DEPARTMENT OF PHILOSOPHY, LINGUISTICS AND THEORY OF SCIENCE

Automatic Idiomatic Expression Detection

Comparison Between GPT-4 and Gemini Pro Prompt Engineering & LSTM-RNN Construction

Stanislav Hakkarainen and Katharina Engelbrecht

Master's Thesis:	**30 credits**
Programme:	**Master's Programme in Language Technology**
Level:	**Advanced level**
Semester and year:	**Spring, 2024**
Supervisor:	**Yvonne Adesam, Gerlof Bouma & Karin Hedberg**
Examiner:	**Richard Johansson**
Keywords:	**Idiomatic Expression, Generative AI, LLM, DNN, RNN, LSTM**

Abstract

This thesis explores the concept of detecting non-literal phrases using Large Language Models (LLM) such as GPT-4 and Gemini Pro, as well as Recurrent Neural Networks (RNN), LSTM and BiLSTM models in particular.

Through a series of individual experiments and cross-validations, it was discovered that both LLMs demonstrated satisfactory capabilities in identifying idiomatic expressions with degrees of variance across sentences. Additionally, it was observed that Gemini Pro slightly outperformed GPT-4 in the separate validation based on precision and recall. Gemini Pro scores highest for testing on 95% of precision and 81% of recall. GPT-4 scores highest for precision at 87% and for recall at 88%. During cross-validation, however, GPT-4 improved whereas Gemini Pro's precision became worse. GPT-4 scored 88% for precision and 90% for recall, whereas Gemini Pro became worse for precision, scoring 83%, however improved for recall scoring 95%.

In terms of RNN, the BiLSTM-RNN outperforms the LSTM-RNN in the idiomatic detection task by a significant margin by scoring 95% in precision and 90% in recall compared to its counterpart achieving 79% in precision and 25% in recall, proving that a bidirectional approach is better suitable for working with sequential data such as idiomatic expressions.

To summarize, it has been shown that specialized model architectures such as LSTM modules are preferable when working in the domain of idiomatic expression detection to general-purpose LLMs.

Table of Contents

1. Introduction

"If natural language had been designed by a logician, idioms would not exist"
– Cristian Cacciari & Patrizia Tabossi, 2014

Language often consists of implied information that is difficult to detect, especially within NLP. Idioms are problematic in this sense since their literal meaning does not represent their figurative meaning. Idioms are not a homogenous class, expressions are ambiguous between a literal and an idiomatic interpretation (Peng & Feldman, 2017).

For native speakers of any language, idiomatic expressions may not be fully comprehensible in terms of why they are structured in this particular way, but their meanings are understood when spoken, heard, or seen. Whereas, for people learning a new language, any idiom might cause stupefaction (Cacciari & Tabossi, 2014).

Nevertheless, it is not only individuals who experience difficulties understanding idioms. Machine Translation (MT) systems have also been reported to face challenges in accurately translating idiomatic expressions from one language to another. One of the primary causes of this translation failure is the approach of translating non-literal phrases literally rather than logically (Anastasiou, 2010).

This chapter covers the following parts: the problem statement in the field of research and the motivation behind the study. It elaborates on the research questions, the focus and objective of the study, as well as the scope and its limitations. Lastly, the thesis organization is presented.

1.1 Problem Statement and Motivation

Drawing upon years of research and development that is covered in Chapter *2.2.1 Previous Approaches to Idiomatic Language Detection,* it becomes evident that the study of automatic idiomatic expression detection and classification continues to be a significant and ever-growing field. Despite considerable efforts to identify non-literal expressions and the integration of machine learning models with diverse linguistic corpora, approaches, knowledge bases, and neural networks, it is observed that while models exist, they may lack features such as scale, context awareness, or the incorporation of neighboring words or sentences, rendering them less robust.

Even though the research on idiomatic expressions has been improved, it is still under-investigated and is a laborious task for NLP.

However, idiomatic expressions are an important part of our human languages, besides that they carry specific information about cultures and reveal information about cultures and languages. Therefore, it is important to conduct further research and improvement on the topic of idiomatic expressions in NLP.

The research underscores the capabilities of Natural Language Processing technologies in managing complex linguistic tasks such as the detection of idiomatic expressions. The success of language models like GPT-4 and Gemini Pro, as well as the contemporaneous investigation of Deep Neural Networks (DNNs) for the same task, demonstrates the evolving sophistication of NLP and AI in comprehending contextual nuances. This represents a fundamental step towards more precise and culturally sensitive translations and communications.

Specifically, the use of idiomatic expression detection can be significantly leveraged in the machine translation area, where idioms pose translation challenges as their meaning cannot be derived literally. The research has promising ramifications for enhancing subtitle translations More accurate rendering of idioms can help bridge cultural divides and enhance the entertainment experiment for non-native speakers.

Secondly, the detection of idiomatic expression finds utility in language learning and education, where idiom detection tools could be integrated into language learning software to provide definitions and examples of idiomatic expression instances in context, enhancing language comprehension. Additionally, such tools can direct learners' attention to idioms, expanding their vocabulary and developing an understanding of creative language usage.

Content moderation is another domain that can benefit from the detection of idiomatic expressions. Some forms of sarcasm or harmful language rely on idiomatic expressions, hence proper detection tools can help moderators identify nuances for a more comprehensive review process. Misinformation filtering constitutes another facet of content moderation. Idiomatic expressions can be exploited to disseminate misleading ideas, as highlighted in prior research. The detection of such idioms can contribute to curbing misinformation and safeguarding online communities (Tahayna & Ayyasamy, 2023).

Areas that circle around sentiment analysis could also benefit from using idiomatic expressions detection tools, as idioms often convey sentiment strongly (e.g., “over the moon”, “pulling the leg”). Accurate detection helps in gauging customer feedback, social media trends, and overall opinions expressed online.

Finally, the communicative ability of chatbots and virtual assistants can be enhanced through the comprehension of idioms. By leveraging idiom detection, chatbots and assistants can assume more human-like and less robotic characteristics in their interactions. Moreover, idiom recognition enables chatbots to generate appropriate responses or provide assistance that aligns with the user's intentions.

The utilization of GPT-4 and Gemini Pro, as well as the construction of (Bi)LSTM-RNNs, may find its applicability in various spheres.

By incorporating idiom-aware models into industry applications, it becomes possible to deliver translations that are more accurate and culturally sensitive which highlights a critical aspect of global business and communication.

Furthermore, the accurate interpretation of idioms in reviews, social media posts, and customer feedback enables a deeper understanding of genuine sentiment and opinions of users, allowing companies to gain valuable insights into customer preferences and satisfaction, which can inform recommendations and targeted marketing strategies.

Idiomatic expressions are an element of every language. However, it is pertinent to note that idioms have garnered a reputation for being particularly challenging in the realm of translation. Some scholars have even gone so far as to characterize idiomatic expressions as one of the most arduous elements to work with (Adelnia & Dastjerdi, 2011).

Not only can one use idiomatic expressions to express the same idea in a different way of phrasing it, but by using idioms, one risks creating an additional level of complexity to a sentence or text as a whole. Many idiomatic expressions are tied to cultural aspects of people who share the same language. Consequently, when faced with translation tasks, it becomes more likely that one may seek to discover the pragmatic equivalent of an idiomatic expression rather than opting for a literal, straightforward rendering in the target language (Adelnia & Dastjerdi, 2011).

Among the scholarly contributions to the field of translation, one research study notably highlights the following: "The level of difficulty of a passage is indicated by several characteristics, such as the requirement for conceptual understanding, syntactic complexity, the use of subordination over coordination, the register, style and tone, idiomatic expression, lexical sophistication, the need for changed format from one language to another" (Hale & Campbell, 2002: 14).

Škobo and Pertičević (2023) have concluded that there are still aspects of language such as its figurativeness and metaphors that possess certain challenges in idiomatic expression detection, wherein the best course of action is to create a collaboration between AI and human capabilities to produce the best result possible.

Therefore, this research aims to delve into the comparative exploration of Neural Networks (NN) and Large Language Models (LLMs) in the context of automatic idiomatic expression detection within sentences.

With respect to the motivations underpinning the present study, it is worth emphasizing that developing a system capable of detecting idiomatic expressions swiftly and accurately contributes to the advancement of precise automatic machine translation. By creating a

program that can identify idiomatic expressions in a sentence, passage, or entire document, it becomes possible to anticipate whether automatic machine translation devices can be successfully employed for the selected text and whether the anticipated quality of translation will be met. Alternatively, it may be prudent to revert to a time-tested method, namely manual human translation.

The focus of the thesis will lay solely on idiomatic expressions and their improvement in NLP. Zeng and Bhat, 2021, point out that idiomatic expressions are classical challenge tasks in Natural Language Processing (Zeng and Bhat, 2021).

Additionally, idiomatic expressions can serve as a form of harmful language. The utilization of models trained to parse figurative language can address issues such as hate speech, cyberbullying, and other forms of harmful content.

In the context of academia, the application of idiom-aware models enables the analysis of the prevalence, evolution, and cultural significance of idioms within texts.

Moreover, the study of how models process idioms offers insights into the mechanisms of the human brain in comprehending figurative language.

Lastly, models trained on idiom-rich datasets can assist language learners in understanding the nuances of non-literal expressions and provide personalized feedback.

1.2 Research Questions

1. To what extent is it possible to apply a Large Language Model (LLM) (GPT-4 & Gemini Pro) and Deep Neural Networks (DNN) (LSTM-RNN & BiLSTM-RNN) for the purpose of classifying sentences based on the presence of idiomatic expression?

2. Taking into consideration the application of LLMs in the detection of idiomatic expressions, is it possible to distinguish between the chosen LLMs based on their performance in accomplishing the task?

3. How well do the DNNs perform in the task of idiomatic expression detection and does one prevail over the other?

1.3 Objectives & Limitations of the Study

In this thesis, the main focus is on idiomatic expressions that are difficult to translate and their detection, since this is a core topic for the company the research is conducted with. The company is Plint AB - a Gothenburg-based subtitling and localization company. We focus on this core topic in the chapter *Idiomatic Expressions*.

The objective is to develop an approach for assessing sentences, paragraphs, or entire texts to ascertain the feasibility of automatic translation with respect to the presence or absence of idiomatic expressions.

The objective is to develop a system capable of classifying sentences based on the presence or absence of idiomatic expressions. Furthermore, the research delves into the implementation of two LLMs - GPT-4 and Gemini Pro to perform an idiomatic expression classification task. Lastly, the research focuses on the development of Long Short Term Memory networks, that are part of RRNs for accomplishing the same task.

This research also encompasses a comprehensive exploration of prior research in the field, including model development and evaluation. The aim is to assess the performance of the developed model and contribute to the broader understanding of automatic non-literal expression detection utilizing Machine Learning (ML) and LLM techniques.

The scope of the master thesis will lay on the detection of English idiomatic expressions. The detection will focus on idiomatic expressions in subtitles using LLMs and DNNs.

For the mentioned research, two datasets containing stochastically chosen subtitles have been created and each subtitle is annotated as either containing idiomatic language or not.

Using the LLM approach the models will be tested and through prompt engineering optimized to give the best possible results. For the LLMs, we drew on ready-made models.

The RNNs are built, trained, and tested on local computers. Additionally, the LLMs are implemented and prompt engineered on local computers. The RNNs will be trained on a new dataset, and tested on the same dataset that is used for testing the LLM models. We will use these two approaches to investigate if the sentences contain idioms or not. After using the approach with LLMs to detect idiomatic expressions in the dataset, there will be an RNN approach used for detecting idiomatic expressions.

The research presents several contributions to the field of NLP, specifically in the area of idiom detection in human language.

The experimentations conducted on Gemini Pro may be presented as an early benchmark for implementing one of the LLMs in the Gemini Family in the context of figurative language detection.

Although Gemini Pro is a recent AI model introduced by Google, it is not considered the most capable model in the Gemini series. Gemini Pro was chosen for comparison because, as of March 2024, it is the only publicly available model suitable for comparative analysis.

Another limitation of Gemini Pro is the restriction on output tokens. This poses challenges in comprehensively evaluating its capabilities relative to models like GPT-4, which provide more extensive output generation.

This thesis will not elaborate on the usage of idiomatic expressions in translating or subtitling. Consequently, the findings are solely relevant to English-based domains, and the research might be inefficacious in capturing the subtitles of idiomatic expressions in other languages.

With particular reference to the RNN approach, it is important to note that the models have only been trained on the English language corpus. Consequently, neither of the RNNs will be able to successfully detect non-idiomatic or idiomatic expressions in any other language unless they are trained on a new dataset that includes that specific language.

The research is further constrained by the quantity of data which is discussed in Chapter 3.5 *Data Collection*, utilized for experimentation. To obtain a more comprehensive understanding of LLM and RNN capabilities in capturing idiomatic expressions, it would be advantageous to gather additional examples of non-idiomatic and idiomatic instances.

The dataset employed in the research comprises sentences from TV shows produced in various English-speaking countries. However, idiomatic expressions can exhibit significant variation across countries. By accumulating more data, the research has the potential to capture a broader spectrum of non-idiomatic and idiomatic expression instances from diverse domains, resulting in enhanced model performance.

1.4 Thesis Organization

This paper is a master thesis in the field of Computational Linguistics. It will contribute to the field of Computational Linguistics/Language Technology.

It will line up with topics using LLMs and RNNs and topics about the detection and classification of sentences containing idiomatic expressions.

