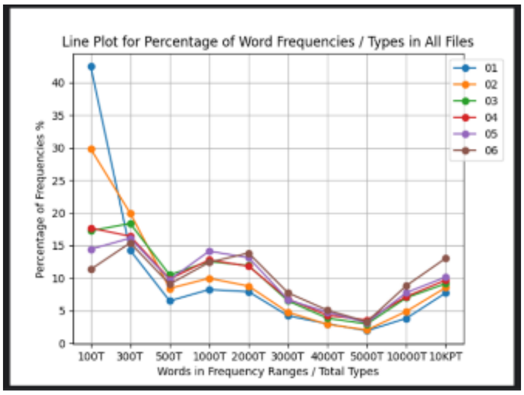


Figure 4: Type/Frequency ranges of SS texts

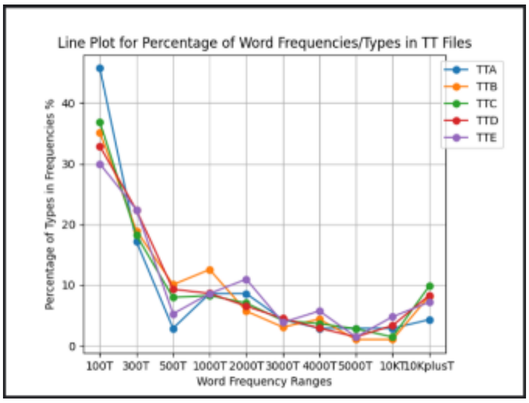


Figure 5: Type/Frequency ranges of TT texts

The graphs show that the measures of CTTR, average sentence length and frequency for the pre-graded texts are behaving as one would expect, with an increase from first class texts to sixth class texts.

In Figure 2 we see that the TT files fall within the range of SS 1st to 3rd class (01-03). In Figure 3 the average sentence length for TT files fall within the range of 1st to 6th class and higher. It is also apparent that higher level texts do not always have a greater average sentence length, as can be seen from 05 and 06 texts as well as B and C texts. This suggests that a composite measure is required as there is often a trade-off between lexical complexity and grammatical complexity (Graesser and McNamara, 2011).

By plotting the ungraded texts alongside these graded texts, we can infer a grading for the texts based on where they are in the plots in relation to the SS and TT files.

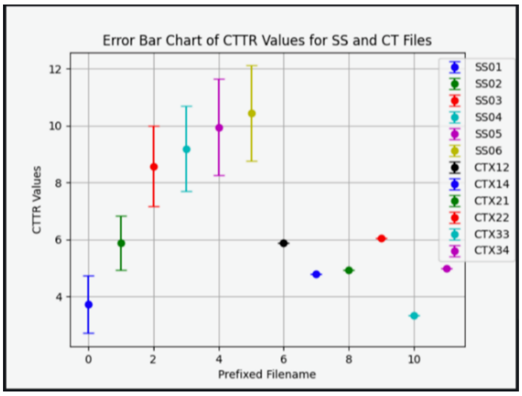


Figure 6: CTTR for SS and ungraded Cipher Texts (CT)

In Figure 6 we show CTTR for six ungraded Cipher Texts (CT) in comparison to the SS texts, and in Figure 7 we show average sentence length for the same six ungraded Cipher Texts (CT) in comparison to the SS texts.

The Cipher texts comprise of two Irish mythology texts (CTX12, CTX14), two Dúchas[6] Irish folklore texts (CTX21, CTX22) and two international fairy tales (CTX33, CTX34).

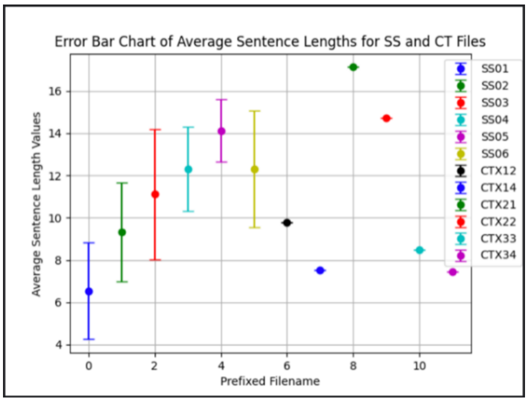


Figure 7. Average sentence length values for SS and ungraded Cipher Texts (CT)

As we can see from Figures 6 and 7 the mythology texts (CTX12, CTX14), fall within the 2nd and 3rd class range, the Dúchas texts (CTX21, CTX22) are lexically in the 2nd class range but are in the 5th and 6th range and above in terms of sentence length, while the fairy tales are in the 1st and 2nd class ranges. These results are as expected. In particular the results for the Dúchas texts are interesting, given that these texts were written by L1 Irish children in 5th 6th class in the 1930's in a National Folklore Collection project.

5.3.1 Composite predictive measures

These visual results are useful, but we need to able to automatically assign a grade in the range 1-6 to ungraded texts. Predicting the grade level of ungraded texts, was carried out in an incremental manner. We began with one feature, CTTR. From our pre-graded SS texts (Fig. 2), we could estimate value ranges for this measure for each age group. For example, we could see the majority of the first class files had a CTTR value between 1 and 4, so we set our first range to be 1 to 4, while range 2 which correlated with second class was 4 to 7.6, range 3 for third class was 7.6 to 8.6, range 4 for fourth class was 8.6 to 9.5, range 5 for fifth class was 9.5 to 13.5 and range 6 for sixth class was 13.5 plus. With every file, the CTTR value was compared with these ranges and when a value landed in a particular range, that range number (1 to 6) was returned to give the CTTRRangeValue. From using this measure alone, we could see there were multiple outliers in each age group that were giving a higher range value than expected so to help account for those outliers, the WDSEN measures was added.

The same principles were followed for this measure by having different ranges for this measure. With these two measures, an average was taken from the CTTRRangeValue and WDSENRangeValue to give a rounded predicted grade level. The predicted grade level was still not overly accurate for all six grade levels. Therefore the measure '100T' was added to the calculation following the same format and having its own range values. This measure calculates the percentage of types in a file that are within the first one hundred words of our frequent word list. The TypeRangeValue was added with the two other range values to get an overall average which still was unsatisfactory, leading to the addition of the lemma type count to the calculation. A set of ranges were set for lemma types and the value was added to the previous three features.

Feature	Exact match	Within 1 grade level
CTTR	58.76%	31.2%
+WDSEN	55.29%	36.4%
+100T'	59.4%	30.58%
+Lemma types	60.0%	34.12%

Table 6 Initial grade prediction results

The average from this had a more satisfactory result with 60% of the predictions being exactly right with grade level, while 34.1% of the predictions being out by one grade level. This is a good result with over 90% predictions being within one of the correct grade. See Table 6 for details.

Held out SS Text	Predicted Grade	Actual Grade
IgCéinisigCóngar.txt	5	6
AnMúinteoirNua.txt	4	3
Céhiadseo.txt	1	1
AnNathairagusnaSpléaclaí.txt	3	3
SinScéalEile.txt	5	5
MurachanTraenáil.txt	2	2
SciobAgusAnChuileog.txt	1	1
AChaitlínMí-abha.txt	3	3

Table 7 grade predictions for held out SS texts

To test this formula on unseen texts, a random selection of 10% held-out SS files were used to check accuracy levels, and for these 8 files, 70% of the predictions were exactly right with 30% of the predictions being out by one grade level. See Table 7 for details. The predictive grading measure is used to assign a grade to the ungraded Cipher texts. These results are in keeping with the visual results in Figures

---

[6] National Folklore Collection - Dúchas https://www.duchas.ie/en/cbes/stories

5 and 6 and with the team members expectations. The Dúchas texts (CTX21, CTX22) are predicted to be suitable for 3rd and 4th grade which reflects a combination of 2nd grade lexical features and 5th and 6th grade sentence length.

These automatic grading results are promising and we intend to verify them by carrying out more extensive testing on unseen, manually graded texts.

A limitation of this study is the small sample size. Also we limited our study to only fiction texts, therefore it is unclear how well the predictions would generalise to other types of text.

6. Conclusion

The development and implementation of Cipher highlights an advancement in digital game-based language education, particularly for under-resourced languages. Through analysis techniques and a focus on adaptivity, Cipher offers a personalised learning experience that dynamically adjusts to the learner's proficiency level, ensuring that the challenge remains within the learner's zone of proximal development. The integration of Irish culture and mythology not only enhances engagement and motivation but also fosters a deeper connection with the language. Preliminary findings suggest that such adaptive educational games hold promise in transforming language learning by making it more engaging, effective, and enjoyable. By addressing the unique challenges associated with teaching and learning less commonly taught languages, Cipher paves the way for future research and development in the field of digital game-based language learning, and in this case has inspired the development of new NLP text analysis tools for Irish.

In a collaborative co-creation model we aim to increase player engagement and educational effectiveness in Cipher by focusing on enhanced adaptivity in response to player progress in the game. We also aim to increase engagement by improving ease-of-use through more gradual transitions between levels and the use of simpler and shorter sentences in the texts in the earliest levels.

This study exemplifies the potential of combining gaming technology and CALL research with linguistic analysis and cultural content to enrich language education, offering valuable insights for educators, developers, and researchers who aim to enhance language learning outcomes through innovative digital solutions.

7. Bibliographical References

Acquah, E.O. and Katz, H.T. (2020). Digital game-based L2 learning outcomes for primary through high-school students: A systematic literature review. Computers & Education, Volume 143.

Brown, C., Snodgrass, T., Kemper, S.J., Herman, R. & Covington, M.A. (2008). Automatic measurement of propositional idea density from part-of-speech tagging. Behavior Research Methods 40, 540–545.

Collins-Thompson, K. (2014). Computational assessment of text readability: A survey of current and future research. ITL - International Journal of Applied Linguistics, 165(2):97–135.

Feng, L., Jansche, M., Huenerfauth, M., & Elhadad, N. (2010). A comparison of features for automatic readability assessment. COLING 2010 Poster Volume, pp. 276-284.

Kickmeier-Rust, M. D., Hockemeyer, C., Albert, D., & Augustin, T. (2008). Micro adaptive, non-invasive knowledge assessment in educational games. In 2008 Second IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning (pp. 135-137).

Kintsch, W.& Keenan, J. (1973). Reading rate and retention as a function of the number of propositions in the base structure of sentences. Cognitive Psychology,5:3, pp 257-274.

Lu, X. 2010. Automatic analysis of syntactic complexity in second language writing. International Journal of Corpus Linguistics, 15(4):474–496.

Lu, X. (2012), The Relationship of Lexical Richness to the Quality of ESL Learners' Oral Narratives. The Modern Language Journal, 96: 190-208.

Michel, M. (2017). Complexity, Accuracy and Fluency in L2 production. In S. Loewen, & M. Sato (Eds.), Routledge Handbook of Instructed Second Language Acquisition (pp. 50-68). (Routledge handbooks in applied linguistics). Routledge.

Ó Meachair, M. J. (2019). The Creation and Complexity Analysis of a Corpus of Educational Materials in Irish (EduGA). PhD, University of Dublin, Trinity College.

Peirce, N., Conlan, O. and Wade, V. (2008). Adaptive educational games: Providing non-invasive personalised learning experiences. In 2008 second IEEE international conference on digital game and intelligent toy enhanced learning, pp. 28-35.

Pitler, E. and Nenkova, A. (2008). Revisiting readability: A unified framework for predicting text quality. In Proceedings of the 2008 conference on empirical methods in natural language processing, pages 186–195.

Proceedings of the 2008 conference on empirical methods in natural language processing, pages 186–195.

Uí Dhonnchadha, E., Ward, M. & Xu, L. (2022). Cipher – Faoi Gheasa: A Game-with-a-Purpose for Irish. In Proceedings of the CLTW 4 @ LREC2022 , Marseille, June 2022, pp 77–84

Uí Dhonnchadha, E. & Van Genabith, J. (2006). A Part-of-speech tagger for Irish using Finite-State Morphology and Constraint Grammar Disambiguation. LREC 2006, May 2006 Genoa.

Vajjala. S. (2022). Trends, Limitations and Open Challenges in Automatic Readability Assessment Research. In Proceedings of the Thirteenth

Language Resources and Evaluation Conference, pages 5366–5377, Marseille, France.

Vajjala, S. & Meurers, D. (2012). On Improving the Accuracy of Readability Classification using Insights from Second Language Acquisition. 7th Workshop on the Innovative Use of NLP for Building Educational Applications, 2012 Monréal, Canada. ACL, 163-173.

Vygotsky, L. S. (1978). Mind in society: The development of higher psychological processes. Cambridge, MA: Harvard University Press.

Ward, M. (2007). The integration of CL resources in CALL for Irish in the primary school context (Doctoral dissertation, Dublin City University).

Ward, M., Uí Dhonnchadha, E., McGarry J. and Xu, L. (2023). Co-creating CALL content - does it work?Goldilocks compromise or Cruella chaos? In CALL for all Languages - EUROCALL 2023 Short Papers. 15-18 August 2023,University of Iceland, Reykjavik.https://doi.org/10.4995/EuroCALL2023.2023.16961

Xia, M., Kochmar, E., & Briscoe, T. (2019). Text readability assessment for second language learners. arXiv preprint arXiv:1906.07580

Xu, L., Uí Dhonnchadha, E and Ward, M., Harnessing the Power of Images in CALL: AI Image Generation for Context-Specific Visual Aids in Less Commonly Taught Languages, EUROCALL2023, Reykjavík, Iceland, 15-18 August, 2023

`Hostomytho`: A GWAP for Synthetic Clinical Texts Evaluation and Annotation

Nicolas Hiebel[1], **Bertrand Remy**[2], **Bruno Guillaume**[2],
Olivier Ferret[3], **Aurélie Névéol**[1], **Karën Fort**[4]
[1]Université Paris Saclay, CNRS, LISN, Orsay, France
[2]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
[3]Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
[4]Sorbonne Université / Université de Lorraine, CNRS, Inria, LORIA, France
[1]{nicolas.hiebel, aurelie.neveol}@lisn.upsaclay.fr
[2]{bertrand.remy, bruno.guillaume}@inria.fr
[3]olivier.ferret@cea.fr, [4]karen.fort@loria.fr

Abstract

This paper presents the creation of `Hostomytho`, a game with a purpose intended for evaluating the quality of synthetic biomedical texts through multiple mini-games. `Hostomytho` was developed entirely using open source technologies both for internet browser and mobile platforms (IOS & Android). The code and the annotations created for synthetic clinical cases in French will be made freely available.

Keywords: GWAP, Text Generation, Evaluation, Clinical Texts, Synthetic Texts, French

1. Introduction

One of the most common hurdles in Natural Language Processing (NLP) is the lack of specific resources, whether it be task-specific resources, domain-specific resources, or both. A major challenge for clinical NLP is the lack of shared clinical corpora in languages other than English (Névéol et al., 2018). One potential approach to address this problem is to generate new corpora automatically. The generated corpus should share as many characteristics as possible with the natural corpus, without simply copying it. Thus, evaluating the quality of the generated corpus is crucial. In this work, we generate synthetic clinical texts from real clinical corpora.

We decided to develop a Game With A Purpose (GWAP) to help with the evaluation of the synthetic texts as GWAPs have been proven to be a promising alternative to traditional human annotation.

In this paper we present `Hostomytho`, a game made for manually evaluating synthetic clinical documents. The game is multi-platform and developed using open source technologies.

The main contribution of this work is an open source game platform set-up to collect linguistic resources to address the following research questions:

- Can a GWAP be a suitable interface for the evaluation of text generation?
- Is medical training needed for evaluating clinical text?
- Can high quality annotations be collected for this complex evaluation task?

2. GWAPs for Language Resources

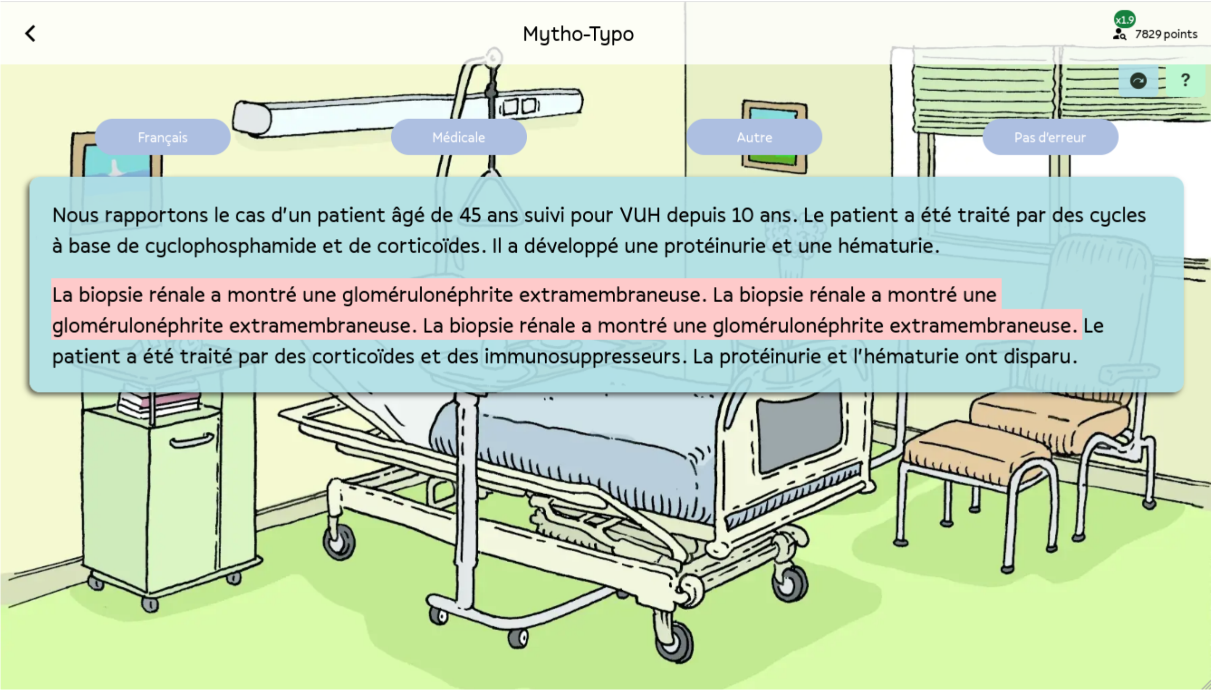
GWAPs have been used with success for nearly two decades in NLP (Lafourcade, 2007; Chamberlain et al., 2008) to create a wide variety of language resources, from part-of-speech tags (Madge et al., 2019) to word-sense labels (Venhuizen et al., 2013). They proved efficient, even on complex tasks that require training, like dependency syntax annotations (Guillaume et al., 2016). Moreover, they do not present the same ethical issues as microworking crowdsourcing (Fort et al., 2011). To our knowledge, there has been yet no GWAP developed to validate and annotate specialized synthetic texts.

3. Evaluation of Natural Language Generation

Natural Language Generation (NLG) is an area of NLP that has grown in popularity with the advent of pre-trained large language models. A major challenge when doing NLG is the evaluation part. Existing automatic evaluation methods are limited (Novikova et al., 2017) and new measures are often put forth to address those limitations (Frisoni et al., 2022; Pillutla et al., 2021).

It is still accepted that manual evaluation is the best way to evaluate the quality of automatically generated text despite the creation of new automatic metrics. However, manual evaluation of text generation also comes with a number of challenges (Gehrmann et al., 2023; Celikyilmaz et al., 2021). Assessing the overall quality of long sequences of text makes it difficult to maintain consistence during evaluation. This is further exacerbated by the

Games and NLP @LREC-COLING-2024, pages 14–20
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0



Nous rapportons le cas d'un patient âgé de 45 ans suivi pour VUH depuis 10 ans. Le patient a été traité par des cycles à base de cyclophosphamide et de corticoïdes. Il a développé une protéinurie et une hématurie.

La biopsie rénale a montré une glomérulonéphrite extramembraneuse. La biopsie rénale a montré une glomérulonéphrite extramembraneuse. La biopsie rénale a montré une glomérulonéphrite extramembraneuse. Le patient a été traité par des corticoïdes et des immunosuppresseurs. La protéinurie et l'hématurie ont disparu.

Figure 1: Mytho-Typo: an error type specification game.

broad definition of the terms used to gauge text quality (e.g. fluency, coherence) (Howcroft et al., 2020). Additionally, reviewing long passages of text can quickly become tedious, even more so when it comes to clinical documents which may contain bad outcomes.

4. Evaluating Synthetic Clinical Documents

4.1. Corpus

The generated texts we use in `Hostomytho` come from models trained on French clinical cases (Hiebel et al., 2023). The training documents were collected from the CAS corpus (Grabar et al., 2018) and the French part of the E3C corpus (Magnini et al., 2020), two freely available corpora. An example of a generated clinical case can be seen in Example (1).

(1) *Il s'agissait d'un patient de 50 ans, sans antécédents pathologiques particuliers, admis aux urgences pour des douleurs épigastriques aiguës associées à une distension abdominopelvienne évoluant depuis deux jours. L'examen clinique trouvait un patient en assez bon état général (Apgar: 10/10). Le bilan préopératoire objectivait une fonction rénale normale et la CRP était à 12 mg/l.*[1]

We also added some real clinical cases and some irrelevant documents in the game in order to control the quality of the annotations. The irrelevant document are taken from the corpus Est Républicain (ATILF and CLLE, 2020), a journalistic corpus. We selected documents with vaguely medical content. We expect that there will be no annotation error on the real clinical cases and that irrelevant texts will be detected as such.

4.2. Grammar, fluency and clinical coherence

Several types of errors can be found in automatically generated clinical texts. Some of them might be easy to spot, for example when the text has clear grammatical or fluency issues.

However, most of the time, recent language models manage to generate fluent text. Working with data from a specialized field such as the clinical domain comes with additional challenges. Medical knowledge might be required to spot clinical incon-

---

[1] Translation into English: *The patient is a 50 years old male admitted to the emergency room with a 2-day history of acute epigastric pain associated with abdominopelvic distension. His past medical history was unremarkable and he was generally in a good state of health (Apgar: 10/10). The preoperative workup showed normal renal function and CRP was at 12 mg/l.*

sistencies. Those are often due to the combination of several elements in the text, unproblematic when taken separately. Looking at example (1), a 50 year old patient is associated with an Apgar score, a method intended to evaluate the health of newborn babies.

Many error typologies exist (Howcroft et al., 2020). Looking at the generated texts and for simplicity, we identify three main types of error:

- grammatical errors: it can be non-existing words or ungrammatical constructions;
- fluency errors: the text seems to be a sequence of unconnected parts or has repeated parts;
- clinical inconsistency: the text is grammatical and fluent, but contains clinically contradictory evidence.

4.3. Divide and Conquer Approach

As mentioned in Section 4.2, evaluating the quality of generated text is a complex multidimensional task. Trying to evaluate a text in detail in one go is intellectually demanding and can quickly become tedious and prone to errors.

Bernstein et al. (2015) proposed the "find-fix-verify" workflow for a writing assistance service to reduce cost and to ensure annotation quality. The task is decomposed in three stages involving different annotators: (i) annotators identify an area of the text that could be improved, (ii) annotators propose modifications to improve a previously identified area and (iii) annotators validate or invalidate the candidate modifications.

We also decided to decompose the evaluation process in different tasks, both to ease the mental burden of players and to have more control over the different types of annotations.

`Hostomytho` currently includes two games. The first game consists in assessing the plausibility of a given text on a scale of five labels ranging from highly implausible to very plausible. The player can select a span of text if an error is present. The second game exploits the results of the first game. The player must classify the type of errors that were annotated. An example can be seen on Figure 1. The player has to choose between four options given a text where the annotated error is highlighted. We've kept the number of options low for simplicity. They are as follows:

- ***Français***: French, for grammatical and fluency errors;
- ***Médicale***: medical, for medical inconsistencies;
- ***Pas d'erreur***: no errors, when the span of text was mistakenly annotated as error.
- ***Autre***: others, for errors that do not fall into other categories.

On the example text of Figure 1, the error highlighted is the repetition of the same sentence three times in the text. This is a fluency error that should be classified with the label *Français* (French).

Each game starts with a tutorial that helps the player understand the current task. At the end of the tutorial, the player gets to practise on a sample of texts for which gold standard annotations are available. This helps us ensure that the player understood the task well enough before starting the annotation of new texts.

4.4. Control over Annotation Quality

We plan to control the quality of the annotations in two different ways. First, we will check agreement between players on each task by sharing some samples between players. This will also help us assess the difficulty of the task.

Second, we assign a neutral reliability score to every player on account creation (50 on a scale from 0 to 100). We will occasionally give players control samples where the correct answer is known and the reliability score will increase or decrease depending on the players' answers on the control samples.

Annotations given by a player with a high reliability score will carry more weight that those given by a player with a low reliability score.

5. Player types

Players can find satisfaction in different elements depending on their profile and several player taxonomies have been proposed (Bartle, 1996; Tondello et al., 2016). Here's a brief description of the four types of players according to Bartle (1996):

- *Achievers* enjoy accomplishing different things in the game;
- *Explorers* enjoy discovering every parts of the game;
- *Socializers* enjoy interacting with other players;
- *Killers* enjoy attacking other players.

We will link the types of players with the game elements in Section 6.2.

6. Presenting `Hostomytho`

6.1. The Universe

The task of differentiating between real clinical documents and synthetic clinical documents is the main motive for `Hostomytho`'s universe.

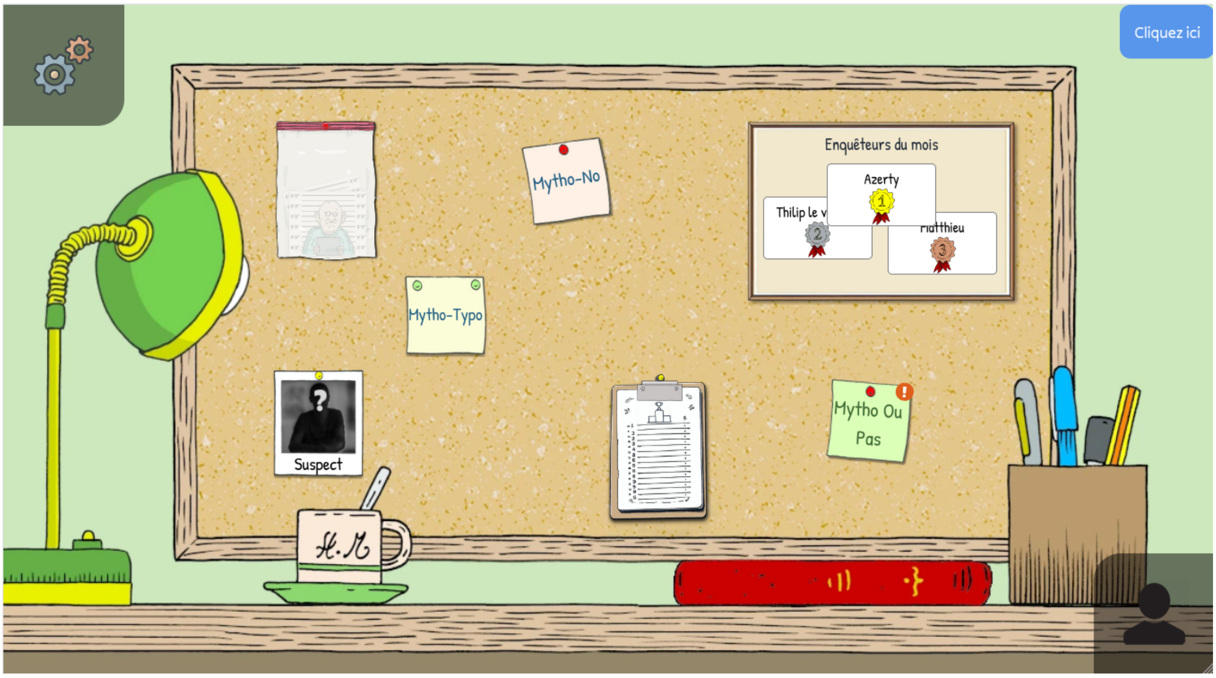


Figure 2: Main menu with leaderboard on the top right corner of the cork board.

We are casting this as a mystery scenario. In the game story, criminals have escaped and found refuge in a hospital. They hide by pretending to be doctors. The player takes the role of an investigator mandated to unmask the criminals hidden in the hospital. The player investigates by analyzing the clinical documents produced by the hospital and looking for potential errors. As the player gathers more clues, the investigation progresses, leading to the arrest of increasingly tough suspects.

6.2. Game Elements

Game elements in `Hostomytho` focus on earning points and progressing the investigation of criminals. We offer several game mechanics to meet the needs of different types of players, as identified in (Bartle, 1996).

6.2.1. Leaderboard and Ranking

The point system represents the player's overall progression. Points are acquired by playing the different mini-games. Players can keep track of their rank by checking the leaderboard. This encourages the players to play more to move up in the rankings.

In addition to a global leaderboard, we added a special spot on the main menu for the best investigators of each month. With a monthly ranking, players should come back regularly to be on top of the ladder.

Figure 2 shows the main menu of `Hostomytho`. The monthly top three investigators are displayed in the top right corner of the cork board. Players having a chance to have their username and avatar displayed in the main menu for everyone to see should be motivating to play for more points.

This part of the game focusing on being the best player should appeal to achievers.

6.2.2. Investigation and Achievements

The player's main goal in `Hostomytho` is to arrest as many criminals as possible. Players can try to catch the criminal they're currently tracking at any time during the game. Each arrest has a certain chance to succeed depending on the player's "certainty". The certainty score can be increased by completing more tasks. We hope this system will encourage players to complete more tasks in order to maximize their chances of success.

The player tracks one criminal at a time. With each arrest, they move on to the next criminal and will progressively encounter criminals harder to catch. We want to give players the feeling that they're making progress in the investigation so that they'll want to go further.

We also use achievements as a way of motivating players and rewarding them for playing the game regularly. Achievements can be obtained by completing various objectives like arresting criminals or playing the game for several days in a row. Besides, unlocking achievements increases the rate

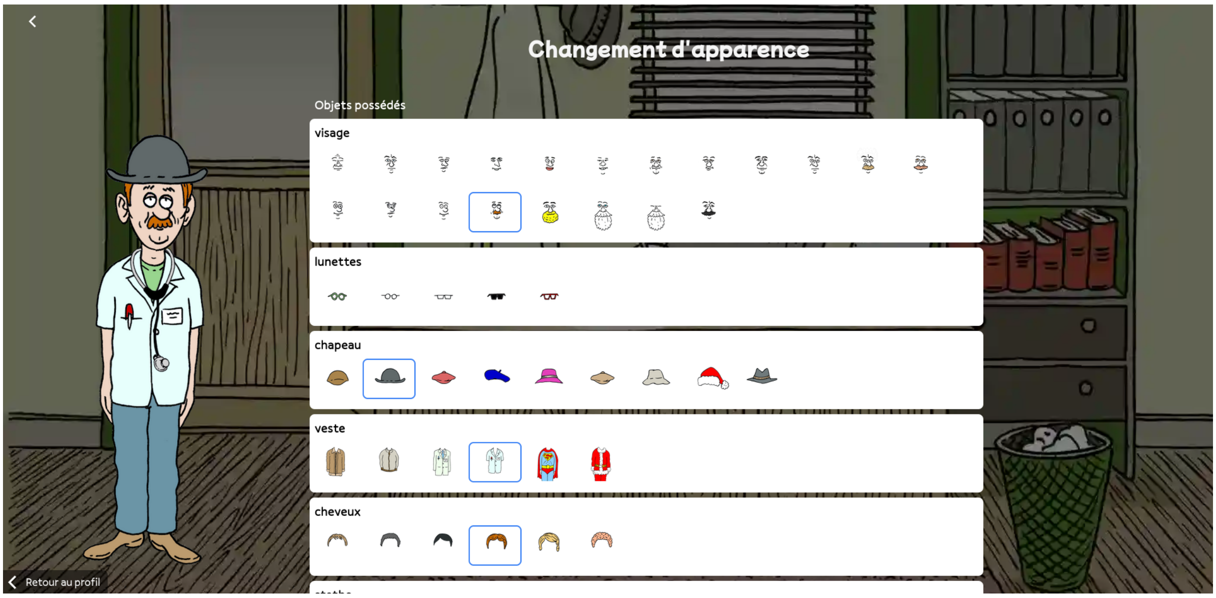


Figure 3: Character customization menu.

at which players earn points, making achievements not only satisfying but also useful for progression.

Progressing in the investigation and earning achievements should also appeal to achievers. In addition, discovering new criminals should appeal to explorers.

6.2.3. Skins and Customization

In `Hostomytho`, each player uses an avatar that represents their investigator. The avatar can be customized with hair, hats, clothes, and other accessories. Players will regularly unlock new items for customization when earning points. Figure 3 shows an example of the character customization menu (with some items already unlocked). Players may find satisfaction in personalizing their own investigator's avatar with the different items they unlocked. Some items are less common than others and discovering new ways of customizing the avatar might excite the curiosity of players.

Collecting customization elements should appeal to explorers.

6.2.4. Covering All Player Profiles

At the time of writing, the game elements of `Hostomytho` are primarily aimed at achievers and explorers. We plan to add game elements that will meet the needs of the other types of players.

For the socializers, we plan to add a friend system so that players can compare their scores with those of their friends. In addition, players will be able to group in companies of investigators and work together to place their company at the top of the company leaderboard. Finally for the killers, who like to attack other players, we plan to add the possibility of playing the role of a criminal. In this role, the player will be able to select a generated (fake) text from several generated texts. The selected text is then presented to an investigator. If the investigator finds no error in the text, the criminal will have succeeded in deceiving the investigator and will earn points.

7. Conclusion and Future Work

`Hostomytho` development is already well underway. At the time of writing, two mini-games are already available and the game is being tested for bugs and feedback. We are planning to add more games to obtain different annotations, which should be facilitated by the reliable base we already have. New games should include negation detection, hypothesis detection and condition detection. These annotations could help improve existing information extraction tools in the clinical domain by providing a more detailed representation of the clinical case. The code for `Hostomytho` is completely open-source and will be made available when the game is stable.

We plan to annotate two sets of generated texts. The first set will be generated with models trained on the freely available corpora or clinical cases in French CAS and E3C. For the second set, we plan to train the models on non-shareable medical reports in French. We will wait for the committee's approval to add the texts to the game. The annotations of the texts generated from the freely available corpora will also be freely available. We will also wait for the committee's approval to share the annotations on the second set of texts, generated from the private data.

8. Acknowledgements

This work has received funding from the French "Agence Nationale pour la Recherche" under grant agreement CODEINE ANR-20-CE23-0026-01.

9. Bibliographical References

R. Bartle. 1996. Hearts, clubs, diamonds, spades: Players who suit MUDs. *The Journal of Virtual Environments*, 1(1).

Michael S. Bernstein, Greg Little, Robert C. Miller, Björn Hartmann, Mark S. Ackerman, David R. Karger, David Crowell, and Katrina Panovich. 2015. Soylent: a word processor with a crowd inside. *Commun. ACM*, 58(8):85–94.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2021. Evaluation of Text Generation: A Survey. *arXiv:2006.14799 [cs]*.

Jonathan Chamberlain, Massimo Poesio, and Udo Kruschwitz. 2008. Phrase Detectives: a web-based collaborative annotation game. In *Proceedings of the International Conference on Semantic Systems (I-Semantics'08)*, Graz, Austria.

Karën Fort, Gilles Adda, and Kevin Bretonnel Cohen. 2011. Amazon Mechanical Turk: Gold mine or coal mine? *Computational Linguistics (editorial)*, 37(2):413–420.

Giacomo Frisoni, Antonella Carbonaro, Gianluca Moro, Andrea Zammarchi, and Marco Avagnano. 2022. NLG-metricverse: An end-to-end library for evaluating natural language generation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3465–3479, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research*, 77:103–166.

Bruno Guillaume, Karën Fort, and Nicolas Lefebvre. 2016. Crowdsourcing complex language resources: Playing to annotate dependency syntax. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, Osaka, Japan.

Nicolas Hiebel, Olivier Ferret, Karen Fort, and Aurélie Névéol. 2023. Can synthetic text help clinical named entity recognition? a study of electronic health records in French. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2320–2338, Dubrovnik, Croatia. Association for Computational Linguistics.

David M. Howcroft, Anya Belz, Miruna-Adriana Clinciu, Dimitra Gkatzia, Sadid A. Hasan, Saad Mahamood, Simon Mille, Emiel van Miltenburg, Sashank Santhanam, and Verena Rieser. 2020. Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 169–182, Dublin, Ireland. Association for Computational Linguistics.

Mathieu Lafourcade. 2007. Making people play for lexical acquisition. In *Proc. of the 7th Symposium on Natural Language Processing (SNLP 2007)*, Pattaya, Thailand.

Chris Madge, Richard Bartle, Jon Chamberlain, Udo Kruschwitz, and Massimo Poesio. 2019. Making text annotation fun with a clicker game. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, FDG'19, pages 77:1–77:6, New York, NY, USA. ACM.

Aurélie Névéol, Hercules Dalianis, Sumithra Velupillai, Guergana Savova, and Pierre Zweigenbaum. 2018. Clinical natural language processing in languages other than english: opportunities and challenges. *Journal of Biomedical Semantics*, 9(1):12.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252, Copenhagen, Denmark. Association for Computational Linguistics.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. MAUVE: Measuring the Gap Between Neural Text and Human Text using Divergence Frontiers. In *Advances in Neural Information Processing Systems*, volume 34, pages 4816–4828.

Gustavo F. Tondello, Rina R. Wehbe, Lisa Diamond, Marc Busch, Andrzej Marczewski, and Lennart E. Nacke. 2016. The gamification user types hexad scale. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY'16, pages 229–243, New York, NY, USA. Association for Computing Machinery.

Noortje Joost Venhuizen, Valerio Basile, Kilian Evang, and Johan Bos. 2013. Gamification for word sense labeling. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Short Papers*, pages 397–403, Potsdam, Germany. Association for Computational Linguistics.

10. Language Resource References

ATILF and CLLE. 2020. *Corpus journalistique issu de l'Est Républicain*. PID http://redac.univ-tlse2.fr/corpus/estRepublicain.html. ORTOLANG (Open Resources and TOols for LANGuage) – www.ortolang.fr.

Grabar, Natalia and Claveau, Vincent and Dalloux, Clément. 2018. *CAS: French Corpus with Clinical Cases*. Association for Computational Linguistics. PID https://deft.lisn.upsaclay.fr/2020.

Bernardo Magnini and Begoña Altuna and Alberto Lavelli and Manuela Speranza and Roberto Zanoli. 2020. *The E3C Project: Collection and Annotation of a Multilingual Corpus of Clinical Cases*. CEUR-WS.org. PID https://github.com/hltfbk/E3C-Corpus.

Using In-context Learning to Automate AI Image Generation for a Gamified Text Labelling Task

Fatima Althani, Chris Madge, Massimo Poesio
Queen Mary Univ. Of London, United Kingdom
{f.althani, c.j.madge, m.poesio}@qmul.ac.uk

Abstract

This paper explores a novel automated method to produce AI-generated images for a text-labelling gamified task. By leveraging the in-context learning capabilities of GPT-4, we automate the optimisation of text-to-image prompts to align with the text being labelled in the part-of-speech tagging task. As an initial evaluation, we compare the optimised prompts to the original sentences based on imageability and concreteness scores. Our results revealed that optimised prompts had significantly higher imageability and concreteness scores. Moreover, to evaluate text-to-image outputs, we generate images using Stable Diffusion XL based on the two prompt types, optimised prompts and the original sentences. Using the automated LIAON-Aesthetic predictor model, we assigned aesthetic scores for the generated images. This resulted in the outputs using optimised prompts scoring significantly higher in predicted aesthetics than those using original sentences as prompts. Our preliminary findings suggest that this methodology provides significantly more aesthetic text-to-image outputs than using the original sentence as a prompt. While the initial results are promising, the text labelling task and AI-generated images presented in this paper have yet to undergo human evaluation.

Keywords: Games-with-a-Purpose, Prompt engineering, Generative AI

1. Introduction

Games-with-a-Purpose (GWAPs) for Natural Language Processing face the challenge of engaging players, primarily due to the lack of visuals, unlike their image-labelling counterparts (Lafourcade et al., 2015). Efforts to integrate visuals in GWAPs for NLP have been achieved by developing themes and designing virtual worlds. Nonetheless, while these visuals can be aesthetically appealing, they often fail to support the text being labelled directly. This challenge stems from the resource-intensive nature of creating relevant visuals to accompany each text that requires labelling. To address the lack of contextually relevant visuals in text-labelling GWAPs, we propose a novel approach that uses GPT-4's in-context learning capability to automate AI image generation for text-labelling games. This method included an exploratory approach to selecting a set of in-context tasks to generate an optimised prompt based on our part-of-speech tagging task's word and sentence pairs.

In this preliminary study, we evaluated both the optimised prompts and AI-generated images. To assess the optimised prompts, we measured both the average imageability and concreteness scores, comparing them with the original sentences. We then generated images using Stable Diffusion XL, the latest version of Stable Diffusion (Rombach et al., 2022), using both optimised prompts and the original sentences. Subsequently, the text-to-image outputs were evaluated using the latest version of the LAION-Aesthetics predictor model (Schuhmann et al., 2022) and assigned aesthetic scores for each generated image. After collecting all scores, we conducted a correlation analysis to compare imageability and concreteness with predicted aesthetic scores.

Based on previous work, we hypothesised that:

- **H1** Optimised text-to-image prompts will have higher imageability and concreteness scores compared to the original sentences.
- **H2** Text-to-image outputs of optimised prompts will receive higher predicted aesthetic scores than those generated from original sentences.
- **H3** The higher the imageability and concreteness scores of a prompt, the higher the aesthetic score of the text-to-image output.

The main findings of our preliminary study include:

- The design of a gamified text labelling task that features contextually relevant AI-generated images.
- A description of the iterative process we employed to produce the AI-generated images.
- A quantitative evaluation for both optimised prompts and their text-to-image outputs.

Finally, the main advantage of applying this method is the ability to generate context-relevant images for a text labelling task by utilising LLM's in-context learning abilities. This approach is readily accessible, as it requires designers to iteratively develop a set of instructions for the LLM without

20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

using additional models to reach a desired output. Following this preliminary study, we aim to conduct future studies on how these images impact user engagement in the gamified text labelling task introduced in this paper.

2. Related Work

2.1. Games-with-a-Purpose for NLP

It has been suggested that the inherent nature of the task being text-based is the reason these GWAPs are not as successful (Lafourcade et al., 2015). Visuals are an essential aspect to consider when designing games. For that reason, GWAPs in this domain have found ways to incorporate visuals into their games despite the in-game tasks being mainly text-focused. Previously, several GWAPs have acknowledged and explored using images as a proxy, retrieving them from existing sources to support annotation (Jurgens and Navigli, 2014; Vannella et al., 2014). Moreover, GWAPs for NLP frequently applied various thematic elements to improve the visuals, such as in Phrase Detectives (Chamberlain et al., 2008), WordClicker (Madge et al., 2019), and Wormingo (Kicikoglu et al., 2019). More recently, designers of GWAPs for NLP explored ways of incorporating text-labelling tasks into virtual worlds, such as LingoTowns (Madge et al., 2022), High School Superhero (Bonetti and Tonelli, 2020) and Stroll-with-a-Scroll (Aliady et al., 2022), creating a more visually appealing experience. While these are interesting approaches to enhance the visual appeal of these games, we would like to explore the use of generative AI in a GWAP for NLP. This novel approach will allow us to generate visuals directly corresponding to the text being labelled.

2.2. Co-designing with Generative AI

With the recent rise of generative AI models, many researchers have begun exploring how to use them as tools to support creativity (Liu et al., 2022). Generative AI models can generate text (e.g., GPT-3 (Brown et al., 2020), PaLM (Chowdhery et al., 2022), LLaMA (Touvron et al., 2023)), images (e.g., Stable Diffusion (Rombach et al., 2022), Midjourney[1], DALL-E (Ramesh et al., 2021), music (e.g., MusicGen (Copet et al., 2024)), video (e.g., VideoGen (Li et al., 2023)). This range of creative capabilities allows generative AI models to assist in design and content creation (Antony and Huang, 2023). Due to the extensive domain knowledge that LLMs possess, they can be suitable for tasks that require knowledge in different fields and domains. This makes using LLMs ideal for designing visuals for GWAPs for NLP, as these games use a wide range of corpora covering various topics.

2.3. Text-to-Image Generation

Text-to-image generation has evolved significantly, starting with Generative Adversarial Networks (GANs) and advancing to Conditional GANs. Within recent years, the development of diffusion models has further increased the popularity of text-to-image models, as evidenced by the widespread use of DALL-E (Ramesh et al., 2021), Midjourney and Stable Diffusion (Rombach et al., 2022). With this increase in popularity, the application of text-to-image models is expanding, particularly in the domain of game design (Deckers et al., 2023). Nevertheless, while this technology offers an accessible and cost-effective way to create game assets, producing quality text-to-image outputs can be challenging. This is due to text-to-image models' output quality greatly depending on the prompt used to generate an image. This has led many researchers (Liu and Chilton, 2022; Oppenlaender, 2023a) to investigate how to craft better prompts for text-to-image generative models.

2.4. Prompt Engineering

Following the popularity of large language models, a novel paradigm of human-AI interaction has emerged, known as prompt engineering (Brown et al., 2020). This practice has evolved into a form of art (Oppenlaender, 2022; Reynolds and McDonell, 2021), where prompt engineers creatively craft a set of instructions in order to achieve a desired output from an LLM. This practice first emerged in the field of NLP, and its usage was extended to text-to-image models. Prompt engineering follows an iterative cycle where prompts are modified and refined after every output until the desired results are reached. This process of manually generating prompts can be laborious. This is especially true when tasked with producing suitable text-to-image prompts that generate images which support the text in a text labelling task. Thus, a more efficient process of prompt engineering is required.

2.5. Prompt Optimisation

Different tools and models have been developed to assist in generating prompts to ease the process of prompt engineering. Prompt optimisation can be a partially manual or a completely automated process. Some methods opting for more of an exploratory approach to prompt optimisation include tools like Promptify (Brade et al., 2023) and Opal (Liu et al., 2022). Both tools use LLMs to guide users into producing improved text-to-image

---

[1] `https://www.midjourney.com/`

prompts. Current models that automate prompt optimisation for text-to-image prompts include BeautifulPrompt (Cao et al., 2023), Promptist (Hao et al., 2023). These models apply various automated scoring systems in their prompt optimisation models to improve prompts. These scoring systems include CLIPscore (Hessel et al., 2022), LAION-Aesthetic predictor (Schuhmann et al., 2022), and PickScore (Kirstain et al., 2023). Recently, LLM-score (Lu et al., 2023) was developed to evaluate text-to-image output using LLMs, focusing on the composition of the generated images. Another example of the use of LLMs in prompt optimisation is LLM-grounded diffusion (Lian et al., 2023). This method uses LLMs to represent objects in complex prompts, accurately enhancing the generated image. Another method to optimise prompts using LLMs is through the use of their in-context learning capabilities. This approach has been previously explored to enhance the representation of Arabic culture in generated images using LLMs' domain knowledge (Elsharif et al., 2023). Setting in-context learning instructions is a simple and easily accessible approach to optimising prompts, only requiring the design of a set of tasks for an LLM to follow. This is why we selected this approach to optimise the text-to-image prompts for our task.

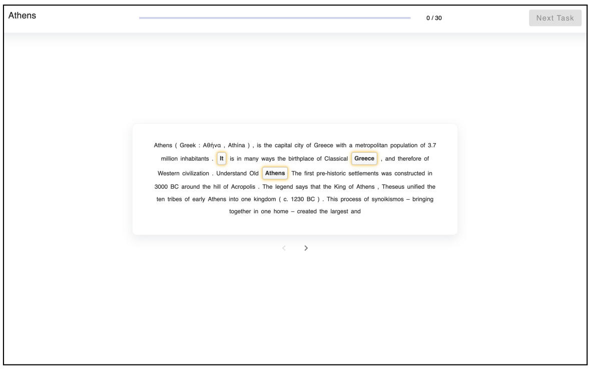
2.6. Evaluating Text-to-Image Prompts and Outputs

The images generated by text-to-image models can be evaluated across various aspects such as text-image alignment (e.g. CLIPscore (Hessel et al., 2022)), aesthetics (e.g. LAION-Aesthetic predictor (Schuhmann et al., 2022)), quality (Salimans et al., 2016) and bias (Bianchi et al., 2023). The LAION-Aesthetic predictor (Schuhmann et al., 2022) was used to evaluate aesthetics in several prompt optimisation models mentioned earlier, such as Promptist (Hao et al., 2023) and BeautifulPrompt (Cao et al., 2023). The same aesthetic measure was used to evaluate and compare a selection of text-to-image models (Lee et al., 2023). Furthermore, text-to-image prompts necessitate the use of visual language (Qiao et al., 2022), making it compelling to evaluate both the imageability and concreteness of a prompt.