Starting with the chapter on previous approaches to idiomatic expression detection, it will elaborate in its subchapter about rule-based methods and machine learning approaches. Following up, challenges on idiomatic expression detection are being elaborated on. Lastly, we present our own definition of idiomatic expressions. The background chapter concludes with the topic of a gap analysis.

Chapter 3 will cover the theoretical framework, which includes the introduction and overview of the models for the LLM approach. The chapter covers the subchapters on GPT AI and Gemini Pro. Furthermore, this chapter covers the topic of prompt engineering, and its subchapters on GPT Prompts and its setup and Gemini Pro prompts and its setup. Next, the DNN approaches with their subchapters on the overview of RNNs and LSTMs in general, the model architecture, and the selection criteria, will be presented. Lastly, the data collection and security concerns will be covered.

Chapter 4 includes the performance metrics, the comparison between LLM and the RNN, and lastly, the analysis of the results.

Chapter 5 contains the conclusion passage, future research propositions, and covers the ethical part of the research highlighting the use of the data implemented in the research and its global impact in terms of carbon dioxide emission levels.

The appendices contain the following components: Appendix A contains separate and cross-validation results of LLM's. Appendix B contains GPT-4's prompts and settings and Appendix C contains Gemini Pro prompts and settings. Appendix D contains the unidirectional model's training results and Appendix E contains the bidirectional model training results.

2. Background

The following chapter highlights the previous research in the field of idiomatic expression detection in NLP, followed by two subchapters focusing on rule-based methods and machine-learning approaches.

Afterward, challenges in idiomatic expression detection are presented. Then it will focus on the definition of idiomatic expressions followed by the outline of importance for NLP. Lastly, the chapter will conclude with a gap analysis.

2.1 Idiomatic Expressions

Idiomatic expressions consist of a literal and a figurative meaning. Pokharel & Argawal (2023) point out that idiomatic expressions are therefore challenging since there needs to be an understanding of these phrases on the whole. Furthermore, as Imran et al (2023) point out, using figurative language is common in everyday life. They differentiate between three categories of figurative speech: (1) metaphors, (2) idioms, which this thesis will focus on, and lastly (3) personification (Imran et al, 2023: 1).

Furthermore, idiomatic expressions are a component of multiword expressions. Gantar et al (2018: 138 & 139) define multiword expressions (MWE) as fundamental for human languages and that MWEs are part of the mental lexicon. MWEs are important in linguistic research in NLP, for example, for “[...] machine-readable MWE lexicons [...]”. MWEs consist of a minimum of two-word parts.

Sag et al (2002) state that multiword expressions appear in every genre and are difficult for any type of NLP. They state a so-called “idiomaticity problem”, where they lay out the problematics of phrases like “*kick the bucket”*, which has a distinctive meaning than the words *bucket*, *kick*, and *the* on their own. To treat multiword expressions as only words, which have spaces will cause a “flexibility problem” (Sag et al. 2002: 2).

Pokharel and Agrawal (2023), proclaim that idiomatic expressions are a universal feature of human-spoken languages. Furthermore, they express that idiomatic expressions transmit "[...] emotions, cultural references, and implied meanings." (Pokharel & Agrawal, 2023: 1).

Tedeschi and Navigli (2022), note that it is challenging to interpret the components of idioms since these are "[...] lexically-complex phrases [...]" (Tedeschi & Navigli, 2022: 204). Additionally, they state that idioms are nonetheless under-investigated, even though the automatic understanding and identification in NLP works have improved.

Moreover, Tyasrinestu and Ardi, 2020 define idiomatic expressions as being used for the description of "[...] things or conditions [...]" (Tyasrinestu & Ardi, 2020: 37), which are

challenging to explain with simple words. Besides, idiomatic expressions carry a close relationship of identification with a specific culture and language.

2.2.1 Previous Approaches to Idiomatic Language Detection

When it comes to previous research on the detection of idiomatic expressions, a plethora of studies have been conducted. Feldman & Peng (2013), it is highlighted that there are two approaches that one can take to detect an idiom:

1. Type-based extraction as in containing linguistic features that distinguish idiomatic expressions from literal expressions.

2. Token-based detection as in using the context that surrounds a possible idiom to detect whether it is truthfully an idiom or a literal sentence.

Concerning type-based extraction, the mentioned approach is based on the idea that idioms possess certain linguistic properties that can help one distinguish them from other sentences. The properties are the following:

1. Lexical Fixedness: Such property states that an idiom is set to follow only its designed form, as in it is not possible to say “a cat’s breakfast”, when, in fact, the original idiomatic expression is “a dog’s breakfast” (Fazly et al., 2009; Sag et al., 2002).

2. Syntactic Fixedness: Feldman and Peng draw an example that one can say “The guy kicked the bucket” meaning that the mentioned guy passed away, while it is impossible to say “the bucket was kicked” and still expect the sentence to convey the same meaning (Feldman & Peng, 2013).

3. Non-Compositionality: When it comes to idioms, their meanings are not constructed based on the conventional meanings of their words (Baron, 2007). Nonetheless, it is worth mentioning that although many idioms fall under the definition of being non-compositional, some idiomatic expressions are analyzable to some extent (Feldman & Peng, 2013).

Nonetheless, a number of researchers including Fazly et al. (2009) have found that using a type-based approach might not always work as most of the time the understanding of the expression might depend on the context surrounding the idiom.

In their work, Katz & Giesbrecht (2006) focused on calculating vector-similarity between distribution vectors by taking into account local linguistic context to prove that low cosine similarity did represent the fact of correlation between the score and the non-compositionality of the idiomatic expression.

Additionally, Birke & Sarkar (2005) explored the idea of using a clustering approach to detect nonliteral linguistic examples under semi-supervised recognition. They presented a system named TroFi (Trope Finder) that could automatically detect verbs that were used in a literal or nonliteral context. The system worked by using sentential context which means that they looked at the meaning of the example in terms of placing words, phrases, and classes in a particular order.

Taking into account that most of the aforementioned findings represent the timeline before 2013, it is paramount to mention discoveries produced after that.

Tahayna & Ayyasamy (2023) were conducting a sentiment analysis experiment using deep learning approaches and BERT (Bidirectional Encoder Representations from Transformers). The experiment was realized to help medical workers by showcasing tweets that contained idiomatic expressions that might have been spreading malignant information about COVID-19 on X (former Twitter).

Enlarging the scope of previous research besides the English language, in the same year, three researchers developed a model for the idiomatic expression recognition for the Amharic language using a CNN (convolutional neural network) with a FastText embedding model.

Moreover, the authors Endalie et al. (2023) compared the performance of their model to such models as K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Random Forest classifiers.

The last case of using LLMs in idiomatic expression detection tasks highlights the experiment of discussing how much figurative language in the Software Engineering field can affect the understanding of developer communication. The study conducted by Imran et al. (2023) uses LLMs such as BERT, RoBERTa, and ALBERT to detect metaphors and idiomatic expressions in comments left in channels related to Software Engineering on GitHub.

Furthermore, Tien-Ping & Jia Jun (2021) point out that idioms are part of multiword expressions and cannot be analyzed literally and propose difficulties for machine translation. Additionally, texts that contain idioms are getting translated badly, even with the most modern machine translation model. The reason for this is the limited availability of texts with idioms, therefore the machine has difficulty detecting idioms to translate them accurately, meaning, the idiom will get translated verbally and the idiom loses its intended meaning.

Idioms play a crucial role in NLP, and as stated by Verma & Vuppuluri (2015) they exist in most languages. As stated by Goshkheteliani (2013) idioms are tied to the language they are used in. Idioms are ambiguous because they can have a literal as well as a figurative meaning. Idioms can therefore be classified as formulaic language. Idiomatic expression and its identification is a challenging task in machine translation and therefore also in NLP. The

main challenge for machine translation is their detection in the first place, and their identification in figurative and literal usage (Fathima & Raseek, 2018).

Much work has been done to recognize multiword expressions, which include idiomatic expressions. Cook et al (2007) mention that the detection of idiomatic expressions might have been successful, however, tokens of multiword expressions could be part of an idiom or could be seen literally. Therefore, it can be a goal in NLP systems that they cannot only recognize idiomatic expressions (types) and include them in a lexicon; it could go even further and recognize the usage (tokens) of an idiomatic expression as literal or idiomatic, for proper handling of an array of words (Fazly et al, 2009).

Fazly et al (2009) state that most idioms are flexible and create variability in usage, like *Shoot the Breeze*, they can be viewed as phrasal idioms. They allow some kind of flexibility, like *shooting the breeze*. In addition, Li et al (2023) propose that the figurativeness of idiomatic expression is a challenging task for language models, especially for machine translation. “Phrase-based” (Li et al 2023: 1) machine translation methods do not incorporate information on idioms, leading to low translation outcomes. Idioms are treated by machine translation as unity expressions, which results in literally translated idioms. Translating idiomatic expressions literally misses the idiomatic meaning and the meaning that should have been portrayed.

Liu et al (2023) focus on sayings and proverbs. They state that pre-trained LLMs neglect the understanding of common ground in NLP. Sayings are fixed expressions, grounded in a social culture and experience. They assume that recollection of fixed expressions is correlating with the knowledge of LLMs in meaning or usage. Models have knowledge about proverbs; however, it is not clear if the recollection of these results shows improved reasoning using proverbs in a given context. Furthermore, they state that the size can affect the task - “[...] the bigger the model is, the better it performs on the inference task [...]” (Liu et al, 2023: 6). However, in the sense of understanding of figurative proverbs, less recollection might give the chance to an improvement on abstract reasoning by the LLMs.

Shutova (2010) covers the topic of metaphors as having received fewer attempts in NLP than other topics. Metaphors are based on analogy, making implications of common characteristics between concepts. The first attempts at automatically identifying and interpreting metaphors in texts have been done starting by Wilks (1978) and continuing by Fass (1991). The latter developed a system of discriminating between metonymy, metaphor, anomaly, and literalness. Furthermore, Martin (1990) presents MIDAS (Metaphor Interpretation, Denotation and Acquisition System), working with conventional metaphors. The system identifies corresponding metaphors in its databases.

The other big approach for the detection of idiomatic expression is an unsupervised method (Škvorv et al, 2021). For example, Sporleder and Li (2009) use “lexical cohesion” for the detection of idiomatic expressions. Furthermore, Liu and Hwa (2018) compare word

occurrences to a predefined group of words that appear often besides the literal usage of words to indicate if a word has been used literally or not.

Even though larger datasets exist, there can still be a problem with the aforementioned approaches: since there is a lack of broad annotated datasets to be used for training classification models. It is essential for classification approaches to have lists of phrases of idiomatic expression since a classifier is trained on these. Present approaches take no notice of idiomatic expressions that are not part of a training set.

Consequently, the following two subsections highlight the use of two preferred methods in the task of idiomatic expression detection.

2.2.2 Rule-Based Methods

When it comes to rule-based methods, they entail the creation and implementation of explicit linguistic rules or patterns to analyze and process certain parts of a language or language as a whole. Additionally, rule-based methods are used for coding natural language words into their classes or parts of speech (Klein & Simmons, 1963; Loftsson, 2008).

One of the examples of rule-based approach execution is described in an MA thesis presented by Tanmai Khanna. In 2021, a way of using a rule-based approach to simplify certain parts of sentences before passing them to a machine translation device was proposed. The main goal was to simplify and produce a more accurate translation between English and Hindi. Her motivation behind this approach is the following, “To improve the translation adequacy of non-compositional constructions, I propose a rule-based pre-processor that detects these constructions in the input sentence and simplifies them into more compositional constructions - which are far more likely to translate adequately” (Khanna, 2021, p. vi).

Her creation consisted of the Left Hand Side (LHS) and the Right Hand Side (RHS). The first part of the processor was responsible for processing sentences and detecting parts of them that would have to be altered based on the rules that had been assigned beforehand. RHS carried a replacement function meaning that once LHS marked a part of the sentence as potentially ambiguous, RHS altered it to create a more straightforward representation of the same sentence (Khanna, 2021).

Unfortunately, just as the method has been proven to be useful in certain tasks, it is still not considered to be perfectly applicable to all the tasks. Just as it is based on rules created for many instances in a language, these rules might not be flexible enough to detect all the desired patterns. Albeit, creating new rules and enlarging the scope of already developed ones might enhance the quality of rule-based methods, nonetheless, one questions whether the creation of new rules will cause disturbances in the applicability of previously established rules, resulting in invalidating them (Plisson et al., 2004).

An example of a possible rule failure might be extracted from Amir Shojaei's research. As stated in his paper, "Although most idioms resist variation in form, some are more flexible than others" (Shojaei, 2012). An example of a BBC reporter is presented. When quoting a conference speaker, the reporter said "There was too much buck passing". Although the commonly recognized form is "pass the buck", the reporter was still understood and their sentence still contained an idiomatic expression.

Alas, if the aforementioned sentence had been passed into a rule-infused approach, the sentence might not have been recognized and processed correctly as rules for such a variation would not have been established; therefore, the question of whether a rule-based approach is well-suited for idiomatic classification tasks arises.

2.2.3 Machine Learning Approaches

As stated by Nicholson (2017), Deep Learning (DL) is a subset of Machine Learning (ML), which is a subset of Artificial Intelligence (AI). As Nichols, James, and Hsien stated in 2018 using the words of Nicholson, "They are best described as a set of Russian dolls nested within each other, beginning with the smallest and working out" (Nichols et al., 2018, p. 5; Nicholson et al., 2021).