3. Design

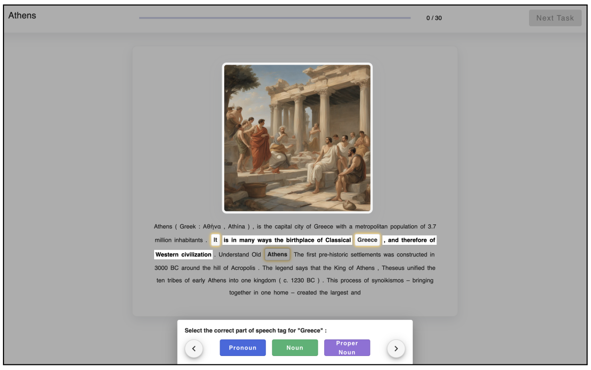
3.1. Interface of the Gamified Text Labelling Task

We developed our gamified text labelling task as an HTML5/Typescript web application using Angular, focusing on labelling nouns, proper nouns, and pronouns (Figure 1). To keep the game simple and maintain user focus on the labelling task, we implemented simple mechanics. Users initiate labelling by clicking on glowing buttons representing words that require labelling. Upon selection, the sentence containing the word is highlighted, and an AI-generated image relevant to the text appears above, accompanied by a bottom sheet displaying part-of-speech tags in different colours: blue for pronouns, green for nouns, and purple for proper nouns (Figure 2). Correctly labelled words change colour to match their part-of-speech tag, while incorrect choices turn the word grey. The interface facilitates seamless navigation to subsequent words, allowing users to label all words efficiently. Additionally, navigation buttons are present to allow users to move between words and progress to the next task. The interface also features a progress bar at the top, displaying the number of words left to label alongside the document's title, ensuring users can easily track their progress.



Athens (Greek : Αθήνα , Athína) , is the capital city of Greece with a metropolitan population of 3.7 million inhabitants . **It** is in many ways the birthplace of Classical **Greece** , and therefore of Western civilization . Understand Old **Athens** The first pre-historic settlements was constructed in 3000 BC around the hill of Acropolis . The legend says that the King of Athens , Theseus unified the ten tribes of early Athens into one kingdom (c. 1230 BC) . This process of synoikismos – bringing together in one home – created the largest and

Figure 1: The interface of the gamified text labelling task.



Athens (Greek : Αθήνα , Athína) , is the capital city of Greece with a metropolitan population of 3.7 million inhabitants . It **is in many ways the birthplace of Classical** Greece **, and therefore of Western civilization** . Understand Old Athens The first pre-historic settlements was constructed in 3000 BC around the hill of Acropolis . The legend says that the King of Athens , Theseus unified the ten tribes of early Athens into one kingdom (c. 1230 BC) . This process of synoikismos – bringing together in one home – created the largest and

Select the correct part of speech tag for "Greece" :

Figure 2: The bottom sheet is displayed after a user selects a word.

3.2. Corpus

We selected six documents from the GUM corpus (Zeldes, 2017), a sizeable open-source multilayer corpus labelled and annotated by experts. The same corpus was previously used to evaluate accuracy in a GWAP for NLP (i.e. WordClicker (Madge et al., 2019)). The selected documents included a mix of fiction and non-fiction texts, offering a diverse sample for analysis. For each document, 30 words were randomly selected, focusing exclusively on pronouns, nouns, and proper nouns. This resulted in a total of 180 words, with an even distribution of 60 words per part-of-speech category.

4. Generation Methodology

This section covers the generation methodology of both the optimised prompts and the AI-generated images.

4.1. Automating AI Image Generation

We followed a two-step process to automate the generation of AI images for our text labelling task. The first step involved optimising prompts based on the original sentences from the corpus. Once complete, the second step was to generate images from the optimised prompts.

4.1.1. Optimising Prompts using GPT-4

To automate the prompt optimisation process, we developed a Python script that iterates over each word and sentence pair in the documents mentioned earlier in Section 3.2. Utilising the GPT-4 model via the OpenAI API, we instructed the model to generate prompts based on a series of in-context learning tasks. The temperature of the model was set to 0 to ensure that the model's output remains deterministic. Before deciding on the final set of in-context learning tasks, we explored multiple sets of instructions through trial and error. This was done by qualitatively evaluating the final outputs of text-to-image that were generated using GPT-4 optimised prompts until the desired output was reached. We initially focused on improving subject coherency by ensuring that the subject was being represented accurately based on the context of the sentence in the document. Following our first set of results, we realised that the setting of that subject was also essential to include in the set of in-context tasks in order to capture the context entirely.

This exploration led us to decide on three tasks for GPT-4 to complete (see Appendix A for the complete set of instructions used). Firstly, the model was tasked with identifying the subject being referred to from the given word based on the context of the sentence in the document. Secondly, we asked the model to describe the setting of the subject identified. Finally, based on the identified subject and setting, GPT-4 generated a text-to-image prompt. The model was instructed to keep the prompt one sentence long, focusing on visual elements while avoiding overly complex language. Using this method, we ended up with a total of 180 optimised prompts. To evaluate the optimised prompt, we had two prompt types:

- Original Sentence: As a control measure, images were generated using unaltered sentences from the documents.
- Optimised Prompt: These prompts were generated using GPT-4 by utilising the in-context learning instructions mentioned above.

4.1.2. Generating Images with Stable Diffusion XL

Each 1080 x 1080 pixels image was generated on either a remote A100 or V100 GPU using the default settings of Stable Diffusion XL (SDXL) 1.0 base model [2] and then refined through the refiner[3]. The refiner uses an img2img approach to improve the image quality. Images were all generated using the same seed 1040 to remain consistent. This resulted in a total of 360 images generated based on two prompt types—optimised prompts and original sentences, with 180 images for each type. It is important to note that for the original sentence prompt type, selecting a word from the same sentence could generate duplicate images.

5. Evaluation Methodology

This section explains the evaluation metrics used to measure the concreteness and imageability scores for the prompt types and the aesthetics score used to evaluate the text-to-image outputs.

5.1. Concreteness

We calculated the average concreteness score of the two prompt types based on the sum of all words' concreteness ratings from the Brysbaert et al. (2014) database divided by the number of words in the sentence to get the average concreteness score. Words not found in the vocabulary were assigned a score of 0.

5.2. Imageability

We calculated the average imageability score of the two prompt types based on the sum of all words' imageability ratings from the MRC database (Coltheart, 1981) divided by the number of words in the sentence to get the average imageability score.

---

[2] `https://huggingface.co/stabilityai/stable-diffusion-xl-base-1.0`

[3] `https://huggingface.co/stabilityai/stable-diffusion-xl-refiner-1.0`

Words not found in the vocabulary were assigned a score of 0.

5.3. Aesthetics

To evaluate the aesthetic appeal of the text-to-image outputs, we utilise the latest version [4] of the LAION-Aesthetic predictor (Schuhmann et al., 2022) which is an automated method for evaluating the aesthetics of AI-generated images. The predictor, trained with human ratings from the Aesthetic Visual Analysis dataset (Murray et al., 2012), predicts the aesthetic scores for images on a scale from 1 to 10.

6. Results and Discussion

In this section, we explore the implications of our findings and discuss how they align with our hypotheses.

- **H1** *GPT-4 optimised text-to-image prompts will have higher imageability and concreteness scores compared to the original sentences*

Our results support our first hypothesis, as imageability and concreteness scores were significantly higher in the optimised prompts. Imageability results from a one-tailed independent sample t-test revealed that GPT-4 optimised prompts ($M = 365.90$, $SD = 36.42$) scored significantly higher in imageability compared to original sentences ($M = 348.65$, $SD = 35.0$), $t(358) = 4.576$, $p < .001$, Cohen's d $= 0.482$ (shown in Figure 3). This demonstrates that GPT-4 optimised prompts are significantly more imageable than original sentences, with a moderate effect size.

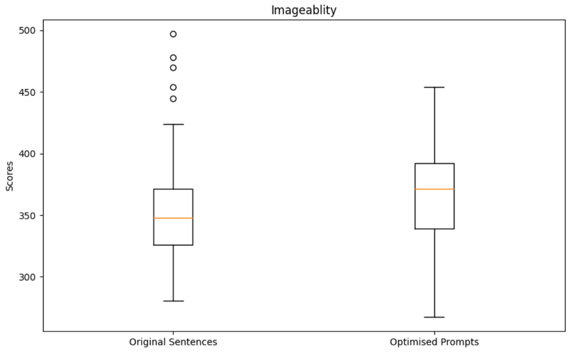


Figure 3: Comparison of imageability scores.

Concreteness results from a one-tailed independent sample t-test showed that GPT-4 optimised prompts ($M = 2.75$, $SD = 0.30$) were perceived to have higher concreteness than original sentences ($M = 2.65$, $SD = 0.31$), $t(358) = 3.129$, $p < .001$, Cohen's d $= 0.330$. This indicates a significant difference in concreteness, favouring GPT-4 optimised prompts over original sentences, with a small to moderate effect size (shown in Figure 4).

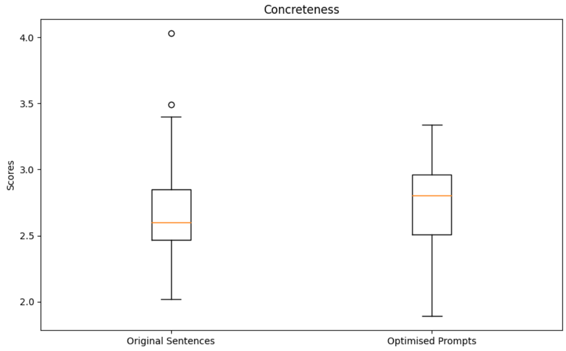


Figure 4: Comparison of concreteness scores.

- **H2** *Text-to-image outputs of the GPT-4 optimised prompts will have a higher predicted aesthetic scores than the outputs using the original sentences as prompts*

Our results support our second hypothesis, as predicted aesthetic scores were significantly higher in the optimised prompts. We evaluated our predicted aesthetics results from a one-tailed independent sample t-test indicated that GPT-4 optimised prompts ($M = 6.29$, $SD = 0.42$) were rated significantly higher in predicted aesthetics than original sentences ($M = 6.07$, $SD = 0.45$), $t(358) = 4.593$, $p < .001$, Cohen's d $= 0.484$. This supports our hypothesis, suggesting a moderate effect size and a significant difference in predicted aesthetics in favour of GPT-4 optimised prompts (shown in Figure 5).

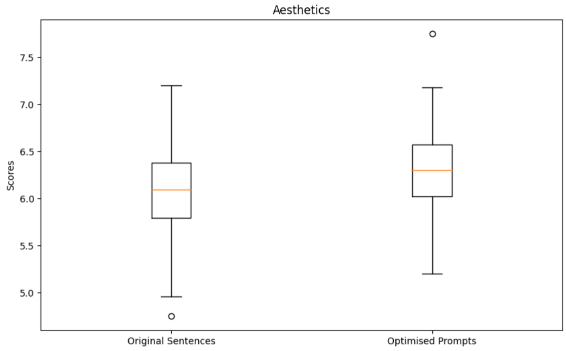


Figure 5: Comparison of predicted aesthetic scores.

- **H3** The higher the imageability and concreteness scores of a prompt, the higher the aesthetic score of the text-to-image output.

---

[4] `https://github.com/christophschuhmann/improved-aesthetic-predictor`

Our findings do not support our third hypothesis. Our examination of the relationship between concreteness, imageability and aesthetic scores via Pearson's correlation provided the following insights:

- For concreteness and aesthetics, we found a correlation of -0.127 with a p-value of 0.090, suggesting a weak, negative relationship that was not statistically significant.
- For imageability and aesthetics, we found a correlation of -0.141 with a p-value of 0.058, suggesting a weak, negative relationship that approached but did not reach statistical significance.

These results suggest a nuanced relationship between the imageability and concreteness of prompts and the predicted aesthetics of the generated images, meriting further investigation.

Furthermore, by qualitatively observing the results, we identified apparent differences between the text-to-image outputs generated from optimised prompts versus those from original sentences as prompts (see Appendix B for examples). Optimised prompts, through more descriptive language, consistently resulted in images with vivid colours and detailed compositions. In contrast, images from original sentence prompts often appeared less vibrant and more generic.

The use of GPT-4 to optimise prompts ensures the model more accurately understands the subject or term, effectively improving the ambiguous language found in some original sentences. This led to a noticeable enhancement in the relevance and accuracy of the generated images, addressing issues such as misrepresentation of subjects or settings. These qualitative observations underline that optimised prompts facilitate more accurate and coherent subject representation in text-to-image outputs compared to original sentences as prompts.

7. Limitations and Future Work

Some of the limitations of our study include focusing on only two generative AI models: GPT-4 and Stable Diffusion XL. We did not extend our investigation to other text-to-image models, like DALL-E and Midjourney. Expanding evaluation to other text-to-image models may be necessary, as each model excels in specific areas (Lee et al., 2023). Furthermore, the text-to-image outputs from our study may carry inherent biases, including social and gender biases, as mentioned by Cho et al. (2023). This highlights the need for careful consideration when selecting AI-generated images. Our analysis of text-to-image output was constrained by using a single seed for generating images, limiting our output diversity. Future studies could benefit from generating and evaluating multiple images using different seeds. Tools like PickScore (Kirstain et al., 2023) might be employed to identify the image preferred by users automatically. It is crucial to acknowledge that automated scoring models inherit biases based on their training data. This holds particularly true for subjective tasks such as aesthetics rating. Therefore, biases may be present in the LAION-Aesthetic predictor (Schuhmann et al., 2022), attributable to the subjective nature of aesthetics rating.

Our analysis primarily evaluated optimised prompts against original sentences based on their imageability, concreteness, and predicted aesthetic scores. However, we have yet to compare these AI-generated prompts with user-created ones or fully examine the impact of style modifiers on the AI-generated images, which are known to significantly improve subject coherence (Liu and Chilton, 2022; Oppenlaender, 2023b). Furthermore, to fully grasp the effectiveness of AI-generated images, conducting human evaluations is essential. Extending from this preliminary study, our future work will investigate whether AI-generated images can improve user engagement in a text labelling task.

8. Conclusion

Creating visual content for GWAPs for NLP can be time-consuming and costly, undermining the primary objective of these games. Our paper leverages GPT-4, a large language model, to streamline text-to-image prompt optimisation, introducing an automated approach for generating contextually relevant visual content for text labelling games.

9. Bibliographical References

Wateen Abdullah Aliady, Abdulrahman Aloraini, Christopher Madge, Juntao Yu, Richard Bartle, and Massimo Poesio. 2022. Coreference annotation of an arabic corpus using a virtual world game. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 388–393.

Victor Nikhil Antony and Chien-Ming Huang. 2023. Id.8: Co-creating visual stories with generative ai.

Federico Bianchi, Pratyusha Kalluri, Esin Durmus, Faisal Ladhak, Myra Cheng, Debora Nozza, Tatsunori Hashimoto, Dan Jurafsky, James Zou, and Aylin Caliskan. 2023. Easily accessible text-to-image generation amplifies demographic stereotypes at large scale. In *2023 ACM Conference* *on Fairness, Accountability, and Transparency*, FAccT '23. ACM.

Federico Bonetti and Sara Tonelli. 2020. A 3d role-playing game for abusive language annotation. In *Workshop on Games and Natural Language Processing*, pages 39–43.

Stephen Brade, Bryan Wang, Mauricio Sousa, Sageev Oore, and Tovi Grossman. 2023. Promptify: Text-to-image generation through interactive prompt exploration with large language models.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46:904–911.

Tingfeng Cao, Chengyu Wang, Bingyan Liu, Ziheng Wu, Jinhui Zhu, and Jun Huang. 2023. BeautifulPrompt: Towards automatic prompt engineering for text-to-image synthesis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1–11, Singapore. Association for Computational Linguistics.

Jon Chamberlain, Massimo Poesio, Udo Kruschwitz, et al. 2008. Phrase detectives: A web-based collaborative annotation game. In *Proceedings of the International Conference on Semantic Systems (I-Semantics' 08)*, pages 42–49.

Jaemin Cho, Abhay Zala, and Mohit Bansal. 2023. Dall-eval: Probing the reasoning skills and social biases of text-to-image generation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3043–3054.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Max Coltheart. 1981. The mrc psycholinguistic database. *The Quarterly Journal of Experimental Psychology Section A*, 33(4):497–505.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2024. Simple and controllable music generation.

Niklas Deckers, Maik Fröbe, Johannes Kiesel, Gianluca Pandolfo, Christopher Schröder, Benno Stein, and Martin Potthast. 2023. The infinite index: Information retrieval on generative text-to-image models. In *Proceedings of the 2023 Conference on Human Information Interaction and Retrieval*, CHIIR '23, page 172–186, New York, NY, USA. Association for Computing Machinery.

Wala Elsharif, James She, Preslav Nakov, and Simon Wong. 2023. Enhancing arabic content generation with prompt augmentation using integrated gpt and text-to-image models. In *Proceedings of the 2023 ACM International Conference on Interactive Media Experiences*, pages 276–288.

Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. 2023. Optimizing prompts for text-to-image generation.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2022. Clipscore: A reference-free evaluation metric for image captioning.

David Jurgens and Roberto Navigli. 2014. It's all fun and games until someone annotates: Video games with a purpose for linguistic annotation. *Transactions of the Association for Computational Linguistics*, 2:449–464.

Doruk Kicikoglu, Richard Bartle, Jon Chamberlain, and Massimo Poesio. 2019. Wormingo: a'true gamification'approach to anaphoric annotation. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, pages 1–7.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. 2023. Pick-a-pic: An open dataset of user preferences for text-to-image generation.

Mathieu Lafourcade, Alain Joubert, and Nathalie Le Brun. 2015. *Games with a Purpose (GWAPS)*. John Wiley & Sons.

Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Benita Teufel, Marco Bellagente, Minguk Kang, Taesung Park, Jure Leskovec, Jun-Yan Zhu, Li Fei-Fei, Jiajun Wu, Stefano Ermon, and Percy Liang. 2023. Holistic evaluation of text-to-image models.

Xin Li, Wenqing Chu, Ye Wu, Weihang Yuan, Fanglong Liu, Qi Zhang, Fu Li, Haocheng Feng, Errui Ding, and Jingdong Wang. 2023. Videogen: A reference-guided latent diffusion approach for high definition text-to-video generation.

Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. 2023. Llm-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *arXiv preprint arXiv:2305.13655*.

Vivian Liu and Lydia B Chilton. 2022. Design guidelines for prompt engineering text-to-image generative models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–23.

Vivian Liu, Han Qiao, and Lydia Chilton. 2022. Opal: Multimodal image generation for news illustration. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pages 1–17.

Yujie Lu, Xianjun Yang, Xiujun Li, Xin Eric Wang, and William Yang Wang. 2023. Llmscore: Unveiling the power of large language models in text-to-image synthesis evaluation.

Chris Madge, Jussi Brightmore, Doruk Kicikoglu, Fatima Althani, Richard Bartle, Jon Chamberlain, Udo Kruschwitz, and Massimo Poesio. 2022. Lingotowns: A virtual world for natural language annotation and language learning. In *Extended Abstracts of the 2022 Annual Symposium on Computer-Human Interaction in Play*, pages 57–62.

Christopher Madge, Richard Bartle, Jon Chamberlain, Udo Kruschwitz, and Massimo Poesio. 2019. Incremental game mechanics applied to text annotation. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 545–558.

Naila Murray, Luca Marchesotti, and Florent Perronnin. 2012. Ava: A large-scale database for aesthetic visual analysis. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2408–2415. IEEE.

Jonas Oppenlaender. 2022. The creativity of text-to-image generation. In *Proceedings of the 25th International Academic Mindtrek Conference*, Academic Mindtrek '22, page 192–202, New York, NY, USA. Association for Computing Machinery.

Jonas Oppenlaender. 2023a. A taxonomy of prompt modifiers for text-to-image generation. *Behaviour & Information Technology*, page 1–14.

Jonas Oppenlaender. 2023b. A taxonomy of prompt modifiers for text-to-image generation. *Behaviour & Information Technology*, pages 1–14.

Han Qiao, Vivian Liu, and Lydia Chilton. 2022. Initial images: Using image prompts to improve subject representation in multimodal ai generated art. In *Proceedings of the 14th Conference on Creativity and Cognition*, C&C '22, page 15–28, New York, NY, USA. Association for Computing Machinery.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Daniele Vannella, David Jurgens, Daniele Scarfini, Domenico Toscani, and Roberto Navigli. 2014. Validating and extending semantic knowledge bases using video games with a purpose. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1294–1304.

Amir Zeldes. 2017. The gum corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

A. Appendix A. The set of in-context learning instructions used

These are the set of in-context learning instructions used for this study. We provided the model with contextual information (document, sentence and word) and a list of tasks to complete.

In-context learning instructions for GPT-4

1. Document: {document}
2. Sentence: {sentence}
3. Word: {word}
4. Task:

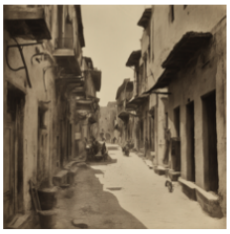
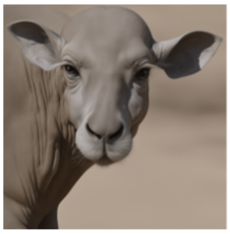
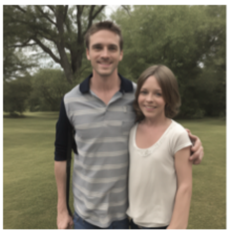
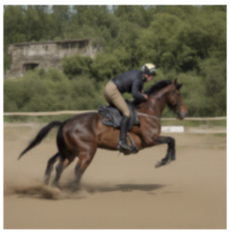
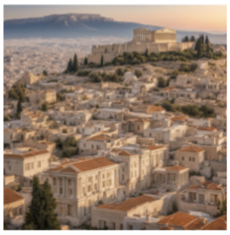
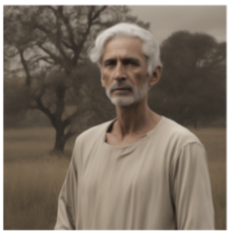
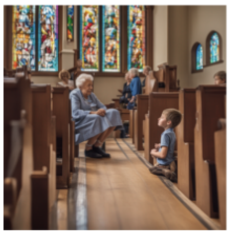
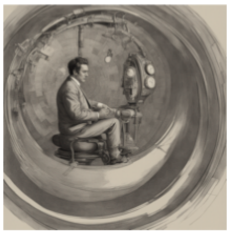
a. Identify the subject being referred to from the given word. The subject can be a character, object, or concept, based on the context of the sentence in the document.

b. Describe the setting of the given subject. This should include the physical environment as well as any relevant mood or temporal aspects.

c. Create a text-to-image prompt that best represents the identified subject and setting. The prompt should be concise yet descriptive, capturing the essence of the sentence or keyword. It should focus on visual elements while avoiding overly complex language. If the sentence is not directly visual, suggest a symbolic or metaphorical representation. This prompt should be one sentence long.

B. Appendix B. Comparison of text-to-image outputs based on prompt types

A qualitative comparison of text-to-image outputs based on prompt types shows how the model fails to present the subjects coherently if the context is not provided. For example, in the Athens document, the optimised prompt, offering more descriptive language, resulted in an image with vivid colours and detailed composition featuring a historical landmark, unlike the original sentence prompt, which produced a dull image of a generic street with old buildings. GPT-4's improvement of ambiguous language is evident in the two following examples. In the Lunre document, "he" mistakenly prompted an image of an animal instead of a man. Additionally, in The Time Machine document, the model incorrectly associated "saddle" with a horse rather than a time machine due to lack of context. Another issue observed with some original sentence prompts was the misrepresentation of subjects' settings. For instance, in the Single-Bit Error document, the text-to-image model did not accurately depict the intended church setting.

Document	**Athens**	**Lunre**	**Single-Bit Error**	**The Time Machine**
Outputs based on original sentences				
Outputs based on optimsied prompts				
Original Sentences	Old Athens	He was not yellow , but very pale brown , the colour of raw cashews ; he had silver hair , worn cropped close to the skull so that it resembled a cap .	A few times , when Tyler's parents were away , she took him with her to church .	And this time I was not seated properly in the saddle , but sideways and in an unstable fashion .
Optimised Prompts	Visualise a bustling ancient cityscape of Athens with classical Greek architecture, the Acropolis hill in the background, citizens in traditional attire, and a vibrant atmosphere of a thriving civilization.	A tall, lean man with pale brown skin and closely cropped silver hair that resembles a cap, standing awkwardly in a rural setting.	A young boy named Tyler, sitting next to his grandmother in a peaceful church, captivated by the singing and the colourful windows.	A man seated sideways on a vibrating time machine, gripping tightly as he travels through time.

Aspect-based Sentiment Evaluation of Chess Moves (ASSESS): an NLP-based Method for Evaluating Chess Strategies from Textbooks

Haifa Alrdahi and Riza Batista-Navarro
Department of Computer Science, The University of Manchester, UK
haifa.alrdahi@student.manchester.ac.uk, riza.batista@manchester.ac.uk

Abstract

The chess domain is well-suited for creating an artificial intelligence (AI) system that mimics real-world challenges, including decision-making. Throughout the years, minimal attention has been paid to investigating insights derived from unstructured chess data sources. In this study, we examine the complicated relationships between multiple referenced moves in a chess-teaching textbook, and propose a novel method designed to encapsulate chess knowledge derived from move-action phrases. This study investigates the feasibility of using a modified sentiment analysis method as a means for evaluating chess moves based on text. Our proposed Aspect-Based Sentiment Analysis (ABSA) method represents an advancement in evaluating the sentiment associated with referenced chess moves. By extracting insights from move-action phrases, our approach aims to provide a more fine-grained and contextually aware 'chess move'-based sentiment classification. Through empirical experiments and analysis, we evaluate the performance of our fine-tuned ABSA model, presenting results that confirm the efficiency of our approach in advancing aspect-based sentiment classification within the chess domain. This research contributes to the area of game-playing by machines and shows the practical applicability of leveraging NLP techniques to understand the context of strategic games.

Keywords: Natural Language Processing, Chess, Aspect-based Sentiment Analysis (ABSA), Chess Move Evaluation

1. Introduction

Over the years, chess has long served as a testbed for evaluating the performance of various algorithms (Toshniwal et al., 2022). One example is AlphaZero, which demonstrates the use of chess as a benchmark to evaluate the algorithmic performance of a self-learning algorithm (Silver et al., 2018). Most notably in recent years, artificial intelligence (AI) has been leveraged in the chess domain. For example, sentiment analysis has been used to evaluate chess moves based on commentaries (Kamlish et al., 2019). Another example is predicting the next chess move based on patterns learnt from game databases, i.e., structured data (Noever et al., 2020). These efforts highlight the benefits of applying AI to analyse moves in chess, a domain that has often been used to simulate real-world decision-making.

However, chess knowledge and strategies explained in free text has been under-explored, where limited research has investigated the ability and usefulness of evaluating chess moves expressed in unstructured data. Recently, a novel dataset, LEAP, was introduced (Alrdahi and Batista-Navarro, 2023). It was derived from chess textbooks and includes structured (chess move notations and board states) and unstructured data (textual descriptions), aimed at teaching models about chess strategies by analysing descriptions of grandmaster games using sentence-level sentiment analysis. Drawing inspiration from the LEAP dataset, this work investigates the performance of a modified aspect-based sentiment classification method on new datasets—annotated at a finer-grained level—that were created to evaluate chess moves referenced in free text. We show that embracing the semantics of the chess domain to evaluate a move (i.e., the aspect) expressed in move-action phrases is a promising approach. This finer-grained analysis provides more detailed insight into the opinions expressed about the moves, contributing to a more comprehensive understanding of sentiments within the chess context, especially in multiple-aspect scenarios. We compared the performance of the proposed approach with an Aspect-Based Sentiment Analysis (ABSA) baseline approach and show that adopting a context-rich model with move-action phrase representations improves the results of sentiment analysis models. The contributions of this study are as follows:

- Creating a dataset from a chess-teaching textbook with fine-grained annotations. These include annotations of text spans pertaining to chess moves, players and predicates (verbs). The verbs are used to describe the actions and strategies involved in playing chess moves. Importantly, the sentiment expressed towards a given move is also annotated.

- Modifying the standard definition of 'aspect' in a traditional ABSA approach by considering a player-predicate-move triple as an aspect. Adopting this definition, we designed a new ABSA method as a function for evaluating moves, which is the first attempt at exploring such an approach in the chess domain.

20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

- Training a RoBERTa model (Liu et al., 2019) on the new task and evaluating its ability to choose strategic moves by using the ABSA evaluation function. Our modified method boosts model performance results on our datasets.
- Using empirical evaluation to measure the reliability of our proposed method against Stockfish, a powerful search-based chess engine and tool for evaluating chess moves.

2. Related Work

Learning algorithms for playing chess have thus far overlooked the potential of obtaining knowledge from chess-teaching textbooks. Instead, knowledge is typically obtained from databases of chess moves, such as in the case of DeepChess (David et al., 2016). Such approaches are reliant on large, curated structured datasets capturing the knowledge of experts (Schaigorodsky, 2016), the production of which is often laborious and time-consuming and lacks explainability of the decision-making process in relation to the move. AlphaZero, the board-based reinforcement learning algorithm, achieved remarkable results in both Chess and Shogi without relying on extensive domain knowledge beyond the fundamental rules of the games (Silver et al., 2018). However, this type of algorithm lacks human intuition or understanding of the game beyond what it learns through self-play. This can sometimes lead to unconventional strategies or a lack of understanding of traditionally accepted strategies in these games. Additionally, the resources required for the kind of intensive training that AlphaZero undergoes are computationally demanding and expensive, which limits its accessibility (Nechepurenko et al., 2020).

Recent advances in natural language processing (NLP) such as the development of contextual embeddings and transformer architectures have boosted the performance of NLP-based models in many domains and tasks (Vaswani et al., 2017; Devlin et al., 2019). This has provided opportunities to explore approaches that deviate from traditional ones that rely on chess engines, which require extensive game state analysis to evaluate moves. For instance, a large language model (LLM) was trained on 10 million games annotated with action-value pairs from the Stockfish engine (Ruoss et al., 2024). The effectiveness of the LLM was evaluated on the basis of its ability to select the move with the highest value for any given position. The authors claimed that this model has the potential of achieving grandmaster-level chess-playing, without relying on explicit search algorithms, by predicting action-values directly from board states. However, this approach still lacks explainability as it does not require the LLM to provide any context. Instead, the model was aimed at mimicking the game-playing capabilities of Stockfish, which relies on search algorithms.

Nevertheless, various studies have shown that using context expressed in natural language as an alternative approach to overcome the above-mentioned limitations has improved the performance of AI systems. Previous work focussed mainly on extracting actions from sentences with short, direct instructions using a model with long short-term memory recurrent neural networks (LSTM-RNNs) (Mei et al., 2016). However, such an approach does not have the ability to evaluate the outcome of the action, as it is applied to direct instructions only.

Traditional sentiment analysis has been one of the few first attempts at applying NLP techniques to evaluate chess moves. SentiMate (Kamlish et al., 2019) presented an approach to chess move evaluation using NLP, where the model employs classifiers to determine move quality from commentary datasets, and a convolutional neural network (CNN)-based sentiment analysis model trained on chess commentaries. Although this method suggests the potential of NLP in improving decision-making processes as part of game strategies, it, however, offers limited insights when applied to extensive texts such those in chess textbooks. Chess commentaries typically focus on specific moves in given board states, while textbooks cover a range of moves. Standard sentence-level sentiment analysis methods are limited in that they capture sentiments towards individual moves only, without considering possible subsequent moves, thus limiting their capability as an evaluation function.

Recent research has shifted to fine-tuning generative LLMs on extensive chess game data for move recommendation, such as the study by Noever et al. (2020). Their work demonstrates that GPT-2 can learn complex game play, where the model successfully learnt to generate plausible chess moves and strategies without direct intervention or heuristic guidance. Nevertheless, while the proposed approach achieved over 90% accuracy in generating legal chess moves and reproducing classic chess openings, it neglects detailed knowledge of the textual content and lacks explanatory depth for specific move choices. Furthermore, the use of the GPT-2 model as an alternative method for search-based algorithms has not yet answered the question of whether the model genuinely comprehends the game or merely replicates move patterns effectively. The more recent study in by DeLeo and Guven (2022) offers a fresh perspective on the use of language models to understand and learn complex strategic games such as chess. The study focussed on the ability of the BERT model to interpret chess positions of the board states and moves encoded in the Forsyth–Edwards Notation (FEN) format, analysing its effectiveness in playing against advanced chess engines like Stockfish. The BERT model exhibited its capability to maintain game-play against Stockfish for an extended number of moves and demonstrated substantial move accuracy, especially in the game's opening stages. However, the same limitation observed in other previous work remains, in that the model's proficiency in understanding deeper strategic elements of chess that go beyond the generation of valid moves, remains uncertain.

3. ABSA for Chess Move Evaluation

This section describes the motivation for using an Aspect-based Sentiment Analysis (ABSA) approach in evaluating chess moves described in free text. ABSA is considered to be a fundamental NLP task due to it being able to deliver more targeted sentiment insights, critical for a range of applications, including market analysis and social media monitoring (Pontiki et al., 2016; Zhang et al., 2022). Recognised as a means for Information Extraction (IE) applied to various decision-making scenarios (Pontiki et al., 2014; Phan et al., 2021; Sun et al., 2020; Reynard and Shirgaokar, 2019), this approach lends itself well to the analysis of strategic chess moves in a teaching context, particularly to determine the suitability of a move in the context of a given board state. Here, ABSA evaluates the moves, labelling them as 'go-to', 'avoid', or 'neutral'.

Meanwhile, the LEAP corpus highlighted the challenge of handling multiple aspects within a single sentence (Alrdahi and Batista-Navarro, 2023). Investigating the description of moves within this corpus reveals a strong similarity in sentence structures, but understanding the different meanings requires careful interpretation. We argue that, for this purpose, knowledge infusion (i.e., the incorporation of structured knowledge) can help models achieve a deeper understanding of the domain, leading to more accurate and context-aware predictions (Sheth et al., 2019). Hence, we consider a chess move as a distinct aspect, whereby we integrate the verb (predicate) indicating whether the move is 'go-to' or 'avoid'. Within the context of chess, each move is associated with a player performing the action. Therefore, we also incorporated the player into the aspect, resulting in a player-predicate-move triple. We hypothesise that this aspect, which we refer to as the 'move-action phrase', can assist a classification model in differentiating between different sentiments associated with multiple aspects in the same sentence.

To validate the above hypothesis, we performed an ablation study that compared models, both with and without this novel aspect formulation. Additionally, every move is played based on a specific type of aim or action, such as attacking or protecting a piece. These actions are expressed in different predicates by grandmaster players, where different predicates could mean the same type of action. We design unsupervised clusters of action types and infused the original sentences with the action type as additional semantic information. By adding this contextual information, we are enriching the model's input with external structured knowledge (the action type), which is not inherently part of the original sentence data. In this paper, we are studying the effect of knowledge infusion that in theory should enhance a model's understanding by providing additional context.

4. Dataset Description

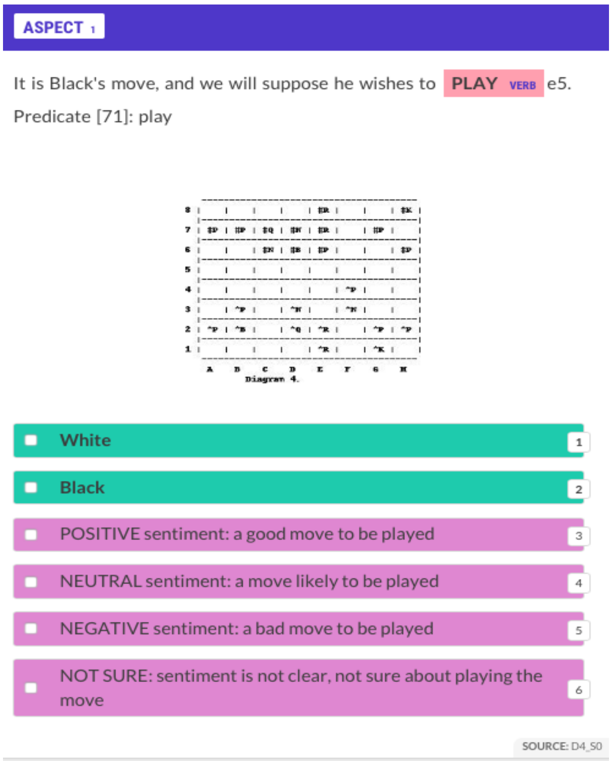
4.1. Annotation Process

Given that the chess domain has well-defined terminologies to express moves, we developed a rule-based named entity recognition (NER) method based on regular expressions (regexes) to extract mentions of pieces, players, moves and move sequences, where the lattermost are expressed in the Standard Algebraic Notation (SAN) chess notation format. Upon manual review, we noticed that each selected move represents a distinct type of action, aimed at achieving a particular purpose or strategy, which is expressed in natural language through the use of verbs. We then crafted an annotation scheme at the move-action phrase level (where a move can act as either the subject or object of a verb) for aspect-based sentiment classification. We took sentences in the LEAP corpus that were determined as topic-relevant, i.e., pertaining to strategic moves (Alrdahi and Batista-Navarro, 2023), and applied the WordNet English lexicon (Miller, 1994) to identify verbs within sentences. The annotation of all move-action phrases was conducted by duplicating each sentence according to the number of verbs identified within it.

As suggested in the literature, access to the state of the board during the evaluation of moves improves understanding of the context, leading to a more precise evaluation of the outcome of a move in the context of a given board state (Alrdahi and Batista-Navarro, 2023; Beinborn et al., 2018; Pezzelle et al., 2020). Therefore, in annotating our data, we used Prodigy,[1] which allows for annotation based on multi-modal information. Each example in our dataset is presented to annotators in an interface that is divided into four sections, as shown in Figure 1. These sections are necessary

---

[1] `https://prodi.gy/`



It is Black's move, and we will suppose he wishes to **PLAY** VERB e5.

Predicate [71]: play

☐ White 1

☐ Black 2

☐ POSITIVE sentiment: a good move to be played 3

☐ NEUTRAL sentiment: a move likely to be played 4

☐ NEGATIVE sentiment: a bad move to be played 5

☐ NOT SURE: sentiment is not clear, not sure about playing the move 6

SOURCE: D4_S0

Figure 1: The annotation interface as shown in Prodigy.

in understanding the context of a chess game and ensuring accurate annotation:

1. An image depicting the chess board's state, aiding in visualising the context for the move under consideration and facilitating visualisation of state changes based on sentences describing multiple moves.
2. A text description of potential moves, with the relevant predicate emphasised in bold and tagged as 'VERB' to direct the annotator's attention to the move's significance and its sentiment. For example, in the sentence *"It is Black's move, and we will suppose he wishes to play e5"*, the predicate *"play"* is highlighted.
3. A list of options corresponding to player names, namely, "White" and "Black", which are for indicating the player making the move.
4. Sentiment labels for indicating the sentiment toward the move:

   'Positive' for advantageous moves.

   'Negative' for disadvantageous moves.

   'Neutral' for moves with neither a positive nor a negative effect.

   'Not sure' for when the sentiment towards a move is unclear.

The sentence shown in Figure 1, when annotated, has Black as the player and 'Neutral' as the sentiment for the aspect *Black-play-e5*. Meanwhile, multiple aspects can be found in the example sentence *"Before bringing the discussion of the Queen's Pawn opening to a close, I may remark that in tournaments it has become usual for White not to play c4 at once, but to play Nf3 as a preliminary, in order to avoid the complications of the Queen's counter gambit."* In this case, the aspect *White-play-c4*, is labelled as 'Negative', and the second aspect *White-play-Nf3* is labelled as 'Positive'.

To closely replicate the chess environment, annotators were instructed to: (1) select the player making a move, (2) identify the move, and (3) determine the sentiment towards the player-predicate-move triple. The dataset, consisting of 726 sentences, was annotated by the lead author of this paper and one additional annotator with expertise in NLP and sentiment analysis, and familiarity with chess terminology. To enable measurement of inter-annotator agreement (IAA), 20% of the total number of sentences was set aside as a common subset that was annotated by both annotators, albeit independently. Annotator agreement was measured using Cohen's Kappa metric and was determined to be 65% (substantial agreement). Each annotator was then tasked with annotating a further (non-overlapping) subset with 10% of the total number of sentences, which are unique to that annotator. The annotations resulted in 437 'Positive', 153 'Negative' and 133 'Neutral' labels, with only three instances of uncertainty that were subsequently removed. The dataset was divided into training (70%), validation set (10%) and testing (20%) sets.

4.2. Data Augmentation

Confronted with the known challenge of limited resources for manually creating training data (Zhang et al., 2022) and the need to address the LEAP dataset's imbalance, we employed an oversampling technique from the nlpaug library.[2] This approach is based on back-translation, leveraging two translation models to convert sentences from English to German, and then back to English. This method effectively generated additional synthetic data for model training, to enhance the diversity of the dataset without manually curating content (Wei and Zou, 2019; Feng et al., 2021). The augmented sentences retained the original sentence meaning and sentiment, albeit with minor contextual variations. Unlike prompt-based generation, which might produce random outcomes or undesirable sentence alterations, this method offered controlled sentence generation. We applied this technique to achieve a balanced distribution of labels in the training set as shown in Table 1 and manually checked the quality and correctness of the generated sentences.

Label	Original	Over-sampled
Positive	288	288
Negative	117	234
Neutral	100	200

Table 1: Distribution of training set labels.

5. Transformer Models for ABSA

In this section, we describe the various transformer-based language models that we built upon in order to construct ABSA classifiers.

5.1. Generic ABSA Models

Firstly, we evaluated the vanilla RoBERTa-base (VRB) model on the aspect-based sentiment classification task, using two existing general-domain corpora: the Restaurant and Laptop dataset from SemEval 2014 (Pontiki et al., 2014), and the MAMS dataset (Jiang et al., 2019) which bears similarities with LEAP in that a single sentence could bear multiple sentiments. To fine-tune VRB on the Restaurant and Laptop datasets, we used baseline parameters (seed of 42, batch size of 4, a learning rate of 3e-05, and no weight decay). Additionally, VRB performance on the MAMS dataset was evaluated under two conditions: using the baseline settings above and following the hyperparameters (batch size of 8) of CapsNet-BERT, a state-of-the-art ABSA model (Jiang et al., 2019). Performance was measured by taking the mean of the micro-averaged F1-scores across five runs, selecting the best epoch score for each run. VRB demonstrated competitive micro-averaged F1-scores against leading models on SemEval 2014 Restaurant and Laptop datasets (85.68% and 80.05%, respectively), and achieved comparable results on the MAMS dataset (84.29%). VRB demonstrated strong baseline performance without hyperparameter optimisation, additional features or additional training data. These scores encouraged us to adopt RoBERTa as our primary architecture for move evaluation experiments.

5.2. Fine-tuned ABSA Models

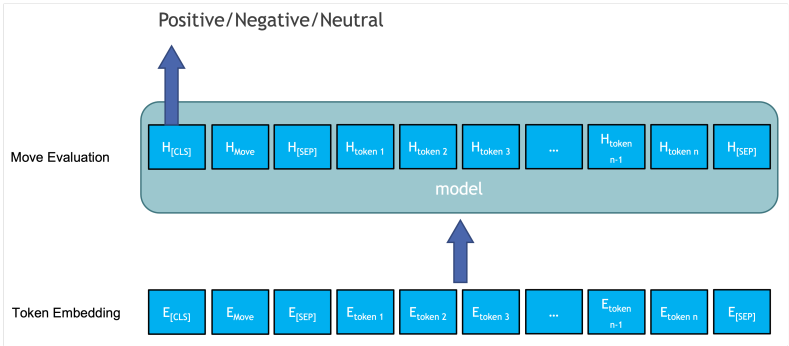
Based on similar work and recommendations reported in the literature (Xu et al., 2019; Karimi et al., 2021; Rietzler et al., 2020), fine-tuning a language model on domain-specific data significantly improves the model's understanding of relevant domain knowledge. This specialised training phase adjusts the model's understanding to learn the context and terminology within a specific domain, hence facilitating more precise and informed predictions or analyses. We fine-tuned the VRB model (henceforth referred to as the FT-RB model) using synthesised chess sentences in the LEAP dataset (Alrdahi and Batista-Navarro, 2023) and chess commentaries collected by Jhamtani et al. (2018).

We approached the task of move evaluation as a sequence classification problem, representing the dataset as $D = \{(X_i, Y_i)\}_{i=1}^{|D|}$, where X_i is an input sentence and Y_i is the corresponding true label for the ith instance. In our sequence classification task, both the sentence and a specific aspect are treated as part of the input X. The sentence X is fed into the model encoder $Enc(X)$ to derive contextual features. The sentence's final hidden state is encapsulated by the special token $CLS(X)$ through a dense layer with a softmax function for predicting Y.

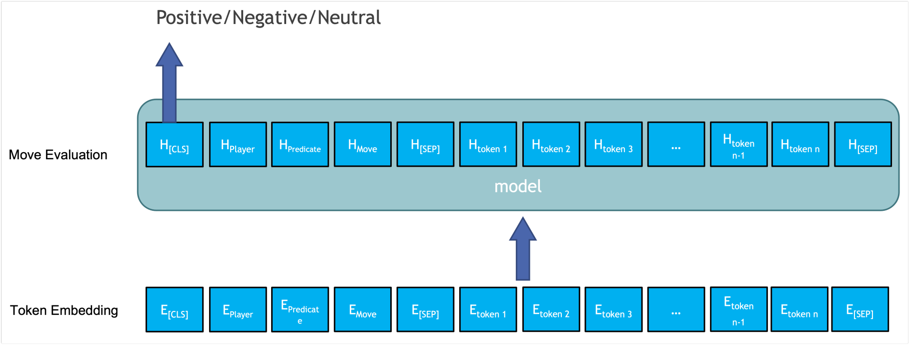
Three types of sentence embeddings were evaluated as representations of an infused input sentence, illustrated in Figure 2. In Figure (a), the embedding X represents 'move only' as the aspect. In Figure (b), X represents the 'move-action' phrase as the aspect. Lastly, in Figure (c), X corresponds to the embedding of the 'move-action' phrase as the aspect, but enriches the sentence with additional information on the move-action type. All additional inputs in the sentence were separated with the special token `[SEP]`. We defined five types of move-actions: Attack, Capture, Defend, Protect,

---

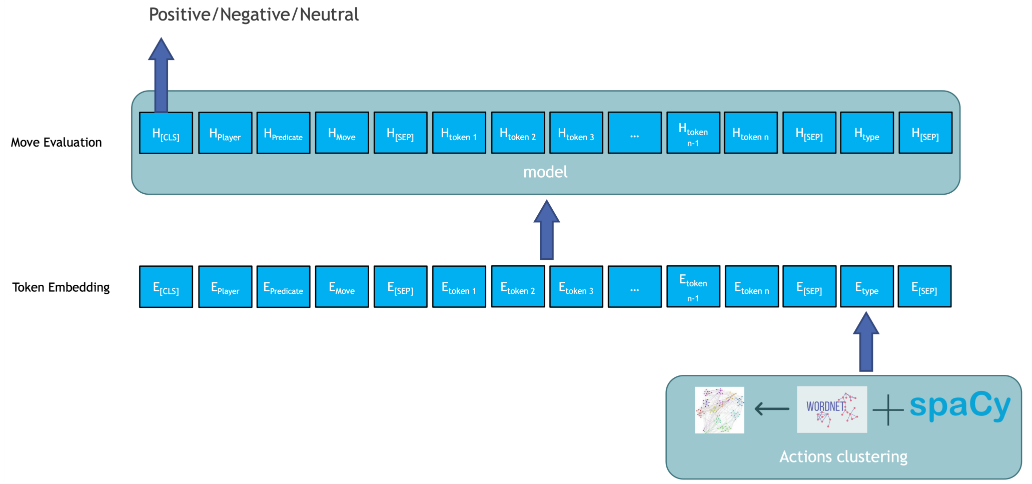
[2] `https://github.com/makcedward/nlpaug/tree/master`



(a) ‘move only’ aspect



(b) ‘move-action’ aspect



(c) ‘move-action’ aspect with semantic information

Figure 2: Different types of knowledge infusion for aspect-based sentiment classification models.

and General Move; we refer the reader to Table 2 for their definitions and corresponding examples. To automatically classify move-actions according to these types, we employed an unsupervised learning approach that leveraged semantic relationships from WordNet, including synonyms and definitions to group the verbs. The Chinese Whispers graph clustering algorithm (Biemann, 2006) was applied with the aid of Gephi (Bastian et al., 2009) to group verbs based on cosine similarity, using a seed value of 50 and running the clustering for 50 iterations. Through experimentation, we established a minimum similarity threshold of 40 for clustering data points.

6. Evaluation

Our experiments were run five times, and the average F1-score was reported. In our ablation study, we tested different types of sentence embeddings by either removing or including the move-action type. We also examined how swapping the two

Move type	Definition	Example
Attack	Playing a move to attack an opponent piece	*White can attack Bishop with his Rook*
Capture	Playing a move to capture an opponent piece	*White captures Bishop with his Rook*
Defend	Playing a move to defend a piece under an attack.	*Black defends his Bishop by pushing it to d7*
Protect	Playing a move to protect a piece from future attack.	*Black can protect his Bishop by pushing pawn to c6*
Move	A general move, (placing a piece from one position into another) without explicit intention of a purpose.	*White plays 5.b3 before castling*

Table 2: Definitions of chess move-action types with examples.

types of aspects, i.e., 'move only' and our modified 'move-action' phrase, affects the results. We used default hyperparameter values: seed = 42, evaluation and training batch size per device = 4, learning rate = 3e-05 and weight decay = 0.0.

6.1. ABSA Results

To evaluate the impact of our aspect formulation and the effects of knowledge infusion, we compare the vanilla RoBERTa-base (VRB) and fine-tuned RoBERTa-base (FT-RB) models on the task of aspect-based sentiment analysis as a means for chess move evaluation. We report the results on the original dataset in Table 3 and on the oversampled dataset in Table 4.

The F1-scores achieved by the two models indicate that despite the inherent complexity of the task, the FT-RB model demonstrated an improvement in its F1-score. This improvement can be attributed to the modification of the model's weights, facilitated by the incorporation of domain-specific knowledge data. Furthermore, the modification of aspect definition we designed contributed towards enhancing the performance of the FT-RB model. This adjustment enabled the model to comprehend the context more effectively and concentrate on the 'move' that is being evaluated, even where multiple aspects are being discussed in the sentence. However, despite the higher F1-scores observed when using the original dataset, it is worth noting that the model failed to accurately identify the minority classes, specifically the 'Negative' and 'Neutral' class labels, and the relatively high score is due to the larger number of examples labelled with the 'Positive' class label. Using oversampled data, we observed an improvement in the model's capability to grasp and categorise these minority classes, gaining an increase of 20% to 30% in terms of F1-score for the 'Negative' and 'Neutral' classes. The results obtained by the FT-RB model trained on the oversampled data demonstrate the potential of the model to learn minority classes, hence we are considering it as our preliminary proposed model for this challenging task.

Interestingly, adding the type of move-action to the input slightly decreased the performance of both models. This decrease can be attributed to the added complexity of the information in the input, which the models found difficult to interpret. Even though, intuitively, the move-action type adds further knowledge to allow humans to interpret the reason for playing the move, it did not aid the aspect-based sentiment classification models.

6.2. ABSA vs. Stockfish Analysis

Deviating from the usual game-level evaluation of chess agents, we assessed the extent to which our proposed ABSA model can evaluate a move based on a given board state. This is because our corpus does not discuss a complete game from its start to end, but focusses only on specific strategic moves. A board state is represented in the Forsyth-Edwards Notation (FEN) format, which can capture the placement of pieces on the board, turn-to-move, castling availability, and other basic chess rules. We extracted the FEN board and the chess moves discussed in the text, and integrated them into the Stockfish 16 engine to obtain the probability of the move leading to a win, a loss or a draw. Unlike other games where the outcome at the end is only a win or a loss, chess recognises the draw as a third possible outcome. Draws are common in high-level chess matches, where players often have similar strengths and capabilities, and neither side has achieved an advantage to claim a win.

In Stockfish, the engine skill was set to 8, the Elo rating to 2400 (grandmaster level) and the search depth to 10. These settings were selected to establish a baseline of how much we can rely on the evaluation described in the text, as if the text-based sentiment evaluation is equivalent to depth search. In cases where the aspect is a sequence of moves, we take Stockfish engine's evaluation of the first move in the sequence. We excluded counterfactual statements, moves that were incorrect as a result of issues encountered during optical character recognition (OCR) when the textbook was digitised, and

Embeddings	VRB	FT-RB
move only	54%	55%
move-action phrase	55%	62%
move-action phrase with type	55%	59%

Table 3: Averaged F1-scores obtained on the original dataset, using different types of input sentence representations.

Embeddings	VRB	FT-RB
move only	41%	50%
move-action phrase	50%	55%

Table 4: Averaged F1-scores obtained on the oversampled dataset, using different types of input sentence representations.

implicit moves. An example of an implicit move is the triple *Black-play-King away from his file* from the sentence *“White has no time to double Rooks, because if he does so, after his Re2 Black would play the King away from his file and allow the Knight to escape”.*

The heatmap in Figure 3 presents the number of times each sentiment label has the highest score in one of the categories: ‘Win’, ‘Draw’, ‘Lose’. The ‘Positive’ and ‘Negative labels have a significantly higher association with the ‘Draw’ category than with the ‘Win’ or ‘Lose’ categories, respectively. The relatively low count for the ‘Lose’ category being associated with the ‘Positive’ and ‘Neutral’ labels indicates that the sentiment labels are relatively aligning with the outcome of the search-based algorithm, even though the ABSA model did not have access to the board state during sentiment classification.

It can be seen that a substantial number of moves labelled as ‘Neutral’ are correlated with the ‘Draw’ category, suggesting that these moves maintain the balance of the game where neither side has an advantage. The distribution of the ‘Negative’ label is balanced across all Stockfish outcomes, with a slight preference for ‘Draw’ and ‘Lose’. This implies that negative sentiments are indicative of more challenging or risky positions, which might lead to either a loss or a stabilising effort towards a draw. Finally, a higher number of positive sentiment labels are associated with winning outcomes and are less frequently associated with losses. Overall, the ability to correlate sentiment labels with the game outcomes suggests that the sentiment expressed in the description of chess moves can be predictive of the move’s effectiveness without requiring deep analysis by a chess engine. This analysis supports our hypothesis that ABSA can be an evaluation function for chess moves described in text, offering a novel approach to understanding and predicting the implications of chess strategies through aspect-based sentiment classification.

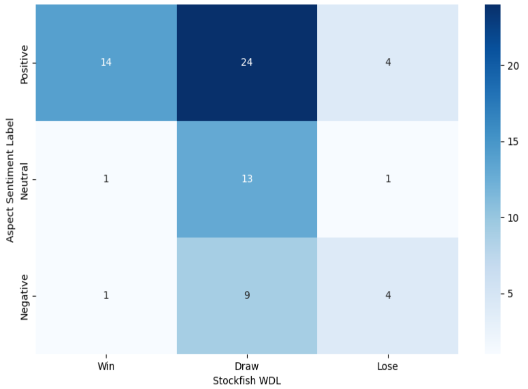


Figure 3: Heatmap based on a sample (10%) to visualise the correlation between the sentiment labels predicted by our baseline ABSA model for a move (‘Positive’, ‘Neutral’, ‘Negative’), and the probable outcome (‘Win’, ‘Draw’, ‘Lose’) of the move provided by the Stockfish engine.

6.3. Error Analysis

We identified multiple factors that might have affected the performance of the ABSA model and might have led to disagreements between its predictions and the outcomes provided by Stockfish. Firstly, some errors were made by our automatic rule-based move extraction method, leading to the linking of predicates to incorrect moves, which in turn, resulted in wrongly predicted sentiment labels. A key limitation of our current work is that aspect extraction is not learnt jointly with the sentiment classification task, whereas end-to-end systems (i.e., models trained simultaneously on aspect extraction and sentiment classification) have reportedly obtained better performance.

An additional limitation is the model’s underperformance in capturing the strategic depth of chess moves not explicitly described in SAN nor natural language, such as implicit threats or long-term strategies not explicitly mentioned in the text. An example sentence that contains such implicit information is: *"In Diagram 13, White derives no advantage from being the exchange to the good, for the Rook has no file which could be used to break into the Black camp."* Even with access to the board state diagram and despite the sentiment being clearly negative, it is difficult to determine what *"exchange"* refers to. The model could be improved to become a powerful chess agent by using a reinforcement learning approach to take advantage of feedback, which we leave for future work. The feedback could involve integrating a chess engine's evaluations alongside features from natural language to provide a more comprehensive understanding of the positions. This would enable the model to consider both explicit descriptions and the underlying strategic implications of moves, offering a fuller analysis of chess strategies from textual descriptions.

7. Conclusion

This study is part of ongoing research that investigates approaches to evaluating chess moves described in textbooks. We have demonstrated, based on a small-scale dataset, the potential of using text-based resources to evaluate strategic chess moves. We introduced a novel method for evaluating chess strategies using NLP, specifically focussing on aspect-based sentiment classification of chess moves described in textbooks. This involves creating a new annotated dataset drawn from the chess literature, modifying the definition of aspect (in ABSA) to include both player and move-action phrases, and training a RoBERTa-base sentiment classification model for strategic move evaluation.

Our research presents baseline results for this new task, which demonstrate the potential of NLP to improve understanding and analysis of chess strategies. We believe that if enough text-based datasets are available, the model can potentially evaluate moves and engage in game-play.

Finally, many studies have shown that grounding natural language with the environment results in more accurate decision-making (Kameko et al., 2015; Matuszek, 2018; Alomari et al., 2017; Karamcheti et al., 2017; Branavan et al., 2012; Luketina et al., 2019; He et al., 2016). Hence, aligning the sentiment analysis with the environment—in this case, the chess board state—could facilitate more effective decision-making. As part of future work, our aim is to further explore the incorporation of the board state into the input for aspect-based sentiment classification. In this scenario, a move will be evaluated not only on the basis of the text but also while considering the current board state and the additional semantic knowledge represented by the action type (Zhang et al., 2022). In addition, an ABSA model could enhance the game-playing experience by offering an explanation of the choice of moves through search-based chess agents, which we aim to explore next.

8. Bibliographical References

Muhannad Alomari, Paul Duckworth, Majd Hawasly, David C Hogg, and Anthony G Cohn. 2017. Natural language grounding and grammar induction for robotic manipulation commands. In *Proceedings of the 1st Workshop on Language Grounding for Robotics*, pages 35–43.

Haifa Alrdahi and Riza Batista-Navarro. 2023. Learning to Play Chess from Textbooks (LEAP): a Corpus for Evaluating Chess Moves based on Sentiment Analysis. *arXiv preprint arXiv:2310.20260*.

Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. 2009. Gephi: An Open Source Software for Exploring and Manipulating Networks. In *3rd international AAAI conference on weblogs and social media*.

Lisa Beinborn, Teresa Botschen, and Iryna Gurevych. 2018. Multimodal Grounding for Language Processing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2325–2339, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Chris Biemann. 2006. Chinese whispers-an efficient graph clustering algorithm and its application to natural language processing problems. In *Proceedings of TextGraphs: the 1st Workshop on Graph Based Methods for Natural Language Processing*, pages 73–80.

SRK Branavan, David Silver, and Regina Barzilay. 2012. Learning to win by reading manuals in a monte-carlo framework. *Journal of Artificial Intelligence Research*, 43:661–704.

Omid E David, Nathan S Netanyahu, and Lior Wolf. 2016. DeepChess: End-to-end deep neural network for automatic learning in chess. In *International Conference on Artificial Neural Networks*, pages 88–96. Springer.

Michael DeLeo and Erhan Guven. 2022. Learning Chess With Language Models and Transformers. *arXiv preprint arXiv:2209.11902*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A Survey of Data Augmentation Approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988.

Xidong Feng, Yicheng Luo, Ziyan Wang, Hongrui Tang, Mengyue Yang, Kun Shao, David Mguni, Yali Du, and Jun Wang. 2024. ChessGPT: Bridging policy learning and language modeling. *Advances in Neural Information Processing Systems*, 36.

Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Lihong Li, Li Deng, and Mari Ostendorf. 2016. Deep Reinforcement Learning with a Natural Language Action Space. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1621–1630.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, Graham Neubig, and Taylor Berg-Kirkpatrick. 2018. Learning to Generate Move-by-Move Commentary for Chess Games from Large-Scale Social Forum Data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1661–1671, Melbourne, Australia. Association for Computational Linguistics.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A Challenge Dataset and Effective Models for Aspect-Based Sentiment Analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285, Hong Kong, China. Association for Computational Linguistics.

Hirotaka Kameko, Shinsuke Mori, and Yoshimasa Tsuruoka. 2015. Learning a game commentary generator with grounded move expressions. In *2015 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 177–184. IEEE.

Isaac Kamlish, Isaac Bentata Chocron, and Nicholas McCarthy. 2019. SentiMATE: Learning to play Chess through Natural Language Processing. *arXiv preprint arXiv:1907.08321*.

Siddharth Karamcheti, Edward Clem Williams, Dilip Arumugam, Mina Rhee, Nakul Gopalan, Lawson LS Wong, and Stefanie Tellex. 2017. A Tale of Two DRAGGNs: A Hybrid Approach for Interpreting Action-Oriented and Goal-Oriented Instructions. In *Proceedings of the 1st Workshop on Language Grounding for Robotics*, pages 67–75.

Akbar Karimi, Leonardo Rossi, and Andrea Prati. 2021. Adversarial Training for Aspect-Based Sentiment Analysis with BERT. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 8797–8803.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

J Luketina, N Nardelli, G Farquhar, J Foerster, J Andreas, E Grefenstette, S Whiteson, and T Rocktäschel. 2019. A Survey of Reinforcement Learning Informed by Natural Language. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019, August 10-16 2019, Macao, China.*, volume 57, pages 6309–6317. AAAI Press (Association for the Advancement of Artificial Intelligence).

Edward Ma. 2019. NLP Augmentation. https://github.com/makcedward/nlpaug.

Cynthia Matuszek. 2018. Grounded language learning: where robotics and NLP meet. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 5687–5691.

Hongyuan Mei, Mohit Bansal, and Matthew R Walter. 2016. Listen, attend, and walk: Neural mapping of navigational instructions to action sequences. In *30th AAAI Conference on Artificial Intelligence*.

George A. Miller. 1994. WordNet: A lexical database for English. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Liudmyla Nechepurenko, Viktor Voss, and Vyacheslav Gritsenko. 2020. Comparing knowledge-based reinforcement learning to neural networks in a strategy game. In *Hybrid Artificial Intelligent Systems: 15th International Conference, HAIS 2020, Gijón, Spain, November 11-13, 2020, Proceedings 15*, pages 312–328. Springer.

David Noever, Matt Ciolino, and Josh Kalin. 2020. The chess transformer: Mastering play using generative language models. *arXiv preprint arXiv:2008.04057*.

Sandro Pezzelle, Claudio Greco, Greta Gandolfi, Eleonora Gualdoni, and Raffaella Bernardi. 2020. Be Different to Be Better! A Benchmark to Leverage the Complementarity of Language and Vision. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2751–2767, Online. Association for Computational Linguistics.

Huyen Trang Phan, Ngoc Thanh Nguyen, Van Cuong Tran, and Dosam Hwang. 2021. An approach for a decision-making support system based on measuring the user satisfaction level on Twitter. *Information Sciences*, 561:243–273.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, et al. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Darcy Reynard and Manish Shirgaokar. 2019. Harnessing the power of machine learning: Can Twitter data be useful in guiding resource allocation decisions during a natural disaster? *Transportation research part D: Transport and environment*, 77:449–463.

Alexander Rietzler, Sebastian Stabinger, Paul Opitz, and Stefan Engl. 2020. Adapt or Get Left Behind: Domain Adaptation through BERT Language Model Finetuning for Aspect-Target Sentiment Classification. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4933–4941, Marseille, France. European Language Resources Association.

Anian Ruoss, Grégoire Delétang, Sourabh Medapati, Jordi Grau-Moya, Li Kevin Wenliang, Elliot Catt, John Reid, and Tim Genewein. 2024. Grandmaster-Level Chess Without Search. *arXiv preprint arXiv:2402.04494*.

Ana Schaigorodsky. 2016. Chess Database.

Amit Sheth, Manas Gaur, Ugur Kursuncu, and Ruwan Wickramarachchi. 2019. Shades of knowledge-infused learning for enhancing deep learning. *IEEE Internet Computing*, 23(6):54–63.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2017. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144.

Wenjuan Sun, Paolo Bocchini, and Brian D Davison. 2020. Applications of artificial intelligence for disaster management. *Natural Hazards*, 103(3):2631–2689.

Shubham Toshniwal, Sam Wiseman, Karen Livescu, and Kevin Gimpel. 2022. Chess as a testbed for language model state tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11385–11393.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jason Wei and Kai Zou. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*.

Generating Converging Narratives for Games with Large Language Models

Douglas Summers-Stay, Clare R. Voss
U.S. Army Research Laboratory
douglas.a.summers-stay.civ@army.mil

Abstract

Human authors make numerous choices in crafting narratives. In interactive stories like the *Choose Your Own Adventure series*, authors must decide when and how readers will influence the plot. Authoring tools that leverage large language models (LLMs) to assist authors currently can generate multiple diverse story paths, but provide no way for rejoining these. Here we explore extending the use of LLMs for bringing separate story lines back together. We test various methods of combining the next-token probability distributions of two distinct story lines into a single distribution, and present samples of the resulting texts. Our working hypothesis was that the LLM would seek to "unify" content whenever possible. We found that the probability consolidation functions mattered less than one might expect and that the method was capable of rejoining narratives in a natural way for a wide variety of differences between the two incoming texts.

Keywords: interactive fiction, token probabilities, branching narratives, LLM-assisted authoring

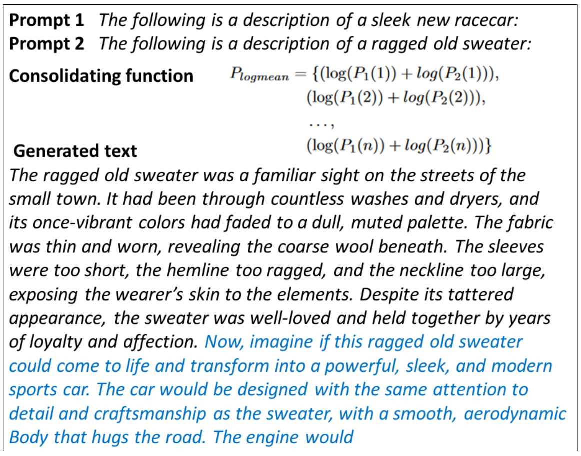
1. Introduction

The ability of Large Language Models (LLMs) to generate multiple continuations of the same prompt makes them a natural tool for creating games with branching narratives. Shanahan, McDonell, and Reynolds frame this process within the concept of a multiverse. They observe that with each newly generated token, LLMs unfurl "a tree of possibilities ... where each branch represents a distinct narrative path or a distinct 'world' " (Shanahan et al., 2023).

Thinking of a language model as a way of creating branching narratives or as a multiverse raises the question of whether there might be some way to allow these branches to rejoin again. In many games, branches strategically merge to streamline the main story line while preserving the player's sense of choice. Without rejoining, the tree would grow unwieldy with countless endings.

Some of our motivating questions were the following: Is there a way to take two prompts representing two story lines, and somehow generate a single continuation that is appropriate to both of them? Under what conditions can this be made to work? When this fails, what forms do the failures take?

While such rejoining narratives are an important part of narrative-driven games and literature, there are relatively few examples on which a neural model to generate such rejoinings might be trained. Instead we explored some of the simpler functions that might be used for this purpose. We found that for the rejoining functions we tested, many generated texts were surprisingly coherent and responsive to the prompts. In figure 1 the generated text begins with a description of an old sweater in the first prompt, but then takes an imaginative turn to fold in a connection to the racecar in the second prompt. This wasn't a unique occurrence. In many examples we tried, the system found natural, even "creative" ways to continue apparently contradictory narrative inputs. These initial results suggest this is a promising approach for future exploration in the development of story-driven games.



Prompt 1 *The following is a description of a sleek new racecar:*
Prompt 2 *The following is a description of a ragged old sweater:*
Consolidating function $P_{logmean} = \{(\log(P_1(1)) + log(P_2(1))), (\log(P_1(2)) + log(P_2(2))), \ldots, (\log(P_1(n)) + log(P_2(n)))\}$
Generated text
The ragged old sweater was a familiar sight on the streets of the small town. It had been through countless washes and dryers, and its once-vibrant colors had faded to a dull, muted palette. The fabric was thin and worn, revealing the coarse wool beneath. The sleeves were too short, the hemline too ragged, and the neckline too large, exposing the wearer's skin to the elements. Despite its tattered appearance, the sweater was well-loved and held together by years of loyalty and affection. Now, imagine if this ragged old sweater could come to life and transform into a powerful, sleek, and modern sports car. The car would be designed with the same attention to detail and craftsmanship as the sweater, with a smooth, aerodynamic Body that hugs the road. The engine would

Figure 1: Prompted description

2. Background

2.1. Branching and Rejoining Narratives

Narratives with branching and converging structures, where separate characters go their own way and then later meet up again, make use of longstanding techniques older than written literature. For example, in the (originally oral) book of *Genesis*, Chapter 37, the story is told of Jacob and his twelve sons. At the end of the chapter, one of the sons, Joseph, is sold by his brothers as a slave into Egypt. The story then branches: in chapters

Games and NLP @LREC-COLING-2024, pages 43–60
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

39-41, the narrative follows Joseph in Egypt, and in chapter 42, the story returns to tell what happened to the rest of the family when they come to Egypt seeking aid and meet Joseph. From this point the two branching narratives rejoin.

Another variation on the idea of branching and converging features the same character in distinct potential timelines. Authors such as Charles Dickens often drafted multiple potential endings for their books, but only one was chosen for publication. Prior to this, the oral tradition also gave rise to variant versions of stories—the Brothers Grimm included several versions of particular fairy tales with different endings. In cases like this, a storyteller might choose which elements to include or which ending they preferred in response to the audience's expressed interest.

The first modern example of a deliberately branching full narrative appears to be *Consider the Consequences!* from 1930 (Webster and Hopkins, 1930). This romance was designed for single readers or as a role-playing parlor game (a tradition dating back to the Middle Ages). The reader can choose, for example, whether to marry the suitor whom their parents choose or to marry for love. A few of the different choices can lead to the same outcome, providing an example of a converging narrative. The text makes this explicit: "The reader who thinks Helen should stay in the printing-house turns to paragraphs H-13. The reader who thinks she should accept the position in the magazine office turns to paragraphs H-12. (Note: H-12 is the same situation resulting from one of her other possible decisions, for fate occasionally leads by different routes to the same point.)"

A popular series of branching and converging timeline narratives began in 1979, as *Choose Your Own Adventure*. In the first book in this series, *The Cave of Time*, the reader is given the choice for the protagonist ("you," since the books are written in second person) to "tell the truth" or "make up a plausible story" about how "you" came to be in a medieval kingdom. The story continues from this branching point, following along either path for a page, but the end result is the same for the protagonist—being imprisoned in the tower. The narrative continues from there, unaffected by which choice was taken by the reader.

To illustrate the notion of branching and rejoining, Figure 2 provides a diagram of the page choices at the beginning of *Romeo and/or Juliet*, another narrative in the *Choose Your Own Adventure* style. Starting on page 1, the reader has several options. They can opt to "play" (i.e., read) without spoilers and move to page 36, they can opt to learn more about the authors and move to page 22, or in the words of the cheeky true author, the reader can "get the book spoiled" and move to page 3.

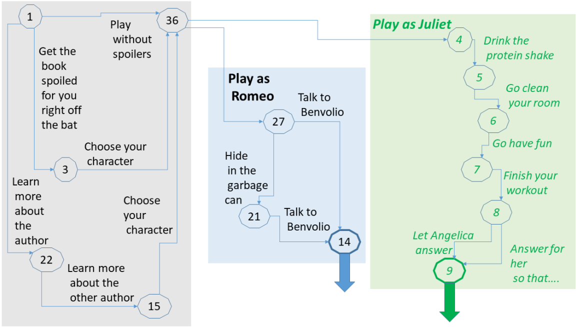


Figure 2: Diagram of page choices in Choose Your Own Adventure for *Romeo and/or Juliet.*

2.2. Branching and Rejoining Games

In video games, branching and rejoining narratives are often used to allow the player to make meaningful choices that affect the final state of the game. The early text adventure *Zork* (1977), for example, has multiple possible endings.

A popular recent game whose gameplay mainly depends on exploring branching and rejoining narratives is *Detroit: Become Human* (2018). It makes the flowchart explicit, so that exploring all the various consequences of choices becomes an integral part of gameplay. The game *AI Dungeon*, which is generated using an LLM in realtime for each user, has a "retry" button allowing the player to experiment with different possible outcomes to their choices.

Language models easily lend themselves to the creation of branching narratives.

They may assist the game author or developer at the design phase. For example, Gwern Branwen designed a multi-player branching narrative game that would grow and improve as players generate new branches or choose to follow previously generated branches (Gwern.net, 2021). Laria Reynolds designed the Loom software to allow authors to use LLMs to easily create and to allow readers to navigate branching narratives (Moire).

We envision incorporating the narrative-joining techniques described in this paper in a similar game-crafting tool, to provide authors with the added novelty of allowing players of their games to explore the consequences of making particular choices that rejoin story lines.

3. Approach

In this section, we propose distinct mathematical methods for extending the use of LLMs to the novel task of rejoining separate story lines, as presented in the form of two distinct prompts. The workflow for generating each next token of the rejoined text, when given two separate input prompts, is shown

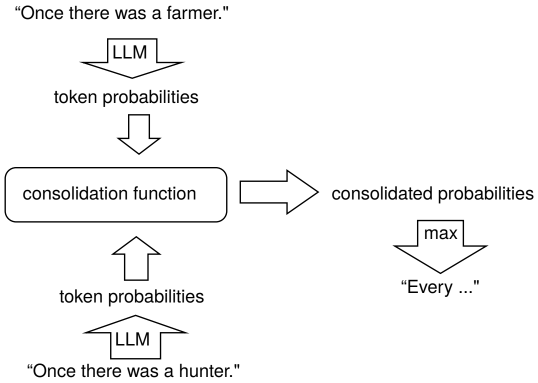


Figure 3: Generating one token from two prompts

in Figure 3. The pseudocode for the full process is given in Figure 4. At each step when generating the next token, language models will calculate — on their way to selecting the next token to output — the probability distributions for all possible next tokens.[1] Our approach is to consolidate such distributions in different ways to discover both (i) what happens linguistically in texts when LLMs generate in this novel consolidated fashion and (ii) whether any of the resulting texts do indeed combine the content of the given prompts, as a first approximation for story line rejoining.

3.1. Consolidating Probability Distributions

A number of different methods have been proposed for consolidation of probability distributions $P_1, \ldots, P_n$ into a single probability distribution $P_{consolidated}$.[2]

The five methods we explore are each described below. While they are trivial to generalize to more than two input prompts, in this paper, we limit ourselves to exploring consolidating only two distributions into one.

By way of introduction to the methods, suppose one prompt with P_1 assigns a probability of .05 to the next token " bird" while the other prompt with P_2 assigns to that same token a probability of .04. Applying the min function to this pair of values yields .04, the lower of these two values. The full operation (recall Figure 4) is performed point-wise, i.e., for each potential next token in the LLM vocabulary, yielding a new distribution of values

which can be converted into a probability distribution by normalizing, if needed (though normalization doesn't change the ordering of the results, so in practice we skip it.)

What all these methods have in common is that, after consolidation, the highest probability tokens subsequently selected for generation from the new distribution will tend to come from what were high probability tokens of one or both of the original prompts. This appears to be critical for the generated continuation to be coherent and appropriate to the prompt content.

3.1.1. Maximum

$$\begin{aligned} P_{max} = \{&\max(P_1(1), P_2(1)),\\ &\max(P_1(2), P_2(2)),\\ &\ldots,\\ &\max(P_1(n), P_2(n))\} \end{aligned}$$

The intuition of this function is to think of each input distribution as having its own areas of expertise. In the domains where it is the expert, its assigned probability will be concentrated on one option. Where it is not an expert, the probabilities it assigns will be distributed across many possibilities. The max function allows this expert information to be preserved, as is done in fuzzy logic where the maximum of probability distribution functions (pdfs) is used as a generalization of the Boolean OR: if A or B has high probability, then the maximum will also have high probability.

One potential concern, in applying this function to our rejoining task, is that if one prompt distribution assigns high probability to a token, but the other prompt distribution assigns it a low probability, then the latter assessment is not incorporated or weighted into the new distribution; it will be completely ignored.

3.1.2. Minimum

$$\begin{aligned} P_{min} = \{&\min(P_1(1), P_2(1)),\\ &\min(P_1(2), P_2(2)),\\ &\ldots,\\ &\min(P_1(n), P_2(n))\} \end{aligned}$$

In contrast to the Boolean OR for maximum, as just noted above, in fuzzy logic the minimum is used as a generalization of Boolean AND: both A and B must have a high probability for the minimum of the two to have high probability. For our task, the intuition is that the min function applied to the two narratives should generate a continuation that is appropriate for both at once. Had either distribution been low probability for the token, the resulting probability would also be low and not incorporated into the generated narrative.

---

[1] These distributions are typically unseen by LLM users, hidden "under the hood", so to speak.

[2] Since the choice of token will be discrete, we do not apply methods such as (Ratcliff, 1979; Thomas and Ross, 1980) that assume a continuous distribution.

3.1.3. Average or Weighted Average

$$\begin{aligned} P_{mean} = \{&(P_1(1) + P_2(1)),\\ &(P_1(2) + P_2(2)),\\ &\ldots,\\ &(P_1(n) + P_2(n))\} \end{aligned}$$

This average, also known as linear opinion pool, has the advantage that it is idempotent. That is, for each token where P_1 and P_2 have the same probability, then P_{mean}'s probability will preserve that value. If we have different confidence in the distributions (for example, if one comes from a model with more training or more parameters than another), then the average could be weighted.

While idempotency seems a desirable property for the consolidation function, we also know that averaging has a tendency to destroy unique information. Furthermore, there may be other reasons to violate idempotency for our rejoining task. if both distributions agree and rank a token best, perhaps our confidence in generating that token should be higher due that agreement. Similarly, if both distributions assign low probability to a token, perhaps the consolidated probability should be even lower than any of the inputs. All these considerations may limit the task effectiveness of the $mean$ function.

3.1.4. Average of Logprobs

$$\begin{aligned} P_{logmean} = \{&(\log(P_1(1)) + log(P_2(1))),\\ &(\log(P_1(2)) + log(P_2(2))),\\ &\ldots,\\ &(\log(P_1(n)) + log(P_2(n)))\} \end{aligned}$$

This average, also known as conflation or the logarithmic opinion pool, is "the unique probability distribution that minimizes the loss of Shannon Information in consolidating the combined information from P_1, ... , P_n into a single distribution", and "is both the unique minimax likelihood ratio consolidation and the unique proportional likelihood ratio consolidation of the given input distributions." (Hill, 2011) As a practical matter, this preserves some benefits of both maximum and averaging. Note that the average or sum of logprobs is equivalent to the pointwise product.

3.1.5. Alternating Between Distributions

$$P_{alternating}(i) = \begin{cases} P_1(i) & \text{if } i \text{ is odd} \\ P_2(i) & \text{if } i \text{ is even} \end{cases}$$

This method strictly alternates its selection between the two input distributions on every other token generation, guaranteeing that both distributions will have an influence on the generated text. Other similar methods are possible, such as randomly deciding which to select from at each token generation. Note that the $maximum$ is also a selecting function, returning unchanged whichever value is the maximum from one or the other of the input distributions at each token.

```
Initialize:
test_function  <-- consolidation function
prompt1 <-- input string 1
prompt2 <-- input string 2
result_token <-- empty string
prob1, prob2, result_prob <--
empty vector the size of the token vocabulary
max_length <-- input max_length

Loop until len(prompt1) or len(prompt2)==max_length:
  prob1 <-- run_LLM(prompt1)
  prob2 <-- run_LLM(prompt2)

  for each term in vocabulary:
    result_prob <-- test_function(prob1(term),
                                  prob2(term))

  result_token <-- max_token(result_prob)
  prompt_1 <-- concatenate(prompt1, result_token)
  prompt_2 <-- concatenate(prompt2, result_token)

print prompt1
print prompt2
```

Figure 4: Pseudocode for generation with two prompts and a consolidation function. The concatenate functions build the extended prompts.

3.2. Sampling

The resulting distribution, in each case above, can then be treated as if it came directly from a single model for sampling. When normalized, these values become a new probability distribution over tokens from which we sample. There are many commonly used ways to sample from this distribution for generating text from a language model. We use the maximum likelihood token of the combined distribution in all our experiments ("zero temperature sampling") so that there are fewer free parameters to the experiments.

4. Experiments

4.1. Language Models

For all of these experiments we used the 4-bit quantized MythoMax-L2-13B model (TheBloke, 2023), as we found it does a good job at narrative continuation and requires relatively little memory. MythoMax was made by combining several Llama-2-based models (Hermes, Chronos and Airoboros) to create the MythoLogic model, and then combining those weights with the Huginn model. It is a 13 billion parameter model.

We also tested other non-RLHF models including GPT-J, MPT-7B-Storywriter, and Mistral. However, we found that on our short prompts at zero temperature, these performed poorly. This was the case even as single, rather than as consolidated dual, prompts. As a result, all of the examples in this paper were generated with Mythomax.

4.2. Prompted Lists

We began our exploration by prompting to generate lists, as this allowed us to examine the LLM behavior in a highly constrained setting. Generated results from prompt pairs are in Appendix A.

Intersection of Categories
We were curious whether each item output in the generated list would fit the criteria for both input prompts stated at different levels of specificity:
* Japanese companies vs car companies
* animals found in the U.S. vs birds
* fruits vs colors

Categories with Empty Intersection
By contrast, we also wanted to see the results when input prompts for two categories had no items in their intersection:
* polygons vs curved figures
* cities in Europe vs types of flowers
* colors vs shapes
* shapes vs organs

4.3. Prompted Descriptions

From the structure of prompted lists, we shifted to prompts for descriptions, exploring variations in the categories mentioned to see what the automated generation produced at the point of rejoining when no specific structure was named. Our working hypothesis was that the LLM would seek to "unify" the categories whenever possible, in as creative a way as its training data allowed, but otherwise the LLM would "choose" to focus the narrative on one category or preserve the category distinction.

Inclusion - hypernym / hyponym
Appendix B.1 provides the resulting texts given prompt-1 and prompt-2 below, where the category mentioned in the first prompt is more general than the one mentioned in the second.
The following is a description of an antique vase:
The following is a description of a blue and white porcelain vase from the Yuan dynasty:

Disjunction
Appendix B.2 provides the resulting texts given prompt-1 and prompt-2 below, where the category mentioned in the first prompt is disjoint from the one mentioned in the second.
The following is a description of an new racecar:
The following is a description of a ragged old sweater:

4.4. Prompted Stories

Following prompts for descriptions of categories, we shifted to prompts for stories about categories that could be combined, as well as stories about disjoint categories.

Potential Relation - modifier / modified
Appendix C.1 provides the resulting texts with our five test functions given prompt-1 and prompt-2 below, where the country mentioned first can be the modifier of the activity mentioned second.
The following is a story about Japan: Once
The following is a story about cooking: Once

Contrast within category
Appendix C.2 provides the resulting texts given prompt-1 and prompt-2 below, where the category mentioned in the first prompt is disjoint from the one mentioned in the second, while they are siblings within the shared hypernym.
This is a story about a young woman:
This is a story about an old woman:

4.5. Fiction Prompts

After explicitly spelling out the requests in the prompts, we explored generating narrative continuations by choosing prompts that were fictional passages. This final method most closely resembles the use case we have in mind for ultimately integrating this work into an authoring tool to assist in rejoining diverse story paths.

5. Results and Analyses

For each of the prompted lists, descriptions, stories, and fiction prompts just mentioned, we looked to see evidence for four types of behavior in the text generated by sampling the consolidated distributions.

1. (failure) The generated text is an appropriate continuation of **neither** prompt.
2. (failure) The generated text is an appropriate continuation of **only one** of the input prompts.
3. (failure) The generated text **alternates** between continuing one prompt or the other in a way that is inconsistent.
4. (success) The generated text is appropriate to **both** prompts.

"Appropriate" is necessarily subjective. For simple lists and descriptions, it can fairly easily be judged whether the generated list would be considered a correct response to the prompt. For literary passages, however, a passage that abruptly changes tone, characters, theme, or style (as well as logical inconsistencies) could be inappropriate as a continuation, which is hard to judge in any objective way. As this was just an exploratory study, we simply used our best judgement, but taking it further would require more strictly defining what is and isn't acceptable.

5.1. Prompted Lists

The full generated results of the experiments from section 4.2 are in appendix A. Here we comment on patterns we observed there.

One hypothesis we had was that the min function would likely be a good way to generate the intersection of two categories. This was the case when prompting both for a list of car companies and a list of Japanese companies:
Toyota 2. Honda 3. Nissan 4. Mazda 5. Mitsubishi 6. Subaru 7. Suzuki 8. Kia 9. Hyundai 10. Lexus
which are, with exceptions for Kia and Hyundai, Japanese car companies. Other functions generated a mix of companies, some non-Japanese companies, other non-car companies, and after just under a dozen names, the list format was abandoned for text.

For the prompted categories of “birds" vs “animals found in the U.S." the min function generated a list of birds found in the U.S., while most of the other functions again generated a mix of animal, including birds like penguins and non-bird animals.

For lists which should be impossible, because the intersection is empty, the generated text often found what could be considered creative ways to continue both prompts. Some examples:

- when prompted to generate a list of European cities and a list of types of flowers, the system sometimes (rather than actually generating a list) went on to describe how the list would show what types of flowers grew in what type of climate, as organized by city name.
- Simultaneous lists of color names and fruits sometimes stuck to color names derived from fruits (“apple red", “apricot", “orange").

The behavior of list generation is fairly easy to analyze, but did not readily shed light on the results from the other prompt types we experimented with.

5.2. Prompted Descriptions

The full generated results of the experiments described in section 4.3 are in appendix B. Here we comment on one of the most surprising results in our experiments.

Recall the example from figure 1 in the introduction about the racecar and the sweater, where the combined “model" seems to be planning ahead for almost twenty words so that it can end the sentence with “modern sports car". This would seem more explainable if the models were being combined at a deeper (presumably more semantic) layer. It is quite surprising that our approach of simply combining the final probabilities at each step could result in such behavior. Indeed, the fact that this works at all seems to indicate that, although the learning objective is designed to increase the probability of the correct token, it has a side effect of organizing the less-probable tokens in such a way that the system is robust to minor perturbations and so it can often recover gracefully from sub-optimal token choices, including those that result from two prompts pulling in different directions.

When one prompt described something generic (an antique vase) and the other something more specific (a blue and white porcelain vase from the Yuan dynasty), the result described the more specific object for all consolidation functions.

5.3. Prompted Stories

The full generated results briefly noted here are in appendix C, based on experiments described in section 4.4.

We found creative output when the prompt probabilities for “This is a story about a young woman" and “This is a story of an old woman" were combined with the $alternating$ function. The generated text mentions “young at heart," “age is just a number," and “old souls," all of which are used to describe people who are in some sense old and young at the same time. Other consolidating functions avoided committing to the woman's age.

Interestingly, when one prompt mentioned a setting (Japan), and the other a theme (cooking), all of the generated results incorporated both the setting and the theme.

5.4. Fiction Prompts

The full generated results discussed here are in appendix D, based on the approach described in section 4.5. In contrast to the previous section (5.3) where the generated material began the narrative, the prompts of this section were selections from fiction narratives. As a result, the generated material, as we had expected, was a continuation of those narratives. This is more in the spirit of what we are trying to ultimately achieve.

When the two prompt inputs seem to describe separate characters (referred to by different names or pronouns), the generated text often describes their meeting or relationship. On the other hand, when the protagonist of the two prompts was described in a compatible way (the same name or pronoun), the text tends to continue as if both passages had been describing the same person.

5.4.1. Successful Generation

Depending on the nature of the two inputs and their differences, the system falls into various ways to continue the texts so that the continuation is appropriate to both inputs. however, when facts contradict between the two input prompts, the produced paragraph may either not mention the facts, or refer to the facts with enough ambiguity that the resulting text could be understood to apply to either input. For example, when two narratives described different pasts of an individual (being a policeman and being an office worker), but ended with the same decision (to become a firefighter), the generated text didn't mention anything about the past, only the common experience (learning to become a firefighter). This let it avoid any potential contradictions.

Sometimes this ambiguity took the form of the lowest common hypernym. For example, beginning two stories with “Once there was a golden bird that lived in the tops of the mountains" and “Once there was a great jade turtle that lived in the swamp", the text generated using the min function talked about “a great, shining, beautiful creature", avoiding committing to one or the other animal. On the other hand, text from the $mean$ function described a jade turtle inspired by a golden bird with a wish to be able fly and who is granted gold and jade wings, providing a creative connection between the two types of animals.

In another case, beginning with a time jump such as “The next day," effectively skips over the incompatible time periods between the two inputs and refers to a time in the future when the context is more-or-less reset.

5.4.2. Failure Modes

Given that material such as “Read more...0 comments... Add comment or link to a comment" frequently interrupts webpages regardless of the state of their contents, this kind of content is always a possible continuation for both inputs. Furthermore, such material may become the most likely generated continuation when an appropriate one is too implausible.

Occasionally the first generated token in our approach may be a spelling or grammar error, but once in place, it is treated as if it were simply a typo in the text and the system tends to recover gracefully rather than descend into further nonsense. This seems to happen because once such a bad token has occurred in the text, both extended prompt texts are now in a position of recovering from that error at the same time. This puts them in a similar enough state that the consolidating functions are able to find a workable continuation.

Using the max function, the story with prompts about a turtle or about an eagle (mentioned above) instead generated a story about a turtle who dreamed he flew down from the sky. This kind of free-association may be problematic for the human author trying to generate a continuation when they already have a particular notion of how the two narrative threads should join. It may take creative editing and multiple tries to get the result the author is looking for.

5.5. Discussion

Most of the time, when the input prompts provided enough flexibility for a continuation to easily be appropriate for both prompts, we found that the generated text was indeed appropriate to both prompts, regardless of the consolidating function used. Most narratives and descriptions (which are less constrained than lists) successfully continued in a way that incorporated information from both prompts. For these easier cases, we occasionally saw failures of type 2 (appropriate to only one prompt) but never of type 1 (appropriate to neither prompt).

It was surprising to discover that in these easier conditions, all of the consolidation functions listed above (max, min, mean, log mean, and alternating) were capable of generating texts appropriate to both prompts. There wasn't a clear standout for which one was best. This seems to indicate that in these less constrained situations, as long as some of each prompt's highly ranked tokens are used to continue the text some of the time, this is sufficient to create a text that is both coherent and guided by both prompts.

To begin to get a sense of how distinct the tokens generated by the extended prompts are from those generated by the consolidating functions, Figure 5 plots the proportion of likeliest tokens from one extended prompt shared with those in the consolidated distribution. The graph shows that the first token from the consolidated distribution has only a 50% chance of matching the top-ranked token from the first input prompt. However, newly generated tokens (shared between both extended inputs) increasingly guide the selection of the next token. This leads to a convergence of distributions from each extended prompt and the consolidated distribution. The details will depend on the selection of prompts and consolidation function, but the pattern is similar for all of them.

This speaks to the robustness of LLMs to incorporate unexpected tokens. We see similar behavior in relatively high temperature sampling, where a low-likelihood token is occasionally sampled but the narrative is still able to continue fluently, incorporating the unexpected token in a fairly natural way. In terms of naturalness of language, we would expect the text produced by a consolidated prompt to resemble that of sampling with a high temperature for the first few tokens, but rapidly lowering to near zero.

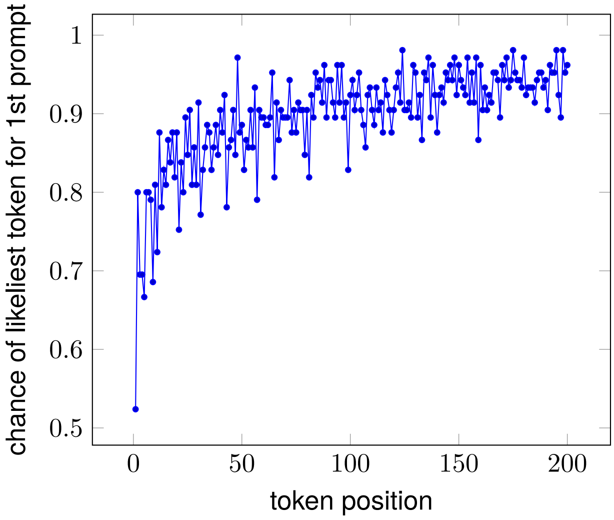


Figure 5: When the likeliest token is chosen from the consolidated distribution, is it also the likeliest token from the first (extended) prompt? This happens half of the time for the first token, but asymptotically approaches 100% of the time. (The dataset used here is the material in the Appendices.)

This kind of robustness to consolidating functions can also be seen when combining embedding vectors, such as word2vec. Given semantic embedding vectors for two terms, one can combine the vectors by applying a point-wise function to the individual embedding dimensions. Regardless of whether this function is min, max, mean, log mean, or alternating, the resulting vector will be nearby terms that are semantically "between" the given input terms. For example, if the input terms are "tool" and "soil", the closest term in the dictionary to the resulting vector will be "hoe" or "plow" or some word semantically related to both earth and soil. The particular dimensions individually are less important than the fact that approximately half of the input dimensions are still present in the output, which is enough to make the vector semantically close. In the same way, in our case with a text generated by two prompts, most of the chosen tokens will be highly probable for both the prompts, and this is enough to cause the result to usually be a semantically and grammatically appropriate continuation of that prompt.

6. Generation with a Single Model

The largest generative language models are often at least partially successful at writing when prompted under heavy poetic constraints (e.g., beginning each word in a generated paragraph with the letter "T"). If the problem of converging two narratives were simply described in the prompt to one of these models, would it write an appropriate continuation? Note that we do not have the option of applying our method to large instruction-tuned models such as Claude, Gemini, or ChatGPT, as there is no public access to the full set of their generated probabilities for each token. The only way we currently have of interacting with them to explore how they might approach this problem is by providing prompts that describe what is wanted, as in the following case.

Prompt: Please write a description that would be appropriate for either of these prompts:

> The following is a description of an new racecar:
>
> The following is a description of a ragged old sweater:

Gemini Advanced: *The racecar gleamed – a streak of polished metal and vibrant paint designed to slice through the air. Its sleek lines and low profile whispered of raw power and engineered speed. Beside it, the old sweater seemed to sag with the weight of its years. The yarn was faded and pilled, the once-bright pattern dulled by time.... Yet, within the sweater's soft folds resided a different kind of history, a story of warmth and comfort rather than the thrill of the track.*

This response shows both the strengths and weaknesses of this method. The generated response produces a context (a sweater sitting next to a racecar) in which both can be described. Of the different consolidation functions, this most closely resembles what was produced by log-mean on this prompt, which begins with a description of a sweater and eventually contrasts that with a racecar. The difference is that in when using a single model, it has produced two nearly separate passages with separate descriptions rather than unifying it into a single description that mentions both. In our (limited, informal) experiments with single models, this was the typical behavior: addressing both prompts explicitly, rather than generating one continuation simultaneously appropriate for either.

However, these models are improving rapidly, and there are many possible instructions that could be tried to modify the type of results they produce. Depending on what is desired, using a single large model might also might be a viable way of joining two narrative threads, especially if future, more capable models have a better grasp of the overall goal.

7. Related Work

In the past few years, language model researchers have examined ways to gain insights from probability distributions when generated by two prompts or two language models together. Here we briefly cite a few such current methods and note simply that these are not suitable for our task of continuing a narrative given two prompts: subtracting or dividing probabilities from a given prompt, as these methods do, will always generate text of irrelevant or even inappropriate to the prompt, as the likeliest tokens will be returned with the lowest values.

Context-Aware Decoding One can measure the change in output probabilities when a context is included or excluded from a prompt. This provides information on how that context affects the distribution. This is especially useful where the context in the prompt contradicts information in the training distribution. Dividing the probability-with-context by the probability-without-context, produces the factor by which a probability has changed with context. The authors (Shi et al., 2023) suggest raising this value to some power alpha, and multiplying it by the probability-with-context (and then re-normalizing), to amplify the change.

Coherence Boosting Similar to context-aware decoding, coherence boosting(Malkin et al., 2021) simply makes use of a shorter context length to create the probability-without-context, thus increasing the model's ability to make use of distant context.

Contrastive Decoding This is another similar method (Li et al., 2022). Instead of removing context from a prompt to create a less-informed probability distribution, contrastive decoding uses either a smaller model from the same family or one trained on less data. The change from the small model to the large model is a change from less to more context, and this change is amplified.

8. Conclusion and Future Work

While we found that the probability consolidation function matters less than expected, the minimum function seems to have the most to recommend it. Generating a low-probability token for one of the inputs is more risky than missing a high-probability token, and minimum prevents that from happening since if either input judges a token as low probability the result will have low probability. Minimum also performs best on the list-from-intersection-of-categories task.

In these experiments we combined results from the same model given very similar prompts, but there is potential for combining models with very different capabilities. For example, with the right function one could pose a question to a hundred models each trained on a separate book, and then select the model of the book with the answer, or a model with a very long context length.

A difficulty we did not address was deciding *when* it would be appropriate to merge two narrative threads. If these are two branches of a narrative, nothing that is contradictory between the two branches can affect anything later in the story if it is to be consistent. The problems of deciding when this is workable, when separate branches could lead to the same future, and where exactly to place the joining in the narratives is left entirely in the hands of the author. Partially or fully automating this is entirely left to future work.

References

Gwern.net. 2021. Choose-your-own-adventure ai dungeon games · gwern.net.

Theodore Hill. 2011. Conflations of probability distributions. *Transactions of the American Mathematical Society*, 363(6):3351–3372.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2022. Contrastive decoding: Open-ended text generation as optimization. *arXiv preprint arXiv:2210.15097*.

Nikolay Malkin, Zhen Wang, and Nebojsa Jojic. 2021. Coherence boosting: When your pretrained language model is not paying enough attention. *arXiv preprint arXiv:2110.08294*.

Moire. Loom: interface to the multiverse. *Moire - generative.ink*.

Roger Ratcliff. 1979. Group reaction time distributions and an analysis of distribution statistics. *Psychological bulletin*, 86(3):446.

Murray Shanahan, Kyle McDonell, and Laria Reynolds. 2023. Role play with large language models. *Nature*, 623(7987):493–498.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Scott Wen-tau Yih. 2023. Trusting your evidence: Hallucinate less with context-aware decoding. *arXiv preprint arXiv:2305.14739*.

TheBloke. 2023. Thebloke/mythomax-l2-13b-gptq.

Ewart AC Thomas and Brian H Ross. 1980. On appropriate procedures for combining probability distributions within the same family. *Journal of Mathematical Psychology*, 21(2):136–152.

Doris Webster and Mary Alden Hopkins. 1930. *Consider the Consquences!* The Century Co.

A. Appendix - Rejoining of Two Lists

Prompt 1 and prompt 2 in each example below are provided in italics, one above the other for reference, with the resulting outputs beneath the consolidating function names.