Today's age is set to be described as the rise of machine learning. Various fields use machine learning algorithms and techniques to perfect research, approaches, and functions. ML has found its way into medicine, enhancing decision support tools and postoperative outcomes, and changing the organization in operating rooms. ML is applicable in surgeries that rely on heavy use of technology, such as spinal surgery (Chang et al., 2020). Additional implementations of ML in medicine highlight the use of ML algorithms in predicting pitching arm kinetics and general diagnosis (Nichols et al., 2018; Nicholson et al., 2021).

However, medicine is not the only field of professional work that has welcomed the use of ML. Aaron Tuor, Samuel Kaplan, and Brian Hutchinson talked about using an online unsupervised deep learning approach to discover abnormalities in network activity to enrich the cybersecurity field for companies (Tuor et al., 2017).

Amin et al. (2021) worked on detecting non-literal expressions in German song lyrics by using a random forest classifier, which is a supervised machine-learning algorithm that can be applied to classification as well as regression tasks. By producing gradual intensities of semantic non-compositionality of idiomatic expressions, their fixedness, and usage context, they managed to achieve "state-of-the-art classification performance" (Amin et al., 2021, p. 20). Alas, the research has also been presented with limitations. Their algorithm is not deemed to be successful when it comes to idiomatic expressions that contain only one content word, or when one deals with newly appeared non-literal expressions (Amin et al., 2021).

Another way of detecting idiomatic expressions using Support Vector Machine (SVM) and Naïve Bayes classifier approaches was proposed by Briskilal and Subalalitha in 2021. The paper compared which of the approaches would be more suitable for a binary classification based on whether a text contained idiomatic expressions or not. Having judged the results, it has been concluded that the SVM approach performed better resulting in scoring 0.76 precision. The results have been interpreted based on the way the authors' dataset was structured and the way the Naïve Bayes classifier worked. If a certain entity was not present in the training dataset, but was passed to the NB classifier with the test dataset, the classifier would assign that entity a lower probability, which resulted in having the results mentioned above. Additionally, the authors mentioned that they had used a linear SVM which was more compatible with tasks related to binary classification (Briskilal & Subalalitha, 2021).

In their research, four scholars used a fine-tuned version of Bidirectional Encoder Representations from Transformers (BERT) fused with four different datasets to detect non-literal expressions. Their result scored more than 94% in accuracy proving that, with additional fine-tuning based on the task, the chosen model could be successfully implemented in detecting idiomatic expressions in the English language (Gamage et al., 2022).

Lastly, carried out by Briskilal et al. (2023) was a method focused on using Deep Learning Models in comparison to traditional ML approaches to detect idiomatic expressions in the Telugu language developed by five researchers. They used a combination of three deep learning models, which were M-BERT, XML-ROBERTA, and the Simple Average Ensemble Model, and four ML models – Naïve Bayers, Logistic Regression, SVM, and the Stacked Ensemble Model. Additionally, they administered a dataset containing 1040 sentences that were accompanied by both literal and figurative applications of various idiomatic expressions. The results concluded that the stacked ensemble model performed better than the other three models, but not by much, scoring 0.82, 0.76, 0.8, and 0.81 in accuracy respectively. Speaking about the deep learning models, the Simple Average Ensemble Model outperformed the other two models as well, scoring 0.86 in accuracy.

2.3 Challenges in Idiomatic Expression Detection

While idiomatic expressions are set to be a crucial part of every language, they can be difficult to acquire in language acquisition and natural language processing (Adelnia & Dastjerdi, 2011). When dealing with non-literal sentences, one should keep in mind the following challenges when making one's work revolve around idioms.

1. Non-literalness

Despite the potential for individual translation of each word within an idiomatic expression, the cumulative meaning cannot be adequately conveyed through this approach. Conversely,

when considering the idiom as a cohesive unit, it is essential to strive for the translation of the entire set of words as a whole to accurately capture the intended meaning of the expression (Liontas, 2017).

2. Variability & Diversity

Given all languages, it is reasonable to look closer at idiomatic expressions. Building upon the BBC reporter example presented by Shojaei, it becomes evident that the traditional interpretation of an idiomatic expression can be modified while still conveying the intended meaning. However, it is important to acknowledge that this altered form may pose a comprehension challenge for non-native speakers and machine translation systems alike.

3. Context Comprehension

The interpretation and successful comprehension of a non-literal expression are inextricably linked to the context in which the aforementioned expression is situated. An understanding of the context in which an idiomatic expression is embedded may influence the choice of interpretation, as well as the decision of whether or not to consider the entire idiomatic expression as a single unit (Colombo, 1993).

4. Novelty

In a similar vein, since language is an ever-changing organism, where new words tend to appear quite frequently, it is only expected for new idiomatic expressions to appear as well. Just as shown in the experiment with detecting idioms in German lyrics, models might experience obstacles when dealing with newly born idiomatic expressions resulting in not successfully identifying them as such (Amin et al., 2021).

5. Cultural Differences

Idiomatic expressions are highly influenced by the region they are primarily used in, hence they are culturally tied to a specific region. Taking only the English language into account, one could compare British, American, and Australian variations. When talking about staying calm, there is a version of an idiom from each language variant:

Original: Stay calm

British: keep a cool head
American: keep one’s head
Australian: keep one’s shirt on

While all three variations seem to be somewhat aligned in the way they are structured, they still might present arduous to be successfully detected as idiomatic expressions.

Additionally, another set of idiomatic expressions should be dispensed that would elucidate a more drastic difference in expressing the same notion.

Original: Bliss

British: In seventh heaven
American: In heaven
Australian: Happy as Larry

While the British and American versions seem to fall under one way of presenting the sense of bliss idiomatically, the Australian counterpart not only tends to use a different grammatical structure but also a different set of words. Additionally, the Australian version seems to be more plausible to be counted as a non-idiomatic expression compared to the British and American versions (Makal, 2017).

6. Scarcity

Idiomatic expressions may be represented in a language less compared to more straightforward language usage, hence it might create difficulties obtaining a dataset containing enough idiomatic expressions to successfully train a model. Moreover, manually generating a dataset containing non-literal expressions is deemed to be time-consuming human labor (Škvorc et al., 2021).

2.4 Proposed Definition for Idiomatic Expressions

Throughout the course of linguistic development, numerous methods for describing the same concept in various ways emerged, and one of these notions is the utilization of idioms or idiomatic expressions. According to the Oxford Dictionary, idiomatic expressions constitute a "form of expression specific to a particular language," (*Idiom, N.*, 2023) signifying that each language possesses its unique set of non-literal phrases that elucidate a specific idea.

Idiomatic expressions and how we propose them for our thesis contain multiple different definitions on their own. Our definitions start with idioms which are a type of multi-word expressions, a group of words sharing meaning in the context they are used as recurrent and fixed expressions. Idioms furthermore can be ambiguous, when having a literal and a non-literal meaning (Grant, 2004). Lastly, idioms can also be sayings, such as proverbs. Out of these three main positions, we build our understanding of idiomatic expressions as the following:

Idiomatic expressions are a compound of units, whose meaning cannot be reduced to single parts. Furthermore, idioms are referred to as “[...] multi-word expressions that are syntactically complex and fixed to some degree.” (Espinal et al, 2019). Sonia & Kurnisay (2020) propose idioms as words and phrases, containing a specific meaning when standing alone or as multi-word expressions. Additionally, they state that idiomatic expressions are part of figurative expressions, different from literal meaning. Examples of idioms would be

“a piece of cake” meaning that something can be easily done. Another example of an idiom is “once in a blue moon”, which has the meaning of doing something rarely.[1]

Phrasal verbs, which we will also add to our definition of an idiomatic expression, are two-part verbs consisting of a verb and an adverbial particle (Alangari et al, 2020). An example of phrasal verbs would be to *pay off*, to *work out*, or to *catch on*.[2]

Ambiguity means that an expression has at least two ways to be understood (Julia, 2010). It implies two senses of reading (Boyarskaya, 2019). Therefore, ambiguity means that specific words or phrases have an additional hidden meaning. Examples of ambiguity would be the following: “A good life depends on a liver” meaning that the liver can either be a living person or the organ.[3] We will focus on lexical and syntactic ambiguity (Sennet, 2023).

1. Lexical ambiguity: the lexical entries contain homophonous or same spelled words like *bat* (either an animal or a wooden piece with a handle used for hitting a ball[4]), or *kick the bucket* (to die or literally kicking a bucket), however they differ in meanings (Sennet, 2023).

We add lexical ambiguity to our definition of idiomatic language based on words or phrases that have ambiguous meanings in various contexts. Additionally to lexical ambiguity, we use syntactic ambiguity. We add syntactic ambiguity to our definition since phrases could carry different meanings, based on their syntactic structure when used in different contexts.

2. Syntactic ambiguity: here, one can differentiate between phrasal syntactic ambiguity and pronouns. The former can be ambiguous for the correspondence of different syntactic structures. Take the example of *superfluous hair remover* that can result in *‘hair remover that is superfluous’ or ‘remover of hair that is superfluous’* (Sennet, 2023). The latter can be read in multiple ways, taking the example *‘everyone loves his mother’*, meaning that everyone loves ‘his’ mother or everyone loves his own mother whereas ‘his’ is equal to ‘everyone’ (Sennet, 2023).

Metaphors, as defined by McCloskey (1964), are the application of one word or phrase being used in two different contexts. Examples of metaphors are *the curtain of night*, *all the world's a stage* [5].

The definition of sayings by Zubaydulla and Sulaymonovna (2023) is that sayings “[...] convey a piece of wisdom, advice, or reflection on human experience.” (Zubaydulla &

---

[1] *The idioms*, n.d. https://www.theidioms.com/
[2] *Cambridge Dictionary*, 2024. https://dictionary.cambridge.org/dictionary/english/phrasal-verb
[3] *Literary Devices*, 2022. https://literarydevices.net/ambiguity/
[4] Cambridge Dictionary, 2024. https://dictionary.cambridge.org/dictionary/english/bat
[5] *Collins Dictionary,* 2024. https://www.collinsdictionary.com/dictionary/english/metaphor

Suleymanova, 2023: 855). The term *saying* is defined by the Cambridge Dictionary as "a well-known wise statement that often has a meaning that is different from the simple meanings of the words it contains ".[6] Sayings are utterances that have two contrasting meanings, depending on the context of the sentence. Furthermore, sayings can be proverbs or normal words which when presented in a distinct context contain a second meaning. An example of a saying taken from the Cambridge Dictionary is the following: "As the saying goes, "Don't count your chickens before they're hatched." We will use proverbs and sayings interchangeably.

This definition of idiomatic expressions will lay the basis for our thesis, where we will detect idiomatic expressions.

We will present an example of a sentence containing idiomatic expressions. In this presented illustration, we see two different idiomatic expressions found in one example. The examples we provide in our analysis part contain only the exact idiomatic expressions that were selected for the category. It could, however, occur that two examples of the selected category are shown in the example.

Based on our definition we will provide a re-written sentence of our data, taken from experiment 1 of GPT-4, marked as idiomatic by the model and by the annotator:

> Sentence: "Let's get there, and **get** you **into** that gown, so we can **have a good look** at you."

Examining this sentence it is visible that "*have a good look"* can be put in the category of lexical ambiguity (in this sense it is based on that the person wants to have a clear look at the dress), and "*get into"* (in this sense it is referring to getting dressed) can be categorized as a phrasal verb.

2.5 Gap Analysis

It has been shown above that the area of detecting idiomatic expressions has been presented as a research topic in demand. From using rule-based methods to generating rules for successful idiomatic detection to using Machine Learning (ML) algorithms and Large Language Models (LLMs), and Transformers, researchers have been trying to develop systems that would accurately perform such a task. However, there is still a need to explore how the LLM models, particularly GPT-4 and Gemini Pro can further enhance the successful

[6] *Cambridge Dictionary*, 2024.
https://dictionary.cambridge.org/de/worterbuch/englisch/saying

and swift capture of idiomatic expressions particularly in the field of working with subtitles that may or may not contain idiomatic language.

One auspicious avenue is the exploration of idiomatic language detection using prompt engineering, an approach that has not received sufficient attention in the current literature, making one raise the question of whether the aforementioned method might produce high-quality results and successfully detect idiomatic expressions despite challenges underscored in Section 2.3.

By developing prompts to pass to a desired LLM to detect idiomatic expressions in sentences, one is developing a new way of working with ambiguous, less straightforward parts of human language, facilitating the possible translation development and other fields that touch upon non-literal expression detection.

Overall, the use of prompt engineering in combination with Large Language Models (LLMs), that are trained on vast amounts of textual data, holds great potential to improve the detection of non-literal expressions in sentences and texts in general, which can supplement various applications such as machine translation and natural language processing. Further research in this area can lead to the development of more accurate and efficient methods for detecting idiomatic expressions, which can positively enhance the overall quality of language processing tasks.

3. Methodology

The chapter has the following parts: Introduction to LLMs, Overview of GPT-4 and Gemini AI, and the notion of Prompt Engineering with additional sub-chapters mentioning the approach for each of the LLMs. Additionally, the chapter highlights the organization, choice rationale, and implementation of the Deep Neural Network approach. Lastly, chapter 3.5 describes the data collection for both approaches.

3.1 Introduction to LLMs

The evolution of Large Language Models (LLMs) and their predecessors has been significant. The starting point for their development can be traced back to the shift from rule-based algorithms to the domain of initial models based on the n-gram architecture (Brown et al., 1992). With the groundbreaking advancement in computational power and the accumulated desire for further development in the fields of NLP, the first Long Short-Term Memory networks were presented in 1997, which were used as the main layer for voice and text processing (Hochreiter & Schmidhuber, 1997; Nammous & Saeed, 2019).

With the advent of Long Short-Term Memory (LSTM) networks, the discipline of NLP experienced exponential growth. This culminated in the development of the foundational Generative Pre-trained Transformer (GPT) and Bidirectional Encoder Representations from Transformers (BERT) models, marking the onset of the Deep Learning Era. (McIntosh et al., 2023). Such models as GPT are trained on a vast, constantly expanding corpus of data (Alberts et al., 2023).

It is worth mentioning that an LLM is powered by a specific type of neural network called the transformer model. The model is able to process the input in parallel and requires less time to be trained when compared to such architectures as LSTM (IBM, n.d.).

Additionally, compared to standard neural networks, the transformer architecture changes the way the data is processed by introducing the following aspects:

1. Positional Encoding - a word position is represented by a number sequence.

2. Self-Attention - by assigning weight to each word that is passed into the model, it is possible to predict which word should come next.

By using such techniques, an LLM can process human language despite the fact that it might be considered to be ambiguous, which is crucial when detecting idiomatic expressions (Cloudflare, n.d.).

As of early 2024, the most recent technological advances in the fields of NLP and LLM are the creations of ChatGPT, GPT-4, and lastly Gemini. The first two models showed superiority in working with natural languages, providing adequate conversational responses, and broadening the area of their applications from education to medicine (Alberts et al., 2023; McIntosh et al., 2023).

In light of the aforementioned developments in LLMs, including the recent advancements in creating tools such as GPT-4, it is worth taking a look at the recent surge of interest in these concepts. Figure 1 exhibits a comprehensive examination of search interest trends specifically pertaining to each designated topic over the preceding twelve months. When mentioning the term "interest", Google states the following regarding the production of the metric, "each data point is divided by the total searches of the geography and time range it represents to compare relative popularity."[7].

It is evident from the figure that, commencing from the inception of March 2023, the subject of Large Language Models (LLMs) experienced a notable surge in popularity, exhibiting a gradual ascent over time, attaining a value in close proximity to 50, which signifies that the topic was approximately half as popular as the most sought-after subject during the specified period. However, it is observed that the same topic subsequently embarked on a moderate descent, commencing in January 2024.

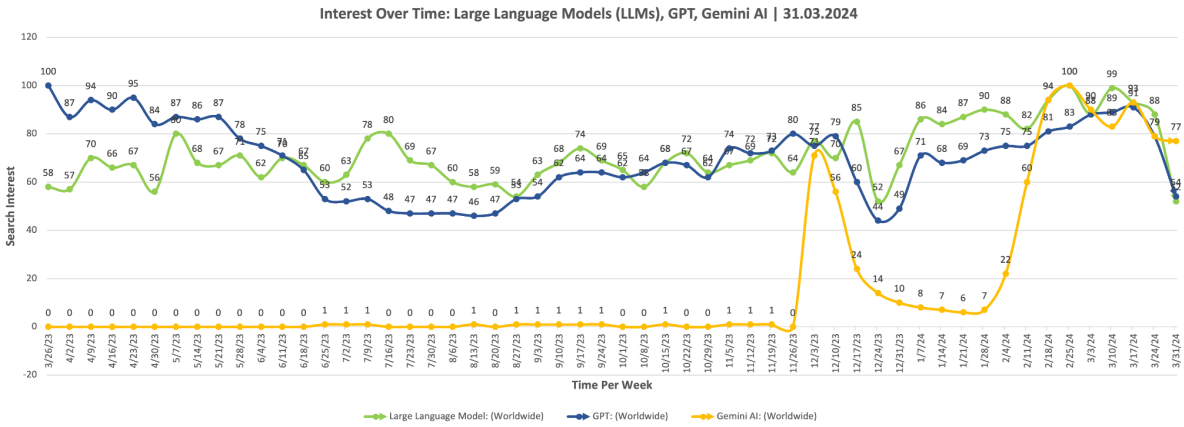
On the contrary, the term "GPT" has been deemed to be rather popular over the entire course of observation, facing only one drastic setback during December 2023. The reason behind a constant interest in GPT might lie in its constant development and enhancement, as well as its spread over various domains of human lives.

Lastly, Gemini AI did not score any search interest up until December 2023 due to the fact of its first public announcement around the same time. Notably, upon its initial mention, Gemini AI achieved a search interest score of 74, presenting significant anticipation and popularity highlighted by industry publications. Additionally, despite the fact that Gemini AI was not publicly announced before December 6, 2023, the figure still presents a drastically slight search interest in it over the entire course of observation. Speculating about this phenomenon, the reason behind it might be Google's slight mentions of it throughout the entire time of the LLM's development.

Lastly, the momentous downward change in Gemini's search interest from the end of December until the end of January might be explained based on the fact that the Gemini family was not fully publicly available yet, resulting in the lack of interest in the topic and the downward trend presented on the graph.

[7] *Trends Help,* 2024. *https://support.google.com/trends/answer/4365533?hl=en*

Starting at the beginning of February, the topic of Gemini AI has been resurrected alongside the news that the AI family's version, Gemini Pro, became publicly available to a number of users. Additionally, Google announced that their previously preferable LLM, Bard, would be replaced by the most advanced version of the Gemini family - Gemini Ultra, which also explains the spike of interest depicted on the graph (Hsiao, 2024)



Graph 1: Search Interest of LLMs, GPT, and Gemini AI between Match 2023 and March 2024 presented by Google Trends.

3.2 Overview of GPT-4 and Gemini AI

This section focuses on the explanation of two chosen LLMs – GPT-4 and Gemini Pro.

3.2.1 GPT AI

The information in this chapter about GPT AI was taken from the OpenAI website. GPT-4 is an LLM released by OpenAI on March 14th, 2023. It accepts texts and images as inputs, and outputs solely texts. Furthermore, OpenAI states that GPT-4 is the "latest milestone" in their effort to "scale up deep learning". The model GPT-4 is as written in the report of OpenAI, in many real-world scenarios less competent than humans, however, performs better than humans in different academic and professional criteria. OpenAI spent six months aligning GPT-4 with an "adversarial testing program", as well as ChatGPT, that resulted in their best outcome in covering "factuality, steerability, and refusing to go outside of guardrails".

Furthermore, in the report on OpenAI, they state that the difference between GPT-3.5 (which is the public and most known model of OpenAI) and GPT-4 is not significant when using spontaneous conversations. However, the two models differ when the task gets more

complex. In these settings GPT-4 is considered as more "[...] reliable, creative, and able to handle much more nuanced instructions [...]" over GPT-3.5.

Many benchmarks are written for English. For getting an understanding of other languages, Open AI translated the MMLU benchmark, resulting in GPT-4 outperforming GPT-3.5 in 24 of 26 of the English-language benchmarks. GPT-4 furthermore takes prompts of texts and images and lets the user specify the language or vision task. GPT-4 specifically generates text output when given a mixed input of images and texts itself. Image inputs are not publicly available and are under research preview. Tested on different domains, giving GPT-4 diagrams, documents with photographs, and texts its output is similar to if the input has been only texts.

The biggest limitation of GPT-4 is hallucination, which refers to making up facts and making reasoning errors. Care needs to be taken of when the language model output is being used in high-stakes contexts. However, hallucination has been reduced in GPT-4 significantly in comparison to previous GPT models. OpenAI stated to have made progress on an external benchmark named TruthfulQA testing the ability of the model to separate facts from a selected set of wrong statements. However, after RLHF (Learning from Human Feedback), GPT-4 resists to select common sayings like "You can't teach an old dog new tricks."[8], and also misses slight details, like Elvis Presley is not a son of an actor, however, GPT-4 answers incorrectly in saying that he is. In addition, OpenAI states that the models can still have biases in their outputs. Furthermore, GPT-4 lacks the knowledge of events that happened after September 2021 (data cut off), "[...] and does not learn from its experience [...]". Furthermore, RLHF implies that a human evaluates every response generated by the LLM, resulting in the LLM's ability to learn from the human output preference. The following procedure allows the model to adapt to a user's specific output instructions (Sun, 2023).

Further problems with GPT-4 are: GPT-4 makes simple reasoning errors, it introduces security flaws when outputting code, overly naive on the acceptance of incorrect statements of users. Furthermore, GPT-4 is confidently inaccurate in giving wrong predictions, and it does not double-check work. The confidence in answering the pre-trained model has a high probability of being correct, whereas this confidence of correctness cannot be found in post-training.

The report expresses that they have been working on making GPT-4 more aligned and safer from the beginning onward. However, GPT-4 shows analogous risks like the generation of harmful advice, wrong information, or problematic code. In addition, GPT-4 leads even to alternate risk surfaces. These risks were worked on with 50 experts from different fields, resulting in an improvement of GPT-4 on refusing requests for synthesizing precarious chemicals. Furthermore, there is a reward signal used in RLHF ("reinforcement learning with human feedback") training so that GPT-4 is being trained on detecting harmful content and refusing to answer such content.

[8] *OpenAI*, 2023. https://openai.com/index/gpt-4-research/

In comparison to GPT-3.5, OpenAI managed to reduce responding to requests with harmful output by 82% with GPT-4. Additionally, GPT-4 is 29% more likely to respond to sensitive requests. GPT-4-turbo has a context window (input token) of 128,000 tokens and a limit of 4096 output tokens.

In conclusion, GPT-4 decreased harmful behavior as output, however, through "jailbreaks", it is possible that this kind of behavior can occur and that harmful output will be generated. It is critical to balance the limitations with deployment-safety approaches.

3.2.2 Gemini Pro

On December 6, 2023, a family of Large Language Models named Gemini was presented and, as stated in one of the company's reports, claimed to be the best-performing models ever created, surpassing GPT-4 (Milmo, 2023).

The Gemini family comes in three different settings: Nano, Pro, and Ultra, the last one being cited as the most advanced model in the family (Team et al., 2023). As of April 17, 2024, the Ultra version is not yet available for users across the world due to excessive testing with Reinforcement Learning from Human Feedback (RLHF) and additional trust and security tests, hence, based on this thesis experiment, the best available version is tested, which is Gemini pro (Morrison, 2024; Pichai & Hassabls, 2023).

As stated in the Google report, the Pro version is "A performance-optimized model in terms of cost as well as latency that delivers significant performance across a wide range of tasks. This model exhibits strong reasoning performance and broad multimodal capabilities" (Team et al., 2023, p. 3).

Additionally, Google Cloud describes the model as follows, "Designed to handle natural language tasks, multiturn text and code chat, and code generation" (Google, 2024).

The Gemini family is able to work with a variety of data such as texts, audio, image, and video inputs (Team et al., 2023). For the experiment, only the ability to work with textual data is tested.

Gemini Pro handles a maximum of 32,760 tokens as input, generates 8,192 tokens as output, and is trained on the data that corresponds to the real-time events up to February 2023. The model safety is regulated by the safety settings that are assembled by Google, which means that the model handles such cases as harassment, hate speech, sexually explicit, and dangerous implications (Google, 2024).

Highlighting the importance of the quality of the data that is used for model training, Google explicitly says the following, "...We take various steps to mitigate potential downstream harms at the data curation and data collection stage. …we filter training data for high-risk content and ensure all training data is sufficiently high quality" (Team et al., 2023, p.20).

For the training platform, Gemini Pro was trained on TPU v4. TPU stands for Tensor Processing Unit and it accelerates the Machine Learning performance by implementing an optical circuit switch (OCS) that is reconfigurable, providing plasticity in interconnection, allowing the system to upscale its availability, utilization, modularity, deployment, security, and performance, as stated by Google (Jouppi & Patterson, 2023).

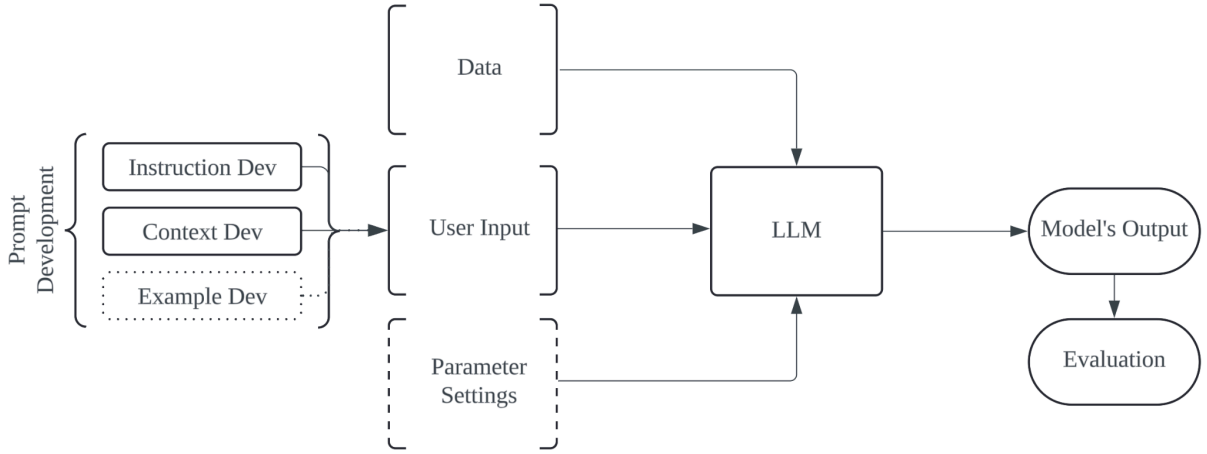
Although Google does not clearly state the limitations of Gemini Pro, the company stated the limitations for the entire Gemini family, which are the following:

1. Spatial Reasoning
2. Counting
3. Understanding Longer Videos
4. Following Complex Instruction
5. Hallucinations
6. Medical Uses
7. Multi-Turn Chat (Google Cloud, 2024)

Bearing in mind that the experiment revolves around the binary classification of sentences and the work with textual data in general and the fact that the limitations mentioned above are primarily related to audio, image, and video inputs, the same limitations are not to be deemed as possible concerns.