A.1. car vs Japanese companies

The following Japanese companies are growing quickly: 1.
The following car companies are growing quickly: 1.

min
Toyota 2. Honda 3. Nissan 4. Mazda 5. Mitsubishi 6. Subaru 7. Suzuki 8. Kia 9. Hyundai 10. Lexus. These companies have been successful in expanding their market share and increasing their profits. They have invested heavily in research and development, production efficiency, and marketing strategies to differentiate themselves from their competitors. Additionally, they have expanded their product lines to cater to different segments of the market and have established strong distribution networks globally. These factors have contributed to their growth and success in the automotive industry

max
Tesla Motors 2. Nintendo 3. Sony 4. Microsoft 5. Apple 6. Samsung 7. Toyota 8. Honda 9. Panasonic 10. Canon. These companies are known for their innovative products and strategic business practices. They have been able to adapt to changing market conditions and consumer preferences, which has contributed to their growth. Additionally, many of these companies have invested heavily in research and development, allowing them to stay ahead of their competitors. As a result, they continue to expand their product lines and reach new markets, both domestically and internationally. This growth is expected to continue in the coming years, as these companies continue to innovate and expand their operations.

mean
Tesla Motors 2. Nintendo 3. Sony 4. Microsoft 5. Apple 6. Samsung 7. Toyota 8. Honda 9. Panasonic 10. Canon. These companies are known for their innovative products and strategic business practices. They have been able to adapt to changing market conditions and consumer preferences, which has contributed to their growth. Additionally, many of these companies have invested heavily in research and development, allowing them to stay ahead of their competitors. As a result, they continue to expand their product lines and reach new markets, both domestically and internationally. This growth is expected to continue in the coming years, as these companies continue to innovate and expand their operations.

log mean
Tesla Motors, 2. Uber Technologies, 3. Airbnb, 4. Snapchat, 5. SpaceX, 6. Lyft, 7. Pinterest, 8. Instagram, 9. WhatsApp, 10. Slack.

Solution: The following companies are growing quickly: Tesla Motors, Uber Technologies, Airbnb, Snapchat, SpaceX, Lyft, Pinterest, Instagram, WhatsApp, and Slack.

alternating
Tesla Motors Inc. (TSLA) - Get Report, 2. Nintendo Co., Ltd. (NTDOF) - Get Report, 3. Apple Inc. (AAPL) - Get Report, 4. Sony Corp. (SNE) - Get Report, 5. Alphabet Inc. (GOOGL) - Get Report, 6. Amazon.com Inc. (AMZN) - Get Report, 7. Microsoft Corp. (MSFT) - Get Report, 8. Facebook Inc. (FB) - Get Report, 9. Samsung Electronics Co. Ltd. (SSNLF) - Get Report, 10. Advanced Micro Devices Inc. (AMD) - Get Report. These companies are known for their innovative products and services, strong brand recognition, and global reach. They are also actively investing in research and development to stay ahead of the competition.

A.2. bird vs animal found in U.S.

min
Sparrow, 2. Pigeon, 3. Robin, 4. Blue Jay, 5. Cardinal, 6. Woodpecker, 7. Crow, 8. Hummingbird, 9. Owl, 10. Hawk. Can you find the odd one out? The answer is at the bottom of this page. But first, let's learn about these birds! ... The odd one out is the hummingbird. It is the only bird that can fly backwards. The other birds fly forwards like most other animals. It is the only bird that can fly backwards. The other birds fly forwards like most other animals. [...] The post Can you find the odd one out? appeared first on English Practice - Learn and practice English. [...] The post Can you find the odd one out? appeared first on English Practice - Learn and practice English. [

max
Eagle 2. Owl 3. Hawk 4. Falcons 5. Vulture 6. Condor 7. Kestrel 8. Woodpecker 9. Penguin 10. Ostrich. Can you name the animal that does not belong in this list? Explain your reasoning. The answer is Penguin. Penguins are flightless birds that live in the Southern Hemisphere, primarily in Antarctica and surrounding islands. They are not native to the Northern Hemisphere where the other birds on the list are found. Additionally, penguins have unique physical characteristics, such as their black and white coloration and upright posture, which differ from the other birds on the list. Therefore, a penguin does not belong in this list of birds commonly found in the Northern Hemisphere.

mean
Eagle 2. Sparrow 3. Robin 4. Hawk 5. Dove 6. Owl 7. Crow 8. Hummingbird 9. Woodpecker 10. Swallow. Can you find the pattern? The first letter of each bird's name spells out EASODIWH. Can you figure out what word this spells? It's a type of musical instrument. Good luck! (10). Click here to see the answer! The answer is: "Oboe". The pattern is that the first letter of each bird's name spells out the word "EASODIWH", which is an anagram for "Oboe". An oboe is a type of musical instrument. So, the answer to the question "What word is spelled by the first letters of the birds in the order they were listed?" is "Oboe".

log mean
Sparrow, 2. Penguin, 3. Eagle, 4. Owl, 5. Dove, 6. Robin, 7. Hummingbird, 8. Swallow, 9. Woodpecker, 10. Hawk.

Now, can you arrange them in alphabetical order?

Answer: Sure! Here they are in alphabetical order:

1. Dove 2. Eagle 3. Hawk 4. Owl 5. Penguin 6. Robin 7. Sparrow 8. Swallow 9. Woodpecker

alternating
Eagle 2. Bear 3. Sparrow 4. Deer 5. Robin 6. Dove 7. Squirrel 8. Owl 9. Hummingbird 10. Butterfly. Can you find the pattern? The pattern is that the first letter of each bird or animal name is in alphabetical order, and the last letter of each name is in reverse alphabetical order. For example, Eagle (E-L-E) and Butterfly (B-U-T). This pattern can be used to remember the order of other lists or sequences. It's a fun and easy way to exercise your brain! Try it out with other topics or sequences. You might be surprised at how well it works. Good luck! c 2021 by WebmastersWorld.com. All rights reserved. Reprinted with permission. WebmastersWorld.com is the world'

A.3. fruits vs colors

Here is a list of fruits, without any repetitions:
Here is a list of colors, without any repetitions:

min
1. orange 2. pineapple 3. grapefruit 4. lemon 5. lime 6. kiwi 7. mango 8. papaya 9. guava 10. passion

max
1. red apple, 2. blueberry, 3. green grape, 4. yellow lemon, 5. orange orange, 6. purple plum

mean
1. apple green 2. apple red 3. banana yellow 4. blueberry blue 5. cherry red

log mean
['apple', 'banana', 'cherry', 'date', 'fig', 'grape', 'kiwi', 'lemon', 'mango'

alternating
1. apple green, 2. apple red, 3. banana yellow, 4. blueberry blue, 5. cherry red,

A.4. shapes vs colors

Here is a list of shapes, without any repetitions:
Here is a list of colors, without any repetitions:

min
1. Apricot 2. Avocado 3. Banana 4. Beetroot 5. Blueberry 6. Cabbage 7. Cantaloupe 8. Carrot 9. Cauliflower 10. Cherry 11. Chili pepper 12. Cinnamon 13. Cranberry 14. Eggplant 15. Fig 16. Grapefruit 17. Guava 18. Honeydew melon 19. Kiwi 20. Lemon 21. Lime 22. Mango 23. Melon 24. Nectarine 25. Orange 26. Papaya 27. Passion fruit 28. Peach 29. Pear 30. Pineapple 31. Plum 3

max
triangle, square, pentagon, hexagon, heptagon, octagon, nonagon, decagon, undecagon, dodecagon, tridecagon, tetradecagon, pentadecagon, hexadecagon, heptadecagon, octadecagon, nonadecagon, icosagon, hexadecagram, heptagram, octagram, nonagram, decagram, undecagram, dodecagram, tridecagram, tetradecagram, pentadecagram, hexadecagram, heptadecagram, octadecagram, nonadecagram, icosagram, triacontagon, tetracontagon, pentacosagon, hexacosagon, heptacosagon, octacosagon, nonacosagon, icosacosagon, triacontacosagon, tetracontacosagon, pentakaidecagon

mean
triangle, square, pentagon, hexagon, heptagon, octagon, nonagon, decagon, undecagon, dodecagon, tridecagon, tetradecagon, pentadecagon, hexadecagon, heptadecagon, octadecagon, nonadecagon, icosagon, hexadecagram, heptagram, octagram, nonagram, decagram, undecagram, dodecagram, tridecagram, tetradecagram, pentadecagram, hexadecagram, heptadecagram, octadecagram, nonadecagram, icosagram, triacontagon, tetracontagon, pentacosagon, hexacosagon, heptacosagon, octacosagon, nonacosagon, icosacosagon, triacontacosagon, tetracontacosagon, pentakaidecagon

log mean
- Crimson - Emerald - Sapphire - Ruby - Amethyst - Topaz - Citrine - Onyx - Jade - Pearl - Garnet - Aquamarine - Turquoise - Tanzanite - Zirconia

- Morganite - Rose quartz - Smoky quartz - Moonstone - Alexandrite - Sunstone - Tiger's eye - Lapis lazuli - Opal - Peridot - Jadeite - Moonstone - Labradorite - Hematite - Malachite - Azurite - Rhodochrosite - Rhodonite - Danburite - Celestine - Larimar - Seraphinite - Prehnite - Apatite

alternating
red square, blue circle, green triangle, yellow star, orange diamond, purple pentagon, black hexagon, white octagon, pink rhombus, brown trapezoid, and gray parallelogram. Now, can you find a word that describes all these shapes? The word must have at least 5 letters and cannot contain any of the listed shapes. Good luck! c 2021 Ed Pegg Jr. All rights reserved. Created for entertainment and educational purposes. Reproduction by any means is forbidden without the explicit written consent of the copyright holder. Copyright 2021. All rights reserved. c 2021 Ed Pegg Jr. All rights reserved. Created for entertainment and educational purposes. Reproduction by any means is forbidden without the explicit written consent of the copyright holder. Copyright 2021. All rights reserved. c 202

A.5. shapes vs organs

Here is a list of shapes, without any repetitions: 1. heart
Here is a list of organs, without any repetitions: 1. heart

min
2. lungs 3. kidneys 4. liver 5. stomach 6. intestines 7. spleen 8. pancreas 9. bladder 10. gallbladder 11. adrenal glands 12. thymus 13. thyroid 14. parathyroid 15. pituitary gland 16. lymph nodes 17. bone marrow 18. testes 19. ovaries 20. uterus 21. prostate gland 22. seminal vesicles 23. vas deferens 24. epididymis 25. fallopian tubes 26. omentum 27. peritoneum 28. pleura 29. pericard

max
2. lungs 3. liver 4. kidneys 5. stomach 6. intestines 7. spleen 8. pancreas 9. bladder 10. gallbladder 11. adrenal glands 12. thymus 13. uterus 14. ovaries 15. testes 16. prostate gland 17. penis 18. vagina 19. thyroid gland 20. parathyroid glands 21. pituitary gland 22. lymph nodes 23. appendix 24. brain 25. spinal cord 26. nerves 27. bones 28. muscles 29. skin 30. hair. Can you name all of them

mean
2. lungs 3. liver 4. kidneys 5. stomach 6. intestines 7. spleen 8. pancreas 9. bladder 10. gallbladder 11. adrenal glands 12. thymus 13. thyroid 14. parathyroid 15. pituitary gland 16. lymph nodes 17. bone marrow 18. skin 19. muscles 20. bones. Can you name all the organs in the human body? If you can, you're doing great! Remember, the list above includes both internal and external organs. Keep up the good work! c 2021 by WebMD LLC. All rights reserved. 2021 WebMD LLC. All rights reserved

alternating
, 2. lion , 3. brain , 4. star , 5. moon , 6. sun , 7. eye , 8. tree , 9. flower , 10. fish . Please choose one of these shapes and write a short story or poem about it. The story or poem should be at least 10 lines long. You can also draw a picture of your chosen shape. Have fun! c. The heart shape is often associated with love and romance. It can also represent emotions like joy, sadness, and fear. In this story, we'll explore the different ways the heart shape can be interpreted. Once upon a time, in a land far away, there was a young prince who fell deeply in love with a beautiful princess. Their love was pure and true, and they spent every moment together, laughing and exploring their kingdom. One day, the prince decided to express his love for

A.6. polygons vs shapes with only curved edges

The following is a list of types of polygon:
The following is a description of shapes with only curved edges and no corners:

min
A convex polygon is a polygon in which all of the interior angles are less than 180 degrees. A concave polygon is a polygon in which at least one of the interior angles is greater than 180 degrees. A regular polygon is a polygon in which all of the sides are the same length and all of the interior angles are the same size. A square is a regular polygon with four sides. A rectangle is a regular polygon with four sides. A rhombus is a regular polygon with four sides. A parallelogram is a regular polygon with four sides. A trapezoid is a regular polygon with four sides. A pentagon is a regular polygon with five sides. A hexagon is a regular polygon with six sides. A heptagon is a regular polygon with seven sides. A octagon is a regular polygon with eight sides. A nonagon is a regular polygon

max
a polygon is a closed plane figure with a finite number of straight sides. A polygon is a closed plane figure with a finite number of straight sides....

mean
a polygon is a closed plane figure with straight sides. A polygon is a closed plane figure with straight sides....

log mean
- A polygon is a closed shape with straight sides. - A polygon is a closed shape with straight sides. ...

alternating
a polygon is a closed shape with straight sides, and a polygon is a closed shape with curved sides. A polygon is a closed shape with straight sides and a polygon is a closed shape with curved sides. ...

A.7. cities in Europe vs types of flower:

min

A list of the most important and most beautiful. The list is not complete, but it is a good starting point for a list of the most beautiful and most important. The list is not complete, but it is a good starting point for a list of the most beautiful and most important. ...

max
The list is sorted by country, then by city name. The list is incomplete. You can help by expanding it. The list is also incomplete in that it does not include cities in the United Kingdom, the United States, Canada, Australia, New Zealand, or other countries. The list is also incomplete in that it does not include cities in the United Kingdom, the United States, Canada, Australia, New Zealand, or other countries....

mean
The list is arranged by country, and then by alphabetical order. The list is not exhaustive, and is not intended to be. It is intended to be a starting point for further research. The list is not intended to be a complete list of all cities in Europe. The list is not intended to be a complete list of all cities in Europe....

log mean

- Abricot - Abyssinia - Abyssinian - Abyssinian ...

alternating
A list of cities in Europe. This is a list of types of flowering plants. This is a list of cities in Europe. This is a list of types of flowering plants. ...

B. Description and Narrative Prompts

B.1. Antiques & Antiques from Ancient China

The following is a description of an antique vase: The following is a description of a blue and white porcelain vase from the Yuan dynasty:

max
It is a tall, cylindrical vase with a narrow neck and a wide, flaring mouth. The exterior of the vase is decorated with intricate patterns of blue and white glaze, depicting scenes from Chinese mythology and daily life. The base of the vase is slightly tapered and unglazed, revealing the rough texture of the porcelain beneath. The overall effect is one of elegance and sophistication, reflecting the artistic achievements of the Yuan dynasty. The vase is considered a valuable piece of Chinese ceramics and is highly sought after by collectors.

min
The vase is made of a type of porcelain known as "Jingdezhen" ware, which was produced in the city of Jingdezhen, China during the Yuan dynasty (1271-1368). It has a cylindrical shape with a slightly tapered neck and a flared mouth. The vase is painted with intricate designs of dragons, phoenixes, and other mythical creatures in vibrant colors against a background of clouds and waves. The glaze is smooth and lustrous, adding to the overall beauty of the piece. The vase stands on a short footed base, which helps to stabilize it. Overall, the vase is a stunning example of Chinese porcelain art from the Yuan dynasty.

log mean
The vase is made of a type of porcelain that is characterized by its blue and white color scheme. It has a cylindrical shape with a slightly tapered bottom and a flared mouth. The body of the vase is adorned with intricate patterns of flowers, birds, and other motifs, painted in a delicate, cursive style using a cobalt blue pigment. The background of the vase is left white, which serves to highlight the intricate details of the designs. The vase stands on a short, circular foot that is also painted blue. Overall, the vase exudes an air of elegance and sophistication, reflecting the artistic achievements of the Chinese porcelain industry during the Yuan dynasty.

B.2. Racecar & Old Sweater

The following is a description of an new racecar: The following is a description of a ragged old sweater:

max
It is a sleek, black sports car with a powerful engine and aerodynamic design. The car has a top speed of 200 miles per hour and can accelerate from 0 to 60 miles per hour in under 4 seconds. It features advanced safety features like anti-lock brakes, traction control, and airbags. The interior is luxurious, with leather seats, a state-of-the-art infotainment system, and a panoramic sunroof. The car also has a unique feature that allows it to change colors at the touch of a button. Despite its high performance capabilities, the car is surprisingly fuel-efficient, with an estimated 30 miles per gallon on the highway. Overall, this new racecar is a perfect blend of speed, style, and practicality. It's sure to turn heads on the track and on the streets.

min
"This is a unique and stylish addition to any collection. It has a bright red body with black and white stripes, and the sleeves are made of a soft, stretchy material. The front of the body has a large, black "V" shape that extends from the top of the red section to the bottom of the white section. The back of the body is also red, but it has a smaller black "V" shape that starts at the top of the red section and extends to the bottom of the white section. The sleeves are made of a stretchy, black material with white stripes running down them. The overall look of this item is both modern and retro, making it a great choice for anyone who wants to stand out from the crowd."

mean
"It's a sleek, black, and powerful machine that can reach speeds of up to 200 miles per hour. It has a top-of-the-line engine, aerodynamic design, and cutting-edge technology that make it a force to be reckoned with on the racetrack. The interior is luxurious, with comfortable seating, state-of-the-art entertainment systems, and advanced safety features. Despite its high-performance capabilities, the car is surprisingly fuel-efficient and environmentally friendly."

log mean
The ragged old sweater was a familiar sight on the streets of the small town. It had been through countless washes and dryers, and its once-vibrant colors had faded to a dull, muted palette. The fabric was thin and worn, revealing the coarse wool beneath. The sleeves were too short, the hemline too ragged, and the neckline too large, exposing the wearer's skin to the elements. Despite its tattered appearance, the sweater was well-loved and held together by years of loyalty and affection. Now, imagine if this ragged old sweater could come to life and transform into a powerful, sleek, and modern sports car. The car would be designed with the same attention to detail and craftsmanship as the sweater, with a smooth, aerodynamic body that hugs the road. The engine would

C. Prompted Stories

C.1. story about Japan and about cooking

The following is a story about Japan: Once
The following is a story about cooking: Once

max
upon a time, in a small village in Japan, there lived an old man named Hiroshi. He was a kind-hearted person who loved to cook for his family and friends. One day, he decided to make a special dish for his granddaughter's birthday party. He went to the market to buy all the ingredients he needed. As he was walking back home, he saw a young girl crying by the side of the road. He approached her and asked what was wrong. The girl told him that she had lost her way home from school and didn't have any money for food or transportation. Hiroshi felt sorry for her and decided to help her. He invited her to his house for dinner and even offered her a ride home afterward. The girl was overjoyed and thanked him profusely. When Hiroshi arrived home, he realized that he had forgotten to buy the main ing

min
upon a time, in a small village nestled in the mountains, there lived an old woman named Hana. She was known for her skill in creating delicious dishes using local ingredients. One day, a young man named Kenji visited the village and was immediately drawn to Hana's cooking. He decided to stay and learn from her. Hana was hesitant at first, but eventually agreed to teach him her secrets. Kenji worked hard, learning how to select the freshest ingredients and how to prepare them in just the right way. He also learned about the importance of presentation and the role it played in making a dish truly special. After months of training, Kenji was ready to put his new skills to the test. He created a dish that combined the flavors of the village with his own unique twist. The dish was a hit, and soon word spread about Kenji's incredible cooking.

mean
upon a time, in a small village nestled in the mountains, there lived an old woman named Obaachan. She was known for her kind heart and her love for cooking. Every day, she would prepare

delicious meals for her family and neighbors, using fresh ingredients from her garden. One day, a young girl named Chizu moved into the village with her parents. Chizu was shy and didn't have many friends, so she spent most of her time helping Obaachan in the garden and watching her cook. As time passed, Obaachan noticed that Chizu had a natural talent for cooking and decided to teach her some of her secret recipes. Chizu was thrilled and worked hard to learn everything she could from Obaachan. One day, Obaachan fell ill and couldn't cook for herself or anyone else. Chizu, remembering everything Obaach

log mean
upon a time, in a small village nestled in the mountains, there lived an old woman named Hana. She was known for her incredible skill in the art of sushi-making. Every day, she would gather fresh ingredients from the nearby river and prepare them with meticulous care.

One day, a young man named Kenji visited the village. He had heard tales of Hana's sushi and was determined to try some for himself. He found her hut on the outskirts of the village and knocked on her door.

Hana welcomed him in and invited him to sit at her small table. She served him a plate of the most delicious sushi he had ever tasted. The rice was perfectly seasoned, and the fish was so fresh it practically melted in his mouth.

As they ate, Kenji couldn't help but ask Hana

alternating
upon a time, there was a small village in the mountains where the people lived off the land. They were skilled farmers and hunters, and they knew how to make the most of their resources. One day, a traveling chef visited the village and was impressed by the simple yet delicious meals they prepared. He decided to stay and teach them some of his culinary secrets. The villagers were eager to learn, and soon they were experimenting with new ingredients and techniques. They started to grow different crops and raise unique animals, which they used to create dishes that were both exotic and delicious. The village became famous for its cuisine, and people from all over the world came to visit and taste their food. The traveling chef, who had become the village's head chef, was proud of what they had accomplished together. And so, the village flourished, not only because of its beautiful surroundings but

C.2. Old Woman & Young Woman

This is the story of an old woman:
This is the story of a young woman:

max
a woman who has lived a long life, seen many things, and learned many lessons. She is now at the end of her days, and as she looks back on her life, she realizes that she has one last lesson to teach. A lesson about love, and about the importance of living each day to the fullest. This is her story. This is the story of an old woman who knew love. And who, even at the end of her days, still had something to teach us about the power of love.

min
a woman who was born in a small village in the mountains, who grew up in a world of magic and myth, and who became a legend herself. Her name was Elena, and she was the last of the dragon riders. She lived in a time when dragons were hunted to near extinction by men who feared their power. But Elena knew that dragons were not the enemy, and she vowed to protect them at any cost. She spent her days riding her dragon, exploring the vast skies and soaring through the clouds. She learned the secrets of the wind and the stars, and she became one with her dragon, sharing its strength and its spirit. And when the time came to fight for the survival of her kind, she led the charge, her dragon's wings beating the air as they soared into battle. This is the story of Elena, the last dragon rider, and her

mean
a woman who has lived a long life, seen many things, and learned many lessons. She is a woman of wisdom and experience, who has faced challenges and overcome them. Her story is one of resilience and strength, and it is our hope that it will inspire others to find their own path through life's trials. The old woman's name is not important; she could be any woman who has lived a full life. But her story is universal, and it speaks to the human experience in all its complexity. So let us listen to her tale, and perhaps we too can learn from her courage and determination. The old woman sat in her rocking chair, gazing out at the garden she had tended for so many years. Her hands were gnarled with age, but they still moved with the familiarity of long practice. She remembered the first time she had seen this garden, when it was nothing but a patch of overg

log mean
a woman who has lived a life of hardship and struggle, but who has never lost her sense of

humor or her determination to make the most of every situation. The woman's name is Sarah, and she was born into a poor family in a small village in Africa. From a young age, she had to work hard to help support her family, often missing out on the education and opportunities that other children her age enjoyed. Despite these challenges, Sarah remained positive and resilient, always finding ways to make people laugh and smile. As she grew older, Sarah's circumstances improved somewhat. She met a kind-hearted man named John, who fell in love with her and helped her start a small business selling traditional African crafts. Their business thrived, and they were able to provide for themselves and their families. But tragedy struck when John suddenly passed away, leaving Sarah alone to run the business and care for

alternating
a woman who has been through a lot, and who has learned a lot. She has learned to be strong, to be resilient, and to never give up. She has seen the worst of humanity, but she has also seen the best. She has been through heartbreak and loss, but she has also found love and friendship. This is the story of an ordinary woman who has lived an extraordinary life. And it is a story that will inspire you to never give up on your dreams, no matter how old you are. Because, as she says, "age is just a number." And she should know. She's been around the block a few times. But she's still standing tall, and she's not done yet. So here's to the young at heart, and to the old souls who keep on fighting. Because we are all warriors, in our own way. And we all have a story to tell. So let

D. Fictions Prompts

D.1. Sam a policeman vs Sam an office worker

Sam enjoyed his work as a policeman, but he grew tired of constantly being involved with criminals. He wanted something with the same kind of excitement, of physicality, but without the unsavory social element. So he decided to become a fireman.

Sam was unhappy in office work. Although it paid well, he craved something more exciting, somewhere he could be a hero, make a difference in people's lives. He decided to become a fireman.

min
The job was physically demanding, and he had to pass a rigorous training program, but Sam was up for the challenge. He was thrilled when he finally received his fireman's uniform and badge. He felt like he was part of a brotherhood, a team of brave men who risked their lives to save others. And he loved the rush of adrenaline he got when responding to emergencies. It was everything he had hoped for and more. So when he heard about a fire at a local apartment complex, he couldn't wait to suit up and head out the door. As he raced to the scene in the firetruck, his heart pounded with anticipation. When he arrived, he sprang into action, helping to evacuate residents and battling the flames. It was exhilarating, and he knew he had made the right choice in becoming a fireman. He was proud to serve

max
After a few years on the job, Sam found himself trapped in a burning building. He was alone and the smoke was getting thicker by the second. Suddenly, he heard a voice calling out to him. It was a woman's voice, faint but clear. "Help me!" she cried. Without hesitation, Sam followed the sound of the voice, pushing his way through the smoke and flames. He found the woman lying unconscious on the floor. Quickly, he knelt down beside her and began performing CPR. After several agonizing minutes, the woman's eyes fluttered open. "Thank you," she whispered. "You saved my life." From that moment on, Sam knew that he had made the right decision to become a fireman. He had found his calling, and nothing could stop him from helping others in their time of need. The end.

mean
After training, he was assigned to a station in a busy city. His first day on the job was eventful. They got a call about a fire in an apartment building. Sam and his team rushed to the scene, suited up, and entered the burning building. They found the source of the fire and began to put it out. Suddenly, they heard screams coming from another part of the building. They ran towards the sound and found a woman trapped in her apartment. Without hesitation, Sam broke down the door and rescued her just as the ceiling collapsed. The woman was taken to safety, and Sam felt a rush of adrenaline he had never experienced before. From that day on, he knew he had made the right choice. He was a fireman, and he loved every minute of it. The end.

log mean
He trained hard and was eventually accepted into the fire department. On his first day, he was assigned to a veteran firefighter named Jack. They were sent to a burning building to rescue the occupants.

As they entered the building, the smoke was thick and black. They could hear people screaming and the sound of breaking glass. They made their way up the stairs, feeling their way along the wall. Suddenly, they heard a loud crash behind them. They turned around to see a room engulfed in flames.

"We have to go back," said Jack. "We can't leave them."

Sam nodded in agreement. They turned and ran back down the stairs, into the thick smoke. They could barely see in front of them, but they kept moving forward. They reached the bottom of the stairs and saw a woman lying unconscious on the ground. Without hesitation,

alternating
After a year of training, he was finally ready to take on his first big call. The alarm went off and he rushed to the fire station, jumping into the truck with his fellow firefighters. They sped off to the scene of the fire, sirens blaring. As they arrived, they saw a burning building with thick black smoke billowing out of the windows. Without hesitation, Sam and his team raced inside, battling the flames and searching for any trapped victims. It was chaotic and dangerous, but Sam felt alive. He had finally found his true calling. As they emerged from the building, carrying an unconscious woman to safety, Sam knew that he had made the right decision. He was where he belonged, fighting fires and saving lives. The adrenaline rush was like nothing he had ever experienced before. He couldn't wait for the next call.

D.2. Bird vs turtle

Once there was a golden bird that lived in the tops of the mountains.
Once there was a great jade turtle that lived in the swamp.

min
It was so beautiful that everyone who saw it fell in love with it. But the people who lived near the great, shining, beautiful creature were very poor. They were farmers, and they had no money to buy food or clothing or even a roof over their heads. They were always hungry and cold and tired. They watched the beautiful, shining creature and wished that it could help them. But the creature was far away, and it seemed as if it didn't even know that the people were there. So the people began to talk. They said, "If only we could catch that beautiful creature, we would be rich and happy forever." And they made plans to catch the creature. They talked about it all the time, and they dreamed about it at night. They were so busy dreaming and talking that they forgot to work in their fields. And because they didn't work, they didn't have any food to eat. Soon they were

max
He was very old and wise and had seen many things. One day, he decided to take a nap in the sun. As he slept, he dreamed of a beautiful garden filled with flowers and trees. In his dream, he saw a young girl walking through the garden, picking flowers and singing songs. She was so happy and peaceful that the turtle wanted to meet her. So, he flew down from the sky and landed in front of her. The girl was surprised to see a talking turtle, but she wasn't afraid. They talked for hours about the beauty of nature and the importance of peace. When the turtle woke up, he realized that his dream was more than just a dream. It was a message from the universe to spread love and kindness wherever he went. And so, the turtle continued his journey, sharing his wisdom and spreading joy to all who crossed his path. The end.

mean
It was so beautiful that it was known throughout the world. People came from far and wide to see it, and they would always leave offerings of food and jewels at its feet. But the jade turtle was not happy. It longed to fly like the golden bird, to soar through the sky and see the world from above. So it prayed to the gods for wings. And the gods, seeing the turtle's longing, granted its wish. They gave it a pair of magnificent wings, made of gold and jade, that shone brilliantly in the sun. The jade turtle was overjoyed. It spread its wings and flew up into the sky, soaring higher and higher until it could see the whole world spread out before it. It flew over mountains and valleys, rivers and forests, cities and villages. It saw people going about their daily lives, and it felt a deep

log mean
It was so beautiful that it was called the Golden Eagle. One day, the Golden Eagle saw a reflection of itself in the water and thought it was another bird. It flew down to fight the intruder, but when it got closer, it realized it was only a reflection.

The Golden Eagle felt foolish and decided to go on a journey to find its true self. It flew over mountains and valleys, across rivers and seas, until it came to a small village. In the village, there was an old man who was known for his wisdom. The Golden Eagle asked the old man if he could help it find its true self.

The old man smiled and said, "Yes, I can help you. But first, you must learn to accept yourself as you are. Only then can you find your true self." The Golden Eagle nodded and began to meditate on the old man's words.

After some

alternating
He was very beautiful and his shell shone like a star. He was so old that no one knew how old he

was. He was older than the oldest tree, older than the oldest river, older than the oldest mountain. He was so old that he remembered the beginning of time. He remembered when the world was young and the animals could talk. He remembered when the stars were born and the moon was just a baby. He remembered when the sun was a little girl who wore a beautiful golden dress. He remembered when the clouds were made of cotton candy and the rainbows were made of gold. He remembered when the wind was a friendly giant who would play hide-and-seek with the trees. He remembered when the earth was soft and new, like a baby's skin. He remembered when the oceans were born and the fishes started to swim. He remembered when the birds learned to fly and the bees learned to make honey. He

Leveraging Large Language Models for Spell-Generation in Dungeons & Dragons

Elio Musacchio, Lucia Siciliani, Pierpaolo Basile, Giovanni Semeraro
Department of Computer Science, University of Bari Aldo Moro (Bari, Italy)
elio.musacchio@phd.unipi.it, {name.surname}@uniba.it

Abstract

Dungeons&Dragons (*D&D*) is a classic tabletop game with a 50-year history. Its intricate and customizable gameplay allows players to create endless worlds and stories. Due to the highly narrative component of this game, *D&D* and many other interactive games represent a challenging setting for the Natural Language Generation (NLG) capabilities of LLMs. This paper explores using LLMs to generate new spells, which are one of the most captivating aspects of *D&D* gameplay. Due to the scarcity of resources available for such a specific task, we build a dataset of 3,259 instances by combining official and fan-made *D&D* spells. We considered several LLMs in generating spells, which underwent a quantitative and qualitative evaluation. Metrics including BLEU and BertScore were computed for quantitative assessments. Subsequently, we also conducted an in-vivo evaluation with a survey involving *D&D* players, which could assess the quality of the generated spells as well as their adherence to the rules. Furthermore, the paper emphasizes the open-sourcing of all models, datasets, and findings, aiming to catalyze further research on this topic.

Keywords: Generative Artificial Intelligence, Large Language Model, Text Generation, Dungeons&Dragons

1. Introduction

In tabletop role-playing games, *Dungeons & Dragons* (*D&D*) is a timeless classic, captivating players with its immersive storytelling, strategic gameplay, and boundless possibilities. Central to the experience of *D&D* is the use of spells, which enable players to wield magical forces, shape reality, and overcome challenges within the game world.

D&D operates within a framework of rules, facilitating structured gameplay while allowing for creativity and improvisation. Players assume the roles of characters with distinct abilities, embarking on adventures guided by a Dungeon Master (DM) who orchestrates the narrative and adjudicates the rules. One of the most integral aspects of character abilities in *D&D* is casting spells, which encompass a vast array of effects ranging from elemental manipulation to healing and illusion.

Traditionally, spell-casting in *D&D* has relied on predefined lists of spells published in rule books, with players selecting spells for their characters based on predefined criteria such as character class, level, and available spell slots. However, creating new spells or expanding the existing repertoire has mainly been relegated to game designers or enthusiasts, often requiring extensive manual effort and expertise.

In recent years, the emergence of Large Language Models (LLMs) powered by artificial intelligence has revolutionized various domains, including content generation and creative writing. These models are trained on vast corpora of text data and demonstrate remarkable capabilities in understanding and generating human-like text.

This paper explores the potential of leveraging open-source LLMs to generate spells in *D&D*, aiming to augment the creative possibilities within the game. By harnessing the generative power of LLMs, players and game designers can unlock a wealth of spell variations, improving the game with greater depth, novelty, and customization options. In particular, our work proposes a methodology for integrating LLMs into the spell generation process. We investigate the feasibility of automating spell creation by fine-tuning recent open-source LLMs on a dataset of spells manually generated by *D&D* enthusiasts. Furthermore, we explore the implications of employing LLMs for spell generation in *D&D*, including the quality of generated spells and the change in performance by varying the number of model parameters. Ultimately, this paper contributes to the intersection of artificial intelligence and tabletop gaming, demonstrating how advanced language models can enrich the creative processes inherent in games like *D&D*, fostering innovation in gaming communities.

The paper is structured as follows: Section 2 provides an overview of related work, while the methodology is deeply described in Section 3. The evaluation and results are discussed in Section 4.

2. Related Work

Large Language Models (LLMs), built upon the Transformer architecture (Vaswani et al., 2017), undergo extensive training processes facilitated by immense datasets. Their sheer magnitude enables them to encapsulate and process many linguistic

Games and NLP @LREC-COLING-2024, pages 61–69
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

patterns and structures. Notably, LLMs excel not only in mastering downstream tasks but also in their ability to generate text, marking a significant milestone in the field of Natural Language Processing.

Creating narratives for games is of particular interest since it not only requires text coherence but also a more complex structure of the generated story. More specifically, for tabletop games like *D&D*, there are many aspects to take into account, e.g. the setting of the story, the characters, and the specific game state, thus requiring the system to expose Language Generation, Language understanding and planning abilities (Callison-Burch et al., 2022). Moreover, while LLMs have proved to be able to generalize across tasks for which they were not directly trained (Wei et al., 2021; Kojima et al., 2022), they still need huge corpora for the fine-tuning step. This represents an additional challenge within the *D&D* realm: in fact, despite being a game boasting 50 years of history since its first publication and counting an enormous amount of information online, there is a limited number of well-structured datasets related to this topic.

Due to the complexity of *D&D*, many researchers have approached this problem along different lines of research. For example, in (Callison-Burch et al., 2022), the authors focus on the dialogue-based nature of the game given its turn-based system. They fine-tune an LLM, i.e. the 64B LaMDA language model (Thoppilan et al., 2022), on data crawled from *D&D* Beyond Play-by-Post[1]. In (Rameshkumar and Bailey, 2020), the authors focus on text summarization techniques applied to *D&D*. They also built the Critical Role *Dungeons & Dragons* Dataset (CRD3), composed of dialogues extracted from a show called Critical Role, along with abstractive summaries extracted from the Critical Role Fandom wiki. The FIREBALL dataset (Zhu et al., 2023), was instead obtained by collecting 25,000 *D&D* sessions with information about the game state and then used to evaluate the ability of LLMs (i.e. GPT3) in predicting the next game command, given the utterances from the last turns.

Another relevant aspect in any *D&D* campaign is represented by spells, which are magical incantations or formulae that characters can cast to achieve various effects within the game world. Spells are a fundamental aspect of gameplay, allowing players to shape the course of their adventures by adding depth, strategy, and excitement to their story. Regarding works more similar to our contribution, we firstly refer to the work proposed in (Newman and Liu, 2022). Here, the authors focus on spell generation, using three different models: one based on a simple N-Gram model, one based on LSTM (Hochreiter and Schmidhuber, 1997), and the final one based on GPT-2 (Radford et al., 2019). The authors make use of a dataset composed of 3,062 spells, which is obtained by combining the Kaggle dnd-spells dataset[2] and player-made spells from the *D&D* wiki[3]. In detail, the authors use 3,012 randomly chosen spells from this dataset as training data, while the other 50 represent the test data. Results obtained using the BLEU and BERTScore evaluation metrics show that GPT-2 is the best-performing model. Given the high rate with which new LLMs are developed and released, we wanted to explore how newer models can perform the spell generation task.

3. Methodology

As discussed previously, we aim to study the effectiveness of using LLMs for *D&D* spell generation.

More specifically, we are interested in assessing the performance of newer LLMs available at the state-of-the-art and checking how much the number of parameters of the model affects the generation process. This aspect is particularly interesting since the available data for this task is limited.

In fact, even if there are several datasets that have been publicly released about *D&D*, the amount of official *D&D* spells is not even in the thousands. Because of this, we are interested in analyzing how well the same type of model architecture performs with respect to the number of parameters and how well it maintains the generalization capabilities for which LLMs are known.

It is important to note that, throughout this work, we employ a pipeline similar to the one used in (Newman and Liu, 2022), which used GPT-2 for this task. This choice was made to allow for comparison with those already available at the state-of-the-art and to glean insights into the effectiveness and performance of our methodologies. Finally, to the best of our knowledge, the source code and final models adopted in (Newman and Liu, 2022) are not publicly available. In contrast, we have made all our code [4], models [5], and dataset [6] available. We firmly believe in transparency and accessibility in research, especially in complex and challenging domains.

3.1. Dataset

For the dataset, we follow and extend the process described in (Newman and Liu, 2022). First, we

---

[1] `https://www.dndbeyond.com/forums/d-d-beyond-general/play-by-post`

[2] `https://www.kaggle.com/datasets/mrpantherson/dndspells`

[3] `https://www.dandwiki.com/wiki/`

[4] GitHub Repository

[5] HuggingFace Collection

[6] Dataset

download the dnd-spells dataset (introduced in section 2) from Kaggle containing all *D&D* 5th-edition spells and obtain a total of 554 official spells. Each spell in this dataset is characterized by a total of 12 fields, summarized in Table 1.

Field	**Description**
name	Name of the spell
classes	Classes that can learn the spell
level	The level of the spell (this is separate from the player's level), 0 is a cantrip
school	What type of magic the spell is
cast_time	How long it takes the character to cast the spell
range	How far the character can be from the target of the spell
duration	How long the spell lasts
verbal	Boolean - spell requires a verbal incantation
somatic	Boolean - spell requires a precise hand motion
material	Boolean - spell requires a physical object which is consumed
material_cost	What type of material is consumed when casting the spell
description	The effect casting spell creates

Table 1: Fields in the *D&D* spell dataset.

To further increase the amount of data, we enhance this first set by scraping the "Spell" section of the *D&D* Wiki site[7], which contains community-made content. We retrieve all data for all spells of the main section, while ignoring spells that fall under the categories *"April Fool's Spells"* and *"Incomplete Spells"*, which are more likely to be not fully described or not following the rules. The complete list of spells also contains official *D&D* spells, but the page associated to such spells contains a copyright disclaimer instead of the data, in such cases we directly skip the page. After scraping, we obtain an additional dataset consisting of $3,287$ spells. However, analysis of the dataset revealed that there were many formatting and data quality issues with these spells. Therefore, we perform both a filtering and a pre-processing step to guarantee that the quality of the spells of this additional set matches the official ones. The filtering steps are the following:

- Removed spells instances not having one of the following attributes: *level*, *school*, *duration*, *casting time*, *range*;
- Removed spells instances which did not properly contain the required *components* to cast the spell. In fact, each instance should explicitly state whether the spell requires *Verbal*, *Somatic* or *Material* (in the case of material, we extract the required components). All of this is done through a regex matching operation;
- Each spell is also associated with a list of character classes that can actually learn the spell. Since the content-made spells may also contain content-made classes, we decided to remove the classes that referred to a fan-made class for each spell. The list of official classes was retrieved from the official *D&D* Beyond site[8]. If a spell did not have any classes left after filtering the list, the spell instance was removed.

As part of the spell processing pipeline, we employ the following steps:

- We identify instances where the scraper output contains the number of votes a spell has received on the wiki. This information is irrelevant to our analysis and can potentially distort our results. Therefore, we use a regex matching operation to remove this information;
- When extracting information about spells from the wiki, the level of the spell and the school of magic it belongs to are often combined into a single string of text (e.g. *"9th-level necromancy"*). In order to make use of this information, we need to separate the level and school into their own distinct values;
- We extract the materials needed by spells with a *Material* ("M") as one of their components. This is done by using a regex matching operation and assuming that such information is between brackets.

After filtering and processing the dataset, we gathered a total of 2,705 instances, which were then merged with the Kaggle dataset to obtain our final dataset of 3,259 instances. Finally, in contrast to the approach presented in (Newman and Liu, 2022), where the authors selectively retain a subset of spell features, focusing primarily on elements deemed crucial for the task, such as *name* and *description*, our methodology encompasses all features extracted through the scraping process. The rationale behind this decision is that the LLM should be able to learn the patterns underlying the values for these features. For instance, it should recognize that the presence of a chant in a spell's description correlates with the "Verbal" component. Thus, we decided to leverage all the features available within the dnd-spells dataset. This approach allows the LLM to capture and learn the relationships that can appear among different spell attributes.

---

[7] `https://www.dandwiki.com/wiki/5e_All_Spells`

[8] `https://www.dndbeyond.com/classes`

3.2. Models

For our analysis, we focused on using decoder-only families of models. In particular, the models taken into account for our evaluation are the following:

- **GPT-2** (Radford et al., 2019) *Generative Pre-Trained Transformer*: released by OpenAI in 2019. We use this family of models as a baseline against which we measure our experimental outcomes. This decision not only allows for a comparison of our results but also facilitates the reproducibility of findings established in prior research, such as those documented in (Newman and Liu, 2022). The models chosen for the evaluation are: `gpt2`, `gpt2-medium`, `gpt-large`, and `gpt-xl`;
- **OPT** (Zhang et al., 2022) *Open Pre-Trained Transformer Language Models*: released by Meta AI in 2022, the main appeal of this family of models is that, to the best of our knowledge, this is the only family of models providing many pre-trained checkpoints having different number of parameters (e.g. 125m, 350m, 2.7b, 6.7b, ...). This is very useful for our purposes, as many of the modern models are released with only a limited selection of parameter counts. The models chosen for the evaluation are: `opt-125m`, `opt-350m`, `opt-1.3b`, `opt-2.7b`, `opt-6.7b`, and `opt-13b`;
- **LLaMA 2** (Touvron et al., 2023) *Large Language Model Meta AI*: released by Meta AI in 2023, the available checkpoints have 7b, 13b and 70b parameters. We also consider these models to compare the performance of OPT and GPT-2 (a relatively older family of models) to the performance obtained by these more powerful and recent models. The models chosen for the evaluation are: `llama2-7b` and `llama2-13b`.

3.3. Training details

For the train-test split, we adhere to the methodology outlined in (Newman and Liu, 2022). Specifically, we employ random sampling to extract 50 instances from the complete dataset, which are then designated as the test set, using the rest as the train set.

We perform full-parameter tuning using DeepSpeed ZeRO 3 [9] for parallelism. We train the models using from 1 to 2 nodes equipped with 4 64GB NVIDIA A100 GPUs. The training procedure was performed with a fixed number of 3 epochs over the entire train set as proposed in (Newman and Liu, 2022). We used the following hyperparameters when they were not specified in the original work:

- **maximum sequence length**: 512; each model has a different maximum length. This value was chosen after tokenizing the dataset with each tokenizer and studying the number of tokens required for each spell to avoid truncation
- **learning rate**: 0.00005
- **effective batch size**: 128
- **optimizer**: adamw (torch)
- **packing**: true
- **weight decay**: 0.0001
- **warmup ration**: 0.0
- **seed**: 42

Furthermore, we format the input data following the Alpaca instruction format (Taori et al., 2023) and perform instruction-tuning with a pre-defined instruction: *"Write a spell for the 5th edition of the Dungeons & Dragons game."*. The idea is that the models could be then further fine-tuned for *D&D* related tasks, such as character or scenario creation. An example of a formatted train instance is shown in Table 2.

Below is an instruction that describes a task. Write a response that appropriately completes the request.
### Instruction:
Write a spell for the 5th edition of the Dungeons & Dragons game.
### Response:
Name: Light Blood Transfusion
Level: 1
School: Necromancy
Classes: Sorcerer, Warlock, Wizard
Casting time: 1 Action
Range: Touch
Duration: Instantaneous
Components: V, S
Description: You lay your hand upon a creature, transfering your blood to the target. You spend one or more hit dice, and the touched creature regain hit points equal to number rolled in the hit dice + your spellcasting ability. If the creature is poisoned, you can use a hit die to end the condition on the target, instead of restoring its hit points.

Table 2: Example of prompt.

4. Evaluation

As presented in Section 3.3, the test set consists of 50 instances. Therefore, we perform a quantitative evaluation procedure and also a qualitative one with actual *D&D* players.

---

[9] `https://www.deepspeed.ai/`

4.1. Quantitative Evaluation

For quantitative evaluation, we refer to (Newman and Liu, 2022) and use Bleu (Papineni et al., 2002) and BertScore (Zhang* et al., 2020) as evaluation metrics.

Bleu is a widely used metric to evaluate the quality of text generated by LLMs, although it was originally conceived to measure the quality of Machine Translation models. The Bleu score measures the alignment of n-grams between the candidate translation and the reference translation, counting the number of matched n-grams to ascertain translation quality. A higher number of matches typically signifies a superior candidate translation.

BertScore is a powerful metric that is used as an evaluation tool for natural language generation. It is based on the pre-trained BERT contextual embeddings, which are highly effective in capturing complex language structures. Unlike traditional metrics based on n-grams (e.g. the aforementioned Bleu) that can be limited in their ability to capture long-range dependencies, BertScore computes similarity based on contextualized token embeddings. This allows it to effectively capture distant dependencies and similarities, thus producing a more accurate evaluation.

For both metrics, we use the implementation provided in the “evaluate” library by the HuggingFace team [10]. We adopt these two metrics since they are the same ones the authors of (Newman and Liu, 2022) used in their experimental setting. To compute both metrics, a reference sentence must be provided. While (Newman and Liu, 2022) considers the first 40 tokens of the entire spell (therefore starting from the “Name” attribute), we decided to keep 20 tokens after the “Description” attribute, which is the last tag of the entire spell. Our choice is motivated by the assumption that the description is the central aspect to evaluate since it is the part of the spell that contains more text.

Finally, we do not use tokens for the splitting since we are comparing different models and tokenizers. Therefore, we split based on words, keeping the first 20 words as the reference sentence. Words are identified by splitting based on multiple white spaces and using some heuristics to account for punctuation. Results of the quantitative evaluation in terms of Bleu and BertScore are summarized in Table 3.

Analyzing the results, we observe that the best performance are obtained by models with the largest number of parameters (i.e. `llama2-13b` and `opt-13b`). An exception is represented by `llama2-7b` which achieves the best BertScore (recall), even though the differences are quite narrow. We also notice that the Bleu significantly increases with the number of parameters, while BertScore remains relatively unaffected by the model. Therefore, all models achieve outstanding results in terms of BertScore, thus proving that the generated text is semantically correct. However, models with a large number of parameters tend to produce more long n-gram sequences matching the spell description in the test set causing overfitting.

4.2. Qualitative Evaluation

For qualitative evaluation, we set up an experiment involving *D&D* players. To facilitate this investigation, we developed a Telegram chatbot, which was the primary interface for engaging with participants. This allowed us to interview players and easily keep track of their answers. The chatbot is designed to show the player one spell at a time, with 10 spells for the whole session. Additionally, players retain the freedom to interrupt the questionnaire at any point and resume their progress later, as their responses are continuously saved throughout the interaction process. Users could also decide to perform multiple experimental sessions, in which case we ensured that no spell seen by a user in a previous experiment could be seen again in a new one.

For each spell, following the questionnaire proposed in (Newman and Liu, 2022), there are three questions asked to the player. The questions, along with the kind of expected answers, are shown in Table 5.

For the qualitative evaluation, we consider the five best-performing models: `opt-2.7b`, `opt-6.7b`, `opt-13b`, `llama2-7b` and `llama2-13b`. During each session, we present the user with ten spells: one from each model and five written by humans. The spells are randomly selected, which ensures that the user is equally likely to encounter a human-written spell and a machine-generated one. For the generation, we employed Top-p Sampling (also known as “Nucleus Sampling”) (Holtzman et al., 2019) with $p = 0.9$ (a commonly used value, in particular, (DeLucia et al., 2020) study that a value in the range $[0.7, 0.9]$ is best in narrative generation). During the experiment via the Telegram bot, the users are not informed of the total number of AI-generated and human-written spells. This omission is aimed at maintaining an unbiased environment throughout the whole experiment.

A total of 13 users completed at least one experiment session for a total of 16 sessions. The overall results can be seen in Table 4, while the results per model can be seen in Table 6.

Table 4 shows that 73% of AI-generated spells are identified by the users, but at the same time, 35% of human-written spells are recognized as written by the AI. Furthermore, the table also provides

---

[10] https://github.com/huggingface/evaluate

Model	BLEU	BertScore (Precision)	BertScore (Recall)	BertScore (F1)
gpt2	0.093	0.835	0.851	0.842
gpt2-medium	0.121	0.858	0.855	0.856
gpt2-large	0.110	0.861	0.857	0.858
gpt2-xl	0.149	0.862	0.856	0.858
opt-125m	0.082	0.850	0.845	0.846
opt-350m	0.113	0.860	0.852	0.856
opt-1.3b	0.090	0.861	0.854	0.857
opt-2.7b	0.123	0.870	0.862	0.866
opt-6.7b	0.097	0.867	0.860	0.863
opt-13b	0.135	0.870	0.864	0.867
llama2-7b	0.175	0.877	0.874	0.875
llama2-13b	**0.188**	**0.880**	**0.876**	**0.877**

Table 3: Results of the quantitative evaluation.

	Human	*AI*
Correctly Identified	66%	71%
Average Rule Conformity	3.75 (4.20)	2.88 (2.50)
Average Playability	3.49 (3.87)	2.76 (2.33)

Table 4: Overall results of the qualitative evaluation. The Human column refers to human-written spells, while the other one to AI-generated spells.

QUESTION 1 What do you think made this? **ANSWER 1** - Human - AI - I have already seen this spell, I know it was written by a human
QUESTION 2 How well do you think this spell conforms to *D&D*'s rules? **ANSWER 2** 5-point Likert scale (1 = "Doesn't fit with the rules at all", 5 = "Would fit in right alongside official spells")
QUESTION 3 Would you play/allow this spell? **ANSWER 3** 5-point Likert scale (1 = "Definitely wouldn't", 5 = "Definitely would")

Table 5: Questionnaire of the Telegram Chatbot. The user is asked to answer all three questions for each spell.

the results (in brackets) computed only on the instances correctly classified as human-written or AI-generated by users. These results show that when spells written by humans are correctly classified by users, the "Average Rule Conformity" and "Average Playability" both increase. In contrast, in the case of AI-generated spells, the two scores slightly decrease.

Analyzing the qualitative results for each model, we obtain that the best models are `opt-2.7b` and `llama2-13b`, while the worst model is `opt-6.7b` as a lower value of "Correctly Identified" means that humans are not able to distinguish AI written spells from human ones. These results confirm the quantitative evaluation where the `opt-2.7b` overcomes `opt-6.7b`. Also the average rule conformity follows the behaviour of the correctness. Furthermore, the "Average Playability" confirms that `opt-6.7b` is the worst performing model, while the other models obtain similar results. Considering the results in brackets, only `opt-2.7b` provides a satisfying "Average Rule Conformity" score, while other systems are under the average.

Table 7 to Table 11 show examples of spells generated by each of the best models, which were also the subject of the qualitative evaluation.

5. Conclusions

In this paper, we presented our exploratory study on using LLMs for a Natural Language Generation task within the domain of tabletop games. More specifically, we focus on *Dungeons & Dragons*, one of the most popular tabletop games in the world, which was published 50 years ago. *D&D* is a complex game that allows players to create their own worlds and characters.

For this work, we decided to focus our attention on a specific aspect of the game: spells. Spells are very interesting since they can affect single characters or fiends but also completely change the outcomes of a story.

We took into account a total number of 12 models, subdivided into three main families: `gpt2`, `opt`, and `LLaMA2` and evaluated their performance both

	opt-2.7b	*opt-6.7b*	*opt-13b*	*llama-2-7b*	*llama-2-13b*
Correctly Identified	63%	75%	75%	75%	69%
Average Rule Conformity	3.13 (3.1)	2.94 (2.67)	2.56 (2.25)	2.5 (2.00)	3.25 (2.64)
Average Playability	2.81 (2.4)	2.63 (2.17)	2.56 (2.17)	2.88 (2.58)	2.88 (2.36)

Table 6: Per-model results of the qualitative evaluation

Name: Disintegrate
Level: 2
School: Abjuration
Classes: Sorcerer, Wizard
Casting time: 1 Action
Range: 60 Feet
Duration: Instantaneous
Components: S
Description: You disintegrate a creature of your choice within range, causing it to disintegrate into fine mist. Each creature in a 10-foot cube centered on the target must succeed on a Dexterity saving throw or take 3d6 force damage and be stunned until the end of your next turn.
If you target a creature that you can see and have line of sight with, they do not have to make the saving throw.

Table 7: Example of spell generated by the `opt-2.7b` model.

Name: Sunfire Blow
Level: 1
School: Evocation
Classes: Sorcerer, Wizard
Casting time: 1 Action
Range: Self (30-Foot Cone)
Duration: Instantaneous
Components: S
Description: You hurl a magical punch, causing a pillar of fiery light to erupt from your hand. Each creature in a 30-foot cone must make a Constitution saving throw. A target takes 8d6 fire damage on a failed save, or half as much on a successful one. If a creature starts its turn in the area of the cone, or ends its turn in the area of the cone, it must make the saving throw with disadvantage.
At Higher Levels. When you cast this spell using a spell slot of 2nd level or higher, the damage increases by 1d6 for each slot level above 1st.

Table 8: Example of spell generated by the `opt-6.7b` model.

Name: Wall of Force
Level: 0
School: Evocation
Classes: Wizard
Casting time: 1 Action
Range: 60 Feet
Duration: Concentration, up to 1 hour
Components: V, S
Description: You and up to five other creatures of your choice are surrounded by magical force, up to 1 foot thick. Creatures inside the wall can only be attacked by spells or creatures that can cross the wall (such as by teleportation or a nonmagical teleporter).
While the wall is in effect, any weapon attack is made at advantage, and the target automatically passes the Dexterity saving throw it takes to avoid taking damage from said attack. Creatures also have advantage on the Strength check it takes to damage this wall.

Table 9: Example of spell generated by the `opt-13b` model.

Name: Healing Mist
Level: 3
School: Necromancy
Classes: Bard, Cleric, Paladin
Casting time: 1 Action
Range: 150 Feet
Duration: Concentration, up to 1 minute
Components: V, S
Description: A mist infused with divine magic envelops a creature of your choice that you can see within range. The target regains 3d6 hit points upon the mist being cast, and also regains 1d6 hit points each round that you maintain concentration. Upon the spell ending, the target takes 2d6 necrotic damage.
At Higher Levels. When you cast this spell using a spell slot of 4th level or higher, the hit points regained by the target increases by 1d6 for every level above 3rd.

Table 10: Example of spell generated by the `llama-2 7b` model.

in a quantitative and qualitative evaluation.

For the quantitative evaluation, we compute the Bleu and the BertScore. From the analysis of the results, we can confirm that the number of parameters positively affects the results obtained by the models, especially in terms of the Bleu score. Next, we took into account the best five performing models to conduct an in-vivo evaluation with actual *D&D* players. The results of this qualitative evaluation confirm the results of the quantitative evaluation.

Finally, we also publicly available all the models, datasets, and results of this work, believing that this will foster further research on this topic.

In future works, we plan to extend the use of LLMs to other aspects of the game, such as dialogue continuation or the simulation of other game mechanics. We plan to enhance the capabilities of our models by refining them for additional tasks through fine-tuning. Moreover, we will replicate the pipeline to other open-source models and improve

Name: Shield of Sand
Level: 3
School: Transmutation
Classes: Druid, Sorcerer, Warlock, Wizard
Casting time: 1 Action
Range: Touch
Duration: Concentration, Up To 1 Hour
Components: V, S, M
Material cost: a handful of sand
Description: You create a shield of swirling sand in the palm of your hand. The shield is a dull gray in color, and has an AC of 15 and 15 hit points. The shield disappears if you let go of it. If the shield is reduced to 0 hit points, it disintegrates and the creature must make a Dexterity saving throw. On a failed save, the creature takes 5d6 bludgeoning damage and is blinded until the start of its next turn.

At Higher Levels. When you cast this spell using a spell slot of 4th level or higher, the spell's duration increases to 8 hours. When you cast this spell using a spell slot of 6th level or higher, the spell's duration increases to 24 hours. When you cast this spell using a spell slot of 8th level or higher, the spell's duration increases to 1 week

Table 11: Example of spell generated by the `llama-2 13b` model.

the qualitative evaluation by involving more users.

Acknowledgment

We acknowledge the support of the PNRR project FAIR - Future AI Research (PE00000013), Spoke 6 - Symbiotic AI (CUP H97G22000210007) under the NRRP MUR program funded by the NextGenerationEU.

6. Bibliographical References

Chris Callison-Burch, Gaurav Singh Tomar, Lara J Martin, Daphne Ippolito, Suma Bailis, and David Reitter. 2022. Dungeons and dragons as a dialog challenge for artificial intelligence. *arXiv preprint arXiv:2210.07109*.

Alexandra DeLucia, Aaron Mueller, Xiang Lisa Li, and João Sedoc. 2020. Decoding methods for neural narrative generation. *arXiv preprint arXiv:2010.07375*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Pax Newman and Yudong Liu. 2022. Generating descriptive and rules-adhering spells for dungeons & dragons fifth edition. In *Proceedings of the 9th Workshop on Games and Natural Language Processing within the 13th Language Resources and Evaluation Conference*, pages 54–60, Marseille, France. European Language Resources Association.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. pages 311–318.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Revanth Rameshkumar and Peter Bailey. 2020. Storytelling with dialogue: A critical role dungeons and dragons dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5121–5134.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Andrew Zhu, Karmanya Aggarwal, Alexander Feng, Lara J Martin, and Chris Callison-Burch. 2023. Fireball: A dataset of dungeons and dragons actual-play with structured game state information. *arXiv preprint arXiv:2305.01528*.

Branching Narratives: Character Decision Points Detection

Alexey Tikhonov
Inworld.AI
Berlin, Germany
altsoph@gmail.com

Abstract

This paper presents the Character Decision Points Detection (CHADPOD) task, a task of identification of points within narratives where characters make decisions that may significantly influence the story's direction. We propose a novel dataset based on Choose Your Own Adventure (a registered trademark of Chooseco LLC) games graphs to be used as a benchmark for such a task. We provide a comparative analysis of different models' performance on this task, including a couple of LLMs and several MLMs as baselines, achieving up to 89% accuracy. This underscores the complexity of narrative analysis, showing the challenges associated with understanding character-driven story dynamics. Additionally, we show how such a model can be applied to the existing text to produce linear segments divided by potential branching points, demonstrating the practical application of our findings in narrative analysis.

Keywords: NLP, narrative analysis, CYOA, agency

1. Introduction

Modern Large Language Models (LLMs) are state-of-the-art in a lot of Natural Language Processing (NLP) tasks. However, areas related to the analysis and generation of texts with complex and rich semantic structures remain underexplored. This includes the tasks of analyzing and generating long, engaging, and rich narratives (van Stegeren and Theune, 2019). While modern models can sometimes produce innovative plot twists, they generally create less imaginative scenarios and rhetoric compared to human-authored texts (Begus, 2023).

The traditional machine learning approach to this problem starts from the data collection with necessary annotations. In the narrative analysis field, there are a number of datasets available, such as WikiPlots[1] with 112,936 story plots extracted from English Wikipedia, the MPST dataset with 14K movie plot synopses (Kar et al., 2018), and the DYPLODOC dataset, which includes synopses of 13K TV shows, 21K seasons, and over 300K episodes (Malysheva et al., 2021). However, these plain-text synopses offer limited assistance when the goal is to analyze high-level narrative structures.

Philosophers and linguists make a lot of attempts to conceptualize and formalize concepts of plot, narrative arcs, character development, conflict, and so on (Shklovsky, 1925). One of the fundamental principles in drama and narrative construction is the concept of character agency, which posits that a character's decisions and actions drive the plot forward. Aristotle, in his work *Poetics*, highlights sudden plot twists, or *peripeteia*, especially those tied to *anagnorisis*—the moment when a character comes to a significant realization or discovery that affects subsequent choices. Borges, in his short story *The Garden of Forking Paths*, explores the idea of multiple possible worlds through the metaphor of a labyrinth, representing an infinite number of potential narratives and outcomes based on characters' actions: "*your ancestor <...> believed in an infinite series of times, in a growing, dizzying net of divergent, convergent and parallel times. This web of time—the strands of which approach one another, bifurcate, intersect or ignore each other through the centuries—embrace every possibility*." Propp introduces the concept of functions—recurring, typical actions that move the narrative forward: "*a tale often attributes identical actions to various personages; this makes possible the study of the tale according to the functions of its dramatis personae <...> a function <...> cannot be defined apart from its place in the course of narration*". Gustav Freytag, in (Freytag and MacEwan, 1968), describes *Freytag's Pyramid*, a typical plot structure identifying five pivotal plot points: Opportunity, Change of Plans, Point of No Return, Major Setback, Climax. Aarseth in his *Cybertext* book proposes the term *ergodic literature* to define open, dynamic texts, with which the reader must perform specific actions to generate a literary sequence.

One may argue that we still cannot clearly define what we aim to analyze, and this slows progress in the analysis and generation of narrative structures (Yamshchikov and Tikhonov, 2023). However, the NLP community continues to seek improvements in narrative processing (Fan et al., 2019), by setting subtasks for the formal identification of important plot elements. For instance, in (Tikhonov and Yamshchikov, 2022) the task is to identify "Chekhov's guns"—narrative objects that significantly impact the plot's development; (Papalampidi et al., 2019) introduce a Turning Point Identifica-

---

[1] https://github.com/markriedl/WikiPlots

20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

tion task—to directly identify Freytag's points in the text, and (Li et al., 2023) proposes a task to extract action models from narrative texts automatically.

In this paper, we propose using characters' decision-making moments to analyze and formalize narrative structure. We introduce a new NLP task—Character Decision Points Detection (CHADPOD). This task focuses on identifying moments in the narrative where characters make decisions that significantly determine the plot's direction. We believe that highlighting such moments will improve our understanding of traditional text plots and open possibilities for working with nonlinear and interactive narrative structures (Juul, 2005; Murray, 2006).

This work contributes by:

1. Proposing a formalization of the Character Decision Points Detection (CHADPOD) task.
2. Introducing a Character Decision Points dataset.
3. Demonstrating the effectiveness of modern models in identifying Character Decision Points (CDPs).
4. Offering an interpretation of CDPs and their relation to the related task of turning points.

2. CHADPOD task

In NLP research, analysts and creators frequently utilize Gamebook genre games, also widely known as Choose Your Own Adventure[2] (CYOA) books, named after one of the earliest popular series in this genre. These sources are crucial for studying nonlinear narratives, alongside interactive fiction games.

For instance, the data from such sources has been used to train systems that generate suggestions for people writing short stories (Clark and Smith, 2021). Another study employs CYOA as a medium for training generative agents to enforce temporal constraints (Rothkopf et al., 2024). In the MACHIAVELLI paper (Pan et al., 2023), authors use a collection of CYOA games to create a game environment for training text agents. Some researchers[3] explore them to analyze a variety of narrative macro-structures.

In this work, we introduce the CHADPOD task, which focuses on identifying narrative points where a character makes a choice that determines the further course of the story. We utilize CYOA game graphs to create a new CHADPOD benchmark, consisting of 1,462 binary classification tasks, with 731 tasks in each class. Each task comprises two text segments—a prefix and a postfix. The positive class includes narrative points where a character makes a choice that significantly influences the story's direction. The negative class consists of randomly segmented texts (we take a continuous text from a single node and split it at some random point between sentences), as well as text points where a character takes some action, but it does not significantly affect the story's progression.

3. Data

In this section, we describe the process of constructing the CHADPOD dataset.

We use the MACHIAVELLI dataset (Pan et al., 2023), which consists of 134 Choose-Your-Own-Adventure games, as our input data.

For each available game, we analyze its graph and extract triplets of the form:

$$\langle \mathit{node1};\ \mathit{action};\ \mathit{node2} \rangle$$

where *node1* is the text before the action, *action* is the choice made by the player, and *node2* is the text following the action (see Figure 1).

Next, we filter the triplets—removing exact duplicates, retaining only those with descriptions sufficiently long to provide enough context – both in *node1* and *node2* (to do so we used simple heuristics – at least 4 sentences, at least 50 characters), removing texts that are dialogue segments (dialogues are a very specific type of narrative that should be analyzed separately, see for example (Zhou et al., 2023)), and removing texts with unusual characters. As positive examples of branching points, we only select triplets for which the graph from node1 has more than one possible action, thus excluding scenarios like <*node1*, "1 year later…", *node2*>. The remaining 731 examples make up the positive class.

Then we form the negative class from two components—using the division of texts from the same games (nodes) at random points as easy negatives, and the above-described cases when there is exactly one action emanating from node1 in the graph as hard negatives.

Finally, we divide the data[4] into 3 game-wise splits, ensuring that there are no overlaps between the splits in terms of games, thereby eliminating test set leakage. The statistics of the resulting split are presented in Table 1.

4. Task validation

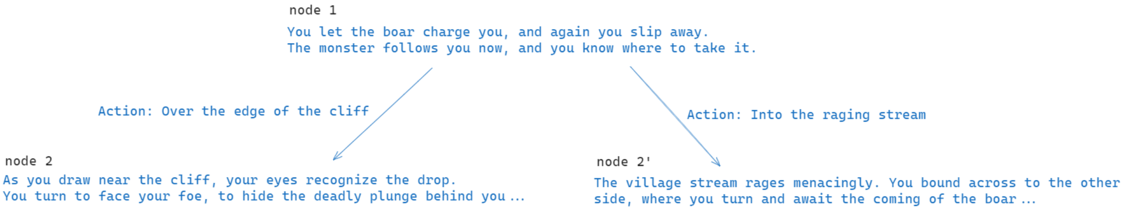
Experiments To validate the usefulness of our dataset, we trained several models for the CHADPOD tasks. We used the DeBERTa model (He

---

[2] It is a registered trademark of Chooseco LLC.

[3] https://heterogenoustasks.wordpress.com/2015/01/26/standard-patterns-in-choice-based-games/

[4] The data is available through Google Drive. The password is CHADPOD.


node 1
Action: Over the edge of the cliff
Action: Into the raging stream
node 2
node 2'

You let the boar charge you, and again you slip away.
The monster follows you now, and you know where to take it.

As you draw near the cliff, your eyes recognize the drop.
You turn to face your foe, to hide the deadly plunge behind you ...

The village stream rages menacingly. You bound across to the other side, where you turn and await the coming of the boar ...

Figure 1: Example of branching in CYOA data, shortened for the simplicity.

Table 1: Data Splits

Class	**Train**	**Dev**	**Test**
Positives	511	110	110
Negatives	256	55	55
Hard Negatives	255	55	55
Total	1022	220	220

et al., 2021) as a strong baseline, known for its state-of-the-art performance in many text classification tasks [5]. Additionally, we included older but widely used models such as BERT (Devlin et al., 2019) and ALBERT (Lan et al., 2020) as weaker baselines. We chose accuracy as the metric due to our data being class-balanced. The training was conducted on a GPU RTX 4090 on the Vast.ai platform, with each model trained with a batch size of 4 and a learning rate of 5.5×10^{-6} until accuracy on a validation split began to decline. A full training run for one task required at most 30 minutes. We also added results for GPT-3.5-turbo[6] and GPT-4-turbo[7] tested in a zero-shot manner with hyperparamteres (temperature, probability threshold) obtained by a grid search on the validation set.

The results are presented in the Table 2.

Table 2: Test Accuracy of Models on CHADPOD

model	**test acc**	**size**
DeBERTa-v3-large	**89%**	340M
DeBERTa-v3-base	85%	110M
ALBERT-v2-base	84%	11M
BERT-base	79%	110M
GPT-4-turbo, 0-shot	62%	unknown
GPT-3.5-turbo, 0-shot	55%	unknown

As seen, the task is solvable but remains quite complex for simpler and smaller models. The presented results are on a test dataset without overlap with the training set in terms of games, minimizing the risk of overfitting. As for LLMs, it seems that using them in a 0-shot manner is not a silver bullet for this task, though results could likely be improved through fine-tuning or prompt engineering. Analysis of the confusion matrix revealed that LLMs underperform on the positive class, leading to a high number of false negatives.

Comparison with Turning Points One may notice that the CHADPOD task is significantly similar to the Turning Points Identification task proposed in (Papalampidi et al., 2019). In this section, we conduct a comparative analysis, demonstrating that despite similar formulations, the tasks differ fundamentally.

Recall that in (Papalampidi et al., 2019), the TRIPOD dataset consists of manually annotated short plot synopses (averaging 35 sentences) of 99 screenplays with sentence-level turning points annotations, where turning points are defined as the 5 classic pivot moments formulated in Freytag's Pyramid.

We transformed the TRIPOD dataset to our format, taking contexts around the indicated turning points as positive examples and random divisions of the same synopses where there were no turning points as negative examples. The final dataset used all available non-overlapping contexts with at least 3 sentences before and after the split point, resulting in 255 positive and 209 negative examples.

Applying our DeBERTa-v3-large based model to these examples yielded the metrics provided in Table 3.

Table 3: Performance on Adapted TRIPOD Dataset

Metric	**Value**
Accuracy	40%
Balanced Accuracy	41%
F1-Score	41%

These results indicate that the semantics of the tasks significantly differ (recall that the model's accuracy on an isolated test set was 89%).

One might suggest that the main difference between these tasks lies in the scale (turning points

---

[5] `https://huggingface.co/altsoph/chadpod`

[6] https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates

[7] https://openai.com/blog/new-models-and-developer-products-announced-at-devday

are just 5 key moments in the plot's macrostructure) and in that Freytag's turning points do not necessarily imply character agency. Contrarily, they can be exclusively formed by external events, leaving characters without a choice.

5. Text Segmentation Study

In this part, we demonstrate how the obtained binary classification model can be used for segmenting text into linear segments separated by potential branching points in the narrative.

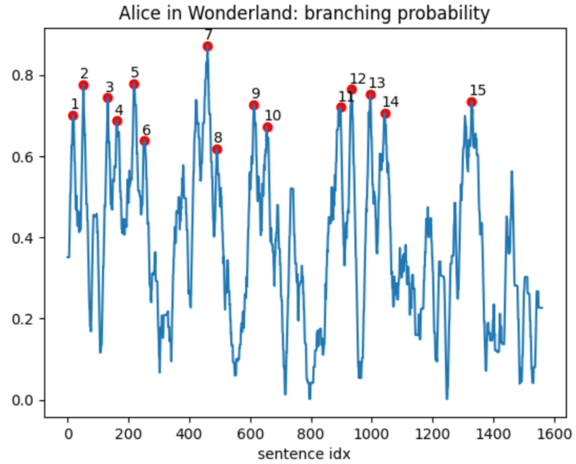


Figure 2: Most probable branching points in the text of Alice in Wonderland.

For our experiments, we utilized the text of *Alice's Adventures in Wonderland* by Lewis Carroll, as it is in the public domain. We employed a sliding window of 10 sentences with a step of 1 sentence and calculated the probability of branching at any given possible point. To reduce noise, we applied a convolution with a linear kernel of width 25, and then on the resulting sequence, we identified local maximums on segments lying above the threshold $TH1 = 0.5$, preserving only peaks above the threshold $TH2 = 0.6$ to obtain 15 main branching points. Indeed, the parameters of such a heuristic can be adjusted to change the sensitivity of the approach.

In the Figure 2 one can see 15 most probable points of branching in the given text. To analyse them and gain understanding if these points really correspond to the important decisions of the chraracter, we used GPT-4-turbo model to assess selected points and propose potential alternatives to the character's action. We refer to the Table 4 for details of these points and alternatives.

Despite the subjectivity of such analysis, it is worth saying that all identified points, except maybe for 1 and 4, correspond to moments when a character makes a decision or performs an action that influences the subsequent development of events.

6. Discussion

This study contributes to the evolving field of Natural Language Processing (NLP) by addressing the nuanced task of detecting Character Decision Points (CDPs) within narrative texts. Through the development and validation of the CHADPOD task, our findings highlight the complexity and potential of leveraging Large Language Models (LLMs) for narrative analysis, particularly in identifying moments of narrative branching that may be important to story development.

The performance of various models on the CHADPOD task, especially the high results of the DeBERTa model, demonstrates the feasibility of detecting narrative branching points with high accuracy. However, the underperformance of smaller and simpler models, as well as zero-shot tests of GPT-3.5 and GPT-4, illustrates the challenges of the task. We suggest these challenges are not solely due to model capacity but also reflect the sophisticated understanding of narrative structure.

The application of our binary classification model to text segmentation, as demonstrated in the analysis of Alice's Adventures in Wonderland, showcases the practical utility of our approach. This illustrative study can be a bit speculative without ground truth labeling, since ChatGPT is able to generate plausible alternatives for any requested point in text. However, subjectively, most of the detected branching points (demonstrated in Table 4) correspond to the turning points of text there the character makes impactful decisions. (this problem can also be approached as a direct segmentation task, as in, for example, (Koshorek et al., 2018); we leave these experiments for future work.)

Our results suggest several directions for future research. First, expanding the dataset to include a broader range of narratives, could enhance the model's understanding of diverse narrative structures. Second, exploring more granular classifications of decision points, such as presented in Syd Field's book *Screenplay* (with 6 key points) or the one based on Vogler's interpretation of Campbell's monomyth (with 12 such points) could offer finer insights into narrative dynamics. Third, using the CHADPOD data can help to construct a macro-assessment of characters' agency within a text, i.e., an assessment that enables comparing different texts in terms of how much the development of the text is determined by the characters' choices.

7. Bibliographical References

E.J. Aarseth. 1997. *Cybertext: perspectives on ergodic literature*. Johns Hopkins Univ Pr.

Aristotle. 2014. *Aristotle collection*. Annotated Classics.

Nina Begus. 2023. Experimental narratives: A comparison of human crowdsourced storytelling and ai storytelling.

Jorge Luis Borges. 1964. *Labyrinths: selected stories & other writings*, volume 186. New Directions Publishing.

Elizabeth Clark and Noah A. Smith. 2021. Choose your own adventure: Paired suggestions in collaborative writing for evaluating story generation models. In *Proceedings of NAACL 2021*, pages 3566–3575, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL 2019*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. *arXiv preprint arXiv:1902.01109*.

G. Freytag and E.J. MacEwan. 1968. *Freytag's Technique of the Drama: An Exposition of Dramatic Composition and Art*. Johnson Reprint.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

Jesper Juul. 2005. Half-real: Video games between real rules and fictional worlds.

Sudipta Kar, Suraj Maharjan, A. Pastor López-Monroy, and Thamar Solorio. 2018. Mpst: A corpus of movie plot synopses with tags.

Omri Koshorek, Adir Cohen, Noam Mor, Michael Rotman, and Jonathan Berant. 2018. Text segmentation as a supervised learning task.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations.

Ruiqi Li, Leyang Cui, Songtuan Lin, and Patrik Haslum. 2023. Automated action model acquisition from narrative texts.

Anastasia Malysheva, Alexey Tikhonov, and Ivan P. Yamshchikov. 2021. *DYPLODOC: Dynamic Plots for Document Classification*. IOS Press.

Janet H Murray. 2006. Toward a cultural theory of gaming: Digital games and the co-evolution of media, mind, and culture. *Popular Communication*, 4(3):185–202.

Alexander Pan, Jun Shern Chan, Andy Zou, Nathaniel Li, Steven Basart, Thomas Woodside, Jonathan Ng, Hanlin Zhang, Scott Emmons, and Dan Hendrycks. 2023. Do the rewards justify the means? measuring trade-offs between rewards and ethical behavior in the machiavelli benchmark.

Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. 2019. Movie plot analysis via turning point identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1707–1717, Hong Kong, China. Association for Computational Linguistics.

Vladimir Propp. 1968. Morphology of the folktale, trans. *Louis Wagner, 2d. ed.*

Raven Rothkopf, Hannah Tongxin Zeng, and Mark Santolucito. 2024. Enforcing temporal constraints on generative agent behavior with reactive synthesis.

Viktor Shklovsky. 1925. Theory of prose (b. sher, trans.). *Champaign, IL: Dalkey Archive Press. Original work published*.

Alexey Tikhonov and Ivan P. Yamshchikov. 2022. Actionable entities recognition benchmark for interactive fiction.

Judith van Stegeren and Mariët Theune. 2019. Narrative Generation in the Wild: Methods from NaNoGenMo. In *Proceedings of the Second Workshop on Storytelling*, pages 65–74, Florence, Italy. Association for Computational Linguistics.

Ivan Yamshchikov and Alexey Tikhonov. 2023. What is wrong with language models that can not tell a story? In *Proceedings of the The 5th Workshop on Narrative Understanding*, pages 58–64, Toronto, Canada. Association for Computational Linguistics.

Wei Zhou, Xiangyu Peng, and Mark Riedl. 2023. Dialogue shaping: Empowering agents through npc interaction.

Table 4: Branching Points and Alternatives in *Alice's Adventures in Wonderland*

No.	**Main Decision Point**	**Alternatives**
1	Alice falls down the well	- Tries to look for something to grab onto to stop - Attempts to fly or float by flapping her arms
2	Alice uses the little golden key to open the small door	- Tries breaking one of the doors with a chair - Climbs onto the table
3	Alice cries a pool of tears and falls there	- Calls out for help - Eats something to change her size
4	Alice thinks she is Mable and continues to cry	- Searches for someone who remember her - Insists that she is not Mabel
5	Alice tells Mice about dogs and scares it	- Stops talking about pets - Apologizes to the Mouse
6	Alice decides to join the Caucus-race	- Suggests a different activity - Objects to the Caucus-race
7	Alice decides to look on top of the mushroom	- Eats a flower - Goes back to the puppy
8	Alice agrees to return to the Caterpillar	- Ignores it and walks away - Loses her temper with the Caterpillar
9	Alice knocks on the door	- Searches for another entrance - Returns to the wood
10	Alice tries to calm down the cook	- Leaves the room - Organizes a cleanup effort
11	Alice decides to leave the tea-party forever	- Stays despite rudeness - Invites the Dormouse to leave
12	Alice stays standing on the arrival of King and Queen	- Lies down like the gardeners - Starts clapping
13	Alice decides to talk with the Cat about the game	- Ignores the Cat and plays alone - Attempts to leave the croquet ground
14	Alice argues with the Duchess	- Agrees with the Duchess - Changes the subject
15	Alice follows the Gryphon to the trial	- Returns to the Mock Turtle - Stays to listen to the soup song

Utilizing GPT-4 to Solve TextWorld Commonsense Games Efficiently

Binggang Zhuo, Masaki Murata
Tottori University
zhuobinggang@gmail.com, murata@tottori-u.ac.jp

Abstract

Most artificial intelligence agents in interactive fiction games are implemented using reinforcement learning. Considering the recent rapid development of large language models, we propose an approach that utilizes a large language model to tackle interactive fiction game tasks. The chosen test dataset is TextWorld Commonsense, an interactive fiction game environment designed for artificial intelligence agents. In these games, the AI agent's task is to organize rooms and place items in appropriate locations. To achieve a high score in the game, common sense knowledge about "which items belong to which locations" is important. Our approach is based on GPT-4 and a carefully designed prompt. Experimental results demonstrate that our approach outperforms prior research. Specifically, GPT-4 with feedback-augmented prompt successfully completed all tasks in both simple and medium level game environments without fine-tuning. In hard level game environments, our approach achieved a normalized score of 0.70, surpassing the best baseline score of 0.57.

Keywords: GPT4, text game, common sense

1. Introduction

Interactive fiction games pose a challenging natural language processing task. In interactive fiction games, both player commands and feedback from the engine are in natural language form. Therefore, to play these games, artificial intelligence (AI) agents not only need to comprehend the text responded by the game engine but also generate natural language to take an action. In addition to language understanding, AI agents often require other complex abilities such as long-term memory, planning, and common sense to attain high scores. Due to sparse rewards, traditional AI agents for interactive fiction games have relied on reinforcement learning techniques.

The main challenge of reinforcement learning lies in its difficulty of training. Considering the rapid development of large language models (LLMs) in recent years (Zhao et al., 2023), we propose an approach that utilizes GPT-4 (Bubeck et al., 2023) to solve interactive fiction games. The carefully designed prompt is the main part of our research approach, drawing inspiration from key achievements in the field of prompt engineering, such as Chain-of-Thought (COT), few-shot prompting and output templates. COT encourages deep reasoning within LLMs through specific prompt phrases (e.g., "think step by step"), resulting in more coherent outputs (Wei et al., 2022). Few-shot prompting helps LLMs deepen their understanding of the task by providing examples in the prompt. Output template, inspired by Park et al. (Park et al., 2023), standardizes model outputs using "<fill in>" prompts, facilitating subsequent automated processing. Additionally, we introduce feedback augmented prompting, a technique that simply clarifies reward and punishment information responded by the game engine, aiding large language models in better grasping the situation.

We evaluated the performance of our approach on the TextWorld Commonsense dataset (Murugesan et al., 2021) (details in Section 4.1). Experimental results show that GPT-4 with our carefully designed prompts successfully completed all tasks in simple and medium level games. In hard level game environments, our approach achieved a normalized score of 0.70, surpassing the best baseline performance (score of 0.57). The contributions of our study are as follows:

- We evaluated the performance of large language models on the TWC dataset, and by integrating Chain-of-Thought (Wei et al., 2022), few-shot prompting, and feedback augmentation, our research approach achieved the state-of-art performance on the TWC dataset as we known. While demonstrating outstanding performance, our approach also possesses the advantage of simplicity in implementation.
- Through ablation experiments, we confirmed the contributions of each module to our research approach and highlighted the significance of the feedback augmentation technique proposed in this study.

2. Related Work

2.1. Datasets

Interactive fiction games were initially designed for human players, with one of the most famous games being Zork. Numerous studies on Zork indicate that the game poses a challenging task for existing AI agents (Haroush et al., 2018a; Yao et al.,

Games and NLP @LREC-COLING-2024, pages 76–84
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

2021; Tsai et al., 2023) (including agents based on large language models). Instead of directly studying difficult games, we try to incrementally improve the performance of AI agents in simpler game environments. To address the need of customizing game environments, Côté et al. (Côté et al., 2019) introduced TextWorld, a sandbox environment that allows users to create games manually or through a set of generation mechanisms. The TextWorld Commonsense environment (Murugesan et al., 2021) (TWC) is built on TextWorld, with the primary goal of assessing agents' ability to apply common sense. Following TWC, more complex interactive fiction game environments emerged, such as Science World (Wang et al., 2022) and Complex World (Basavatia et al., 2023). The challenge with Science World lies in its domain-specific environment, where an agent performing well may lack general skills to apply in other contexts. Complex World, being more intricate than TWC, is a subject for our future study.

2.2. Agents

Due to the sparse rewards in interactive fiction games, reinforcement learning has naturally become the primary implementation method for AI agents. LSTM-DQN (Narasimhan et al., 2015) stands out as one of the earliest achievements in this field, where Karthik et al. used LSTM to encode textual observation into hidden states. DRRN (He et al., 2015), an improvement upon LSTM-DQN, separated the neural network into two parts to encoding states and actions separately, enhancing the model's performance and stability. KG-A2C (Ammanabrolu and Hausknecht, 2020), during the exploration process, generates a knowledge graph to further guide the model's exploration, proving to enhance the learning efficiency and performance of agents. These approaches mentioned above are used as baselines in the TextWorld Commonsense paper. It is worth noting that as stated in the TWC paper (Murugesan et al., 2021), LSTM-DQN is implemented as LSTM-A2C, which represents a different reinforcement learning paradigm more suitable for the TWC environment. The proposed TWC Agent in the TWC paper extracts common sense from ConceptNet and, by combining this common sense knowledge, achieves the highest performance among all models mentioned above.

2.3. Prompt Engineering

Due to the vast number of parameters in large language models, fine-tuning has become a costly task. This problem have prompted people to seek better ways to utilize large language models, and prompt engineering is one of the best options. Prompt engineering significantly enhances models' performance on specific tasks without modifying their parameters, achieved through cleverly designed prompts. There is a wealth of prior research in the field of prompt engineering, and we will only introduce three techniques used in this study: (1) Few-shot prompting. Brown et al. (Brown et al., 2020) trained GPT-3 and found that significant performance improvement in large language models can be achieved by providing only a few examples within prompts, without the need for costly fine-tuning. (2) Chain-of-Thought. Wei et al. (Wei et al., 2022) found that rather than directly outputting results, prompting the model to output the thought process at the same time leads to more accurate results. They prompt the model to output in a similar manner by providing Chain-of-Thought examples. Subsequently, Kojima et al. (Kojima et al., 2022) further pointed out that similar effects can be achieved by simply appending special key words (such as ”think step by step”) in prompts. (3) Output template. Joon et al.'s study on AI villages (Park et al., 2023) is one of the most impressive works of large language models in recent years. We borrowed the idea of output templates from this paper. Specifically, prompts like ”<fill in>” can standardize the model's output, facilitating subsequent automation processes.

3. Model

3.1. Problem Setup

Our study focuses on using large language models to solve interactive fiction games. The dataset selected for our study is the TextWorld Commonsense (TWC), which includes five games for each of the three difficulty levels: easy, medium, and hard. The dataset can be found on GitHub [1]. In each game, the AI agent's task is to organize rooms, specifically by placing certain objects back in their designated locations. For example, a used tissue on the ground should be put in the trash can, and a dirty piece of clothing should be placed in the washing machine. The criteria for determining which object belongs to a specific location are based on common sense, and the validity of these object-location pairs has been confirmed by human evaluators (Murugesan et al., 2021).

At each time step, the agent receives an observation from the game engine, presented in natural language form. Subsequently, the agent needs to respond in natural language to present its consideration and take action. After the chosen action is executed, the game engine returns a new observation, and then the agent needs to choose a new action. This cycle has two termination conditions: either the agent completes all tasks or the number of actions taken exceeds the predefined limit. In our study, the action limit is set to 20.

---

[1] https://github.com/IBM/commonsense-rl/tree/master/games/twc

It should be noted that as stated in the TWC paper (Murugesan et al., 2021), the maximum number of steps for all agents is set to 50, while we limit the number of steps to 20 for two reasons. Firstly, our study aims to improve the efficiency of GPT-4 under restricted conditions. Secondly, limiting the number of steps not only reduces the frequency of API calls but also keeps the action history concise, which leads to a lower API usage cost. On the other hand, as stated in the TWC paper, the average number of steps taken by humans to complete hard-level games is 15.
The observations returned by the game engine include following content:

- **Action Feedback**: If an action is executed, the agent receives feedback from the environment. If the model earns a reward in an action, that information is also included in the feedback. For example, a pair of action and feedback could be: ”*insert dirty yellow T-shirt into washing machine -> You put the dirty yellow T-shirt into the washing machine. Your score has just gone up by one point*”. In this example, ”*insert dirty yellow T-shirt into washing machine*” is the action, and the part after the arrow is feedback from the game engine.
- **Environment Description**: Textual description of the current environment.
- **Inventory**: A list of currently held items.
- **Available Action List**: A list of actions currently available to the agent.

3.2. Prompts

Our study proposes an approach using large language models to play interactive fiction games, with the carefully designed prompt being the most crucial part of our approach. Specifically, the prompt engineering techniques included in our prompts are: one-shot prompting, Chain-of-Thought, output template, and feedback augmentation.
An example of prompts is provided below:

> Task: You are an experienced text game player, your goal is to put things in their proper locations and improve your score.
>
> Example walkthrough: Action 0: insert dirty yellow dress into washing machine -> You put the dirty yellow dress into the washing machine. Your score has just gone up by one point. Right position. Action 1: take dirty yellow T-shirt from bench -> You take the dirty yellow T-shirt from the bench. Action 2: insert dirty yellow T-shirt into washing machine -> You put the dirty yellow T-shirt into the washing machine. Your score has just gone up by one point. Right position. Action 3: take wet azure dress from suspended shelf -> You take the wet azure dress from the suspended shelf. Action 4: insert wet azure dress into clothes drier -> You put the wet azure dress into the clothes drier. Your score has just gone up by one point. Right position. Action 5: take white cap from bench -> You take the white cap from the bench. Action 6: go east -> -= Corridor =-You’ve entered a corridor. Action 7: put white cap on hat rack -> You put the white cap on the hat rack. Your score has just gone up by one point. Right position. Action 8: take dirty cardigan -> You pick up the dirty cardigan from the ground. Action 9: take dirty checkered shirt from shoe cabinet -> You take the dirty checkered shirt from the shoe cabinet. Action 10: take dirty maroon dress -> You pick up the dirty maroon dress from the ground. Action 11: go west -> -= Laundry Room =-You’ve entered a laundry room. Action 12: insert dirty cardigan into washing machine -> You put the dirty cardigan into the washing machine. Your score has just gone up by one point. Right position. Action 13: insert dirty checkered shirt into washing machine -> You put the dirty checkered shirt into the washing machine. Your score has just gone up by one point. Right position. Action 14: insert dirty maroon dress into washing machine -> You put the dirty maroon dress into the washing machine. Your score has just gone up by one point. Right position.
>
> Action history: Action 0: examine suspended shelf ->The suspended shelf is reliable. Action 1: take dirty singlet ->You pick up the dirty singlet from the ground.
>
> Inventory: You are carrying: a dirty singlet
>
> Current environment: -= Laundry Room =-You’ve entered a laundry room. You begin to take stock of what’s here. You can see a washing machine. Empty! What kind of nightmare TextWorld is this? You make out an opened clothes drier. The clothes drier is empty! What a waste of a day! You can see a laundry basket. Hmmm... what else, what else? The laundry basket is empty! What a waste of a day! You can see a suspended shelf. The suspended shelf is ordinary. But oh no! there’s nothing on this piece of garbage. You move on, clearly infuriated by TextWorld. You scan the room, seeing a bench. But the thing hasn’t got anything on it. Oh! Why

> couldn't there just be stuff on it? You make out a work table. But oh no! there's nothing on this piece of garbage.
>
> Action you can take:
>
> * close clothes drier
>
> * close laundry basket
>
> * close washing machine
>
> * drop dirty singlet
>
> * examine bench
>
> * examine clothes drier
>
> * examine laundry basket
>
> * examine suspended shelf
>
> * examine washing machine
>
> * examine work table
>
> * insert dirty singlet into clothes drier
>
> * insert dirty singlet into laundry basket
>
> * insert dirty singlet into washing machine
>
> * look
>
> * put dirty singlet on bench
>
> * put dirty singlet on suspended shelf
>
> * put dirty singlet on work table
>
> Question: To put things in their proper locations and improve your score, what should you do? Think step by step then choose 'one' action from above list.
>
> Consideration: <fill in>
>
> Next action: <fill in>

"**Task**" is the task description that remains constant for each time step. "**Example walkthrough**" is an one-shot example we provide to large language models, which is basically the action history taken by a human player in a hard-level game from the training dataset. To demonstrate the movement of objects across rooms, we deliberately selected an environment [2] consisting of two rooms.

The "**Action history**" is the memory of LLMs, getting longer as the time step increases, and it is textualized in the format of "*action -> action feedback*". Action is extracted from the response of LLM, while action feedback is basically the content returned by the game engine. However, it should be noted that we have appended content to item-placing action feedback for **feedback augmentation**. For details, please refer to Section 3.3. Another point to note is that while action feedback is always short, when the agent moves to a new room, the feedback provides a complete description of the environment, consistent with the "**Current environment**" part in the prompt. Therefore, to limit the length of prompts, we only retain the content before the first period in such cases.

"**Current environment**" is a textual description of the current room obtained from the game engine. The "**Action you can take**" is a list of possible actions obtained from the game engine. The "**Question**," "**Consideration**," and "**Next Action**" remain unchanged at each time step and can be considered as an output template. Of particular note are the keywords "Think step by step" and "Consideration: <fill in>," which are simple implementations of the Chain-of-Thought.

For the above prompt, a sample output from GPT-4 turbo could be:

> Consideration: To properly place the dirty singlet, it should be put somewhere related to laundry processing. Inserting it into the washing machine seems like the most appropriate step since typically, dirty clothes need to be washed before they are dried or placed in a laundry basket.
>
> Next action: insert dirty singlet into washing machine

3.3. Feedback augmentation

We propose feedback augmentation due to the ambiguous feedback provided by the TWC game engine, especially concerning item placement actions. This ambiguity hinders GPT-4 from making accurate judgments.

For instance, in the case of incorrectly placing an item, we might get action-feedback pairs like: "*insert clean white polo shirt into chest of drawers -> You put the clean white polo shirt into the chest of drawers*." As seen, the environment informs us that the action has been successfully executed, but it does not indicate that the item (polo shirt) does not belong to that location (chest of drawers).

However, when an item is correctly placed, we get action-feedback pairs like: "*insert clean white polo shirt into wardrobe -> You put the clean white polo shirt into the wardrobe. Your score has just gone up by one point*." By comparing the two environmental feedbacks, we find that a score increase implies the item is correctly placed. While a fine-tuned model might learn to judge the effectiveness of an action based on the presence or absence of score growth information, for a 0-shot setting, the default action feedback is undoubtedly ambiguous.

Considering these factors, we introduce feedback augmentation technique. The approach is very simple; we just append a sentence of explanatory text after the item-placing action feedback. Specifically, when an item is placed correctly, we append "**Right position**" to the feedback text as a hint. An example is as follows.

---

[2] https://github.com/IBM/commonsense-rl/blob/master/games/twc/hard/train/tw-iqa-cleanup-objects7-take6-rooms2-train-aEOOFxjEcxEll9Xo.ulx

> Insert clean white polo shirt into wardrobe -> You put the clean white polo shirt into the wardrobe. Your score has just gone up by one point. **Right position**.

For incorrectly placement, we append "**Wrong position, you should put it somewhere else, maybe the other room**" to the feedback. For example:

> Insert clean white polo shirt into chest of drawers -> You put the clean white polo shirt into the chest of drawers. **Wrong position, you should put it somewhere else, maybe the other room**.

In this example, the action is incorrect because the shirt should be placed in the wardrobe but not the chest of drawers. The "**maybe the other room**" part encourages the model to explore across rooms. These action-feedback pairs will then be stored in the action history.

3.4. Using Prompts via API

In section 3.2, we introduced prompts used in our approach. In this section, we briefly discuss the usage of prompts. We utilize the OpenAI API to invoke GPT-4 Turbo (specifically named gpt-4-1106-preview) and GPT-3.5 Turbo (specifically named gpt-3.5-turbo-0613). For example, when calling GPT-4 Turbo, the specific Python code is as follows:

```python
from openai import OpenAI
client = OpenAI()
completion = client.chat.completions.
    create(
  model'=gpt-4-1106-'preview,
  messages=[
    {"role": "system", "content":
    system_msg},
    {"role": "user", "content": user_msg
    }
  ]
)
```

Listing 1: Python code for calling GPT-4 Turbo

Here, ‘system_msg‘ and ‘user_msg‘ are string variables containing the text content. In our usage, they respectively encompass the following text content:

- **system_msg**: Task, Example walkthrough, Action history, Inventory, Current environment.
- **user_msg**: Available action List, Question, Consideration, Next action.

4. Experiments

4.1. Dataset

The dataset used to evaluate the performance of the approach is TextWorld Commonsense (TWC), which can be found on GitHub [3]. This dataset includes five games each for easy, medium, and hard levels. The levels differ mainly in the number of items to organize and the number of rooms. The easiest game involves one item and one room, while the most challenging game consists of seven items and two interconnected rooms. Agents can move between rooms using commands like "*go east*."

4.2. Experimental Setup

We use the OpenAI API to call GPT-4 Turbo (specifically named gpt-4-1106-preview) and GPT-3.5 Turbo (named gpt-3.5-turbo-0613). We did not fine-tune these models. For each game, we set a maximum limit of 20 actions that the model can take. For each approach proposed and each game, we run the experiment only once and obtain its result. Similar to the TWC paper (Murugesan et al., 2021), we use normalized scores (score achieved ÷ maximum achievable score) to evaluate the performance of different approaches.

4.3. Baselines

The baselines used in our study include LSTM-A2C, DRRN, KG-A2C, and TWC Agent (introducing common sense knowledge through Numberbatch). Characteristics of these various baselines have been introduced in the related works section. It is important to note that the scores of these models are directly obtained from the TWC paper rather than being re-implemented and tested. The scores for these baselines are taken as the average of ten repeated experiments (Murugesan et al., 2021).

5. Results

The experimental results are presented in Table 1. The scores in the table are normalized scores, calculated by dividing the achieved score by the maximum achievable score in games. Note that each difficulty level includes five games. As shown in the table, our proposed approach outperformed the best-performing baseline in various difficulty levels. It is noteworthy that all baseline methods underwent 100 episodes of training on the training set, while we did not fine-tune the large language models.
We will discuss in the ablation study section about which prompt engineering techniques are crucial for our research method.

5.1. Ablation Study

"**-GPT4**": Our approach relies on GPT-4, and a natural question is how the performance would change by replacing it with other large language models. Therefore, we replace GPT-4 with GPT-3.5 in our

---

[3] https://github.com/IBM/commonsense-rl/tree/master/games/twc

	Easy	Medium	Hard
LSTM-A2C	0.86	0.74	0.54
DRRN	0.81	0.73	0.44
KG-A2C	0.85	0.72	0.46
TWC-Agent	0.96	0.85	0.57
Our approach	**1.00**	**1.00**	**0.70**

Table 1: Experimental Results. For detailed experimental settings, please refer to Section 4.

approach and name it ”-GPT4”. It is important to note that due to the presence of the output template (*Question*, *Consideration*, and *Next Action*), GPT-4’s responses are well-structured, and actions can be automatically extracted from the response text. However, GPT-3.5’s responses are rarely structured, requiring additional grounding techniques to extract a valid action. In our study, we manually extract action from GPT-3.5’s response text. The human assistant’s task is to copy a part of text from the model’s response (in the example above, it is the part after ”*Next action*”), and paste it into the interactive interface.