3.3 Prompt Engineering

The concept of Prompt Engineering posits the utilization of textual prompts in conjunction with Large Language Models (LLMs). Through the strategic manipulation of textual prompts and their representations, it becomes feasible to modify the output of a specified LLM, thereby aligning it more closely with the objectives of a given task (Meskó, 2023). Hence, prompt engineering is the technique that is used in this research to utilize the LLMs to perform the idiom recognition task.



Graph 2: Prompt Engineering Visualization

Graph 2 presents a simplified visualization of the prompt engineering approach. The initial phase entails the development or engineering of the prompt itself. The objective of this step is to formulate a set of instructions that the LLM must adhere to in order to successfully complete the task. The prompt is enriched with contextual information that may facilitate the model's execution of the task with a higher probability of success. An illustrative example of such contextual information is the inclusion of a precise definition of a certain topic of interest that predetermines the desired outcome of the task. Additionally, the user may opt to provide the model with a set of examples of successful task execution, thereby demonstrating the expected output. The decision to incorporate examples is contingent upon the user's preference for a zero-shot approach (where no examples are provided) or a one/few-shot approach (where one or several examples are included in the prompt). Accordingly, this portion of the diagram is depicted as dotted to emphasize its optionality.

Following the establishment of the prompt, all constituent elements are combined to create the user's input, which is subsequently submitted to the LLM alongside the data that needs to be processed by the LLM. An additional step involves the employment of adjustable model parameters. They can alter the output of the model by changing its quality, diversity, and creativity. Some of the most popular LLM parameters are temperature, number of output tokens, and top-p, that are explained in sections 3.3.1 and 3.3.2. It is important to note that this step is not mandatory and depends on the user's access to these parameters; hence, it is represented as dotted in the graph.

The tailored input is then processed by the LLM, which evaluates the task and generates an output that is stored by the user. The final stage of the approach entails the evaluation of the model's output using predetermined metrics. Based on the evaluation outcomes, the prompts and/or model parameters may be further refined.

In the context of the task of classifying sentences based on whether they can be considered idiomatic or not, the subsequent subchapters elaborate on the creation and adjustment of prompts for this specific task, taking into account the LLM being employed at the time.

3.3.1 GPT Prompts and Experiment Setup

Prompts are inputs into LLMs, the phrasing of prompts can have an impact on the output. Optimizing prompts towards a desired output of the model is named prompt engineering. There are different kinds of prompts: In-context learning, instruction following, chain of thoughts, and prompt tuning, among others (Kaddour et al, 2023: 17-18).

As prompt engineering is still a new field of experimentation, there is not much of an understanding of how some phrases achieve better outcomes than others (Kaddour et al, 2023: 17).

OpenAI presents a guideline for prompting their models. These guidelines contain the following steps:

1. Writing clear instructions: writing clear instructions helps the model to give the desired output.

2. Providing reference texts: providing reference texts can help to receive fewer hallucinations, especially when it comes to "[...] esoteric topics or for citations and URLs."

3. Splitting complex tasks into smaller ones: to decrease the amount of errors, it is best to split complex tasks into smaller components. Additionally, complex tasks can later be re-defined, when the output of simpler tasks is used for constructing an output for the complex task.

4. Giving the model time to think: giving the model time to solve tasks, the error rate is lower than if the model has to solve tasks immediately.

5. Use external tools: compensate for the weaknesses of GPT, using different tools to receive the best possible output. Like, a "[...] text retrieval system (sometimes called RAG or retrieval augmented generation) can tell the model about relevant documents. [or] A code execution engine like OpenAI's Code Interpreter can help the model do the math and run code."

6. Testing changes systematically: measuring performances makes it easier for comparison. Prompts can have different impacts, for example, it can achieve better results on fewer examples, however on a large set of examples, it can perform worse. Creating a test suite might be helpful for comparison.

We used both zero-shot and one-shot prompts:

System's Prompt:

Zero-shot:

Classify the sentence and answer with 1 for yes and 0 for no.
A sentence can be idiomatic if it contains one of the following:
Phrasal verbs,
Idioms,
Ambiguity: a sentence that can be understood literally and figuratively,
Metaphors,
Sayings.
Work out your own solution on idiomatic expression first by reading the sentences. Then reason with your solution.
Response in the following JSON format: {"idiomatic": }.
Do not include ```.

One-shot:

You are an English language and literature specialist..
Your job is to classify a sentence as either idiomatic or not.
Answer with 1 for yes and 0 for no.
A sentence can be idiomatic if it contains one of the following.
Phrasal verbs: example: put down (idiomatic: to kill; literal: to put something down).
Idioms: example: (A dog's breakfast: something that is disorganized).
Ambiguity: a sentence that can be understood literally and figuratively: example: (A good life depends on a liver: liver can be considered either to be an organ or a person).
Metaphors: example: (curtain of the night: metaphor that expresses the way the night came at that area).
Sayings: example: (Don't count your chickens before they're hatched).
Answer in the following JSON format: {"idiomatic": }.
Do not include ```.

User Prompt:

The sentence to classify: {sentence}

The user prompt was the same for zero-shot and one-shot systems prompts.

Giving the model no personality is a possible strategy for getting better results. In the task it is clearly stated that idiomatic expression should be detected by the model, what is meant by idiomatic expression. Furthermore, the answer requirement was mentioned, being integers

and only answering using these. Then the model was encouraged to read each sentence, and reason with its solution. Lastly, it should not have any words included in the response and should save the answers in a JSON format. In the one-shot example, the model adopts a personality being an English language and literature specialist. Part of the prompt is to specify sentences being idiomatic or not. To detect the idiomaticity of the sentences examples are being added to make it clear to the model. The model should follow the answer requirement only answering using integers and store the answers in a JSON format.

The following model parameters are used in combination with the prompt above:

1. Temperature: 0.8,
2. Max Tokens: 1,
3. Model: gpt-4-1106-preview
4. seed: 1

The parameters that are given to the model are the seed, the max tokens, the temperature, and lastly, the model itself. The seed stays always at 1 and is not changed, trying to get deterministic responses. The max token can vary, when it is higher than 1 (1 is the lowest) the model is prone to give a more elaborate answer. The longest max token is based on the context length of the model. The temperature controls the randomness of the output by the model, having a low temperature (2 is the highest temperature) outputs a less creative answer. Lastly, it is possible to change the model for different outcomes. For this thesis, the model gpt-4-1106-preview was chosen because when the thesis was being started at the beginning of January, this was the latest model.

The system prompt and the user prompt, which contains the sentence that should be analyzed, are separated by different files. The last step is the comparison of the predictions with the gold standard file.

3.3.2. Gemini Prompts and Experiment Setup

The following tactics are used when developing prompts for Gemini Pro.

1. Impersonation: the model is asked to adopt a person that fits best for the task given. Given the objective of the task, one of the possible personas could be a specialist in the English language and literature or a professional translator.

2. Use of Delimiters: the prompt is created with distinct compartmentalizations such as persona to adopt, task requirements, examples, and a sentence to analyze.

3. Examples: considering the fact that a definition of idiomatic expression has been created for the sake of this experiment, the same definition is given to the model with examples for each instance mentioned in the definition.

4. Output Requirements: the model is asked to produce an output in the JSON format with desired information embedded into the JSON.

5. Time to think: the model is given clear instructions to reach the conclusion on its own and reason with it.

6. Model's Parameters: the effectiveness of the prompts and the model is tested depending on the parameters that are passed to the model. The following parameters are considered:

    a. Temperature: the parameter controls the severity of randomness in token selection when generating a response from the LLM.

    b. Max output tokens: maximum number of tokens that can be present in the response.

    c. Top-K: the setting changes the behavior of choosing tokens for output. A value of 1 sets the model to choose the next selected token as the most probable, while a value of 4 results in the model choosing a token among the four most probable tokens by applying the temperature parameter.

    d. Top-P: the parameter sets the probability of the selected tokens. As mentioned before, tokens are selected from the most to least probable. For instance, if the model selected A, B, and C tokens with the probability score of 0.4, 0.3, and 0.1 and the Top-P value is set to 0.7, the system chooses token A or B as the next token based on the set temperature.

The following are the examples of zero- and one-shot prompts used in the experiments:

Zero-Shot:
System Prompt:
You are an English language and literature specialist.
Your job is to classify a sentence as either idiomatic or not.
Answer with 1 for yes and 0 for no.
A sentence can be idiomatic if it contains the following:
Phrasal verbs.
Idioms.
Ambiguity: a sentence that can be understood literally and figuratively.
Metaphors.

Sayings.
Give the answer in the JSON format like {sentence: “” , prediction: “” }.
Present your decision reasoning in the JSON.

User Prompt:
The sentence to classify: {sentence}

One-Shot:
System Prompt:
You are an English language and literature specialist.
Your job is to classify a sentence as either idiomatic or not.
Answer with 1 for yes and 0 for no.
A sentence can be idiomatic if it contains the following:
Phrasal verbs: example: put down (idiomatic: to kill; literal: to put something down).
Idioms: example: (A dog's breakfast: something that is disorganized).
Ambiguity: a sentence that can be understood literally and figuratively: example: (A good life depends on a liver: liver can be considered either to be an organ or a person).
Metaphors: example: (curtain of the night: metaphor that expresses the way the night came in that area).
Sayings: example: (Don't count your chickens before they're hatched).
Answer in the JSON format that looks like this: {"prediction": , "sentence": }.
Do not include ``` .

User Prompt:
The sentence to classify: {sentence}

The following model parameters are used in combination with the prompt above:

1. Temperature: 0.4,
2. Max Output Tokens: 500,
3. Top-K: 40
4. Top-P: 0.8

Based on the prompt example, it is visible that the model has been asked to adopt a personality by presenting itself as an English language and literature specialist, the task, requirements, and the sentence to analyze are separated, and every notion of the idiomatic expression definition has been given, apart from the examples themselves. Additionally, the output requirements have been mentioned and the model has been asked to explain its decision which gives it time to think.

Lastly, once the LLM has classified every given sentence, the model’s predictions are compared against the gold standard to calculate the accuracy number.

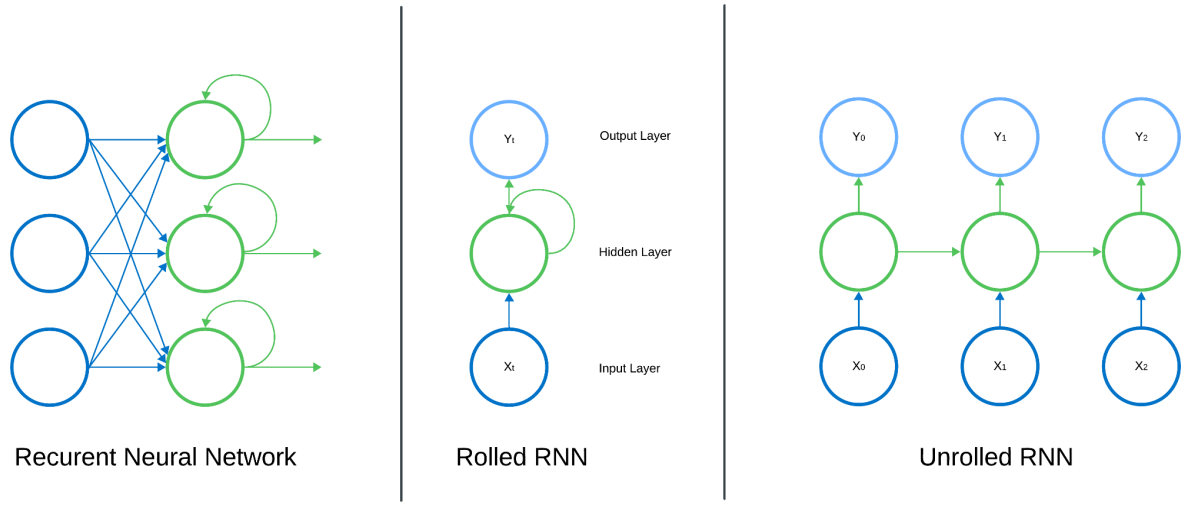
3.4 DNN

This section covers the second approach used in this research – Deep Neural Networks. Within this approach, two Recurrent Neural Networks have been built to test the ability of Neural Networks (NN) to detect idiomatic expressions in sentences. The RNNs diverge in their method of processing training and testing datasets, model architecture, and the partial evaluation of the model's performance.

3.4.1 RNN

A Recurrent Neural Network (RNN) is a type of artificial neural network that is used to process sequential or time series data. Some of the most popular use cases of these networks are machine translation, automatic speech recognition, and image processing.

Just like any other neural network, an RNN is dependent on training data in order to learn. One of the main factors in leveraging RNN abilities to work with data is its memory. While traditional neural networks treat every bit of information as a separate piece, RNNs are able to take information from prior learning to influence their judgment when working with new information (Pascanu et al., 2014).