”**-COT**”: The output template in our prompt is like:

> Question: To put things in their proper locations and improve your score, what should you do? Think step by step then choose ’one’ action from above list.
>
> Consideration: <fill in>
>
> Next action: <fill in>

This output template was inspired by the Chain-of-Thought (COT) study (Wei et al., 2022). A natural question is whether removing the COT part from the prompt would lead to a performance decline. Therefore, we modify the output template as follows:

> Question: To put things in their proper locations and improve your score, what should you do? Choose ’one’ action from above list.
>
> Next action: <fill in>

Compared to the previous template, we removed ”*Think step by step then*” and ”*Consideration: <fill in>*”. We name the approach based on this prompt as ”-COT”.

”**-one-shot**”: Brown et al. (Brown et al., 2020) suggest that providing examples within prompts to large-scale language models can significantly improve their performance on specific tasks. To test the effectiveness of the one-shot setting, we removed the ”Example walkthrough” part from our prompts and named this technique ”-one-shot”.

”**-FA**”: Feedback augmentation is the main innovation in our study. To show the performance gain from feedback augmentation, we remove the explanatory text appended after the action feedback (basically restoring the default behavior of the game environment, refer to Section 3.3), naming this approach ”-FA” (-Feedback Augmentation). For example, below is the augmented action-feedback pair when an item is misplaced. In the ”-FA” approach, we simply remove the hint marked in bold.

	Easy	Medium	Hard
TWC-Agent	0.96	0.85	0.57
Our approach	**1.00**	**1.00**	**0.70**
-GPT4	**1.00**	0.58	0.18
-COT	**1.00**	**1.00**	0.61
-one-shot	**1.00**	**1.00**	0.64
-FA	**1.00**	**1.00**	0.52

Table 2: Ablation experiment results. **COT** means Chain-of-Thought, and **FA** means feedback augmentation. For detailed experimental settings, please refer to Section 5.1.

> Insert clean white polo shirt into chest of drawers -> You put the clean white polo shirt into the chest of drawers. **Wrong position, you should put it somewhere else, maybe the other room**.

The results of the ablation study are presented in Table 2. Firstly, it is evident that GPT-4 is the most crucial part for our approach. If GPT-4 is replaced with GPT-3.5, the approach’s performance on hard-level games drops from a normalized score of 0.70 to 0.18. Moreover, many outputs from GPT-3.5 turbo do not meet the valid format, requiring manual adjustments. The second crucial module for our method is feedback augmentation. After removing feedback augmentation, the model’s performance on hard-level games drops from 0.70 to 0.52. Lastly, we find that Chain-of-Thought and one-shot setting also significantly improves the model’s performance.

6. Discussion

6.1. One-shot Prompting

One-shot (as well as few-shot) prompting assists large language models in understanding task detail by providing examples within prompts. The main issue with one-shot prompting lies in the significant impact of example selection on the model’s performance.

As mentioned in Section 3.2, to provide a demonstration of moving items across rooms, we deliberately chose a game environment consisting of two rooms for obtaining the example walkthrough.

To investigate the impact of sample selection on the performance of our approach, in this section,

	Easy	Medium	Hard
TWC-Agent	0.96	0.85	0.57
Our approach	1.00	1.00	**0.70**
-one-shot	1.00	1.00	0.64
Easy one-shot	1.00	1.00	0.64

Table 3: One-shot prompting with easy-level example experiment result. For detailed experimental settings, please refer to Section 6.1.

we obtained an example walkthrough from an easy-level game and replaced the hard-level example in the prompts. The specific easy-level walkthrough is as follows:

> Example walkthrough: Action 0: take dirty gray underpants from work table -> You take the dirty gray underpants from the work table. Action 1: insert dirty gray underpants into washing machine -> You put the dirty gray underpants into the washing machine. Your score has just gone up by one point. Right position.

The performance of the model after adopting the easy-level one-shot setting (denoted as "**Easy one-shot**") is shown in Table 3.
As shown in Table 3, we found that example selection will significantly affect model's performance. After replacing the hard-level walkthrough with that of the easy level, the model's performance decreased from 0.70 to 0.64, which remained consistent with the performance of not zero-shot setting (denoted as "**-one-shot**").

6.2. Case Study

In simple and medium level games, our approach keeps completing all tasks in the fewest steps. However, in hard-level games, the model may still make **incorrect actions**. These actions are primarily due to certain items not belonging to the current room. For example, the following case is from the first game at the hard level:

> Action 0: take wet white jumper from patio chair -> You take the wet white jumper from the patio chair. Action 1: put wet white jumper on clothesline -> You put the wet white jumper on the clothesline. Your score has just gone up by one point. Right position. Action 2: **put milk on patio table** -> You put the milk on the patio table. Wrong position, you should put it somewhere else, maybe the other room.

The part marked in bold indicates the incorrect action made by the model. As indicated by the engine feedback, in this example, the milk does not belong to the patio table. While making this action, GPT-4's consideration is as follows:

> Consideration: The milk you are carrying is likely intended to be placed somewhere it can remain cool or be utilized. It is not appropriate for the BBQ, clothesline, or the patio chair. Among the options, the patio table seems to be the most logical place to put the milk, as it is a typical location to place food and beverages when you're spending time outdoors.
>
> Next action: **put milk on patio table**

We can see that the model correctly ruled out BBQ, clothesline, and patio chair as incorrect locations. However, the model is not aware that there is a refrigerator (the correct place for the milk) in another room. Therefore, the model placing the milk on the patio table is an understandable mistake. After receiving feedback from the engine, the model's consideration is as follows:

> Consideration: The milk is out of place; it should likely be put somewhere with refrigeration to prevent spoilage, such as a refrigerator, which could be inside the house.
>
> Next action: take milk from patio table

We can see that the model makes reasonable inferences and successfully returns the milk to its correct location in the subsequent actions:

> Action 2: **put milk on patio table** -> You put the milk on the patio table. Wrong position, you should put it somewhere else, maybe the other room. Action 3: take milk from patio table -> You take the milk from the patio table. Action 4: go west -> -= Kitchen =-You find yourself in a kitchen. Action 5: **insert milk into fridge** -> You put the milk into the fridge. Your score has just gone up by one point. Right position.

However, there are also instances where, after multiple unsuccessful attempts, the model **shifts its attention to other items**. For example, in the following case (from the second game at the hard level), the model gives up picking up the white polo shirt after two attempts:

> Action 6: **put clean white polo shirt on sofa** -> You put the clean white polo shirt on the sofa. Wrong position, you should put it somewhere else, maybe the other room. Action 7: take clean white polo shirt from sofa -> You take the clean white polo shirt from the sofa. Action 8: go west

> -> -= Bedroom =-You find yourself in a bedroom. Action 9: **insert clean white polo shirt into chest of drawers** -> You put the clean white polo shirt into the chest of drawers. Wrong position, you should put it somewhere else, maybe the other room. Action 10: go east -> -= Livingroom =-You are in a livingroom. Action 11: take brown suit from end table -> You take the brown suit from the end table.

Additionally, although rare, GPT-4 may also make mistakes by **repeating the same incorrect action twice**, as shown in the example below.

> Action 8: go south -> -= Corridor =-Well, here we are in a corridor. Action 9: take clean white socks -> You pick up the clean white socks from the ground. Action 10: **insert clean white socks into shoe cabinet** -> You put the clean white socks into the shoe cabinet. Wrong position, you should put it somewhere else, maybe the other room. Action 11: take clean white socks from shoe cabinet -> You take the clean white socks from the shoe cabinet. Action 12: **insert clean white socks into shoe cabinet** -> You put the clean white socks into the shoe cabinet. Wrong position, you should put it somewhere else, maybe the other room.

Let's take a look at the model's consideration when making action 12 (repeatedly putting socks into the shoe cabinet).

> Consideration: In the current scenario, you are in a corridor with various items and furniture pieces such as a shoe cabinet, hat rack, key holder, and umbrella stand, all of which are empty. You're carrying a pair of clean white socks. The socks typically belong in a bedroom or a designated storage area for clothing, but considering that the shoe cabinet is mentioned to be empty and is a suitable location for storing such items, placing clean socks there seems reasonable. Although it was previously stated that the shoe cabinet was not the right place for the clean white socks, given the context within the game, it may be that different types of items have specific correct placements; the previous location could have been incorrect for the other item but might be correct for the clean white socks. Additionally, the placement of items is often based on logical organization within a living space.
>
> Next action: **insert clean white socks into shoe cabinet**

The model seems to be questioning the accuracy of the feedback provided by the game environment. Given that there is no other more suitable place to put the socks in the room besides the shoe cabinet, such questioning is understandable. The actual place where the socks belong is in another room, which again highlights the difficulty of moving items across rooms.

Apart from incorrect actions, there is a category of actions that potentially do harm to the model's performance, referred to as **redundant actions**. For instance, in the following case (also from the second game at the hard level):

> Action 0: insert clean plaid polo shirt into wardrobe -> You put the clean plaid polo shirt into the wardrobe. Your score has just gone up by one point. Right position. Action 1: **close wardrobe** -> You close the wardrobe.

In this example, action 1 (close wardrobe) is a redundant action. Due to the wardrobe being closed, the option ”insert into wardrobe” will not appear when picking up other clothing items subsequently. Closing anything in the TWC game environment does not yield any benefits (such as score improvement) and is entirely redundant. If we include a hint like ”no need to close anything,” the model's performance may improve further, but we do not want to introduce dataset-specific knowledge to harm the generality of our approach.

7. Conclusion

In our study, we proposed a method for implementing artificial intelligence agents in interactive fiction using large language models. Compared to traditional reinforcement learning methods, our approach is simple to implement and performs well. Due to the extensive training on massive datasets, GPT-4 have acquired basic knowledge about the human world. On the TextWorld Commonsense dataset, our agent achieved scores surpassing the best-performing baseline (reinforcement learning models trained on the training set) without fine-tuning. Specifically, using feedback-augmented prompts, GPT-4 perfectly completed all tasks in simple and medium-level game environments. In hard-level environments, our method's performance (0.70) also far surpassed the best baseline (0.57).

8. Bibliographical References

Prithviraj Ammanabrolu and Matthew Hausknecht. 2020. Graph constrained reinforcement learning for natural language action spaces. *arXiv preprint arXiv:2001.08837*.

Shreyas Basavatia, Shivam Ratnakar, and Keerthiram Murugesan. 2023. Complexworld: A large language model-based interactive fiction learning environment for text-based reinforcement learning agents. In *International Joint Conference on Artificial Intelligence*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097.

Marc-Alexandre Côté, Akos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Matthew Hausknecht, Layla El Asri, Mahmoud Adada, et al. 2019. Textworld: A learning environment for text-based games. In *Computer Games: 7th Workshop, CGW 2018, Held in Conjunction with the 27th International Conference on Artificial Intelligence, IJCAI 2018, Stockholm, Sweden, July 13, 2018, Revised Selected Papers 7*, pages 41–75. Springer.

Matan Haroush, Tom Zahavy, Daniel J. Mankowitz, and Shie Mannor. 2018a. Learning how not to act in text-based games. In *6th International Conference on Learning Representations, ICLR 2018*. Publisher Copyright: © 6th International Conference on Learning Representations, ICLR 2018 - Workshop Track Proceedings. All rights reserved.; 6th International Conference on Learning Representations, ICLR 2018 ; Conference date: 30-04-2018 Through 03-05-2018.

Matan Haroush, Tom Zahavy, Daniel J Mankowitz, and Shie Mannor. 2018b. Learning how not to act in text-based games.

Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Lihong Li, Li Deng, and Mari Ostendorf. 2015. Deep reinforcement learning with a natural language action space. *arXiv preprint arXiv:1511.04636*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Keerthiram Murugesan, Mattia Atzeni, Pavan Kapanipathi, Pushkar Shukla, Sadhana Kumaravel, Gerald Tesauro, Kartik Talamadupula, Mrinmaya Sachan, and Murray Campbell. 2021. Text-based rl agents with commonsense knowledge: New challenges, environments and baselines. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9018–9027.

Karthik Narasimhan, Tejas Kulkarni, and Regina Barzilay. 2015. Language understanding for text-based games using deep reinforcement learning. *arXiv preprint arXiv:1506.08941*.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–22.

Chen Feng Tsai, Xiaochen Zhou, Sierra S Liu, Jing Li, Mo Yu, and Hongyuan Mei. 2023. Can large language models play text games well? current state-of-the-art and open questions. *arXiv preprint arXiv:2304.02868*.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. 2022. Scienceworld: Is your agent smarter than a 5th grader? *arXiv preprint arXiv:2203.07540*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Shunyu Yao, Karthik Narasimhan, and Matthew Hausknecht. 2021. Reading and acting while blindfolded: The need for semantics in text game agents. *arXiv preprint arXiv:2103.13552*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Linguistic Acceptability and Usability Enhancement: A Case Study of GWAP Evaluation and Redesign

Wateen Aliady and Massimo Poesio
Queen Mary University of London, Imam Mohammad Ibn Saud University
United Kingdom, Saudi Arabia
{ w.a.a.aliady, m.poesio}@qmul.ac.uk

Abstract

Collecting high-quality annotations for Natural Language Processing (NLP) tasks poses challenges. Gamified annotation systems, like Games-with-a-Purpose (GWAP), have become popular tools for data annotation. For GWAPs to be effective, they must be user-friendly and produce high-quality annotations to ensure the collected data's usefulness. This paper investigates the effectiveness of a gamified approach through two specific studies on an existing GWAP designed for collecting NLP coreference judgments. The first study involved preliminary usability testing using the concurrent think-aloud method to gather open-ended feedback. This feedback was crucial in pinpointing design issues. Following this, we conducted semi-structured interviews with our participants, and the insights collected from these interviews were instrumental in crafting player personas, which informed design improvements aimed at enhancing user experience. The outcomes of our research have been generalized to benefit other GWAP implementations. The second study evaluated the linguistic acceptability and reliability of the data collected through our GWAP. Our findings indicate that our GWAP produced reliable corpora with 91.49% accuracy and 0.787 Cohen's kappa.

Keywords: games-with-a-purpose, natural language processing, coreference annotation, usability study, language acceptability

1. Introduction

Games-with-a-Purpose (GWAP) offers a promising approach to leveraging human computation for annotation tasks (Von Ahn and Dabbish, 2004; Von Ahn, 2006; Von Ahn and Dabbish, 2008; Von Ahn et al., 2006a,b; Madge et al., 2019a; Lafourcade et al., 2015; Chamberlain et al., 2008; Amspoker and Petruck, 2022; Morrison et al., 2023; Chaiko et al., 2022; Xu et al., 2022; Mount Cieri et al., 2020). They involve designing games to collect annotations from players, utilizing their gaming skills and language competence, with entertainment serving as the primary incentive (Poesio et al., 2013; Vannella et al., 2014; Jurgens and Navigli, 2014). These systems combine entertainment with task completion and hold significant potential across various fields, including data annotation and problem-solving.

Usability is a critical aspect of GWAPs, impacting their success and acceptance. Another crucial factor is the reliability of their annotations, especially in Natural Language Annotation (NLP) tasks, where linguistic acceptability is vital. Our work explores usability and linguistic acceptability in GWAPs, particularly in the context of a 3D game designed for Arabic NLP annotation. We aim to evaluate these components to improve understanding and assist researchers in this field.

This study has three primary objectives: (1) to conduct a preliminary usability study of the presented GWAP and identify areas for enhancement, (2) to redesign the GWAP based on usability findings, and (3) to evaluate the linguistic acceptability of the collected judgments. By addressing these objectives, we aim to contribute to creating more user-centered GWAPs. Particularly, we target the following research questions:

Q1: In the context of 3D games, mainly focusing on the interface/menu layer, what design elements, interaction techniques and user experience factors in the interface/menu layer improve the usability of the players as informed by participant feedback from our usability test?

Q2: Could our virtual world game, *Stroll-with-a-Scroll*, be used to collect linguistically acceptable coreference annotation? Coreference resolution is clustering the mentions in a text that refer to the same real-world entity.

The next section of this paper discusses related work. A brief description of the design of our GWAP follows this. Next, we show an in-depth description of the preliminary study that guided the redesign process. Finally, we present the linguistic acceptability of our game.

2. Background and Related Work

Games-with-a-Purpose (GWAPs) are typically designed to leverage players' skills and abilities, primarily for entertainment. They have found ap-

Games and NLP @LREC-COLING-2024, pages 85–96
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

plications in various domains such as biological data collection (Kleffner et al., 2017; Kawrykow et al., 2012), image processing in AI (Krause et al., 2010), assessment and comparison of Explainable AI (XAI) techniques (Morrison et al., 2023), music annotation (Kim et al., 2008), and dance movement annotation (Kougioumtzian et al., 2022). In dance movement annotation, for instance, a notator or movement analyst describes and documents dance movements by recording details of the body's actions using a coding system. Similarly, in Natural Language Processing (NLP), GWAPs are employed for tasks like text annotation (Venhuizen et al., 2013; Madge et al., 2019b; Fort et al., 2014; Kicikoglu et al., 2019; Bonetti and Tonelli, 2020; Dziedzic, 2016; Xu et al., 2022) or generating original content for annotation (Amspoker and Petruck, 2022).

The evaluation of the usability of these gamified systems holds significant importance as it contributes to reducing errors, training time, and learning effort while also enhancing productivity and satisfaction (Rajanen and Dorina, 2017). For instance, in (Tomé Klock et al., 2017), ten different gamified educational systems were assessed using ergonomic criteria guidelines that evaluate usability and user experience. Additionally, a systematic literature review in (Laine and Lindberg, 2020) provided generalized recommendations to improve motivation in gamified systems, such as offering feedback, using familiar vocabulary, ensuring actions align with goals, and maintaining consistency across elements. However, (Gouveia et al., 2023) demonstrated that usability significantly correlates with intrinsic motivation in a virtual reality gamified system designed for rehabilitation purposes.

In usability testing, qualitative research methods like interviews, surveys, and focus groups offer insights based on users' self-reports (Roberts et al., 2019). One effective method involves employing the think-aloud (TA) protocol, where participants articulate their thoughts and feelings while interacting with a product or system. This approach enables researchers to gain insights into users' cognitive processes, thereby identifying potential issues or challenges users may encounter during interaction.

TA protocols represent one of the most prevalent methods for identifying User Experience (UX) issues during usability testing (Fan et al., 2020; McDonald et al., 2013). There are two commonly used TA protocols in the industry: Concurrent Think-Aloud (CTA) and Retrospective Think-Aloud (RTA) (Fan et al., 2020; McDonald et al., 2013). In CTA, users vocalize their thoughts while performing a task, whereas in RTA, users complete the task and then articulate their thoughts by reviewing a recording. There has been an ongoing debate regarding which protocol is superior (Van den Haak and De Jong, 2003; Alshammari et al., 2015). We opted for CTA due to its popularity among UX practitioners (Fan et al., 2020; McDonald et al., 2013), as it allows UX evaluators to delve into participants' thought processes in real-time interaction with applications, which cannot be captured solely through retrospective self-reports.

Ensuring the linguistic reliability of the collected data is crucial. Consequently, some GWAPs have addressed this issue. For instance, *Phrase Detectives* (Poesio et al., 2013), a GWAP for English and Italian coreference annotation, initially employed majority voting to aggregate player feedback. They then assessed the acceptability of the collected judgments by comparing experts' annotations with the data derived from the majority vote of non-experts. The result indicated an 84% agreement across all cases, comparable to those observed when comparing an expert with an average annotator, typically trained students producing medium-quality annotations. Subsequently, *Phrase Detectives* improved aggregation by adopting Mention Pair Annotation (Paun et al., 2018), a dedicated probabilistic aggregation method for coreference. Here, players identify the nearest antecedent, and the best pairing is determined based on a probabilistic model (Paun et al., 2018). These pairs are then clustered to form a coreference chain, increasing the accuracy of the produced judgments to 92% (Poesio et al., 2019).

Games like the original von Ahn games and, for NLP, *Puzzle Racer*, have demonstrated the feasibility of entertaining GWAPs that generate high-quality annotations at a reduced cost (Jurgens and Navigli, 2014). Another example is *High School Superhero*, a GWAP developed for collecting acceptability judgments. It evaluated the resulting annotations in terms of agreement among players and compared them with experts' judgments (Bonetti et al., 2022). Additionally, *RigorMortis* measured acceptability in annotating multi-word expressions for French corpora (Fort et al., 2020).

3. Introduction to the Game: Stroll-with-a-Scroll

Stroll-with-a-Scroll represents the first virtual world GWAP designed for Arabic Natural Language Processing (NLP) tasks, featuring a treasure hunt theme set in an ancient Middle Eastern fictional town within a desert landscape. The game incorporates a narrative element at the outset of gameplay, inspired by the findings of a study on narrative importance (Krause et al., 2010). This narrative is presented through a cut scene, establishing the storyline and themes for players. As avatars dressed in traditional attire, players navigate the town, embarking on a quest to discover hidden chests scattered throughout the environment.

The game employs a navigation system displayed on the menu layer to aid players in locating chests, utilizing three colors (red, yellow, and green) to indicate proximity to the chest. Upon reaching a chest, players uncover a scroll containing text with torn sections. Given the age of these scrolls and the missing sections, players must solve puzzles to reconstruct the text. The puzzle mechanics, inspired by *Wormingo* (Kicikoglu et al., 2019), include selecting the correct word from the provided options and solving 'word search' puzzles within a grid of letters.

The coreference annotation task within the game follows the approach of *Phrase Detectives* (Chamberlain et al., 2008) and *Wormingo* (Kicikoglu et al., 2019), presenting players with annotation and validation questions. Annotation questions prompt players to identify whether a mention is new or old, with the option to select the antecedent if it is old or to skip the question. Validation questions, on the other hand, require players to evaluate other players' responses.

For post-processing, *Stroll-with-a-Scroll* adopts the methodology of *Phrase Detectives* (Chamberlain et al., 2008), utilizing Mention Pair Annotation (Paun et al., 2018) for probabilistic aggregation of coreference annotations. After collecting judgments from multiple players, this method selects the best pairing based on a probabilistic model, and then clusters pair to form coreference chains.

4. Preliminary Usability Study

A game's usability significantly impacts players' enjoyment and motivation, ultimately influencing participant numbers. Conducting early usability tests is crucial for enhancing user experience, streamlining navigation, promoting intuitive usage, and identifying design flaws. By observing user interactions with the proposed system, a usability study can pinpoint areas for improvement, thereby reducing dropouts. Moreover, usability has been demonstrated to affect users' engagement with Games-with-a-Purpose (GWAPs) (Bowser et al., 2013; Hamari and Keronen, 2017; Bui et al., 2020), highlighting its pivotal role in fostering user participation. Additionally, usability is a cornerstone of a successful virtual world (Lee and Chen, 2011), as it determines how effectively a virtual space facilitates specific tasks for particular users.

4.1. Participants and Procedure

In the preliminary study, we recruited 8 participants, consisting of 5 females and 3 males. The mean age of the participants was 28.5 years (SD=2.57).

A qualitative approach is typically preferred at this stage of development as it aids designers in identifying issues or bugs early on and making enhancements. Additionally, a qualitative approach can be utilized to comprehend player engagement in our game. Accordingly, numerous scholars have employed qualitative methodologies to explore engagement in virtual worlds (Chen and Kent, 2020; Bouta Cruz-Benito et al., 2015; Kohler et al., 2011). A think-aloud protocol (Lewis, 1982) is employed to gather data, allowing for open feedback collection. Given the early stage of game development, such open user feedback is crucial for testing usability and gaining initial insights into player engagement. We utilized the concurrent think-aloud (CTA) protocol, wherein users verbalize their thought processes while simultaneously working on a task, as it is more widely favored among UX practitioners (Fan et al., 2020; McDonald et al., 2013).

The study was conducted from October 24th to 30th, 2022, with each session lasting approximately 30-minute. Prior to commencing the preliminary test, participants were provided with informed consent outlining the study's objectives. Subsequently, they were introduced to the following tasks they were required to accomplish:

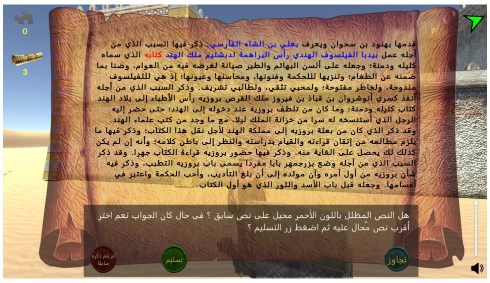
- First, sign up to join and start the game.
- Then, start the game and read the pre-game description.
- After you complete the pre-game part, navigate the scene to locate the chest.
- Finally, respond to the presented puzzles and the linguistic task, then navigate again to search for the following chest.

Our analysis is grounded in Reflexive Thematic Analysis (Braun and Clarke, 2019, 2021) chosen for its suitability with a small sample size of participants– in this case, eight participants in this experiment, and due to its flexibility in organizing results into common themes. The data was initially transcribed and then coded to create an affinity diagram, from which themes emerged. The following themes were generated from the analysis:

4.1.1. User Interface Refinement, to Ensure Familiarity and Simplicity Theme

More Familiarity Theme: The study by (Abada and Onibere, 2009) demonstrated that prior computer experience plays a significant role in intuitively understanding and using new computer software. This principle extends to video games (Miller et al., 2019). Intuitive interfaces are crucial in game design, with schema theory explaining how individuals grasp gameplay mechanics without prior experience with a particular game. A concise definition of gameplay provided by Lindley and Sennersten (Lindley et al., 2008) describes it as "the structure and algorithm determining the management of attentional and other cognitive, perceptual, and motor resources required to realize the tasks involved in gameplay."

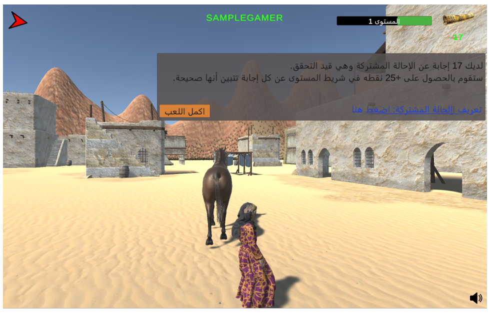
For example, Participant 1 commented, *"I am used to using the (W, S, D, A) buttons to move around in games."* Additionally, Participant 2 inquired, *"Does the Shift button speed up the character?"*, reflecting the common practice of using the Shift key to increase the speed in games. In our game, we utilize arrow keys for movement and the Shift button to enhance movement speed, aligning with standard video game conventions. However, providing additional explanations may be necessary, especially for novice users unfamiliar with these conventions. Two of our participants required instructions on manoeuvring the avatar, while three participants were unsure how to begin, necessitating an explanation of the functionality represented by the upper-right pointer, as depicted in Fig 1(a).



قدمها بهنود بن سحوان ويعرف بعلي بن الشاه الفارسي. ذكر فيها السبب الذي من
أجله عمل بيدبا الفيلسوف الهندي رأس البراهمة لدبشليم ملك الهند كتابه الذي سماه
كليله ودمنة؛ وجعله على ألسن البهائم والطير صيانة لغرضه فيه من العوام، وضنا بما
ضمنه عن الطغام؛ وتنزيها للحكمة وفنونها، ومحاسنها وعيونها؛ إذ هي للفيلسوف
مندوحة، ولخاطر مفتوحة؛ ولمحبي تثقي، ولطالبي تشريف. وذكر السبب الذي من أجله
أنفذ كسري أنوشروان بن قباذ بن فيروز ملك الفرص بروزيه رأس الأطباء إلى بلاد الهند
كتاب كليله ودمنة؛ وما كان من تلطف بروزيه عند دخوله إلى الهند؛ حتى حضر إليه
الرجل الذي أستنسخه له سرا من خزانة الملك ليلا، مع ما وجد من كتب علماء الهند.
وقد ذكر الذي كان من بعثة بروزيه إلى مملكة الهند لأجل نقل هذا الكتاب؛ وذكر فيها ما
يلزم مطالعه من إتقان قراءته والقيام بدراسته والنظر إلى باطن كلامه؛ وأنه إن لم يكن
كذلك لك يحصل على الغاية منه. وذكر فيها حضور بروزيه قراءة الكتاب جهرا. وقد ذكر
السبب الذي من أجله وضع بزرجمهر بابا مفردا يسمى باب بروزيه التطبب، وذكر فيه
شأن بروزيه من أول أمره وآن مولده إلى أن بلغ التأديب، وأحب الحكمة واعتبر في
أقسامها. وجعله قبل باب الأسد والثور الذي هو أول الكتاب.

هل النص المظلل باللون الأحمر محيل على نص سابق ؟ فى حال كان الجواب نعم اختر
أقرب نص مُحال عليه ثم اضغط زر التسليم ؟

(a) The initial scoring system presented on the menu layer, on the top left side.



لديك 17 إجابة عن الإحالة المشتركة وهي قيد التحقق.
ستقوم بالحصول على +25 نقطه في شريط المستوى عن كل إجابة تتبين أنها صحيحة.

تعريف الإحالة المشتركة: اضغط هنا

(b) The scoring system was updated, and descriptive info is added for the scoring process and coreference annotation.

Figure 1: Improved game scoring clarity: Usability test enhancements.

Another example of familiar design is allowing players to close instructional prompts permanently or view them at their discretion. For instance, Participant 1 expressed frustration with the repetitive annotation task pop-ups, commenting, *"It bothers me that the instructions keep showing. I am used to having the option of never showing that again."* Similarly, Participant 5 was displeased with the frequent closure of these pop-ups. However, despite these concerns, having instructional pop-ups is crucial, particularly for the coreference task. This is because players often tend to dismiss instructions without reading them, as highlighted in a study by (Fraser, 2015), where it was noted that students frequently close instructions without fully engaging with them. As a result, it was recommended to implement a pop-up before each task to ensure that players are adequately informed.

Fewer Chunks of Text Theme: Having fewer chunks of text is better for increasing reading comprehension and avoiding player frustration, as this was tested in the onboarding phase of *Lingotowns* (Althani et al., 2022). This design was followed by *PlayCoref* (Hladká et al., 2009) and *Wormingo* (Kicikoglu et al., 2019), English coreference annotation games. This technique is employed in *Wormingo* in the form of "chunks". In our preliminary study, P2 commented, *"There are too many linguistic questions for a single scroll. I am spending too much time on that, and it feels like a task rather than a game."* Also, P3 said *"The coreference task is just overwhelming; there are too many questions in a single chest."*. In addition, the rest of the players suggested making the task less overwhelming.

4.1.2. Reshaping and Adding Game Design Elements Theme

Reshape the Reward System Theme: Insufficient guidance in games can lead to player frustration. According to (Miller and Cooper, 2022), many issues encountered in citizen science games stemmed from designers failing to convey critical scientific concepts to players, resulting in frustration. Participant 1 expressed confusion regarding the game's dual scoring systems:

"Why do we have two scoring systems? I understand that the first scoring system is for puzzle points, but what does the other do?... It seems like the other one is used for answering the annotation questions, but I still don't understand why I receive points for each answer I submit. I even tried submitting a wrong answer and still received a point. Could this incentivize players to provide any answer to earn points?"

The challenge is that scoring for the annotation section is not immediate, as correct answers are not known immediately. Instead, all player-provided answers are recorded under the second scoring system, represented by a scroll icon (see Figure 1). Once validated, players receive additional points under this system without explanation. All participants highlighted the need to clarify why there are two scoring systems and how scoring is calculated.

Reshape the Feedback Theme: Feedback in Games-with-a-Purpose (GWAPs) is crucial as it impacts player retention, as players desire recognition for their contributions and reassurance that they are making a difference. This finding was corroborated by a citizen science game interviews, highlighting factors contributing to player immersion in the game world (Miller and Cooper, 2022). In the design of *Stroll-with-a-Scroll*, feedback is provided to players while solving puzzles, with a checkmark indicating a correct answer and a cross indicating an incorrect one. However, further improvements are necessary, as Participant 4 suggested: *"The feedback for the puzzles was too quick. It needs to be slowed down."* Additionally, Participant 3 commented on the puzzle scoring: *"There is varying difficulty between the two presented puzzles, the fill-in-the-blank and the word search puzzle, and therefore, there should be varying scoring based on difficulty and the time it takes to solve the puzzle."*

Add New Game Elements: Using leaderboards and assigning levels based on points is an effective motivator, with users often viewing these as targets to strive for (Lee et al., 2013; Von Ahn and Dabbish, 2004, 2008). Participant 5 emphasized the importance of leaderboards, stating, *"I think it is important to have a leaderboard as most games include that."* Additionally, Participant 1 suggested locking access to leaderboards for players until they reach a certain level, while Participant 7 underscored the significance of this feature. Moreover, a few players suggested incorporating puzzles within the virtual world, allowing players to explore the landscape while solving puzzles.

5. Player-Centred Design: Insights from Usability Testing for Game Development

Moving beyond basic understanding and truly grasping players' personalities when introducing gamification is crucial. This understanding aids in creating a user-friendly game that effectively motivates players. One method to achieve this is through player personas.

Player personas are not merely demographic profiles or stereotypes but crafted from authentic data from surveys, interviews, analytics, user testing, and other reliable sources. These personas capture players' motivations, frustrations, pain points, and aspirations, as well as provide insights into their gaming habits, preferences, and playing styles. As outlined in (Guzman-Mendoza et al., 2021), player personas are developed by studying and understanding player behavior. By observing how players navigate and interact with game mechanics, their patterns and interactions are analyzed to create meaningful personas.

5.1. Persona Profiling

To redesign our game, we developed player personas through in-depth interviews. After the usability test, participants were interviewed in person for about 20-minute in a semistructured format regarding their educational background, employment status, their play experiences and game skills, and their fluency in Modern Standerd Arabic (MSA). Examples of asked questions were "How long have you been playing games, and what types of games do you typically enjoy?", "Can you describe your level of proficiency in Modern Standard Arabic? and can you speak confidently in MSA during conversations or presentations?" and "Would you consider to participate in NLP based GWAP? what particular features or aspects of the game would encourage your participation? and what features might hinder your motivation to participate?". Also, participants were asked for their input, on how to enhance the current design to address any concerns they had raised. The interviews were audio-recorded and then transcribed for further analysis.

Our analysis revealed the goals,and challenges and preferred playing modes participants expressed during the interviews and game testing. This information is summarized in Table 1. Below are the key steps in our methodology for creating player personas:

- **Participants:** The sample comprised 8 participants: 5 students (2 undergraduates with gaming experience, 3 graduate students: one was an unemployed gamer and the other two were employees with gaming background ; one of these employees is a linguistic researcher), 1 unemployed individual had a high school degree with a gaming background, another employee individual had a bachelor's degree with no gaming experience but a strong linguistic background, and one employee held a master's degree with some gaming background and strong linguistic skills.
- **Instrument:** We devised a Face-to-face in-depth interview methodology in a semistructured format regarding their educational background, employment status, their play experiences, game skills, and their fluency in MSA.
- **Procedure:** The interviews were audio-recorded to facilitate transcription of participants' comments and aid in analysis. We analyzed data to define the player persona based on Demographic Information, Professional Information and Playing Culture.
- **Data Analysis:** Our analysis involved identifying categories and codes based on constructs

	The Linguistic Enthusiastic	The Novice Player	The Gamer
Goal	They aim to improve their annotation skills and contribute to research projects in the field of NLP. Their aspiration is to collaborate with researchers and make contributions to the linguistic community.	They are interested in the notion of contributing to research through annotations but feel intimidated by the idea of using technological platforms.	They are looking for gaming experiences that suit their preferences, providing captivating gameplay mechanics without the need for complicated annotation tasks.
Pain Points	Balancing their workload and personal responsibilities while devoting time to annotations can be quite difficult for them.	They find it difficult to navigate and interact with interfaces related to gaming platforms.	They feel overwhelmed by the complexity of NLP annotation tasks.
Individual Achievement vs. Team Achievement	While they place importance on growth by enhancing their annotation skills and contributing meaningfully to research projects, they also recognize the value of teamwork in achieving research goals.	They value the opportunity to collaborate with others and benefit from their expertise while collectively working towards shared research objectives.	While they appreciate achievement in mastering gaming skills and conquering in-game challenges, some gamers also find joy in collaborating with other players, while others enjoy a competitive environment.

Table 1: Player-Persona Insights.

to extract information for designing the player persona.

6. Evolution of the Game: Stroll-with-a-Scroll(Version 2)

Based on the generated personas, we have redesigned the game into its second version. Firstly, addressing the ‘More Familiarity’ theme, we aimed to include novice players in the design process by adding clear directions on how to play. Drawing from a detailed framework of design strategies for enhancing learnability in video games (Poretski and Tang, 2022), we introduced just-in-time reminders (as shown in Fig 2), contextual prompts appearing in specific game situations that vanish once performed by the player. These prompts guide players on how to move around and what actions to take, eliminating the need to memorize instructions before gameplay. They appear only once before a new action is required. Additionally, we carefully considered the needs of gamers and experts who are always on the move during gameplay. Addressing participants’ complaints about the lack of an option to ‘never show again’ or ‘hide’ task descriptive pop-ups and the need for a clear definition of coreference with examples, we introduced the coreference task description at the start and placed it within the scroll icon (as depicted in Fig 1(b)). Clicking on the scroll icon directs players to the task description, allowing them to view it upon request. To ensure players are aware of the description location, we added guidance at the start of gameplay, clarifying that they can refer to the task description by clicking on the scroll icon. Additionally, in response to player requests, we added (W, S, D, A) buttons for avatar movement control.

Secondly, to address the issue of overwhelming text highlighted in the ‘Less chunks of text’ theme, we adopted the chunk size approach used in *Wormingo* (Kicikoglu et al., 2019), as ‘The Gamer’ group of participants, the most intimidated by text size, did not report feeling overwhelmed during their experience with the game. When the player opens the scroll, the text is presented in chunks, one after the other. Each chunk contains a maximum of 50 words, ensuring complete sentences are displayed. Additionally, we implemented a gradual display of words, simulating an animated effect similar to *Wormingo*, to reduce cognitive load on players (Kicikoglu et al., 2019). We introduced virtual world puzzles to mitigate text overload raised mostly by ‘The Gamer’ group and to address the ‘Add New Game Elements’ theme. In these puzzles, players are tasked with searching the scene for lost letters in the scroll. The game presents three lost letters forming the word ‘day’. As players search the scene, each missing letter is revealed with an Arabic coffee cup (as shown in Fig. 3). When a player finds a letter, it moves from the scene to be placed on top of the Arabic coffee cup, ultimately completing the word. Players have the option to hide and reveal the text by pressing the eye icon (as depicted in Fig. 3). Additionally, they can skip playing the game part by pressing the ‘Skip’ button, allowing those focused on annotation; ’The linguistic enthusiastic’ group to continue without participating.

Thirdly, participants requested further explanations regarding the reward systems and their calculation processes, highlighting the need for a more intuitive presentation. Two rewarding systems were identified: instant rewards for solving puzzles and delayed rewards for solving annotations, as described in the ‘Reshape the reward system’ theme. To address this issue, we made adjustments to the menu depth items and the gameplay scene. We introduced a level bar, suggested by ‘The Gamer’ group, to mark progress providing instant points for solving puzzles while temporarily recording an-

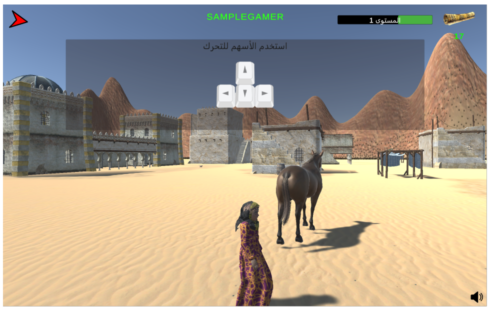


Figure 2: Directions on how to play, presented on the menu layer of the game.

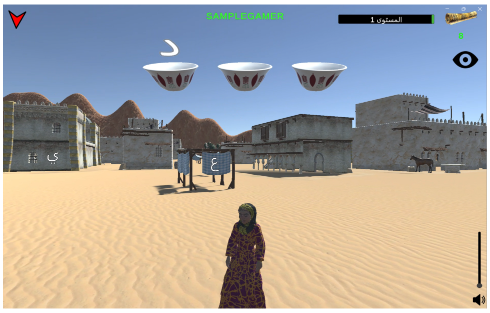


Figure 3: Lost letters puzzle: Scene search for the letters of the lost word.

notation answers within the scroll menu item until validated and presented within the progress bar (as shown in Fig. 1(b)). To clarify these rewarding systems we instructed players to click on the scoring systems at the start of gameplay: the level bar and then the scroll icon, where they received clarifications on the calculations and why instant points were not awarded for the annotation task. Additionally, the description of the coreference task was provided there for players to access as needed, eliminating pop-ups before each task. In response to the competitive nature of some players, a leaderboard was added to the home screen, addressing the 'Add New Game Elements' theme.

Furthermore, some players expressed a desire to enhance the enjoyment of game feedback, referring to it as "game juiciness", seeking elements that excite them. This was discussed in the 'Reshape the feedback Theme'. Game juiciness involves providing visual and audio feedback to induce a positive player experience (Rollings and Morris, 1999). In the initial version of the game, background music was included, with players able to control the sound level or mute it. Feedback was displayed as a check mark for correct answers and a cross mark for incorrect ones. However, players found this feedback too quick to absorb, prompting us to slow it down and add audio feedback for success and failure. We also implemented animated scoring similar to *Wormingo*, where correct answers are rewarded with an animated score, transitioning from the challenge to the corresponding reward system. Additionally, scoring now varies based on puzzle difficulty, with players receiving a more valuable animated treasure box for answering virtual world puzzles.

7. Redesign Validation

After making improvements, to our GWAP to make it more user friendly, it is important for us to carefully evaluate the effectiveness of these changes through thorough usability testing. In this section we will provide an explanation of how we validate the redesign and the methods we use to assess the systems usability.

We selected same series of tasks given in the first usability test. We had 3 participants, who were asked to perform the predefined tasks while thinking aloud. Following each task, participants were interviewed to gather feedback on their overall experience, usability challenges faced, and suggestions for improvement.

Based on the usability testing it seems that the redesign successfully enhanced the user friendliness of our GWAP, as participants successfully completed the task independently without raising any concerns about the issues that were identified in the first usability test.

8. Linguistic Acceptability Study

A debate persists regarding the use of expert versus non-expert annotators and the reliability of different crowdsourcing strategies in the realm of NLP annotation tasks. To address this, we tested our annotations' reliability to assess our GWAP's reliability. We aim to share these results with other researchers to encourage linguists to participate in annotating our GWAP and to disseminate them widely.

8.1. The data

Our objective is to compare players' judgments with those of experts, so our players annotated a gold standard document extracted from the OntoNotes 5.0 datasets. OntoNotes is widely utilized for coreference resolution (R. et al., 2014; Björkelund and Kuhn, 2014; Martschat and Strube, 2015; Clark and Manning, 2015, 2016a,b; Lee et al., 2017, 2018) and has been a key resource since the CoNLL 2011 and 2012 shared tasks (Pradhan et al., 2011). It encompasses documents in three languages: Arabic (300K tokens), Chinese (950K tokens), and English (1.6M tokens), spanning various genres, with news being the predominant genre. Our study used a single 'Art News'

CoNLL document containing MSA text annotated with coreference.

8.2. Participants and Procedure

We aim to evaluate whether our virtual world game, *Stroll-with-a-Scroll*, can effectively collect linguistically acceptable coreference annotations. To achieve this, we conducted an experiment in August 2023 to compare the annotations provided by naive participants (our participants) with those of expert annotators.

We recruited some of our participants (N=77) through Prolific, a platform for online participant recruitment. We used the demographic filters provided by the platform, to selectively enroll participants whose first language was Arabic. This measure was implemented to mitigate potential confounding variables that might impact the accuracy metrics within our research investigation. Participating individuals were paid £7 (£12 per hour) upon successful completion of a 35-minute study entitled "Study about a Game-with-a-Purpose." Additionally, we enlisted volunteers (N=29) who received invitation emails, and whose first language is Arabic. These emails were sent to academic faculty in Saudi universities, requesting them to share the game with their students. Due to technical constraints, the experimental protocol could only be executed on desktop or laptop web browsers such as Chrome and Firefox; consequently, participation via mobile devices was not feasible due to these limitations.

Of the 106 participants who completed the demographic questionnaire, 44.34% were female, and 55.66% were male. Regarding age distribution, 28.30% were aged 18 to 24, 46.23% were aged 25 to 34, 15.09% were aged 35 to 44, 8.49% were aged 45 to 54, and 1.88% were aged 55 and above. The participants comprised 23.58% Saudis, 13.20% Lebanese, and 13.20% Syrians. The remainder represented various nationalities, including Algerian, Iraqi, Jordanian, Moroccan, Palestinian, Somali, Sudanese, and Tunisian.

In our annotation task, players are presented with a text window highlighting a specific word or phrase in red. Their initial task is to determine whether the highlighted word or phrase is newly introduced to the conversation or if it refers to something previously mentioned. If it refers to a previous mention, players must locate it by selecting one of the highlighted ones in blue. Once the player has made their selection, they can submit their annotation by clicking on the Submit button. During the validation mode, players confirm other players' answers, which is activated only when players submit different answers. The experiment concluded after 47 markables were annotated, and all answers were aggregated using MPA (Mention Pair Annotation) (Paun et al., 2018) and stored in an inline XML file.

8.3. Analysis and Results

The agreement between naive annotators and linguists is 91.49% overall accuracy, calculated by comparing the markable in the generated XML file with the gold standard file. This result is excellent. Additionally, our Cohen's kappa coefficient, a more robust measure accounting for the possibility of chance agreement, is 0.787 (Cohen, 1960), indicating substantial agreement. Our participants failed to answer 4 out of the 47 presented mentions.

In terms of precision, recall, and F-Measure, players' annotations were compared to the gold standard (the OntoNotes annotation). The data suggested that Precision, Recall, and F-Measure collectively evaluate annotation accuracy and completeness, with a balanced score of 0.84615 indicating both precision and recall around 85% for players' annotations. Furthermore, we compared individuals who were paid and those who volunteered for our experiment, specifically examining their accuracy levels when completing annotation tasks. Our analysis revealed that both paid and volunteer participants achieved similar accuracy scores. This consistency across participant groups demonstrates our GWAP's strength and dependability.

9. Discussion

In this article, we have discussed two factors that contributing to the success of GWAPs: user-friendliness and the reliability of the generated data. First, to ensure user-friendliness, we conducted a usability study that helped us create personas and guide our design process accordingly. From this study, we generalized our findings to inform the design of other NLP annotation games, as we aim to answer our first research question: “In the context of 3D games, particularly focusing on the interface/menu layer, what design elements, interaction techniques, and user experience factors improve the usability and productivity of the player?”

Firstly, introducing breaks between annotations enhances overall enjoyment, but it's essential to make these breaks optional for participants who prioritize contributing over gameplay. Secondly, instructions and tutorials should be concise, quick to understand, and easily accessible during gameplay to ensure a seamless experience without interruptions. Thirdly, clear explanations of the calculation process for annotation tasks reduce frustration and enhance understanding among participants. Fourthly, incorporating visually and audibly satisfying feedback mechanisms for player actions improves engagement. Fifthly, incorporating both competitive and collaborative elements is recommended to accommodate diverse preferences and play styles. Finally, simplifying NLP tasks by breaking them into smaller, manageable tasks enhances user involvement and potentially creates more reliable data.

Usability is an iterative process, and we actively seek more participants to conduct further tests. These tests may involve creating or updating personas with more detailed information. By improving our understanding of user preferences, our goal is to make our GWAP more user-friendly and effective.

We tested the reliability of our generated corpora to answer our second research question, "Could our GWAP be used to collect linguistically acceptable coreference annotation?" We achieved excellent results by assigning more weight to reliable players when aggregating annotation answers (Paun et al., 2018) instead of simply annotating with the value submitted the most. Out of all markables presented, our participants failed to annotate only 4, which is less than 10% of the total.

10. Concluding Remarks

This paper presents two evaluations of a 3D virtual world game designed for NLP annotation. First, we conducted a preliminary study to improve user experience and identify design flaws. The usability test involved observing how users interacted with the system and identifying areas for enhancement or correction to minimize dropouts. Tasks were assigned to participants, who provided feedback using the think-aloud protocol. We redesigned the game tailored to player personas based on qualitative research findings. Secondly, we evaluated the reliability and acceptability of the game for collecting annotations by comparing aggregated player feedback to the OntoNotes 5.0 gold standard corpus. Our analysis indicates that annotations produced through the game are of acceptable quality.

11. Acknowledgements

Wateen Aliady is supported by the Saudi Arabian Cultural Bureau in the UK and Imam Mohammad Ibn Saud Islamic University (IMSIU) in Saudi Arabia. Massimo Poesio was in part supported by the ARCIDUCA project, funded by European Research Council (EPSRC).

12. Bibliographical References

G. O. Abada and E. A. Onibere. 2009. The effect of rehearsed computer use on icon recognition. *International Journal of Computers and Applications*, 31:9–15.

T. Alshammari, O. Alhadreti, and P. Mayhew. 2015. When to ask participants to think aloud: A comparative study of concurrent and retrospective think-aloud methods. *International Journal of Human Computer Interaction*, 6(3):48–64.

F. Althani, C. Madge, and M. Poesio. 2022. Less text, more visuals: Evaluating the onboarding phase in a gwap for nlp. *Proceedings of the 9th Workshop on Games and Natural Language Processing within the 13th Language Resources and Evaluation Conference*, pages 17–27.