Graph 3: RNN Visualization (Graph recreated from IBM, n.d.)

To better understand the logic behind an RNN, graph 3 represents the visualization of it. Based on the visualization, it is possible to determine why this type of network is termed recurrent, as the hidden state is recurrently fed back into the network with each new input sample.

Following the input of data denoted as x into the input layer, the hidden unit receives the previous state, represented as h_{prv}. Within the hidden layer, two quantities are calculated.

The first value corresponds to the new or updated state, denoted as h_{new}, which will be used for the next data point in the sequence. The second output is the network's output, denoted as y . Ultimately, the new state is determined as a function of the previous state and the input data.

During the initial phase of model training, a default initial hidden state is utilized. This default state varies according to the type of data under analysis, but generally, it is initialized to comprise entirely of zeros.

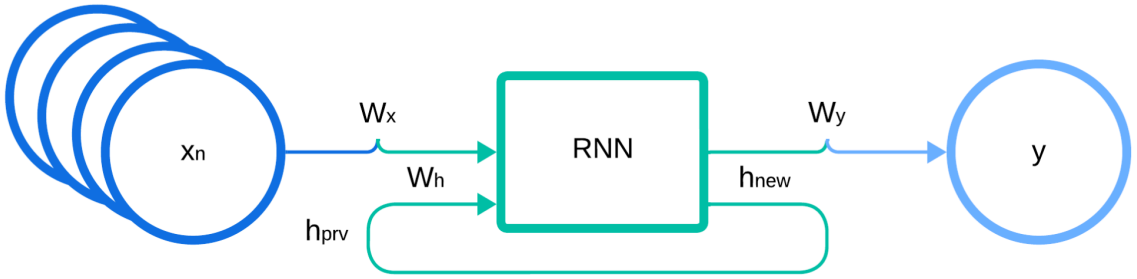
Additionally, the network incorporates supplementary parameters, including W_x, which is the weight matrix between the input and the hidden units. Another parameter, W_h, reflects the weights that are multiplied by the preceding hidden state. Given these established values, the mathematical expression for the new hidden state can be formally expressed as follows:

$$h_{new} = tanh(W_h . h_{prv} + W_x . x)$$

The output of the hidden layer is calculated by multiplication of the output weight matrix and the new hidden state, which may be represented as the following formula:

$$y = W_y . h_{new}$$

As the main topic of this research is the classification based on the fact whether a sentence contains an idiomatic expression or not, and, as established before, an instance of idiomatic expression may spawn from one to several words at once, a specific type of RNN is used which is called many-to-one.



Graph 4: many-to-one RNN visualization

Graph 4 represents such a type of network. It consumes a sequence of data and produces just one output. In this paradigm, the data flows in a single direction, from the input layer through intermediate hidden layers to the output layer. Each neuron in the network performs a simple mathematical operation, typically a weighted sum of its inputs followed by a non-linear activation function. The weights of the connections between neurons are adjusted during the

training process to minimize the error between the network's predictions and the desired outputs.

Feedforward neural networks are widely used for supervised learning tasks, where the network is trained on a dataset of labeled examples. The network learns to map the input data to the corresponding output labels by adjusting its weights to minimize the error. Once trained, the network can be used to make predictions on new, unseen data.

3.4.2 LSTM

Long Short-Term Memory (LSTM) is formally recognized as a distinct type of RNN specifically designed to process sequential data. The LSTM architecture's primary advantage over a standard RNN lies in its ability to avoid the occurrence of unstable behavior, such as vanishing and exploding gradients, during the backpropagation phase. In the LSTM network, the calculation of gradients for weight values follows the Chain Rule, which dictates that the gradients of earlier stages undergo multiplication by the gradients of later stages (Hochreiter & Schmidhuber, 1997).

Referring to Graph 2, the unrolled representation of an RNN illustrates how a basic RNN handles the processing of sequential data. Within the hidden layer, input data is passed through weights along a feedback loop, and the input data is subsequently multiplied by the weight values.

For illustrative purposes, consider the scenario where the weight value equals 2 and the number of sequential points is 65. In this instance, the multiplication formula takes the following form:

$$Gradient = Input_1 . 2^{65}$$

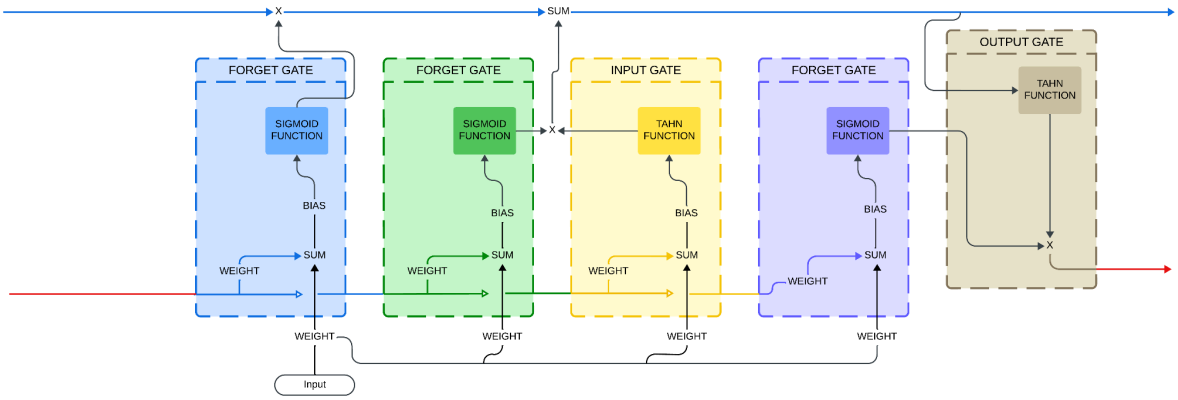
During the backpropagation process, if the gradient of the preceding layer increases, the weights of the subsequent layer exceed their intended values, prompting divergence in the gradient descent algorithm.

Conversely, when the weight value during backpropagation falls below 1 (e.g., 0.5), the gradient value approaches zero. The formula in such a case would be:

$$Gradient = Input_1 . 0.5^{65}$$

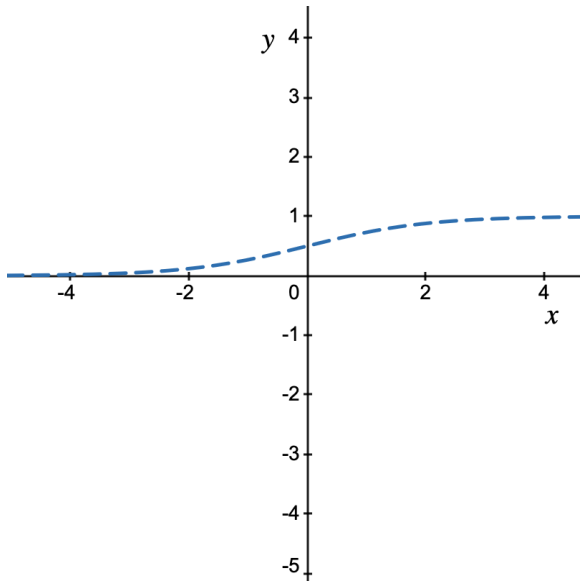
The resulting value gravitates towards zero, exhibiting vanishing gradient behavior. Vanishing gradients imply that when the gradients diminish, approaching a value close to zero, the new model weights will mirror those of the preceding layer without any alterations. (Wang & Jiang, 2016).

To mitigate these challenges, the utilization of LSTM architectures is employed. One of the main differences between a vanilla RNN and an LSTM network is their approach to data. Unlike the RNN, which relies on a solitary feedback loop for handling sequential data, LSTM introduces a dual-path architecture for sequential data processing. One path is dedicated to Long-Term data, while the other is reserved for Short-Term memories. Hence it is visible where the network's name is derived from. Graph 5 represents a unidirectional LSTM unit, which allows one look at each of its segments separately.

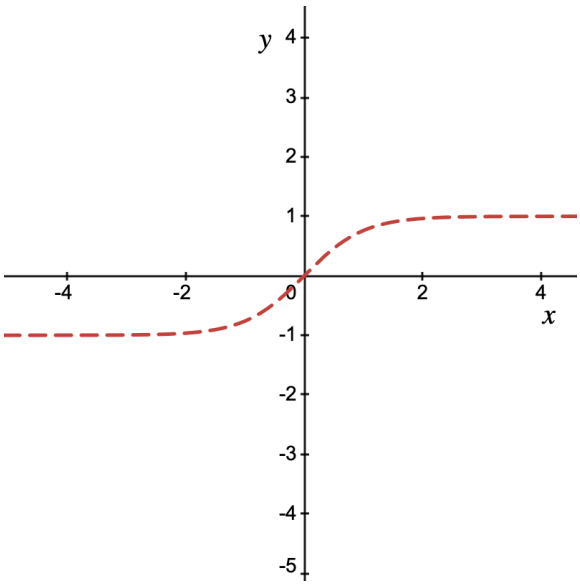


Graph 5: Unidirectional LSTM Module Visualization

The LSTM module uses both Sigmoid and Tanh activation functions and for a better understanding of the model's architecture, it is warranted to elucidate on these functions.



Sigmoid Activation Function



Hyperbolic Tangent (Tanh) Activation

Function

Graph 6: Sigmoid and Tanh Activation Function Visualizations

Referring to graph 6, the Sigmoid Activation Function takes any x-axis coordinate and converts it into a y-axis coordinate between 0 and 1 (Bonnell, 2011). The function can be represented as the following formula:

$$f(x) = \frac{e^x}{e^x+1}.$$

For illustrative purposes, by putting an x-axis coordinate as 10 into the function, the y-coordinate would be approximately 0.9995 and the formula would be the following:

$$f(10) = \frac{e^{10}}{e^{10}+1} \approx 0.99995.$$

Alternatively, by putting a value that is lower than 0, the y-axis coordinate would be closer to zero. For instance:

$$f(-5) = \frac{e^{-5}}{e^{-5}+1} \approx 0.00669285.$$

When it comes to the tanh function, the function takes any x-axis coordinate and turns it into a y-axis coordinate between -1 and 1 (Biswas et al., 2020). In the same vein as the sigmoid function, having inserted a positive x-axis value (e.g. 5) into the tanh activation formula, the y-axis coordinate would be the following:

$$f(5) = \frac{e^5-e^{-5}}{e^5+e^{-5}} \approx 0.9999.$$

Additionally, if a negative x-axis value (e.g. -3) is put into the activation function, the y-axis coordinate would be the following:

$$f(-3) = \frac{e^{-3}-e^{-(-3)}}{e^{-3}+e^{-(-3)}} \approx -0.995.$$

Having established the relationship of the activation functions to the whole LSTM module, one can run a mock test to understand how the module works.

The blue line above the module is named the cell state and represents the Long-Term memory of the module. Despite the fact that the state can be altered by the multiplication and the sum functions depicted in the graph, it does not contain direct weight and biases, allowing the data to be processed without causing the gradient to explode or vanish.

Additionally, the lines passing through each gate below are named the Short-Term Memories amd are directly modified by weights.

For the sake of the test, one could assume that the previous Long-Term Memory (LTM) is equal to 3, the Short-Term Memory (STM) is 0.5 and the input value is 1. Moreover, for the sake of the mock experiment, random weights and biases will be introduced to complete the test cycle. Upon entering the forget gate of the module, it is calculated how much the network should remember from the previous input.

$$(0.5 \times 2.00) + (1 \times 1.94) + 1.30 = 4.24.$$

1. 0.5 - STM value
2. 2.00 - Weight value
3. 1 - Input value
4. 1.94 - Weight value
5. 1.30 - Bias value

The equation below calculated the x-axis coordinate for the sigmoid activation function by using the values of LTM, STM, input, and bias values. By utilizing the x-axis coordinate value, it is possible to calculate the y-axis coordinate:

$$f(4.24) = \frac{e^{4.24}}{e^{4.24}+1} \approx 0.985797.$$

The last stage of the forget gate is to multiply the value of the LTM by the y-axis coordinate:

$$3 \times 0.985797 \approx 2.95.$$

A. 2.95 - LTM value

The following result indicates that the value of the LTM is reduced, hence this part of the module is named the forget gate, as it determines the percentage of the previous memory to be remembered.

Having passed the forget gate, the data enters the second part of the module where the input gate generates the value of the Potential Long-Term Memory, which denotes how much of the newly processed information might be retained, and the second Forget State determines how much of the Potential Long-Term Memory should be remembered in the end.

The calculations for this stage would be the following:

1. Potential Long-Term Memory:

$$(0.5 \times 2.74) + (1 \times 2.45) - 0.24 = 3.58.$$

   A. 0.5 - STM value
   B. 2.74 - Weight value
   C. 1 - Input value
   D. 2.45 - Weight value
   E. -0.24 - Bias value

   Having calculated the x-axis value for the tanh activation function, the y-axis value is the following:

$$f(3.58) = \frac{e^{3.58} - e^{-3.58}}{e^{3.58} + e^{-3.58}} \approx 0.998447.$$

   The calculations show that the new potential memory value is approximately 0.99.

2. Percentage of Potential Memory to remember:

$$(0.5 \times 2,74) + (1 \times 1.65) + 0.54 = 3.56.$$

   A. 0.5 - STM value
   B. 2.74 - Weight value
   C. 1 - Input value
   D. 1.65 Weight value
   E. 0.54 - Bias value

   By calculating the x-axis value for the sigmoid function, the y-axis value is the following:

$$f(3.56) = \frac{e^{3.56}}{e^{3.56}+1} \approx 0.972348.$$

3. Value to remember:

$$0.998447 \times 0.972348 \approx 0.97.$$

4. New LTM:

$$0.97 + 2.95 \approx 3.92.$$

Finishing the calculations, the value of the new LTM has been established.

The final state of the LSTM module is the update of the STM value.

1. The Output Gate Value

   The LTM value is put into the tanh activation function of the output gate as the x-axis value to produce the y-axis value:

$$f(3.92) = \frac{e^{3.92}-e^{-3.92}}{e^{3.92}+e^{-3.92}} \approx 0.999213.$$

2. Percentage of Potential Memory to remember:

   The last forget gate is activated to calculate how much of the new potential STM should be remembered:

$$(0.5 \times 1.53) + (1 \times (-0.25)) + 0.57 = 0.32.$$

   Having calculated the x-axis value, the y-axis value is the following:

$$f(0.32) = \frac{e^{0.32}}{e^{0.32}+1} \approx 0.57.$$

3. New STM:

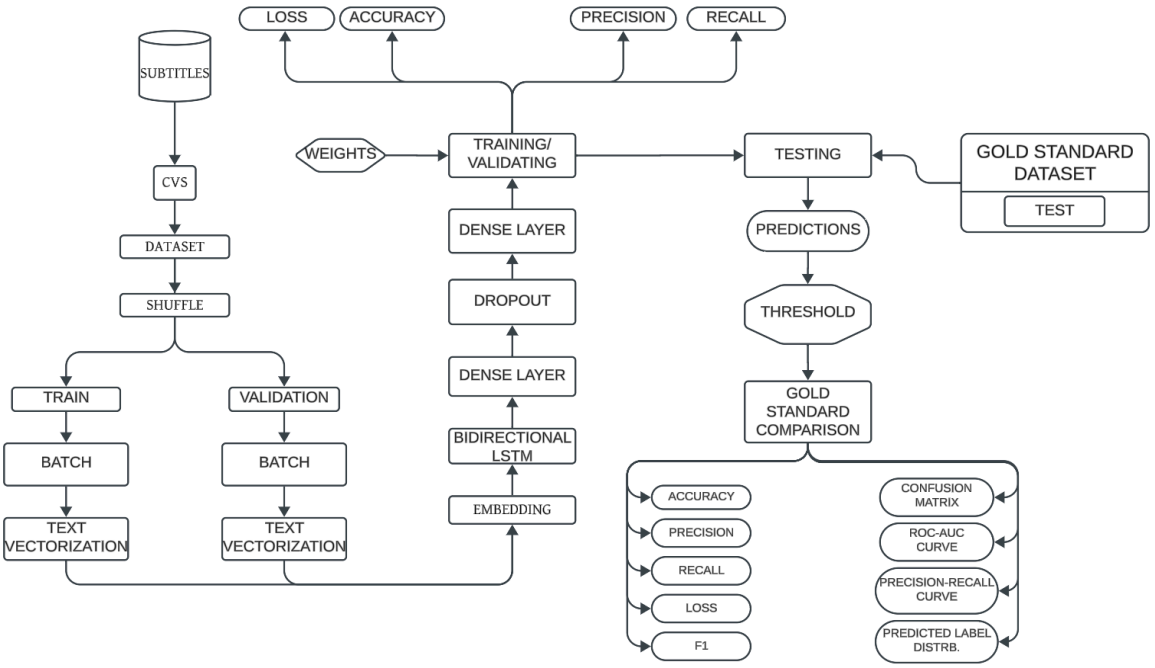
$$0.57 \times 0.999213 \approx 0.56.$$

The new STM value is 0.56, which is also the final output from the entire LSTM module. Additionally, the output value of the first LSTM module is the input value for the second module.

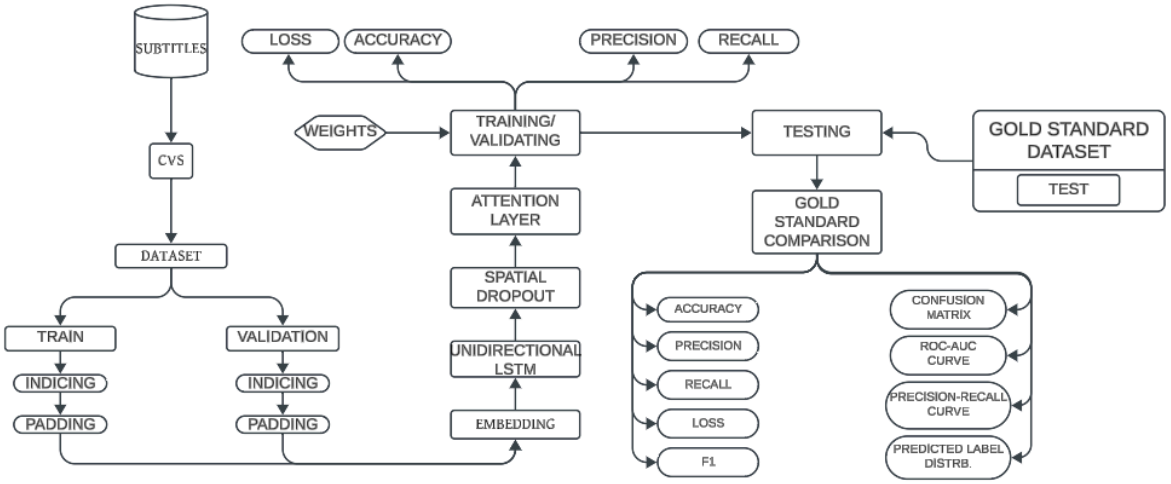
Having completed the explanation of the LSMT module cycle, it is crucial to outline the full model's architecture used for the sake of idiomatic expression detection.

3.4.3 Model Architecture

In the context of the DNN approach, two Recurrent Neural Networks (RNN) are used for a binary classification approach. While both models adhere to the same fundamental approach, several distinctions contribute to their differentiation such as the processing of the train and test data, the core of the model, and its evaluation approach.



Graph 7: RNN with a bidirectional LSTM layer architecture visualization



Graph 8: RNN with a unidirectional LSTM and activation layers architecture visualization

The model depicted in Graph 7, while passing the data preparation stage, loads the data, removes the duplicates, and shuffles the dataset for more randomization before splitting it into train and validation Additionally, model one implements batching of train and test datasets in the data processing part, which means that the model does not process each instance at once, but a grouped number of instances. The following approach helps to enhance computational efficiency. After that, both train and validation datasets are sent into the model's core where the data goes through the text vectorization and embedding layers for training and validating purposes. Consequently, the data goes through a bidirectional LSTM layer, which implies that the data is processed in both forward and backward directions, which should improve the model's understanding of the relationship between sequences (Graves et al., 2005).

Having passed the LSTM layer, the processed sequences are put through the dense layer, dropout, and another dense layer, which constitutes the final stage of the model's training.

Since the dataset that is used for this model is highly imbalanced, weights are added to the training step to secure the model's ability to pay greater attention to the underrepresented class. The weight value is generated based on the distribution of each class in the dataset.

Upon the successful training cycle, the following metrics are produced based on the validation data: Accuracy, Precision, Loss, and Recall.

Entering the testing state of the model, a dataset that has been used to conduct experiments on LLMs is used to test the model's ability to detect idiomatic expressions in sentences. The model returns predictions which are applied to determine a label for the test sentences based on a threshold value.

In comparison, the model presented in Figure 8 preprocesses the entire dataset by splitting it into training and validation and applying inducing and padding.

After processing the datasets, the training and validation parts are put into the core model where the sequences are processed through an embedding layer. The next 2 layers are different from the model presented above. The embeddings are passed through a unidirectional LSTM layer, implying that the data is processed only in the forward direction. After that, the processed data is sent into the spatial dropout layer, which determines how much of the data should be forgotten.

Additionally, having passed the dropout layer, the data is moved through an attention mechanism layer, which is supposed to apply a focus on various parts of the sequences produced by the LSTM output layers (G. Liu & Guo, 2019).

The training results, accuracy, loss, precision, and recall, are generated based on the output of the attention layer.

The last node of the model is its testing based on the test dataset that has been used for the LLMs testing practices. The testing part of the model classifies sentences whether they contain idiomatic expressions or not, which is in the later stage of testing compared to the gold standard labels.

The last step is the assessment of both models' abilities by producing the following metrics:

1. Accuracy
2. Recall
3. F1
4. Precision
5. ROC-AUC
6. Precision-Recall Curve
7. Confusion Matrix
8. Predicted Label Distribution based on count

3.4.4 Rationale for Selection

There are several factors contributing to the choice of the LSTM-infused RNN approach.

The first notion that should be taken into account is the fact that idiomatic expression, particularly idioms, exhibits long-range dependencies wherein the meaning of the entire idiom is derived from a combination of words instead of individual components (Espinal et al, 2019:1).

Secondly, the advantage of the RNNs is their ability to process sequential data by retaining information from previous parts of the sequence in their memory, which makes them a potent candidate in idiomatic expression detection (Keren & Schuller, 2016).

Lastly, traditional RNNs experience the issue of the vanishing gradient, which creates an arduous problem of preserving previously learned information. Hence, the implementation of Long Short-Term Memory (LSTM) mitigates the issue of the vanishing/exploding gradient by regulating information within the cells to be retained and updated over longer sequences (Hu et al., 2018).

3.5 Data Collection

The data used for the research has been acquired from Plint AB. The preferred information, English subtitles, was extracted from the dataset. The dataset itself contained subtitles in English. Within the scope of this research sentences might be referred to as instances.

The interest within the LLM approach focuses exclusively on English subtitles, therefore no other information from the data has been extracted. After having extracted all English sentences, all formatting was removed and the sentences were lowercased. Additionally, all of the sentences were lowercased. However, punctuation was retained.

Since all of the sentences that were present in the dataset were subtitles, most of the sentences were not presented as full instances. Therefore, sentences that did not end with a period were concatenated with the next subtitle until full sentences emerged.

After having extracted and cleaned the sentences, they were annotated following a binary classification method – (0) being a sentence without an idiomatic expression and (1) being a sentence with an idiomatic expression. Idiomatic expression, as defined earlier, consists of the definitions of idioms, phrasal verbs, metaphors, ambiguity, and sayings. The annotated sentences are used as the experiment's gold standard dataset containing approximately 2500 sentences. Out of these sentences, 500 sentences were extracted and split into two sub-datasets and annotated by each of the authors. Then, the sub-datasets were exchanged between the authors to be annotated again. The decision came from the idea of calculating a Cohen's Kappa to determine how arduous it would be for two annotators to agree on sentence annotation before passing it to the models.

After establishing a solid understanding of what an idiomatic expression was and raising the value of Cohen's Kappa, the remaining 2000 were annotated separately.

Having completed the annotation, it was discovered that upon the concatenation of sentences, not all instances were presented as full sentences, which resulted in their deletion from the dataset. The final version of the dataset contained 2409 examples.

	LABELS		EVAL. METRICS	
DATASET	1	0	NUMBER OF SENTENCES	% OF POSITIVES
	511	1898	2409	21.2%

Table 1: LLM Dataset Label Distribution

As shown in Table 1, the dataset that is used for the LLM experiment execution exhibits a considerable imbalance, with 1898 instances classified as non-idiomatic and 511 instances as idiomatic. This means that the dataset comprises approximately 21,2% idiomatic instances.

	LABELS		EVAL. METRICS	
DATASET	1	0	NUMBER OF SENTENCES	% OF POSITIVES
	348	1244	1592	21.8%

Table 2: RNN Dataset Label Distribution

In order to recreate the same experiment environment for the RNN approach, an additional 1592 sentences were extracted and annotated to train the model. The dataset created for the training cycle resembles the quality of the LLM dataset by containing 21.8% of positively labeled sentences.

After the successful RNN training on the newly synthesized dataset, the LLM dataset is passed into both networks to re-create the experiments conducted during the LLM testing and produce the results of the models' testing cycles.

3.5.1 Security

With regard to the creation of datasets and their annotation for training and testing purposes, it is essential to underline the measures implemented to mitigate potential disturbances on the datasets and model.

In the context of this research, a disturbance involves the deliberate corruption of data used for either training or testing. Furthermore, it can be orchestrated on the entire model's architecture, resulting in a degradation of its performance (Huang et al., 2017).

There are two main approaches one could follow to conduct a disturbance:

1. White box - one has access to the model's architecture and/or parameters.

2. Black box - one does not possess access to the model's architecture or parameters but can alter the data that is used for the model's training and testing cycles (Huang et al., 2017).

Regarding the style of the disturbance, two main approaches are considered in the scope of this research:

1. Poisoning - one corrupts the training data or its labels to worsen the model's performance (Biggio et al., 2012).

2. Evasion - one tempers with the model's output by corrupting the data sent to the trained model (*IBM Documentation*, 2024).

Regarding this research, the authors have access to both LLMs' parameters, meaning that any possible disturbance conducted on either GPT-4 or Gemini Pro would be considered white-box. Additionally, the authors have access to both the architecture and parameters of both RNNs, which can also be deemed as white-box.

Both datasets were openly available to both authors, which implies that the datasets were open to either poisoning and/or evasion.

To ensure that these disturbances were not conducted on either the models or the datasets, the following regulations were implemented:

LLM:

1. The parameters of each experiment are saved as a separate JSON file, thus, if one is unsure about the results of the experiment, the experiment can be recreated.