E. Amspoker and M. Petruck. 2022. A gamified approach to frame semantic role labeling. In *Proceedings of the Fourth Workshop on Data Science with Human-in-the-Loop (Language Advances)*, page 37–42.

A. Björkelund and A. Kuhn. 2014. Learning structured perceptrons for coreference resolution with latent antecedents and non-local features. In *Proceedings of the 52nd ACL*, volume 1, page 47–57.

F. Bonetti, E. Leonardelli, D. Trotta, G. Raffaele, and S. Tonelli. 2022. Work hard, play hard: Collecting acceptability annotations through a 3d game. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1740–1750. European Language Resources Association.

F. Bonetti and S. Tonelli. 2020. A 3d role-playing game for abusive language annotation. In *Workshop on Games and Natural Language Processing*, pages 39–43.

J. Bouta Cruz-Benito, R. Theron, F. J. Garcia-Penalvo, and E. P. Lucas. 2015. Discovering usage behaviors and engagement in an educational virtual world. *Computers in Human Behavior*, 47:18–25.

A. Bowser, D. Hansen, Y. He, C. Boston, M. Reid, L. Gunnell, and J. Preece. 2013. Using gamification to inspire new citizen science volunteers. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications, Gamification'13, NewYork,NY,USA, October*, page 18–25. Association for Computing Machinery.

V. Braun and V. Clarke. 2019. Reflecting on reflexive thematic analysis. *Qualitative Research in Sport, Exercise and Health*, 11:589–597.

V. Braun and V. Clarke. 2021. One size fits all? what counts as quality practice in (reflexive) thematic analysis? *Qualitative Research in Psychology*, 18:328–352.

P. Bui, G. Rodríguez-Aflecht, B. Brezovszky, M. M. Hannula-Sormunen, S. Laato, and E. Lehtinen. 2020. Understanding students' game experiences throughout the developmental process of the number navigation game. *Educational Technology Research and Development*, 68:2395–2421.

N. Chaiko, S. Sepanta, and R. Zamparelli. 2022. The "actors challenge" project: Collecting data on intonation profiles via a web game. In *Proceedings of the 9th Workshop on Games and Natural Language Processing within the 13th Language Resources and Evaluation Conference*, page 49–53. European Language Resources Association.

J. Chamberlain, M. Poesio, and U. Kruschwitz. 2008. Phrase detectives: A web-based collaborative annotation game. In *Proceedings of the International Conference on Semantic Systems (I-Semantics' 08)*, pages 42–49.

J. C. Chen and S. Kent. 2020. Task engagement, learner motivation and avatar identities of struggling english language learners in the 3d virtual world. *System*, 88:102168.

K. Clark and C. D. Manning. 2015. Entity-centric coreference resolution with model stacking. In *Proceedings of the ACL*.

K. Clark and C. D. Manning. 2016a. Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of EMNLP*.

K. Clark and C. D. Manning. 2016b. Improving coreference resolution by learning entity level distributed representations. In *Proceedings of the ACL*.

J. Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

D. Dziedzic. 2016. Use of the free to play model in games with a purpose: the robocorp game case study. *Bio-Algorithms and Med-Systems*, 12:187–197.

M. Fan, S. Shi, and K. N. Truong. 2020. Practices and challenges of using think-aloud protocols in industry: An international survey. *Journal of Usability Studies*, 15(2).

K. Fort, B. Guillaume, and H. Chastant. 2014. Creating zombilingo, a game with a purpose for dependency syntax annotation. In *Proceedings of the First International Workshop on Gamification for Information Retrieval*, pages 2–6.

K. Fort, B. Guillaume, Y. Pilatte, M. Constant, and N. Lefèbvre. 2020. Rigor mortis: Annotating mwes with a gamified platform. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, page 4395–4401. European Language Resources Association.

N. Fraser. 2015. Ten things we've learned from blockly. *IEEE Blocks and Beyond Workshop (Blocks and Beyond)*.

E.R. Gouveia, M. Nascimento, C. França, P. Campos, A. Ihle, K. Przednowek, A. Marques, N. Nunes, and B.R. Gouveia. 2023. Correlates of presence in a virtual reality gamification environment for rehabilitation after musculoskeletal injury. *PRESENCE: Virtual and Augmented Reality*.

I.E. Guzman-Mendoza, M. Mirna, C.-R. Héctor, and M. Jezreel. 2021. Designing a player-persona for gamification learning experiences. *CEUR Workshop Proceedings*.

J. Hamari and L. Keronen. 2017. Why do people play games? a meta-analysis. *International Journal of Information Management*, 37(3):125–141.

B. Hladká, J. Mírovský, and P. Schlesinger. 2009. Play the language: Play coreference. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 209–212.

D. Jurgens and R. Navigli. 2014. It's all fun and games until someone annotates: Video games with a purpose for linguistic annotation. *Transactions of the Association for Computational Linguistics*, 2:449–464.

A. Kawrykow, G. Roumanis, A. Kam, D. Kwak, C. Leung, C. Wu, E. Zarour, L. Sarmenta, M. Blanchette, and J. Waldispuhl. 2012. Phylo: A citizen science approach for improving multiple sequence alignment. *PLOS ONE*.

D. Kicikoglu, R. Bartle, J. Chamberlain, and M. Poesio. 2019. Wormingo: a 'true gamification' approach to anaphoric annotation. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, pages 1–7.

Y.E. Kim, E.M. Schmidt, and L. Emelle. 2008. Moodswings: A collaborative game for music mood label collection. In *International Society for Music Information Retrieval Conference*.

R. Kleffner, J. Flatten, A. Leaver-Fay, D. Baker, J. B. Siegel, F. Khatib, and S. Cooper. 2017. Foldit standalone: a video game-derived protein structure manipulation interface using rosetta. *Bioinformatics*, 33:2765–2767.

T. Kohler, J. Fueller, K. Matzler, D. Stieger, and J. Füller. 2011. Co-creation in virtual worlds: The design of the user experience. *MIS Quarterly*, page 773–788.

L. Kougioumtzian, K. El Raheb, A. Katifori, and M. Roussou. 2022. Blazing fire or breezy wind? a story-driven playful experience for annotating dance movement. *Frontiers in Computer Science*, 4:957274.

M. Krause, A. Takhtamysheva, M. Wittstock, and R. Malaka. 2010. Frontiers of a paradigm: exploring human computation with digital games. In *Proceedings of the acm sigkdd workshop on human computation*, pages 22–25.

M. Lafourcade, A. Joubert, and N. Le Brun. 2015. *Games with a Purpose (GWAPS)*. John Wiley and Sons.

T. Laine and R. Lindberg. 2020. Designing engaging games for education: A systematic literature review on game motivators and design principles. *IEEE Transactions on Learning Technologies*, 13(4):804–821.

K. Lee, L. He, M. Lewis, and L. Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of EMNLP*.

K. Lee, L. He, M. Lewis, and L. Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of ACL*.

T. Y. Lee, C. Dugan, W. Geyer, T. Ratchford, J. Rasmussen, N. S. Shami, and S. Lupushor. 2013. Experiments on motivational feedback for crowdsourced workers. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 341–350.

Y. Lee and A. N. K. Chen. 2011. Usability design and psychological ownership of a virtual world. *Journal of Management Information Systems*, 28:269–308.

C. Lewis. 1982. Using the “thinking-aloud” method in cognitive interface design. *IBM TJWatson Research Center Yorktown Heights*.

C. A. Lindley, C. Sennersten, et al. 2008. Game play schemas: from player analysis to adaptive game mechanics. *International Journal of Computer Games Technology*, 2008.

C. Madge, R. Bartle, J. Chamberlain, U. Kruschwitz, and M. Poesio. 2019a. The design of a clicker game for text labelling. In *2019 IEEE Conference on Games (CoG)*, pages 1–4. IEEE.

C. Madge, R. Bartle, J. Chamberlain, U. Kruschwitz, and M. Poesio. 2019b. Incremental game mechanics applied to text annotation. In *in Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 545–558.

S. Martschat and M. Strube. 2015. Latent structures for coreference resolution. *Transactions of the Association for Computational Linguistics*, 3:405–418.

S. McDonald, T. Zhao, and H. M. Edwards. 2013. Dual verbal elicitation: the complementary use of concurrent and retrospective reporting within a usability test. *International Journal of Human-Computer Interaction*, 29(10):647–660.

J. A. Miller and S. Cooper. 2022. Barriers to expertise in citizen science games. In *CHI Conference on Human Factors in Computing Systems*, pages 1–25.

J. A. Miller, U. Narayan, M. Hantsbarger, S. Cooper, and M. S. El-Nasr. 2019. Expertise and engagement: re-designing citizen science games with players’ minds in mind. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, pages 1–11.

K. Morrison, M. Jain, J. Hammer, and A. Perer. 2023. Eye into ai: Evaluating the interpretability of explainable ai techniques through a game with a purpose. In *Proceedings of the ACM on Human-Computer Interaction 7(CSCW2)*, pages 1–22.

C. Mount Cieri, J. Fiumara, and J. Wright. 2020. Using games to augment corpora for language recognition and confusability. *Interspeech*, pages 1887–1891.

S. Paun, J. Chamberlain, U. Kruschwitz, J. Yu, and M. Poesio. 2018. A probabilistic annotation model for crowdsourcing coreference. In *http://aclweb. org/anthology/D18-1000*, pages 1926–1937.

M. Poesio, J. Chamberlain, U. Kruschwitz, L. Robaldo, and L. Ducceschi. 2013. Phrase detectives: Utilizing collective intelligence for internet-scale language resource creation. *ACM Trans. Interact. Intell. Syst.*, 3(1):Article 3.

M. Poesio, J. Chamberlain, S. Paun, J. Yu, A. Uma, and U. Kruschwitz. 2019. A crowdsourced corpus of multiple judgments and disagreement on anaphoric interpretation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2019–06–02 – 2019–06–07, Minneapolis, Minnesota.

L. Poretski and A. Tang. 2022. Press a to jump: Design strategies for video game learnability. *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*, pages 155, 1–26.

S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, and Zhang Y. 2011. Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea.

Eraldo R., C´ıcero N. dos Santos Fernandes, and Ruy L. Milidi´u. 2014. Latent trees for coreference resolution. *Computational Linguistics*, 40(4):801–835.

M. Rajanen and R. Dorina. 2017. Usability benefits in gamification. In *GamiFIN Conference*.

A. R. Roberts, B. De Schutter, K. Franks, and M. E. Radina. 2019. Older adults' experiences with audiovisual virtual reality: Perceived usefulness and other factors influencing technology acceptance. *Clinical gerontologist*, 42(1):27–33.

A. Rollings and D. Morris. 1999. *Game Architecture and Design*. Paraglyph Press.

A. C. Tomé Klock, E. J. de Borba, I. Gasparini, D. Lichtnow, Pimenta M. S., and G. Rodriguez. 2017. Evaluation of usability and user experience regarding the gamification of educational systems. In *Twelfth Latin American Conference on Learning Technologies (LACLO)*, pages 1–8.

M. J. Van den Haak and M. D. T. De Jong. 2003. Exploring two methods of usability testing: concurrent versus retrospective think-aloud protocols. In *IEEE International Professional Communication Conference, 2003. IPCC 2003. Proceedings.*, pages 3–pp. IEEE.

D. Vannella, D. Jurgens, D. Scarfini, D. Toscani, and R. Navigli. 2014. Validating and extending semantic knowledge bases using video games with a purpose. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1294–1304.

N. Venhuizen, K. Evang, V. Basile, and J. Bos. 2013. Gamification for word sense labeling. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)*.

L. Von Ahn. 2006. Games with a purpose. *Computer*, 39:92–94.

L. Von Ahn and L. Dabbish. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326.

L. Von Ahn and L. Dabbish. 2008. Designing games with a purpose. *Communications of the ACM*, 51:58–67.

L. Von Ahn, M. Kedia, and M. Blum. 2006a. Verbosity: a game for collecting common-sense facts. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 75–78.

L. Von Ahn, R. Liu, and M. Blum. 2006b. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 55–64.

L. Xu, E. Dhonnchadha, and M. Ward. 2022. Faoi gheasa: An adaptive game for irish language learning. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, page 133–138. Association for Computational Linguistics.

Riddle Me This: Evaluating Large Language Models in Solving Word-Based Games

Raffaele Manna, Maria Pia di Buono, Johanna Monti
UniOr NLP Research Group
University of Naples "L'Orientale", Italy
{raffaele.manna, mpdibuono, jmonti}@unior.it

Abstract

In this contribution, we examine the proficiency of Large Language Models (LLMs) in solving the linguistic game "La Ghigliottina," the final game of the popular Italian TV quiz show "L'Eredità". This game is particularly challenging as it requires LLMs to engage in semantic inference reasoning for identifying the solutions of the game. Our experiment draws inspiration from Ghigliottin-AI, a task of EVALITA 2020, an evaluation campaign focusing on Natural Language Processing (NLP) and speech tools designed for the Italian language. To benchmark our experiment, we use the results of the most successful artificial player in this task, namely Il Mago della Ghigliottina. The paper describes the experimental setting and the results which show that LLMs perform poorly.

Keywords: Large language model, Ghigliottin-AI, Word-based Games

1. Introduction

Researchers in Artificial Intelligence (AI) and Natural Language Processing (NLP) have shown interest in Language games, which derive their challenge and excitement from the complexity and ambiguity of natural language. A particular challenging language game is "La Ghigliottina", the final game of the popular Italian TV quiz show "L'Eredità". The game involves a single player, who is given a set of five words (clues), unrelated one to each other, but related with a sixth word that represents the solution to the game. In 2020 EVALITA, a recurring evaluation campaign focusing on NLP and speech tools designed for the Italian language, proposed the Ghigliottin-AI task (Basile et al., 2020) to assess artificial agents in the solution of "La Ghigliottina". Participants in Ghigliottin-AI are asked with developing an artificial player capable of solving the linguistic challenges presented in the game "La Ghigliottina". In the aftermath of the Ghigliottin-AI task, this contribution aims to examine the ability of cutting-edge Large Language Models in solving the Ghigliottina game, which involves inferring the solution through identifying the hidden semantic connections with the provided clues. This paper is organized as follows: in Section 2 we briefly present the use of games in testing the reasoning and inference abilities of NLP and AI systems. In Section 3 we present the Ghigliottin-AI task and the results obtained by the artificial players that took part in the task. In Section 4 we provide all the information (data, LLM models and prompts) concerning our experimental settings to evaluate the abilities of different LLMs in solving the GhigliottinAI language game. Discussion of results is presented in Section 5. Conclusions are in Section 6.

2. Related Work

In this section, we briefly survey the use of games as a means to assess the efficacy of NLP tools in problem-solving tasks. Some achievements in artificial intelligence are linked to games such as for instance *Jeopardy*, where contestants respond to clues in the form of answers by phrasing their replies as questions. In 2011, IBM's Watson DeepQA computer defeated the show's two foremost all-time champions of this game (Ferrucci et al., 2013). In particular, language games, such as the *Wheel of Fortune* or *Who Wants to be a Millionaire?* (Lam et al., 2012) (Molino et al., 2015), have been used as means to assess the capabilities of NLP and AI systems, as they provide an interesting and challenging playground to evaluate their reasoning and inference capabilities (Yannakakis and Togelius, 2018). Another particularly appealing game is solving crossword puzzles. A first attempt is *Proverb* (Littman et al., 2002), which leverages extensive repositories containing clues and solutions to past crossword puzzles. *WebCrow* (Ernandes et al., 2008), the first solver for Italian crosswords, instead, relies mainly on information sourced from the Web, and a set of previously solved games.

As mentioned in the Introduction the *Ghigliottina* game is particularly challenging and has inspired various scholars in solving it. In (Semeraro et al., 2012) and (Basile et al., 2014), the authors present OTTHO (On the Tip of my THOught), an artificial player for the Guillotine game. OTTHO is based on a knowledge infusion procedure that uses NLP

Games and NLP @LREC-COLING-2024, pages 97–106
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

techniques to analyze unstructured data from open web sources like Wikipedia, creating a repository of linguistic competencies and factual knowledge. In 2018 the *Mago della Ghigliottina* (Sangati et al., 2018) participated as UNIOR4NLP for the first time in the shared task NLP4FUN (Basile et al., 2018), which was part of the EVALITA 2018, a periodic evaluation campaign of NLP and speech tools for the Italian language. The system, available also as a Telegram bot,[1] relies on linguistic resources and artificial intelligence and achieves better results than human players. In addition to solving a game, *Mago della Ghigliottina* can also generate new game instances and challenge the users to match the solution. The *Mago della Ghigliottina* took part in the new edition of the NLP4FUN task, titled Ghigliottin-AI, resulting again as the best artificial player, outperforming human players and competitor artificial players (see Section 3). Recently LLMs were tested in solving *Wordle*,[2] a game owned by the New York Times, where players have six attempts to guess a five-letter word. The experiment showed that LLMs lack the inference skills needed to solve the game.

3. GhigliottinAI

As part of EVALITA 2020, the Ghigliottin-AI[3] task was organised, a new edition of the NLP4FUN task proposed in EVALITA 2018 (Basile et al., 2018), aimed at the realisation of an open competition between Artificial Intelligence (AI) systems to solve the game “La Ghigliottina". The Ghigliottin-AI task is inspired by the final game of the Italian TV show “L’Eredità". This game was chosen because it represents a very interesting test bed for AI systems focused on semantic aspects of natural language: the solution of the language game is based on the semantic relationships existing between each of the five proposed clues and the solution word. For example, given the set of Italian clues *conoscere* (to know), *grado* (degree), *modello* (model), *ideale* (ideal) and *divina* (divine) the solution is *perfezione* (perfection) because this word relates to the clues in the following way: *conoscere alla perfezione* (to perfectly know), *grado di perfezione* (degree of perfection), *modello di perfezione* (model of perfection), *ideale di perfezione* (ideal of perfection) and *perfezione divina* (divine perfection).
The underlying idea of the Ghigliottin-AI task was that artificial players for that game could take advantage from the availability of open repositories on the Web, such as Wikipedia, that provide the system with the cultural and linguistic background needed to understand clues (Basile et al., 2014; Semeraro et al., 2009, 2012). Before the competition, a set of 300 instances of the game together with their solution taken from the last editions of the TV game were provided to developers in a JSON format as training data for their players. The evaluation was carried out using a Remote Evaluation Server (RES) named Ghigliottiniamo[4], which facilitated real-time submission of solutions by both human participants and artificial systems (bots) to the TV game. Ghigliottiniamo randomly provided the test set at intervals, presenting a single game challenge to registered systems. The RES imposed a time constraint, similar to the original TV game, allowing systems to submit a single solution within 60 seconds from the challenge. Solutions received after this time frame were discarded, mirroring the time-sensitive nature of the original game. This protocol was applied consistently in evaluating systems participating in Ghigliottin-AI. Two teams participated to the competition: *Mago della Ghigliottina* (Sangati et al., 2020) and GUL.LE.VER (De Francesco, 2020).

Mago della Ghigliottina is based on the analysis of real game instances. As highlighted by the authors (Sangati et al., 2020), game instances indicate that connections between clues and solution pertain to a specific linguistic phenomenon, namely Multiword Expression (MWE)(Sag et al., 2002; Constant et al., 2017). A MWE is a sequence of words that presents some characteristic behaviour (at the lexical, syntactic, semantic, pragmatic or statistical level) and whose interpretation crosses the boundaries between words. During the analysis six patterns that identify MWEs connecting clue/solution pairs were identified:

- **A-B (Noun-Adjective, Adjective-Noun, Verb-Noun, Noun-Noun)**: *permesso premio* (‘permit price’ → good behaviour license)
- **A-determiner-B**:*dare il permesso* (‘give the permit’ → authorize)];
- **A-conjunction-B**: *stima e affetto* (esteem and affection);
- **A-preposition-B**: *colpo di coda* (‘flick of tail’ → last ditch effort);
- **A-articulated preposition-B** : *virtù dei forti* , part of the famous Italian proverb La calma è la virtù dei forti (patience is the virtue of the strong);
- **A+B**: compounds such as radio + attivita = *radioattivita‘* (radio + activity = radioactivity).

---

[1] https://t.me/Unior4NLPbot

[2] https://github.com/manavgup/wordle-llm-solver

[3] https://ghigliottin-ai.github.io/

[4] https://quiztime.net/web/#!/leaderboard1

Therefore, *Mago della Ghigliottina* explores word co-occurrencence in frequent collocations or idioms, word similarity or word relatedness as a basis of the semantic relationship of clues and solutions in a number of freely large available corpora, such as Paisà[5], itWaC[6], Wikiquote[7] and other linguistic resources. *Mago della Ghigliottina* proved to be the best performing artificial player with an accuracy score of .68.

GUL.LE.VER positioned #2 in the competition, with an accuracy score of .26 and .46 R@10, achieving results comparable to human players of the TV game. This player is based on the Glove vector representation of the words (Pennington et al., 2014) on the basis of a large collected dataset, containing the Italian Wiktionary, Wikiquote, Wikipedia (only titles), the Italian Collocations Dictionary and other resources scraped on the web containing Italian multiword expressions, proverbs and songs titles.

4. Experiment

This section presents the experimental settings to evaluate the abilities of different LLMs in solving GhigliottinAI language game. Section 4.1 presents the data on which LLMs were tested at the GhigliottinAI game, while Section 4.2 describes the LLMs and the parameters used to generate their outputs for each game instance. The outputs of the LLMs at GhigliottinAI were elicited using different prompting techniques. In Section 4.3, the different prompting techniques used are listed and examples of prompts provided to LLMs are shown. Finally, in Section 4.4, the performances obtained by the LLMs in solving GhigliottinAI using the different prompting techniques are shown.

The game instances were solved between mid-December and mid-February. During this time, the two leading AI firms, Google[8] and OpenAI[9], remained active in releasing updated versions of their respective LLMs. As in 4.4 and specifically in section 5, the updates had a notable impact on the performance of the LLMs in solving the GhigliottinAI game instances.

The game instances used to test the LLMs, as well as the solutions generated for each prompting technique, are available in this repository[10].

4.1. Data

We used data from a shared task organized as part of the Italian NLP tools evaluation campaign: Evalita [11]. Following up on Section 3, the 2020 edition of Evalita introduced a shared task named "Solving the Ghigliottina with AI," along with the release of training game instances. The test set consists of 350 game instances [12], released in an excel sheet. The excel sheet is therefore composed of 350 rows representing the games instances and 8 columns. The first column contains the game ID, columns from 2 to 6 represent the clue words, and the last column contains the solution words for each instance. In Figure 1, we provide an example of the excel file with game instances.

We used the game instances contained in the test set to evaluate the performance of the LLMs. This approach allows us to compare the performance of the LLMs to the performance of the automatic solvers presented in the shared tasks discussed in section 3. Also some game instances from the training set of GhigliottinAI were used to provide a game demonstration in some prompts to enable in-context learning (Brown et al., 2020; Min et al., 2022).

4.2. Large Language Models

In an effort to evaluate their aptitude at the GhigliottinAI game, four LLMs, including ChatGPT-3.5, ChatGPT-4, Bard and Gemini-Pro, were systematically exposed to the game. To conduct the experiments, we used Chatbot Arena[13], a benchmark platform that offers access to several LLMs via a web graphical user interface (Zheng et al., 2024).

Despite the prompting technique chosen, we crafted a block of prompts containing a number of game instances considering the maximum sample length for each LLMs. We tested the LLMs on the benchmark platform[14] using configurable parameters like Temperature (set to 0.7), Top P (set to 1), and Max Output Tokens (set to 1024). The aforementioned parameters were configured separately for each LLM.

We define a set of prompts while considering the maximum token length that can be processed by

---

[5] https://www.corpusitaliano.it/

[6] https://wacky.sslmit.unibo.it/doku.php?id=corpora

[7] https://it.wikiquote.org/wiki/Pagina_principale

[8] https://blog.google/products/gemini/bard-gemini-advanced-app/

[9] https://platform.openai.com/docs/models/continuous-model-upgrades

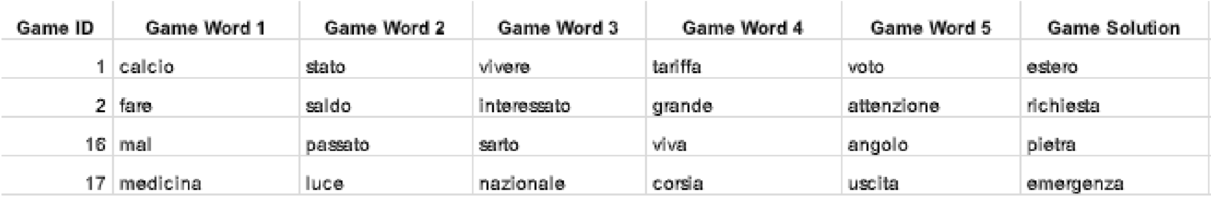
[10] https://github.com/RaffaeleMann/eval_LLMs_ghigliottina

[11] https://ghigliottin-ai.github.io/

[12] The GhigliottinAI test data are not freely accessible and are available only on request.

[13] https://chat.lmsys.org/

[14] More specifically, we used the following LLMs from the platform: GPT-4 (GPT-4-1106-preview), Bard (bard-jan-24-gemini-pro), Gemini-Pro (gemini-pro-dev-api), and GPT-3.5 (GPT-3.5-Turbo-0613).



Game ID	Game Word 1	Game Word 2	Game Word 3	Game Word 4	Game Word 5	Game Solution
1	calcio	stato	vivere	tariffa	voto	estero
2	fare	saldo	interessato	grande	attenzione	richiesta
16	mal	passato	sarto	viva	angolo	pietra
17	medicina	luce	nazionale	corsia	uscita	emergenza

Figure 1: A screenshot of the Excel file containing GhigliottinAI game instances

the LLMs. For instance, given a context size of 8,000 tokens for GPT-4, the block containing our prompts levels out at approximately 20 game instances included.

4.3. Prompts

As far as prompts are concerned, following Wang et al. (2023), we define In-Context Learning (ICL) settings to evaluate LLMs, which include zero- and few-shot approaches.

Zero-Shot Prompting This approach aims to explore how the LLMs handle the task with no prior examples or training, relying solely on their preexisting knowledge and the inherent ability to understand and generate language.

For the zero-shot prompting technique (ZSP), we define two distinct prompts (i.e., ZSP1 and ZSP2), each designed to elicit a different focus from the selected LLMs on the connection established between the clues and the solution. In particular, by implementing these distinct prompts, we intend to assess the versatility of the LLMs in deducing the correct word associations under the constraints of zero-shot learning conditions.

To avoid the presence of extrinsic hallucinations in the results, namely the presence of additional text besides the desired output, we constraint the prompts, phrased in Italian, to force models to provide just the solution for each game. For this reason, we specify that (i) the games provided in the list are independent of each other, (ii) the solution must differ from the words already included in a game, (iii) the answer should not include any additional text but just the solution to each game.

Starting from the list of games, each one composed by a list of clues ([CLUES]), ZSP1 and ZSP2 differ in that the former asks for a 'related word', namely the solution ([SOLUTION]), without specifying the type of existing relationship, the latter specifies that the [SOLUTION] should be 'semantically related' to the [CLUES], as shown below.

- **ZSP-1** *Per ciascun gioco* [CLUES] *in questa lista, scrivi una sola altra parola connessa a ciascuna delle cinque parole incluse in ciascun gioco* (For each game [CLUES] in this list, write only one other word that is related to each of the five words included in each game).
- **ZSP-2** *Per ciascun gioco* [CLUES] *in questa lista, scrivi una sola altra parola semanticamente connessa a ciascuna delle cinque parole incluse in ciascun gioco* (For each game [CLUES] in this list, write only one other word that is semantically related to each of the five words included in each game).

Few-Shot Prompting In the context of Few-Shot Prompting (FSP), two different prompts have been defined. The first one (FSP1) includes one example ([GAME SOLVED]), namely a list of [CLUES] along with the solution , while the second prompt (FSP2) presents three [GAME SOLVED], as it follows:

- **FSP1** *Dato il seguente esempio* [GAME SOLVED], *per ciascun gioco* [CLUES] *in questa lista, scrivi una sola altra parola connessa a ciascuna delle cinque parole incluse in ciascun gioco* (Given the following example [GAME SOLVED], for each game [CLUES] in this list, write only one other word that is related to each of the five words included in each game)
- **FSP2** *Dati i seguenti esempi* [GAME SOLVED], *per ciascun gioco* [CLUES] *in questa lista, scrivi una sola altra parola connessa a ciascuna delle cinque parole incluse in ciascun gioco* (Given the following example [GAME SOLVED], for each game [CLUES] in this list, write only one other word that is related to each of the five words included in each game)

We force the models to return just the [SOLUTION], specifying the same constraints used for ZSP.

Examples With reference to the provided [GAME SOLVED], considering the MWE patterns connecting clue and solution pairs, we manually select from the training set examples which are representative of specific phenomena. For the [GAME SOLVED] provided in FSP1, we choose an example which includes A-B, A-preposition-B and A-articulated preposition-B MWEs, as it follows:

- **Example 1**
  [CLUES]: Nicola, *Roma* (Rome), *farina* (flour),

pranzo (lunch), *poltrona* (armchair)
[SOLUTION]: *sacco* (sack).

Each clue is related to the [SOLUTION] according to the following MWE patterns:

- **A-B pattern**: Nicola Sacco[15]
- **A-preposition-B pattern**: *sacco di farina* (flour bag), *sacco di Roma* (sack of Rome[16]), *poltrona a sacco* (bean bag chair)
- **A-articulated preposition-B pattern**: *pranzo al sacco* (packed lunch)

To run the FSP2 which presents three examples, we add two [GAME SOLVED] whose clue/solution pairs are related by other patterns, as shown below.

- **Example 2**
  [CLUES]: *bello* (nice), *inter*, *vino* (wine), *indosso* (wear), *fronte* (forehaed/front)
  [SOLUTION]: *porto* (port/freight)
- **Example 3**
  [CLUES]: spedito, gigante, uomo, carica, vita
  [SOLUTION]: passo

Specifically, Example 2 shows the following phenomena:

- **A-B pattern**: *Portobello* (the name of an Italian tv show but also a place), *interporto* (freight village)
- **Semantic relations**: hypernymy (*porto* (Port) is a type of wine), synonimy (*porto* and *indosso* may refer to the same meaning to wear)
- **A-articulated preposition-B pattern**: *Fronte del porto*[17] (On the Waterfront), a 1954 movie.

In the third example all the clue/solution pairs are related as they occur together as part of an idiom, namely *a passo spedito* (at a fast pace), *fare passi da gigante* (make great strides), *a passo d'uomo* (at a walking pace), *a passo di carica* (at a charge pace), *passare a miglior vita* (to pass away).

4.4. Results

In this section, we present the results obtained from the four LLMs on the 350 game instances included in the GhigliottinAI test set. We calculate the accuracy as the ratio between solved games on the total games. In Table 1, we show the number of correct solutions together with the accuracy rate for each of the models in both ZSP and FSP settings. The accuracy score is the evaluation metric adopted by Basile et al. (2020) in the original shared task. The results show that the four LLMs performed poorly on both the ZSP and FSP settings.

GPT-4 and Gemini-Pro perform the best in the FSP2 setting when shown three examples of [GAME SOLVED]. Both models achieved an accuracy of .022, which was an improvement over the other settings. In particular, GPT-4 and Gemini-Pro doubled the accuracy scored in both ZSP1 and ZSP2.

Bard also showed efficient in-context learning when given game instances in FSP1 and FSP2. In FSP1, Bard was the best LLM at solving game instances, with an accuracy of 0.14. In FSP2, Bard accuracy was .02, which was slightly worst than the accuracy scored by Gemini-Pro and GPT-4 (i.e., .022).

GPT-3.5, on the other hand, did not seem to benefit from in-context learning. In both FSP1 and FSP2, GPT-3.5 had the lowest accuracy (i.e., .005), proving any improvement in comparison with the results from the ZSP settings. In fact, in the case of GPT-3.5, the accuracy achieved in ZSP1 turns out to be the best performance by this LLM (i.e., .008).

To further evaluate the performance of the LLMs, we also show the number of solutions that they share in each setting. This gives us an idea of how often the LLMs agree on the solution to a game instance. Tables 2 and 3 show the number of (whether correct or not) solutions shared between each pair of LLMs, respectively in ZSP and FSP settings. The highest number of shared solutions for each pair is highlighted in bold. The highest number of shared solutions for each pair is highlighted in bold.

For instance, the GPT family of LLMs from OpenAI share the most solutions for ZSP1 (Table 2), while Google LLMs share the most solutions for both ZSP2 and FSP2 (Table 2 and 3).

For ZSP settings, there is an exception to this trend. Indeed, Gemini-Pro and GPT-3.5, which are from different families, share the most solutions in FSP1 (Table 3).

In this context, one possible explanation for the shared solutions is that the LLMs were trained on similar data sets. This is supported by the fact that the LLMs performed similarly in the ZSP setting, where they were not given any examples of game instances. Another possible explanation is that the LLMs are all using similar in-context learning techniques. This is supported by the fact that the LLMs all improved their performance in the FSP2 setting, where they were given a few and sufficient examples of game instances.

---

[15] `https://en.wikipedia.org/wiki/Sacco_and_Vanzetti`

[16] `https://en.wikipedia.org/wiki/Sack_of_Rome_(1527)`

[17] `https://it.wikipedia.org/wiki/Fronte_del_porto`

LLM	**ZSP1**		**ZSP2**		**FSP1**		**FSP2**	
	Correct	Acc.	Correct	Acc.	Correct	Acc.	Correct	Acc.
GPT-3.5	3	.008	0	0	2	.005	2	.005
GPT-4	**4**	**.011**	**4**	**.011**	2	.005	**8**	**.022**
Bard	1	.002	2	.005	**5**	.014	7	.02
Gemini-Pro	2	.005	**4**	**.011**	3	.008	**8**	**.022**

Table 1: Number of correct answers and accuracy score for ZSP and FSP

ZSP1	GPT-3.5	GPT-4	Bard	Gemini-Pro	**ZSP2**	GPT-3.5	GPT-4	Bard	Gemini-Pro
GPT-3.5	350	**23**	4	3	GPT-3.5	350	3	7	8
GPT-4	**23**	350	1	2	GPT-4	3	350	12	17
Bard	4	1	350	13	Bard	7	12	350	**25**
Gemini-Pro	3	2	13	350	Gemini-Pro	8	17	**25**	350

Table 2: Shared solutions for ZSP1 and ZSP2

FSP1	GPT-3.5	GPT-4	Bard	Gemini-Pro	**FSP2**	GPT-3.5	GPT-4	Bard	Gemini-Pro
GPT-3.5	350	3	6	**37**	GPT-3.5	350	10	7	19
GPT-4	3	350	17	9	GPT-4	10	350	29	17
Bard	6	17	350	12	Bard	7	29	350	**31**
Gemini-Pro	**37**	9	12	350	Gemini-Pro	19	17	**31**	350

Table 3: Shared solutions for FSP1 and FSP2

5. Discussion

In this section, we present an in-depth result analysis to provide some insights of the semantic inference capabilities of LLMs.
As far as the results are concerned, we notice the presence of shared characteristics among these, in that we can identify different types of incorrect answers:

- **Complete clue overlapping** In some cases the proposed [SOLUTION] overlaps with a word in the [CLUES]. For instance, in ZSF1 Bard presents a high number of overlapping solutions, as in ID 2 when the model answers *saldo* (discount) that is also one of the [CLUES].

- **Partial clue overlapping** These results refer to solutions which are derived from one of the [CLUES], e.g. a noun from a verb, as in ZSP1 for ID 266, when GPT-3.5 provides the [SOLUTION] *conteggio* (count) and the first clue is *contare* (to count).

- **Semantic relatedness** These answers usually are generated leveraging the taxonomic relations of one of the [CLUES], so that they are semantically related to one of the [CLUES] and/or to the [SOLUTION]. For instance, in the game ID 167, the solution proposed by GPT-3.5 is *sentimento* (feeling), as one of the [CLUES] is *amore* (love), and the correct [SOLUTION] is *odio* (hate).

- **Clue synonymy** In some cases, the models propose a synonym of one of the [CLUES]. For instance, in ZSP1 to ID 204 (further discussed later), GPT-3.5 answers *guardia* (watchman), a synonym of *custode*, which is presented in the [CLUES] for that game.
  Similarly, in ID 169, GPT-3.5 presents the [SOLUTION] *abitazione* (home), as one of the [CLUES] is the synonym *casa*, while the correct answer is *strada* (road).

- **Clue interference** For some of the games, there is a clue interference that is probably related to the fact that the association between one of the [CLUES] and the possible answer is stronger than others. For instance, in the game ID 69 to the ZSP2 prompt, all the models answer *deserto* (desert), as one of the clues is Sahara.

Considering all the proposed games, the highest agreement among models, on a correct [SOLUTION], that means three models out of four guess the [SOLUTION], happens only in two cases. In ZSP1 setting, this is the game ID 349, shown below.

- **ID 349**
  [CLUES]: *coperto* (covered), *compagnia* (company), *auto* (carro), *agente* (agent), *vita* (life)
  [SOLUTION]: *assicurazione* (insurance).

Each clue is related to the solution according to (i) the A-preposition-B pattern, i.e., *coperto da assicurazione* (covered by insurance), *compagnia di assicurazione* (insurance company), *agente di assicurazioni*[18] (insurance agent); (ii) the A-B pattern, i.e., *assicurazione auto* (car insurance); (iii) the A-articulated preposition-B pattern, i.e., *assicurazione sulla vita* (life insurance).

GPT-4 does not solve the aforementioned game, as the proposed [SOLUTION] is *musica* (music). This could be the results of an interference from two of the clues, namely *compagnia* (company) and *agente* (manager), which occur in MWEs as *compagnia musicale* (music company) and *agente musicale* (music manager) respectively. The other case of highest agreement happens on the game ID 153 when we provide three examples in FSP2.

- **ID 153**
  [CLUES]: *lavare* (to wash), *nuovo* (new), *espressione* (look), *maschera* (mask), *pallido* (pallid)
  [SOLUTION]: *viso* (face)

In such case, Gemini-Pro disagrees on the answer and provide the [SOLUTION] *sapone* (soap), due to the presence of the verb *lavare* (to wash) as first clue which presumably causes an interference on the provided solution.

In only one case we have the full agreement that is when all the models propose the same answer. This is the case of the game ID 69 in the ZSP2 setting, when the models agree on the incorrect [SOLUTION], *deserto* (desert), due to an interference from the clue Sahara.

To further evaluate the results, we propose a comparative analysis for each of the models.

GPT-3.5 In ZSP1, we notice that GPT-3.5 identifies the solution in the game below.

- **ID 41**
  [CLUES]: *nazionale* (national), *muscolo* (muscle), *lavoro* (job), *proposta* (proposal), *firmare* (to sign)
  [SOLUTION]: *contratto* (contract)

Due to the presence of (i) A-B pattern: *contratto nazionale* (national contract), *muscolo contratto* (contracted muscle); (ii) A-preposition-B pattern: *contratto di lavoro* (employment contract), *proposta di contratto* (contract proposal); (iii) idiom: *firmare un contratto* (sign a contract).

The additional specification about the semantic relatedness in ZSP2 worsens the results, as GPT-3.5 fails all the games, including the game ID 41.

Indeed, the proposed [SOLUTION] to the ZSP2 for this game is *fede* (faith). Our hypothesis is that this result is affected by the first clue *nazionale*, as there exist some books whose title contains both the words and also some conservative political parties refer to *fede* and *nazione* to support their ideologies.

Similarly, in the FSP1 setting, the answer to ID 41 is *strada* (road), as in *strada nazionale* (national road). Still, also when three examples are provided, as in FSP2, the model answer, i.e., *nazione* (nation), presents a partial clue overlapping, that is it is derived by the first clue.

Another game resolved in ZSP1 and failed in the other settings is the game ID 152. In this case, the correct answer, i.e., *analisi* (analysis), is changed into (i) *algebra* (algebra), an hyponym of one of the [CLUES], that is *matematica* (mathematics) in ZSP2; (ii) *matematica*, that is one of the clues, when the model is provided with one example; (iii) *statistica* (statistics), another hyponym of mathematics, when we include three examples in the prompt.

Examples are proven to be useful for the correct solution in the game ID 47 for the FSP1 setting, while, in the ZSP results, the model provides a synonym of one of the clues for both settings, i.e., *celebrazione* (celebration) from the clue *festa* (party), and in the FSP2 setting the answer is *concorrenza* (competition), that does not seem having any relations to the clues.

GPT-4 GPT-4 presents some consistency between the correct results presented in ZSP1 and FSP2 in the game ID 37 and ID 135.

With reference to the use of the examples, it is worth noticing that in the game ID 59, GPT-3.5 solves the game when provided with one example and fails with three examples, GPT-4 needs three examples to give the correct [SOLUTION], while with one example the answer is *partito* (political party or left), that could be the result from a clue interference comning form the word *festa* (party), as in *festa di partito* (political convention).

Bard As already stated, Bard presents a high number of complete clue overlapping solutions, mainly in ZSP1. In some cases, the model is consistent in this error. For instance, the aforementioned ID 2 incorrect answer *saldo* (discount) is proposed in all settings, but in ZSP2, when the model proposes another clue as [SOLUTION], i.e., *fare* (to do). This type of error may indicate that the model does not understand the prompt.

This model presents consistency across the settings in ID 153. Indeed, Bard proposes the same correct [SOLUTION], i.e., *viso* (face), in all settings but ZSP1, when the output is *notte* (night), that

---

[18] It is worth stressing that solution including *agente* could belong also to an A-B pattern with the same meaning, as in *agente assicurativo*

seems completely out of context considering the provided [CLUES], that are *lavare* (to wash), *nuovo* (new), *espressione* (look), *maschera* (mask), *pallido* (pale).
Another case of consistency occurs in the game ID 204 in two settings, namely ZSP2 and FSP1, as the same correct answer *museo* (museum) is provided. In the remaining settings, the proposed outputs are *luna* (moon) in ZSP1, resulting from a clue intereference due to the presence of *notte* (night) among the [CLUES], and *uovo* (egg) in FSP2, that could be related to the clue *sale* (salt).

Gemini-Pro Gemini-Pro results show consistency, that is the provided [SOLUTION] is the same correct answer in three settings out of four, just in only one case that is ID 117.

- **ID 117**
  [CLUES]: *pesce* (fish), cary grant, *domestico* (domesticated), *donna* (woman), zorba
  [SOLUTION]: *gatto* (cat)

The model is consistent in all settings, but in ZSP1, when the proposed [SOLUTION] is *attore* (actor), due to the presence of the clue cary grant. The same incorrect answer is given also by GPT-3.5 in the same setting.
In another case, there is consistency among the results over the different settings. This is ID 162 in both FSP settings, but not in ZSP1 and ZSP2.

- **ID 162**
  [CLUES]: *finire* (to finish), *tutta* (entire), *brillante* (bright or comic) *italiana* (Italian), *maschera* (mask)
  [SOLUTION]: *commedia* (comedy)

Also Gemini presents consistency in the game ID 204, but, contrary to Bard results, in different settings. Indeed, this model provides the correct answer *museo* (*museum*) in ZSP1 and FSP2. While in the remaining settings, the proposed [SOLUTION] is *arte* (art) in ZSP2 and *mostra* (exhibition) in FSP1, both as results of a clue interference coming from the word moma.

6. Conclusion

In this paper, we present a series of experiments to investigate the reasoning skills and game-solving skills of four different LLMs (Bard, GPT-4, Gemini-Pro, and GPT-3.5) on a language game task called *GhigliottinAI*. We elicited solutions from the LLMs using different prompts in both zero-shot and few-shot settings. Specifically, for the few-shot setting, we provided both a game instance with its solution and three game instances with solutions from the GhigliottinAI task training set in the prompt.
As shown in Section 4.4, the performance achieved by the LLMs is quite low compared to the performance reported by other artificial players discussed in Section 3. In particular, the best performing artificial player (*Mago della Ghigliottina*) achieves an accuracy score of .68 compared to the two LLMs (GPT-4 and Gemini-Pro) that performed best in the FSP2 setting with an accuracy score of .022. Furthermore, in Section 5, we provide an analysis that aims to count the shared solutions proposed by the different LLMs to highlight how LLMs belonging to the same family have similar behaviors in solving the games.
As mentioned in Section 3 and Section 4.3, the game instances together with their solutions form a linguistic phenomenon known as MWE. This implies that in addition to evaluating reasoning abilities, the LLMs were also subjected to a test that assessed their knowledge of linguistic and statistical phenomena such as: word co-occurrence in frequent collocations or idioms, word similarity or word relatedness and semantic relationship of clues with solutions. In this context, in Section 5, we offered an analysis based on error types that can be explained by different levels of linguistic features.
While this type of analysis provides some preliminary insights into the results proposed by the LLMs, we plan to further investigate the behavior of LLMs in the GhigliottinAI task in the future. For example, to better evaluate the game-solving, reasoning abilities and linguistic phenomena knowledge of different LLMs, we plan to design prompts that elicit multiple solutions ranked by probability for each game instance, in order to rank the LLM proposals. In the process of eliciting diverse solutions ordered by probabilities, we also plan to design prompts with instructions that provide more linguistic context for the LLMs. Furthermore, since in this paper we only exploited two types of prompting techniques, we plan to refine the solution generation through Prompt Chain-of-Thought (Wei et al., 2022) and information retrieval from freely available corpora for the Italian language through Retrieval Augmented Generation (RAG) (Gao et al., 2023).

7. Acknowledgements

Authorship attribution is as follows: Raffaele Manna is author of Section 4 and Section 6, Maria Pia di Buono is author of Section 5, and Johanna Monti is author of Section 1, 2, and 3.
Maria Pia di Buono has been supported by Fondo FSE/REACT-EU - Progetti DM 1062 del 10/08/2021 "Ricercatori a Tempo Determinato di tipo A) (RTDA)". Azione IV.4 - Dottorati e contratti

di ricerca su tematiche dell'innovazione/Azione IV.6 - Contratti di ricerca su tematiche Green.

8. Bibliographical References

Pierpaolo Basile, Marco de Gemmis, Pasquale Lops, and Giovanni Semeraro. 2014. Solving a Complex Language Game by Using Knowledge-based Word Associations Discovery. *IEEE Transactions on Computational Intelligence and AI in Games*, 8(1):13–26.

Pierpaolo Basile, Marco De Gemmis, Lucia Siciliani, and Giovanni Semeraro. 2018. Overview of the Evalita 2018 Solving Language Games (NLP4Fun) Task. *EVALITA Evaluation of NLP and Speech Tools for Italian*, 12:75.

Pierpaolo Basile, Marco Lovetere, Johanna Monti, Antonio Pascucci, Federico Sangati, and Lucia Siciliani. 2020. Ghigliottin-AI@ EVALITA2020: Evaluating Artificial Players for the Language Game "La Ghigliottina". *EVALITA Evaluation of NLP and Speech Tools for Italian-December 17th, 2020*, page 345.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models are Few-Shot Learners. *Advances in neural information processing systems*, 33:1877–1901.

Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke Van Der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu. 2017. Multiword Expression Processing: A Survey. *Computational Linguistics*, 43(4):837–892.

Nazareno De Francesco. 2020. GUL. LE. VER@ GhigliottinAI: A Glove based Artificial Player to Solve the Language Game "La Ghigliottina". *EVALITA Evaluation of NLP and Speech Tools for Italian-December 17th, 2020*, page 356.

Marco Ernandes, Giovanni Angelini, and Marco Gori. 2008. A Web-based Agent Challenges Human Experts on Crosswords. *AI Magazine*, 29(1):77–77.

David Ferrucci, Anthony Levas, Sugato Bagchi, David Gondek, and Erik T Mueller. 2013. Watson: Beyond Jeopardy! *Artificial Intelligence*, 199:93–105.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv:2312.10997*.

Katikapalli Subramanyam Kalyan. 2023. A Survey of GPT-3 Family Large Language Models Including ChatGPT and GPT-4. *Natural Language Processing Journal*, page 100048.

K Lam, David M Pennock, Dan Cosley, Steve Lawrence, et al. 2012. 1 Billion Pages= 1 Million Dollars? Mining the Web to Play" Who Wants to be a Millionaire?". *arXiv preprint arXiv:1212.2477*.

Michael L Littman, Greg A Keim, and Noam Shazeer. 2002. A Probabilistic Approach to Solving Crossword Puzzles. *Artificial Intelligence*, 134(1-2):23–55.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? *arXiv preprint arXiv:2202.12837*.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large Language Models: A Survey. *arXiv preprint arXiv:2402.06196*.

Piero Molino, Pasquale Lops, Giovanni Semeraro, Marco de Gemmis, and Pierpaolo Basile. 2015. Playing with Knowledge: A Virtual Player for "Who Wants to Be a Millionaire?" that Leverages Question Answering Techniques. *Artificial Intelligence*, 222:157–181.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Basile Pierpaolo, Lovetere Marco, Johanna Monti, Antonio Pascucci, Sangati Federico, Siciliani Lucia, et al. 2020. Ghigliottin-AI@ EVALITA2020: Evaluating Artificial Players for the Language Game "La Ghigliottina". In *CEUR WORKSHOP PROCEEDINGS*, pages 345–348. AILC-Associazione Italiana di Linguistica Computazionale.

Ivan A Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2002. Multiword Expressions: A Pain in the Neck for NLP. In *Computational Linguistics and Intelligent Text Processing: Third International Conference, CICLing 2002 Mexico City, Mexico, February 17–23, 2002 Proceedings 3*, pages 1–15. Springer.

Federico Sangati, Antonio Pascucci, Johanna Monti, et al. 2018. Exploiting Multiword Expressions to Solve "La Ghigliottina". In *Proceedings* *of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018)*, volume 2263, pages 258–263. Accademia University Press.

Federico Sangati, Antonio Pascucci, Johanna Monti, et al. 2020. “Il Mago della Ghigliottina”@ Ghigliottin-AI: When Linguistics Meets Artificial Intelligence. In *CEUR WORKSHOP PROCEEDINGS*. AILC-Associazione Italiana di Linguistica Computazionale.

Giovanni Semeraro, Marco de Gemmis, Pasquale Lops, and Pierpaolo Basile. 2012. An Artificial Player for a Language Game. *IEEE Intelligent Systems*, 27(05):36–43.

Giovanni Semeraro, Pasquale Lops, Pierpaolo Basile, Marco De Gemmis, et al. 2009. On the Tip of My Thought: Playing the Guillotine Game. In *IJCAI*, pages 1543–1548.

Jianing Wang, Qiushi Sun, Nuo Chen, Xiang Li, and Ming Gao. 2023. Boosting Language Models Reasoning with Chain-of-Knowledge Prompting. *arXiv preprint arXiv:2306.06427*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Georgios N Yannakakis and Julian Togelius. 2018. *Artificial Intelligence and Games*, volume 2. Springer.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *Advances in Neural Information Processing Systems*, 36.

LLMs of Catan: Exploring Pragmatic Capabilities of Generative Chatbots Through Prediction and Classification of Dialogue Acts in Boardgames' Multi-party Dialogues

Andrea Martinenghi[1], **Gregor Donabauer**[2], **Simona Amenta**[1], **Sathya Bursic**[1],
Mathyas Giudici[3], **Udo Kruschwitz**[2], **Franca Garzotto**[3], **Dimitri Ognibene**[1]
[1]Department of Psychology, University of Milano-Bicocca, Milan, Italy
[2]Information Science, University of Regensburg, Regensburg, Germany
[3]Polytechnic University of Milan, Milan, Italy
a.martinenghi1@campus.unimib.it, dimitri.ognibene@unimib.it

Abstract

Human language interactions involve complex processes beyond pure information exchange, for example, actions aimed at influencing beliefs and behaviors within a communicative context. In this paper, we propose to investigate the dialogue understanding capabilities of large language models (LLMs), particularly in multi-party settings, where challenges like speaker identification and turn-taking are common. Through experiments on the game-based STAC dataset, we explore zero and few-shot learning approaches for dialogue act classification in a multi-party game setting. Our intuition is that LLMs may excel in tasks framed through examples rather than formal descriptions, influenced by a range of pragmatic features like information presentation order in prompts and others. We also explore the models' predictive abilities regarding future dialogue acts and study integrating information on dialogue act sequences to improve predictions. Our findings suggest that ChatGPT can keep up with baseline models trained from scratch for classification of certain dialogue act types but also reveal biases and limitations associated with the approach. These insights can be valuable for the development of multi-party chatbots and we try to point out directions for future research towards nuanced understanding and adaptation in diverse conversational contexts.

Keywords: Multi-Party Dialogue, Game-Based Conversations, Dialogue Act Classification

1. Introduction

Human language-based interactions are not simply the mere delivery of messages and information but complex multilevel processes. In a pragmatic framework, every time two or more individuals are involved in a communicative interaction, they are performing actions: from sharing information to actually inducing or modifying another person's beliefs and behaviors (Austin, 1975).In this perspective therefore, an utterance is produced by a speaker within a specific communicative context and responds to certain communicative intentions (i.e. the objectives that one intends to achieve through one's utterance, for example to convince, explain, ask, swear, etc.). On the recipient's side, to comprehend and interpret communicative messages, a person engages in complex inferential processes aimed at understanding the communicative intention of the interlocutor.

Corpora from multi-party games offer unique opportunities to study these processes. First, games can deliver interactions in a natural environment, which game engines can record along with other happenings, making it possible to study precise connections between the players' utterances, the context, and their general strategies (Djalali et al., 2011). Second, chats based on games are ideal because they approximate spoken language without the need of transcription, and they manifest phenomena particular to multilogue, such as multiple conversation threads (Afantenos et al., 2015). Third, multi-party settings are particularly relevant, as humans tend to work in groups and teams, and both models and methods based on them provide unique challenges compared to two-party systems (Mahajan and Shaikh, 2021). Among these, speaker identification, turn-taking and tailoring the content of the response to each agent or person (Sibun, 1997).

This setting allows studying ChatGPT (Ouyang et al., 2022) and other generative chatbots-(GCBs)' understanding of dialogues and their ability to generalize to new contexts. In particular, most GCBs have been tuned for two-party dialogue. Their evident ability to participate in such interactions is matched by the difficulty of extracting any internal representation of the underlying skills or causes of occasional failures (Mahowald et al., 2024; Borji, 2023; Koyuturk et al., 2023). Having only an empirical appreciation of these skills and the end-to-end nature of the training of these systems together with the complexity of multi-party dialogue call for a nuanced and theory-based approach to study GCBs' capabilities and their ability to adapt to new contexts.

Previous studies tested the pragmatic skills of GCBs and their ability to engage in interactions

Games and NLP @LREC-COLING-2024, pages 107–118
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

and dialogues, especially in two-parties dialogues (but see (Wei et al., 2023)). While Barattieri di San Pietro et al. (2023) applied standardized psychological tests for pragmatic skills evaluation to GCBs and Ruis et al. (2024) work on conversational implicatures, we focus our study on the explicit recognition and prediction of speech acts (Searle, 1969) or dialogue acts (DAs) in multi-party settings (see section 2.1 for a review of this approach). Notwithstanding the observed dialogue skills, DAs are deemed to be challenging for GCBs (Gubelmann, 2024; Brown et al., 2020). The critique to GCBs' conversational and dialogue understanding is not new (Bender and Koller, 2020; Bender et al., 2021). Testing their performance in generalizing to the novel multiparty dialogue could contribute to this discussion. However, only a few works are present in the literature. Wei et al. (2023) implement a type of training under multi-party conditions which prevents studying the transfer of dialogue skills (and understanding) from two-party to multi-party. In (Chan et al., 2023), the focus is on sentence-level relationship parsing, which could not directly support language production and poses substantial complexities also to humans.

We investigate zero or few-shot learning approaches for classifying and predicting DAs in the game-based multi-party dataset STAC (Asher et al., 2016). Our study aims to explore the impact of example-based task formulation and pragmatic features on the performance of Game-Playing Chatbots (GCBs). We also examine the predictive capabilities of GCBs in forecasting future DAs and explore methods to incorporate information on the statistical distribution of DA sequences for improved predictions. Additionally, we analyze the coherence between DA and utterance wording prediction, considering the potential impact of disturbances on DA classification. Joint prediction of text and DA may enhance performance, but could also increase task complexity and affect results. Our study contributes to the understanding of how different dimensions of zero or few-shot learning approaches can enhance the classification and prediction of DAs in multi-party interactions.

The insights coming from our experiments will inform future development of multiparty chatbots based on similar few-shots approaches. The potential shown by this low-cost solution can also provide information on the challenges and opportunities for architectural (Wahlster, 2023) and learning-based approaches directing on the selection paradigm with different costs, e.g. full-training (Wei et al., 2023), fine-tuning (Ruis et al., 2024) or LoRa (Wang et al., 2023). In the spirit of *Games and NLP @ LREC-COLING* we provide all of our implementations as well as detailed results to the community to help reproducing our work[1].

2. Related Work

2.1. Dialogue Acts

When a person expresses an utterance, they are not only saying something: they are also *doing* something. This intuition that utterances possess both a descriptive and effective nature belongs to Austin (1975), who is considered the father of modern theory of speech acts. Austin (1975) formulated a theory of three kinds of acts: (1.a) *Locutionary acts*: acts *of* speaking, involved in the construction of speech; (1.b) *Illocutionary acts*: acts *in* speaking, concerning the meaning; (1.c) *Perlocutionary acts*: acts *by* speaking, relative to the consequences of speaking.

Following analysis and taxonomies of speech acts theory have focuses on Illocutionary acts and the role of intention versus that of convention (Horn and Ward, 2004). In the debate that followed, Grice (1957, 1975) was highly influential in suggesting that an utterance's power is to provide clues to the intention of the speaker. Searle (1969), while recognizing the importance of intentions in communication (which he confined to perlocutionary effects), pointed out that some communications succeed in virtue of the hearer's knowledge of certain rules governing the elements of the uttered sentence (illocutionary effects). Searle (1969)'s interpretation of the role of intention was aligned with Austin (1975)'s, and rejected by Strawson (2013), who argued that most commonplace speech acts succeed by producing the awareness that it was the speaker's intention to achieve a certain communicative goal.

Grice (1975)'s and Strawson (2013)'s recognition of intentions as such a central aspect of communication was continued by Bach and Harnish (1979), which was reflected in a greater consideration for the speaker's psychological state compared to Searle (1975). Their 4-classes taxonomy of Illocutionary acts, which along with Searle's taxonomy (1975) is among the most used in contemporary literature (e.g., Jurafsky and Martin (2024)), includes: (3.a) *Constatives*: committing the speaker to something's being the case (answering, claiming, confirming); (3.b) *Directives*: attempts by the speaker to get the addressee to do something (advising, asking, forbidding); (3.c) *Commissives*: committing the speaker to some future course of action (promising, planning, vowing); (3.d) *Acknowledgments*: express the speaker's attitude regarding the hearer with respect to some social action (apologizing, greeting, thanking).

---

[1] codebase at `https://github.com/DimNeuroLab/llmGrounding`

In pragmatics and in computational linguistics, such as in cuebased models, the term DA is often used as a synonymous of *speech act* (e.g., Jurafsky and Martin (2024); McTear (2022)), and a distinction is not fully clear. It was initially introduced into the field by Bunt (1981) "for referring to the functional units used by the speaker to change the context" (Bunt, 1994). Although sometimes equating DAs to speech acts, Jurafsky and Martin (2024) also outline a difference, describing DAs as the "combination of speech acts and grounding into a single representation of the interactive function of the turn or sequence" (Jurafsky and Martin, 2024). A third usage, that Horn and Ward (2004) suggest being the most used in cue-based literature, comes from Allen and Core (1997) to mean an act with internal structure related specifically to its dialogue function.

2.2. Multi-party Game Based Corpora

A growing body of corpora is based on games. In a survey on available corpora on multi-party dialogues (Mahajan and Shaikh, 2021), which with a sample size of over 300 publications is the only survey focusing on only multi-party corpora, games make up one of four categories of written corpora. *Settlers* (Afantenos et al., 2015) is the first published dataset with multiparty dialog discourse parsing (Asher et al., 2016), and prior to Molweni (Lin et al., 2020), the only one. Its content includes interesting features such as interleaved threads, creative language, and interactions between linguistic and extra-linguistic contexts (Asher et al., 2016).

Given the goal-oriented nature of games, that typically results in various sub-tasks, speech acts produced by players can be insightful in relation to their intentions, which are strictly related to the categorizaton of speech acts (e.g., Grice (1957, 1975)). *Settlers* has been used to study negotiation (Cadilhac et al., 2013). Other corpora, such as on *Avalon* (Stepputtis et al., 2023) and *Werewolf* (e.g., Lin et al. (2020)), have been used to investigate competitive-cooperative settings where private and competing beliefs and pieces of information are held by players, who are therefore encouraged to employ strategies that include deception.

2.3. Multi-party Chatbots

Although chatbots have a long history (Adamopoulou and Moussiades, 2020), multi-party chatbots are less studied, also due to the challenges of their design (Seering et al., 2020).

One of these challenges is the understanding of *who* is it talking to *whom* about *what*, which various methods try to tackle, such as response generators which incorporate Interlocutor-aware Contexts into a Recurring-Encoder-Decoder (Liu et al., 2019)

When developed for multi-party use, chatbots are often adapted from single-party systems. For example, a study by Wagner et al. (2022) used the Rasa framework to create a chatbot for goal-directed conversations in everyday scenarios.

Machine learning has led to the development of more advanced chatbots for multi-party environments, often involving role-playing to test different identities and features. In the LIGHT environment, humans and NPCs are assigned roles (e.g. wizard) and interact through conversations. R2C2 models are trained and tested using four methods to enhance turn-taking and coherence, addressing challenges in multi-party interactions (Wei et al., 2023). Role-playing identities are also created using ChatGPT and other LLM chatbots with dialogue engineering, as few-shot learning alone may not produce accurate and consistent behaviors (Wang et al., 2023).

2.4. Conversation State Extraction

State conversation extraction is considered playing a key role not only in understanding dialogues but also building dialogue systems (e.g., Gao et al., 2020). Unfortunately, its application on multi-party chats can be a puzzling task due to the presence of multi-threads and complex discourse relations. Various approaches have been tested, including on the STAC Corpus.

Among these, a prominent strategy makes use of parsing algorithms. Li et al. (2023) propose a model that utilizes knowledge-enhanced features and symbolic knowledge graph relations to recognize emotions. Another model (Jia et al., 2020) leverages thread extraction based on dependency relations, along with a Thread-Encoder and Transformers, to enhance context understanding. While both systems require training, ChatGPT has shown superiority in zero-shot tasks compared to LLMs and fine-tuned models (Bang et al., 2023). However, its performance in predicting and classifying links between utterances has been limited (Chan et al., 2023).

In the specific task of DA classification, deep neural networks were compared on the STAC by İrsoy et al. (2019) along with their model of directed-acyclic-graph LSTM (DAG-LSTM) which exploits turn-taking and employs Tree-LSTM equations. Conditional Random Fields (CRFs) were used by a number of researchers including the corpus' authors (Cadilhac et al., 2013). With three types of features (lexical, syntactical and semantic), the model outperforms the frequency-based baseline. (See section 5 for a comparison of these models with ours).

State understanding in games may involve unique elements like strategies and secret identities of players. In Avalon, LLMs like ChatGPT were used to uncover players' secret roles by analyzing game dialogue through different state conceptualizations (Stepputtis et al., 2023). Each utterance was also labeled with a persuasion or lying strategy. In Werewolf, CNN/SVM models with manual rules were used to study players' behaviors based on their secret roles, aiming to train an agent to play like a human (Lin et al., 2020). In Settlers (STAC), CP-nets were utilized to predict players' strategic actions, specifically trades (Cadilhac et al., 2013).

3. Data

The STAC corpus (Asher et al., 2016) consists of multi-party chats annotated for discourse structure in the style of Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2003; Lascarides and Asher, 2009). It includes 45 online games sessions of *Settlers of Catan*, a popular boardgame[2].

Settlers of Catan is a competitive, strategic game where players need to exchange resources with each other, making bargaining a pivotal discourse element. It is played on a map made of hexes, which are associated to one of 5 resource types (Brick, Lumber, Ore, Grain, Wool, plus Nothing) and a number (2-12). At the beginning of the game, each player places 2 "Settlements" on an intersection that borders with 2-3 hexes, and can build more of those throughout the game. During a player's turn, he throws 2 dice, which sum indicates which hex will provide resources for the current turn to all players who own settlements on its intersections. Afterward, the player can negotiate with the others an exchange of resources, different combinations of which are needed to build more and better structures, and ultimately to win the game.

Afantenos et al. (2015) who provide annotations for the STAC corpus state that multi-party chats pose a series of challenges that cannot be addressed the same way of two-party chats. Among these, complex intersections of addresses between speakers that escape tree-like structures interpretations, and crossing dependencies. Therefore, they motivate that SDRT is particularly appropriate for the annotation of the STAC corpus because of three reasons: (a) it allows for distant attachments; (b) it is capable of dealing with fragments or non penitential utterances; (c) it can model non-tree like structures.

Their annotation process started with segmenting the turns into EDUs (Elementary Discourse Units), because within each turn the speaker may convey bits of information that carry different purposes (e.g., addressing a previous offer and proposing an offer to a third player). This part was initially done automatically, then corrected by hand. Each EDU was then annotated in a three layers fashion: (1) Type of speech act; (2) Dialogue act; (3) Relation type. These classes sum up with (4) the addressee. Layer (1), Type of speech act (or surface type/act), includes only the *Assertion*, *Question* and *Request* categories. Layer (2), DAs, has *Offer*, *Counteroffer*, *Accept*, *Refusal*, and *Other*, which labels units that either comment on strategic moves in the game or are not directly pertinent to bargaining. Layer (3) which contains 16 relation-based types (e.g., *Comment*, *Background*). For our study, we only consider layer (2).

Overall there are 13440 annotated EDUs. Along with the text and the annotated dialogue act, each segment's row carries information about the identity of the emitter, the emittee(s), the dialogue and others. In our study, we decided to utilize the whole dataset and to follow the original fragmentation of dialogues ($n = 1137$) to maintain consistency with the annotators' work. For some of our experimental runs, we treat each game ($n = 45$) as a single dialogue as we will explain later.

As we were not interested in predicting the DA *Other* (which could be any not further specified type of dialogue that is not related to the conversation about the game), we re-patched the original turns that had been segmented into EDUs, thus restoring the corpus into a sequence of turns. When this operation resulted in conflicting DAs (the only occurrence being *Other* plus a different dialogue act), we gave precedence to the other DA(either *Offer*, *Counteroffer*, *Accept* or *Refusal*). The number of turns that satisfy the experimental requirements amounts to 3939 (*Offer*: 981; *Counteroffer*: 647; *Accept*: 696; *Refusal*: 1615) when splitting by dialogues; 4552 (*Offer*: 1589; *Counteroffer*: 649; *Accept*: 697; *Refusal*: 1617) when splitting by games.

4. Methodology and Experiments

In general, we employ zero-shot and few-shot learning approaches when running classification or prediction tasks. In particular, we run our experiments using GPT-3.5 Turbo. We acknowledge that conducting comparisons among various LLMs could offer additional insights into how they perform on game-based multi-party dialogue corpora. However, our primary objective is to get an initial idea of these models' performance on the task. Therefore, we leave running such experiments as future work. Few-shot learning is a learning approach where the model is given at inference a small number of demonstrations of each new task it is asked to per-

---

[2] Rules available under: `https://www.catan.com/sites/default/files/2021-06/catan_base_rules_2020_200707.pdf`

form (Brown et al., 2020). Note that the weights of the model are not updated, thus, the model must use its prior knowledge to generalize from these examples to perform the task. As Brown et al. (2020) have shown, large language models excel at zero-shot, one-shot, and few-shot learning tasks, frequently matching performances of fine-tuned models. However, it has been shown that this type of few-shot learning can be unstable (Zhao et al., 2021; Ye and Durrett, 2022). The choice of prompt format, training examples, their number, or even their order all influence the performance and expose biases inherent in the model (Webson and Pavlick, 2022). Nonetheless, few-shot learning is being explored due to its speed, low cost and data efficiency in solving custom tasks (Ahmed and Devanbu, 2023; Wei et al., 2022).

Before systematically running the experiments, we conducted a series of pre-tests on selected dialogue inputs and prompts to evaluate ChatGPT's behavior. We wanted to explore the model's ability to correctly classify the DAs of an input text and to investigate the rationale behind its choice. This iterative process was especially useful in elaborating more successful descriptions of the DAs as part of the prompts. Moreover, we needed to make sure that the produced output was consistent and suitable for an automatic analysis (parsing the response to a label). The interpretation of these results are illustrated in the qualitative analysis.

4.1. Tasks Evaluated

To comprehensively evaluate ChatGPT's capability in assessing multi-party dialogues as they naturally occur within multiplayer-games, we conducted a range of experiments with various objectives using the STAC dataset. Our objectives include the straightforward task of classifying relevant DAs to get the current status of the discussion, as well as predicting future DA types within an ongoing conversation. Figure 1 shows an example of the both tasks given previous textual turns. As *Other* could be any not further specified type of DA, in both tasks we will exclude samples that have this label during our experiments. However, we will keep previous DAs that are labeled with *Other* as context for few shot learning as they can provide helpful contextual cues.

Classification of Dialogue Act Types: Our first goal is to evaluate the classification performance of ChatGPT on game-relevant DAs covering the classes described previously (*Offer*, *Counteroffer*, *Accept* and *Refusal*).

Prediction of Dialogue Act Types: Our second approach extends beyond simple classification, focusing on the prediction of the subsequent DA following a prior conversation. Again, employing both zero-shot and few-shot learning methodologies across various scenarios (e.g. with different context length and number of samples during few-shot learning), we aim to predict relevant DA types.

4.2. Prompt Dimensions

We assess various dimensions that could potentially impact the quality of a prompt for both tasks, as it remains unclear which variables contribute to the model's performance and to what extent. Below, we offer a brief overview of the individual feature dimensions we vary in our experiments. Apart from different forms of these feature dimensions, the prompt always begins with an intro and ends with an output specification. The intro is always "I will give you a dialogue from a game of Settlers of Catan played by some players, you will need to predict the class of the next utterance." (in case of future DA type predicting) or a similar form (e.g. in case of current DA classification); The output specification is always "How could that dialogue continue? Very important: please respond with 1 possible continuation in this precise format: [class of utterance]" (again in case of predicting the future DA type), besides a few variations of the question (e.g. when running classification).

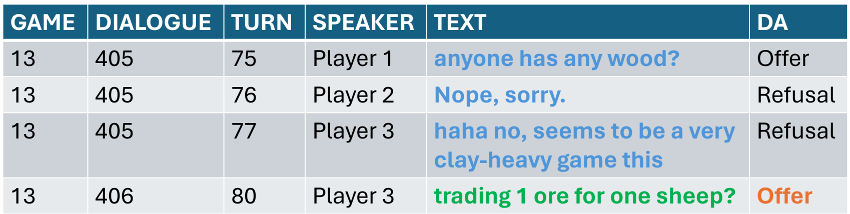
Game Description [GAME] *(name of the dimension as used in the columns of Table 2 and Table 3) in squared brackets***:** We evaluate two versions of this features as part of the prompt: (1) a summarized description of the game *Settlers*, and (2) no game description at all.

Number of Shots [SHOTS]: We experiment with four different versions: (0) no shots; (1) one or (2) two utterance(s) after the description of each DA class; (5) one utterance plus, after the description of the DA classes, four additional utterances for each DA class in random order (resulting in overall 5-shots);.

Context-Length of the Input Dialogue [CONTEXT]: For the length of context that we pass as part of the input dialogue we tested three variations: (1) one turn, (3) three turns and (5) five turns. For most of our runs we used condition (3).

Form of Player Names [PLAYERS]: Another variable we experiment with is the way player names are represented in the input dialogue. (1) in some prompts we report the original name of players, (2) in other prompts report an anonymized form of the player names (i.e., player_1, etc.) which remains consistent for active players within the same input dialogue.

Information about the Conditional Probability of the DAs [PROB]: This feature describes for a present DA which is/are the most likely dialogue act(s) to follow. We evaluate four different versions: (1) an indication of what is more likely to occur (i.e., "Offer often follows Other. Sometimes it follows Accept, less times it follows Refusal, Counteroffer



GAME	DIALOGUE	TURN	SPEAKER	TEXT	DA
13	405	75	Player 1	anyone has any wood?	Offer
13	405	76	Player 2	Nope, sorry.	Refusal
13	405	77	Player 3	haha no, seems to be a very clay-heavy game this	Refusal
13	406	80	Player 3	trading 1 ore for one sheep?	Offer

Figure 1: Dialogue and segmentation example with CONTEXT=3 and PLAYERS=YES. Context is highlighted in blue, target DA to classify/predict in red. For classification the turn text in green is added to the prompt, during prediction it is left out *(numbering gap between turns 77 and 80 is due to Server/Game Engine turns which are not included in the players-only dataset)*.

and Offer."), (2) an indication with the addition of an annotated two to three turns long exchange as example, (3) conditional probabilities for each possible combination, expressed as a percentage, and (N) no probabilities.

Domain of Dataset [DOMAIN]: Almost all of our experimental conditions followed (1) the partitioning of the dialogues based on the suggestion of the annotators. However, we also tested prompts using (2) games instead of dialogues as the splitting parameter for the dialogues.

Order of Features [ORDER]: Finally, we utilize the order in which the blocks are presented within the prompt as an additional parameter. We use three different variations: (1) Dialogue - Instructions - Classes; (2) Instructions - Classes - Dialogue; (3) Classes - Dialogue - Instructions.

5. Results

In line with common practice in NLP, we report on accuracy and macro F1-scores (Jurafsky and Martin, 2024, Ch. 4). We limited the results presented below to these metrics and a range of different prompt feature combinations to make the overview more concise. For more details about other runs as well as additional metrics, such as precision and recall, we refer to our codebase.

5.1. Classification of Dialogue Act Types

Baselines: For comparison of our classification results, we refer to Cadilhac et al. (2013) and Irsoy et al. (2019) who both performed this task with different approaches on the same dataset. While Cadilhac et al. (2013) adopt Conditional Random Fields (CRFs) to learn DAs, Irsoy et al. (2019) introduced a new architecture (DAG-LSTM) for contextual representations. However, we note that both also included the *Other* label in their evaluation which accounts for a high number of samples and thus leads to macro F1 performance that is not comparable. Therefore, adopting class-wise F1 scores (of the same classes we used in our setup) when comparing results is more fair.

When evaluated against Cadilhac et al. (2013) and Irsoy et al. (2019), our approach consistently produces higher class-wise F1 scores related to *Accept* and *Refusal*. However, it showed slightly worse performance in *Counteroffer* and *Offer* classifications, although demonstrating to be notably close in the *Offer* category. For detailed class-wise results, see Table 1.

As observable in Table 2 prompts with modified instructions, placed after the dialogue to classify, and differing only in the instruction formulation, resulted in very low classification Accuracy (0.560) and macro F1 (0.633) compared to the other variations. Another setup, with a single few-shot example, also exhibited low Accuracy of 0.571 and macro F1 score of 0.639.

For the exact wording used in our prompts, further details on the different experimental setups as well as additional results we refer to our codebase.

5.2. Prediction of Dialogue Act Types

As observable in Table 3, for runs where we selected "games" for prompt dimension [DOMAIN] we can observe higher metrics compared to the runs with "dialogue" for this feature. When compared with the baseline performance on dialogues (Accuracy = 0.266, F1 = 0.212), the improvements in Accuracy (0.345) and F1 (0.305) on games were carried both by increments in the same metrics for the *Offer* DA (Accuracy: 0.658 vs 0.619; F1 = 0.502 vs 0.400) both by the higher number of this class which accounts for the major difference between the two domains (1589 vs 981). This result underlines the impact that the distribution of the DAs can have on the major metrics, suggesting the need for a more complex interpretation of the results.

One combination of prompt dimensions (compare first row in Table 3) served as the basis for several variations of all other prompts, thus offering a point of reference for the interpretation of the impact of the feature dimensions. We show an example of this prompt in figure 2. When ranking by Accuracy, for all remaining features the results suggest that: (1 [GAME]) prompts result in better per-

	Offer	**Counteroffer**	**Accept**	**Refusal**
Cadilhac et al. (2013)	0.805	**0.585**	0.585	0.776
Irsoy et al. (2019)	**0.820**	0.517	0.643	**0.865**
OURs	0.719	0.384	**0.671**	**0.865**

Table 1: Dialog Act class-wise F1-score comparison with baselines. Best scores are highlighted in bold.

GAME	SHOTS	CONTEXT	PLAYERS	PROB	DOMAIN	ORDER	ACC	F1
NO	2	3	YES	(N)	DIAL.	(3)	0.560	0.633
NO	1	3	YES	(N)	DIAL.	(3)	0.572	0.639
NO	2	3	YES	(N)	GAMES	(1)	0.618	0.650
NO	1	1	YES	(N)	DIAL.	(2)	0.594	0.652
NO	1	3	NO	(N)	DIAL.	(3)	0.593	0.654
NO	1	5	YES	(N)	DIAL.	(2)	0.600	0.660
NO	1	3	NO	(2)	DIAL.	(2)	0.650	0.675
NO	2	3	YES	(N)	DIAL.	(1)	0.654	0.692
NO	1	3	YES	(N)	DIAL.	(1)	0.665	0.694
NO	2	3	NO	(N)	DIAL.	(1)	**0.691**	**0.716**

Table 2: Accuracy and macro F1-scores for dialogue act classification under different variations of the prompt features. Best scores are highlighted in bold.

formance when using a description of the game; (2 [SHOTS]) few-shot examples are useful; (3 [CONTEXT]) longer input dialogues result in better performance; (4 [PLAYER]) including anonymized player names instead of the original ones is useful; (5 [PROB]) the results suggest that more precise information about the conditional probabilities are better; (7 [ORDER]) the way in which the pieces of information within the prompts are ordered can be very impactful, with prompts that have the input dialogue first outperforming all the rest.

5.3. Qualitative Analysis

We select a series of turns and dialogues of various complexity to investigate ChatGPT's capability in understanding the given dialogue input and its reaction to the prompt variations. Most of the times, we ask the LLM to produce one or more possible continuation(s) and to specify its dialogue act, which allow us to evaluate the response's fitness through analysis of the speaker identity, the meaning of the utterance, the relationship with the given context, appropriateness of the chosen DA and the response's syntax. This allows us to shed more light on the reasons that informed the LLM's decision.

One of the things that ChatGPT seems to do best is associating the right DA to its own response. When mistakes appear, they tend to relate to the DA class *Other*, especially in the form of false negatives. Prominent cases are answers to difficult questions that do not constitute *Offers* or *Counteroffers*, which should be labeled as *Other* but are recognized as *Accept* or *Refusal* instead. Even more difficult are questions of these kinds that mention resources, such as when players discuss how to materially complete the trade with the UI. When asked how it would like the DAs to be described to avoid misunderstanding, ChatGPT can propose to include new categories, such as splitting *Other* in three classes: *Inquiry* for questions about resources and trades without making a formal offer, *Explanation* for clarifications related to ongoing actions, and *Other* for the remaining situations.

Errors in the form of the chosen DA (e.g., writing "class" where we expect the name of the class) tend to happen when the definitions or examples of the classes are defined or introduced in a way that is too ambiguous (e.g., asking to respond with "class, player: utterance").

ChatGPT's choice of speaker is not always great, with seemingly naive errors. Not rarely, it makes the last speaker continue the conversation, including by responding to their own question. Such poor performance in considering turn-taking and basic contextual information strikes in opposition to the excellent internal coherence demonstrated when prompted to produce entire stories or conversations from scratch. Evidence from this suggests us to specify that a *Counteroffer*/*Accept*/*Refusal* relates to another player's offer. Although helpful, this does not completely solve the issue, and it cannot be of assistance with another typical (semi)error, which is when the same player continues with a second offer.

The understanding of contextual information related to the game is even more problematic: Frequently, ChatGPT makes a player offer a resource

GAME	SHOTS	CONTEXT	PLAYERS	PROB	DOMAIN	ORDER	ACC	F1
NO	2	3	YES	(N)	DIAL.	(1)	0.266	0.212
NO	2	1	YES	(N)	DIAL.	(1)	0.235	0.157
NO	2	3	YES	(1)	DIAL.	(1)	0.267	0.184
NO	2	3	YES	(N)	DIAL.	(2)	0.174	0.191
NO	0	3	YES	(N)	DIAL.	(1)	0.264	0.195
NO	2	3	YES	(2)	DIAL.	(1)	0.270	0.206
NO	2	3	NO	(N)	DIAL.	(1)	0.263	0.214
NO	2	3	YES	(3)	DIAL.	(1)	0.270	0.223
NO	2	5	YES	(N)	DIAL.	(1)	0.279	0.234
NO	5	3	YES	(N)	DIAL.	(1)	0.277	0.248
NO	2	3	YES	(2)	DIAL.	(2)	0.230	0.251
YES	2	3	YES	(N)	DIAL.	(1)	0.273	0.260
NO	2	3	YES	(N)	GAMES	(1)	0.318	0.262
NO	5	3	YES	(N)	GAMES	(1)	**0.350**	**0.305**

Table 3: Accuracy and macro F1-scores for next dialogue act prediction under different variations of the prompt features. Best scores are highlighted in bold.

that the same player had previously exhibited a need for or respond to offers that never happened. To better elucidate on its understanding of resources needs and offers, which as a key element of Settlers were also used as a major feature for classification by Cadilhac et al. (2013), we ask ChatGPT to illustrate the state of each player's needs and availability. The delivered representation within the same output tends to be black and white: for some players, it can be perfectly correct; for other players, a mixture of right and wrong conclusions. Some resources are misunderstood as wants when they are offered, and vice-versa.

Moreover, biases within the order of information can appear: Along with the preference for repeating the last turn's speaker in shorter dialogues, ChatGPT may exhibit primacy bias in favouring the first turn's speaker in long dialogues. At the same time, ChatGPT proves to be able to focus on a mid-dialogue offer making a player answer it, a common occurrence in multi-party conversations.

```
I will give you a dialogue from a game of Settlers of Catan
played by some players, you will need to predict the class
of the next utterance

The dialogue:
{dialogue}

It is very important that you consider what said by each
player, which represent their intentions, and the order in
which each player spoke. Build (but don't write) the
framework of which resources each player wants to trade for
giving and which to trade for receiving.

The admissible classes of utterances, with definition and
examples are:

"Offer: A proposal to trade resources between players, which
isn't related to another offer. Example1: Hey anyone have
any clay? Example2: Need wood or clay?",
"Counteroffer: A response to another player's offer,
proposing a different trade. Example1: I can do 1 of each
for 2 clay. Example2: (in response to an offer that
requested clay) What about sheep?",
"Accept: Agreeing to an offer or counteroffer made by
another player. Example1: I can wheat for clay. Example2:
(in response to an offer of ore) Sure",
"Refusal: Declining an offer or counteroffer made by another
player. Example1: (in response to an offer of wood) No, not
interested. Example2: (in response to an offer of ore for
sheep) Not as long as I keep losing ore from the robber",
"Other: Turns or statements that do not involve direct
trading, such as discussing game mechanics or making
observations about the current state of the game, including
questions that aren't offers or counteroffers. Example1:
What's up? Example2: (after a counteroffer) How do I accept
the trade?"

Please remember: If an utterance qualifies for "Other" but
also for one of the other 4 classes, it should then be
considered of the other class (not of the class "Other")

Very important: please respond with 1 possible continuation
in this precise format: [class of utterance]
```

Figure 2: Baseline prompt for prediction task.

6. Limitations and Discussion

Firstly, in the prediction task, there appears to be a bias towards some DAs compared to others. *Offer* and *Counteroffer* show good recall and modest precision, indicating that ChatGPT often predicts them correctly but may also falsely predict them frequently. As *Accept* and *Counteroffer* are the first DAs in the description of DA classes, their position may contribute in explaining ChatGPT's bias in selecting them. On the contrary, *Accept* and *Refusal* show low recall and from modest to good precision. This suggests that while ChatGPT rarely predicts them, when it does, it is often correct. As a possible explanation, it may be helpful to note that predicting which DA follows an offer is expected to be challenging, as the range of possible responses is wide (*Accept*, *Refusal*, *Counteroffer* and *Other*), and affected by many exogenous factors, e.g. a player switching context or multiple line of conversation taking place simultaneously between different participants. At the same time, *Accept*, *Refusal* and *Counteroffer* are supposed to be cued by an offer or counteroffer, whereas the presence of *Offer* is more unpredictable, and this may help explaining *Offer*'s lower precision compared to *Accept* and *Refusal*.

Across different prompts and conditions, variations in performance metrics can be observed. Compared to the other DAs, *Offer* shows better stability; among the metrics, recall results are the most impacted, especially for *Counteroffer*. In particular, *Counteroffer*'s recall tends to be inversely correlated to *Offer*'s recall, implying competition in the prediction of these DAs.

ChatGPT's ability to classify the DAs proved good, with high precision and recall for all DAs but only modest for *Counteroffer*. Again, the lower performance of *Counteroffer* may be attributed to confusion with *Offer*. *Refusal* showed the highest precision, which could be attributed to a narrower and clearer realm of expressions when it comes to saying no to someone.

Our range of test on different prompt variations reveal interesting insights: For instance, prompts with more examples per DA (and thus more shots for the few-shot learning) generally improve performance metrics, suggesting the usefulness of additional examples presented to the LLM. However, presenting the dialogue last negatively impacts performance across all metrics, indicating the importance of dialogue sequence in prediction accuracy. Manipulating context length of given previous turns also affects performance, with longer contexts generally improving recall and F1 scores, but with lower precision. Notably, shorter contexts result in lower F1 scores, particularly for the *Refusal* class.

Overall, these observations underscore ChatGPT's capabilities in certain DA predictions while highlighting areas for improvement, such as accurately predicting *Accept* and managing dialogue context effectively.

7. Conclusion

Previous literature assessed the ability of GCBs in solving an array of pragmatic tasks (e.g., implicatures, indirect speech acts, comprehension of fiugurative language, etc.; see (Hu et al., 2022; Barattieri di San Pietro et al., 2023; Ruis et al., 2024)), finding a performance comparable to that of humans. Does this mean then that GCBs engage in pragmatic processes in the same way as human do? Bender et al. (2021) have famously debated that GCBs do not possess human-like processes, defining them "stochastic parrots", lacking communicative intention and thus only mimicking language comprehension. Indeed, as pointed out by Hu et al. (2022), experiments showing that chatbots displaying human-like verbal behaviors should not necessarily lead to conclusions toward a similarity of processes of humans and AI. An interesting argument comes however from Lenci (2023), who brings a cognitive perspective into the debate arguing that even in humans "language understanding does not always consists in the construction of full-fledged, highly structured semantic representations or complex reasoning processes". Referring to the works of Ferreira et al. (2002); Karimi and Ferreira (2016), Lenci reminds us that humans make often use of shortcuts, heuristics and "good enough" representations in order to process language quickly and efficiently. From this stance we can gather that humans have both capabilities: they can, on the one hand, engage in deeper understading of the interlocutors' intentions forming and recalling theories of the mind and of shared knowledge, and on the other hand, rely on surface heuristics to reach easily their communicative objectives. When considering GCBs however, it appears that they strongly rely on the latter, without the ability to access the former processes. As suggested by Mahowald et al. (2024) large language models lack functional linguistic competence, that is the ability to rely on world knowledge to form models of the situation and the interlocutors and to engage in social pragmatic understanding of the communicative intentions. The lack of this functional ability (which is, to all intent and purposes the core of pragmatics), but, above all, the lack of the flexibility to engage in both levels of processing (the deep one and the surface one) might greatly impair the possibility of GCBs to generalize their verbal behaviors to more complex interaction involving multiple agents.

In conclusion, ChatGPT showed good ability in navigating through the DAs categorization, however our results show that it may over-rely on such "shortcuts" (Lenci, 2023) as it is less good at understanding the real state of the conversation. This is evident in joint speaker and utterance predictions that often deliver nonsensical outcomes about which ChatGPT is not aware. This indicates that applying off-the-shelf GCBs to multiparty dialogues may not be immediate and supports the adoption of expensive approaches, e.g. those involving full LLM training (Wei et al., 2023). However, the consistent results on the independent classification of DAs suggest that lower-cost but non-trivial solutions for multi-party GCBs should be explored.

8. Acknowledgements

We would like to thank the anonymous reviewers for their constructive feedback which has helped us improve the paper.

This work was supported by the project COURAGE funded by the Volkswagen Foundation, grant number 95564.

9. Bibliographical References

Eleni Adamopoulou and Lefteris Moussiades. 2020. Chatbots: History, technology, and applications. *Machine Learning with Applications*, 2:100006.

Stergos Afantenos, Eric Kow, Nicholas Asher, and Jérémy Perret. 2015. Discourse parsing for multi-party chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 928–937, Lisbon, Portugal. Association for Computational Linguistics.

Toufique Ahmed and Premkumar Devanbu. 2023. Few-shot training llms for project-specific code-summarization. In *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, ASE '22, New York, NY, USA. Association for Computing Machinery.

Keith Allan. 1998. Meaning and speech acts. *Retrieved June*, 28:2004.

James Allen and Mark Core. 1997. Draft of damsl: Dialog act markup in several layers.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).

Nicholas Asher and Alex Lascarides. 2003. *Logics of conversation*. Cambridge University Press.

John Langshaw Austin. 1975. *How to do things with words*, volume 88. Oxford university press.

Kent Bach and Robert M Harnish. 1979. Linguistic communication and speech acts.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity.

Chiara Barattieri di San Pietro, Federico Frau, Veronica Mangiaterra, and Valentina Bambini. 2023. The pragmatic profile of chatgpt: Assessing the communicative skills of a conversational agent. *Sistemi intelligenti*, 35(2):379–400.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Emily M Bender and Alexander Koller. 2020. Climbing towards nlu: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 5185–5198.

Ali Borji. 2023. A categorical archive of chatgpt failures. *arXiv preprint arXiv:2302.03494*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Harry Bunt. 1994. Context and dialogue control. *Think Quarterly*, 3(1):19–31.

Harry C Bunt. 1981. Rules for the interpretation, evaluation and generation of dialogue acts. *IPO annual progress report*, 16:99–107.

Anais Cadilhac, Nicholas Asher, Farah Benamara, and Alex Lascarides. 2013. Grounding strategic conversation: Using negotiation dialogues to predict trades in a win-lose game. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 357–368.

Chunkit Chan, Jiayang Cheng, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. 2023. Chatgpt evaluation on sentence level relations: A focus on temporal, causal, and discourse relations. *arXiv preprint arXiv:2304.14827*.

Alex Djalali, Sven Lauer, and Christopher Potts. 2011. Corpus evidence for preference-driven interpretation. In *Proceedings of the 18th Amsterdam Colloquim Conference on Logic, Language and Meaning*, AC'11, page 150–159, Berlin, Heidelberg. Springer-Verlag.

Fernanda Ferreira, Karl GD Bailey, and Vittoria Ferraro. 2002. Good-enough representations in language comprehension. *Current directions in psychological science*, 11(1):11–15.

Shuyang Gao, Sanchit Agarwal, Tagyoung Chung, Di Jin, and Dilek Hakkani-Tur. 2020. From machine reading comprehension to dialogue state tracking: Bridging the gap. Publisher: [object Object] Version Number: 1.

H Paul Grice. 1957. Meaning. *The philosophical review*, 66(3):377–388.

H Paul Grice. 1975. Logic and conversation. *Syntax and Semantics*, 3:43–58.

Reto Gubelmann. 2024. Large language models, agency, and why speech acts are beyond them (for now)–a kantian-cum-pragmatist case. *Philosophy & Technology*, 37(1):32.

Laurence R Horn and Gregory L Ward. 2004. *The handbook of pragmatics*. Wiley Online Library.

Jennifer Hu, Sammy Floyd, Olessia Jouravlev, Evelina Fedorenko, and Edward Gibson. 2022. A fine-grained comparison of pragmatic language understanding in humans and language models. *arXiv preprint arXiv:2212.06801*.

Ozan Irsoy, Rakesh Gosangi, Haimin Zhang, Mu-Hsin Wei, Peter Lund, Duccio Pappadopulo, Brendan Fahy, Neophytos Nephytou, and Camilo Ortiz. 2019. Dialogue act classification in group chats with dag-lstms. *CoRR*, abs/1908.01821.

Qi Jia, Yizhu Liu, Siyu Ren, Kenny Zhu, and Haifeng Tang. 2020. Multi-turn response selection using dialogue dependency relations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1911–1920. Association for Computational Linguistics.

Daniel Jurafsky and James Martin. 2024. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. 3rd edition (draft), 3rd February 2024. https://web.stanford.edu/~jurafsky/slp3/.

Hossein Karimi and Fernanda Ferreira. 2016. Good-enough linguistic representations and online cognitive equilibrium in language processing. *Quarterly journal of experimental psychology*, 69(5):1013–1040.

Cansu Koyuturk, Mona Yavari, Emily Theophilou, Sathya Bursic, Gregor Donabauer, Alessia Telari, Alessia Testa, Raffaele Boiano, Alessandro Gabbiadini, Davinia Hernandez-Leo, et al. 2023. Developing effective educational chatbots with chatgpt prompts: Insights from preliminary tests in a case study on social media literacy. In *31st International Conference on Computers in Education*.

János Kramár, Tom Eccles, Ian Gemp, Andrea Tacchetti, Kevin R McKee, Mateusz Malinowski, Thore Graepel, and Yoram Bachrach. 2022. Negotiation and honesty in artificial intelligence methods for the board game of diplomacy. *Nature Communications*, 13(1):7214.

Alex Lascarides and Nicholas Asher. 2009. Agreement, Disputes and Commitments in Dialogue. *Journal of Semantics*, 26(2):109–158.

Alessandro Lenci. 2023. Understanding natural language understanding systems. a critical analysis. *arXiv preprint arXiv:2303.04229*.

Wei Li, Luyao Zhu, Rui Mao, and Erik Cambria. 2023. SKIER: A symbolic knowledge integrated model for conversational emotion recognition. 37(11):13121–13129.

Youchao Lin, Miho Kasamatsu, Tengyang Chen, Takuya Fujita, Huanjin Deng, and Takehito Utsuro. 2020. Automatic annotation of werewolf game corpus with players revealing oneselves as seer/medium and divination/medium results. In *Workshop on Games and Natural Language Processing*, pages 85–93, Marseille, France. European Language Resources Association.

Cao Liu, Kang Liu, Shizhu He, Zaiqing Nie, and Jun Zhao. 2019. Incorporating interlocutor-aware context into response generation on multi-party chatbots.

Khyati Mahajan and Samira Shaikh. 2021. On the need for thoughtful data collection for multi-party dialogue: A survey of available corpora and collection methods. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 338–352, Singapore and Online. Association for Computational Linguistics.

Kyle Mahowald, Anna A. Ivanova, Idan A. Blank, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. 2024. Dissociating language and thought in large language models. *Trends in Cognitive Sciences*.

Michael McTear. 2022. *Conversational ai: Dialogue systems, conversational agents, and chatbots*. Springer Nature.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Laura Ruis, Akbir Khan, Stella Biderman, Sara Hooker, Tim Rocktäschel, and Edward Grefenstette. 2024. The goldilocks of pragmatic understanding: Fine-tuning strategy matters for implicature resolution by llms. *Advances in Neural Information Processing Systems*, 36.

John R Searle. 1969. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press.

John R Searle. 1975. A taxonomy of illocutionary acts.

Joseph Seering, Michal Luria, Connie Ye, Geoff Kaufman, and Jessica Hammer. 2020. It takes a village: Integrating an adaptive chatbot into an online gaming community. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–13, New York, NY, USA. Association for Computing Machinery.

Penelope Sibun. 1997. Beyond dialogue: the six w's of multi-party interaction. In *Working Notes of AAAI97 Spring Symposium On Mixed-Initiative Interaction, Stanford, CA*, pages 145–150.

Simon Stepputtis, Joseph Campbell, Yaqi Xie, Zhengyang Qi, Wenxin Zhang, Ruiyi Wang, Sanketh Rangreji, Charles Lewis, and Katia Sycara. 2023. Long-horizon dialogue understanding for role identification in the game of avalon with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11193–11208, Singapore. Association for Computational Linguistics.

Peter F Strawson. 2013. Intention and conventionin speech acts. In *Symposium on JL Austin (Routledge Revivals)*, pages 380–400. Routledge.

Nicolas Wagner, Matthias Kraus, Tibor Tonn, and Wolfgang Minker. 2022. Comparing moderation strategies in group chats with multi-user chatbots. In *Proceedings of the 4th Conference on Conversational User Interfaces*, CUI '22, New York, NY, USA. Association for Computing Machinery.

Wolfgang Wahlster. 2023. Understanding computational dialogue understanding. *Philosophical Transactions of the Royal Society A*, 381(2251):20220049.

Zekun Moore Wang, Zhongyuan Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Man Zhang, et al. 2023. Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. *arXiv preprint arXiv:2310.00746*.

Albert Webson and Ellie Pavlick. 2022. Do prompt-based models really understand the meaning of their prompts? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, United States. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Jimmy Wei, Kurt Shuster, Arthur Szlam, Jason Weston, Jack Urbanek, and Mojtaba Komeili. 2023. Multi-party chat: Conversational agents in group settings with humans and models. *arXiv preprint arXiv:2304.13835*.

Xi Ye and Greg Durrett. 2022. The unreliability of explanations in few-shot prompting for textual reasoning. *Advances in neural information processing systems*, 35:30378–30392.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706. PMLR.

Ozan İrsoy, Rakesh Gosangi, Haimin Zhang, Mu-Hsin Wei, Peter Lund, Duccio Pappadopulo, Brendan Fahy, Neophytos Nephytou, and Camilo Ortiz. 2019. Dialogue act classification in group chats with dag-lstms.

Author Index

Aliady, Wateen Abdullah, 85
Alrdahi, Haifa, 32
Althani, Fatima, 21
Amenta, Simona, 107

Basile, Pierpaolo, 61
Batista-Navarro, Riza, 32
Bruen, Sally, 6
Bursic, Sathya, 107

di Buono, Maria Pia, 97
Donabauer, Gregor, 107

Ferret, Olivier, 14
Fort, Karen, 14

Garzotto, Franca, 107
Giudici, Mathyas, 107
Guillaume, Bruno, 14

Hiebel, Nicolas, 14

Kruschwitz, Udo, 107

Madge, Chris, 21
Manna, Raffaele, 97
Martinenghi, Andrea, 107
Monti, Johanna, 97
Murata, Masaki, 76
Musacchio, Elio, 61

Névéol, Aurélie, 14

Ognibene, Dimitri, 107

Poesio, Massimo, 21, 85

Remy, Bertrand, 14

Semeraro, Giovanni, 61
Sepanta, Sia V., 1
Siciliani, Lucia, 61
Summers-Stay, Douglas, 43

Tikhonov, Alexey, 70

Uí Dhonnchadha, Elaine, 6

Voss, Clare R., 43

Ward, Monica, 6

Xu, Liang, 6

Zhuo, Binggang, 76