2. Author 1 is only responsible for the Gemini Pro experiment environment creation (e.g. code construction) and does not have access to the environment of author 2.

3. Author 2 is only responsible for the GPT-4 experiment environment creation (e.g. code construction) and does not have access to the environment of author 1.

RNN:

1. The parameters of each experiment are saved as a separate JSON file, thus, if one is unsure about the results of the experiment, the experiment could be recreated.

2. Author 1 is only responsible for the creation of BiLSTM-RNN architecture and does not have access to the model's architecture of author 2

3. Author 2 is only responsible for the creation of LSTM-RNN architecture and does not have access to the model's architecture of author 1.

Datasets:

1. Once the gold standard dataset has been annotated, neither of the authors can change their labels, mitigating the possibility of poisoning the dataset.

2. Once the dataset for the RNN training and validation purposes has been created and annotated, neither of the authors can change their labels, mitigating the possibility of poisoning or envisioning the training and validation data.

By following these rules, all of the experiments become reproducible and neither the neural networks' architectures nor the datasets are prone to any of the aforementioned disturbances during the research.

4. Experimental Results

The following chapter will cover the performance metrics for our chosen models GPT-4 and Gemini. Furthermore, it focuses on the comparison between the two models and our RNN approaches. The chapter concludes with the analysis of the results.

4.1 Performance Metrics

The initial phase of the experiments, wherein the classification task was undertaken by LLMs, entailed the evaluation of each experiment's outcomes utilizing the following metrics:

1. Precision: shows the number of all correctly classified sentences that contain idiomatic expressions based on all sentences that were classified as containing idiomatic expressions.

2. Recall: out of all sentences containing idiomatic expression, how many are correctly detected by the models.

3. F 0.5: shows the balance between recall and precision, putting more focus on precision.

The selection of the F 0.5 metric reflects a prioritization of minimizing false positives within the model's output. Given the research context – a corporate setting where human verification of translation accuracy is employed – reducing false positives directly translates to decreased workload for human reviewers, which aligns with the need for efficiency in real-world applications.

Additionally, the mean values of each metric have been calculated and incorporated specifically for graph 18 to make both approaches comparable.

The rationale behind the utilization of the aforementioned metrics varied based on the specific metric. Normally, while the accuracy score presents the overall representation of how often the LLM’s classifications are correct, it can be potentially misleading due to the highly imbalanced nature of the dataset employed in the experiments. Precision is useful in this classification, as it showcases how reliable the output of the LLM and/or RNN can be when it predicts a positive instance. Moreover, it aids in minimizing false positive instances. Lastly, the F 0.5 score depicts the output of both precision and recall and is considered a valuable parameter to determine the model’s performance with regard to false positive and false negative instances.

In the context of the RNN methodology, various additional metrics were introduced to supplement the previously mentioned evaluation criteria in order to assess the performance of the models. These additional metrics include:

1. Receiver Operating Characteristic - Area Under the Curve (ROC-AUC): This metric provides an overall assessment of the model's performance without relying on a specific classification threshold. It gauges the model's ability to distinguish between positive and negative classes.

2. Precision-Recall - Area Under the Curve (PR-AUC): Given the imbalanced nature of the dataset, it is important to consider the Precision-Recall curve in conjunction with the ROC-AUC. This curve illustrates the model's capacity to accurately classify true positive instances, regardless of the true negative rate.

The use of these additional metrics ensures a comprehensive evaluation of the models' performance, taking into account both the overall classification abilities and the model's behavior in the context of the imbalanced dataset.

4.2 Experiment Results

4.2.1 LLMs

Both GPT-4 and Gemini Pro have been tested by conducting individual experiments using various prompts and parameters. 32 experiments were conducted for GPT-4 and 24 for Gemini Pro.

For the further experiment analysis, only the top five experiments of each model from training and testing were selected. The experiment selection happened based on the selection of experiments by scoring the highest precision score.

Referring to GPT-4, in all five experiments, all prompts share a common core:

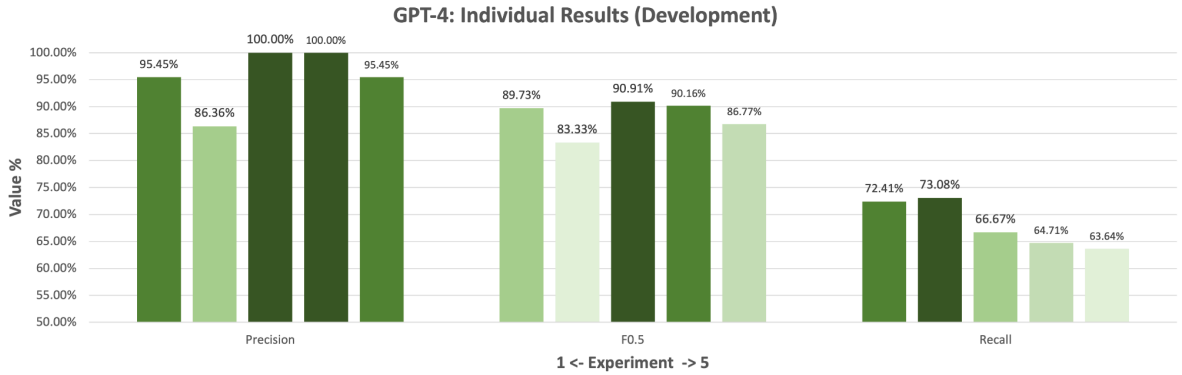
1. Definition of idiomatic expression
2. Examples of figurative elements (idioms, phrasal verbs, ambiguity, etc)
3. Instructions to classify a sentence

However, four out of the five experiments were conducted using zero-shot prompts, which means that the model was not given examples of each figurative element. Instead, the model

relied solely on its internal knowledge and understanding of language to perform the classification task.

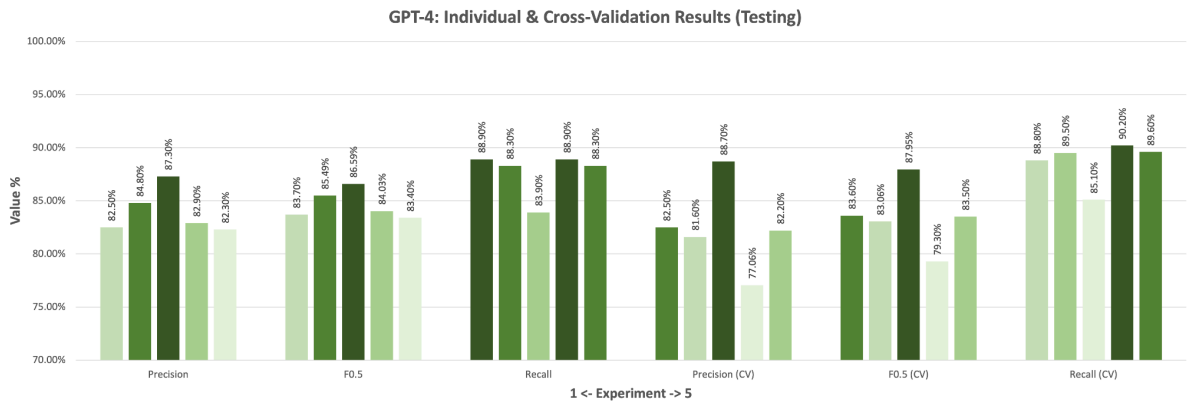
Additionally, each experiment employed a different temperature setting, which influenced the model's behavior during generation. A higher temperature setting allows the model to be more creative and unpredictable, whereas a lower temperature setting makes the model more straightforward and conservative in its responses. The 5 best prompts and its parameters can be found in Appendix B for GPT-4 and Appendix C for Gemini Pro.

Moreover, as part of the cross-validation testing, the prompts witten for GPT-4 were tested on Gemini Pro and vice versa, including the corresponding parameters. Additionally, the 5 best prompts and parameters are given in the appendix.



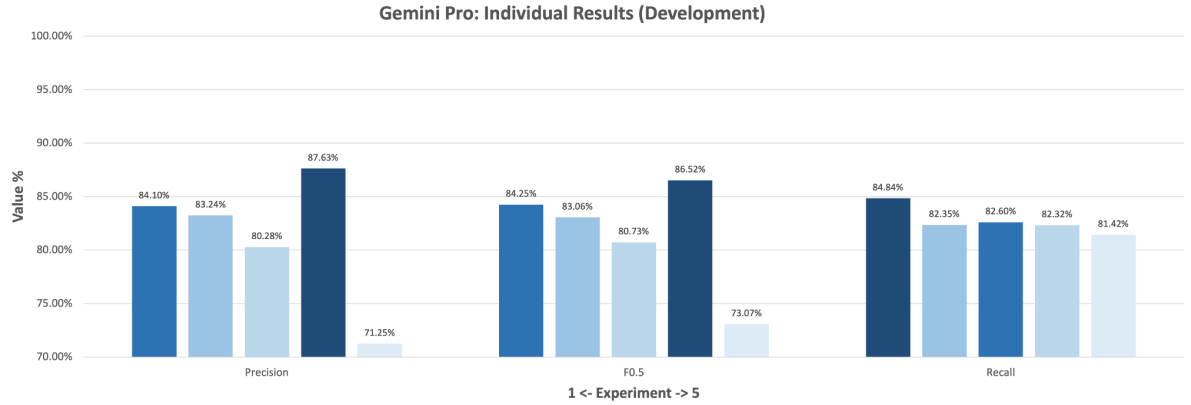
Graph 9: GPT-4 Individual Results (Development)

Graph 9 shows the results of top 5 experiments conducted on GPT-4 when developing prompts. Looking at the graph and its individual experiments, it can be seen that the precision is very high, twice it reaches 100%. This means that twice the model detected all sentences that it classified as having idiomatic expression as correctly. Even though the precision is high, interestingly, the recall is comparatively low. Especially at experiment 3 and 4, where the precision scores 100% it can be seen that the recall is particularly low compared to the precision result. This means that all of the sentences that have been recognized by GPT-4 to contain idiomatic expression were actually idiomatic; however, a plethora of idiomatic sentences in the entire dataset have not been recognized by the LLM. The weighted average between recall and precision is F 0.5, indicating that the balance between them is high.



Graph 10: GPT-4: Individual & Cross-Validation Results (Testing)

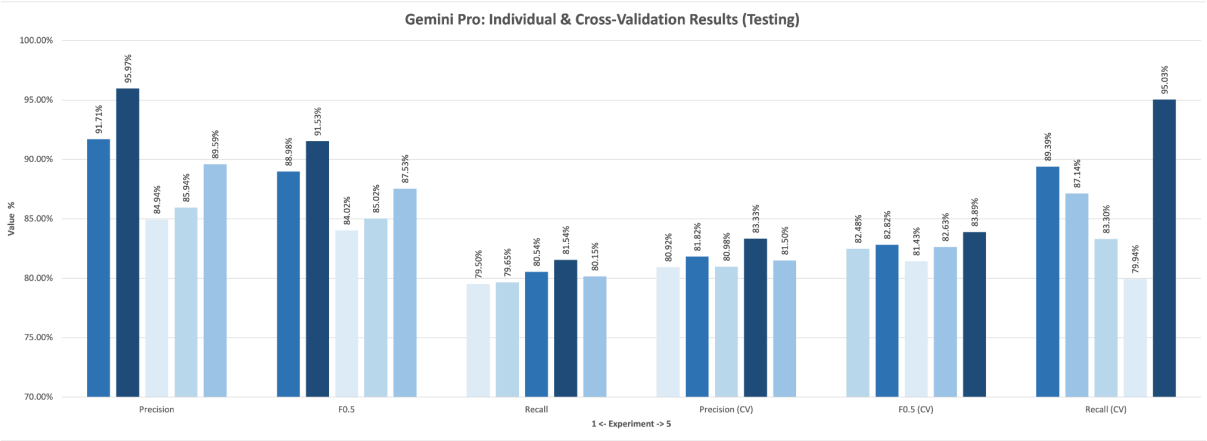
Graph 10 shows the results of GPT-4 individual testing and its cross-validation with Gemini Pro prompts and settings. Looking at the graphs, it can be seen that overall the precision testing with GPT-4 prompts and settings is always above 80%, ranging from 82% to 88.9%. This indicates that all sentences that the model marked as containing idiomatic expression, get detected within this range correctly as containing idiomatic expression. Comparing the precision using GPT-4 prompts and settings and the cross-validation results, it can be seen that the precision varies. In only 3 experiments out of 5 experiments the precision in cross-validation is approximately similar to the precision when using GPT-4 own prompts and settings. Especially experiment 4 shows a low precision using cross-validation. Comparing recall there cannot be seen any significant difference between recall-testing with GPT-4 prompts and settings, and using cross-validation. Comparing the testing results to the development results, in the test results precision is lower and recall is higher than during development.



Graph 11: Gemini Pro Individual Results (Development)

Graph 11 shows the results of top 5 experiments conducted on Gemini Pro when developing prompts. Looking at the graph and its individual experiments it can be seen that the precision ranges from 80% to 87%, indicating that the model is detecting all sentences that it classified as having idiomatic expression as correctly. It can be seen that the recall is also relatively high compared to the precision measurement, which means that, out of all of the idiomatic sentences present in the dataset, Gemini Pro is able to detect a relatively high number of them. This indicates that, out of the entire dataset, 81.42% of all the idiomatic sentences are

detected. Additionally, experiment 5 shows that, under certain parameters used for this experiment, out of all of the sentences that the model deemed as idiomatic, only 71.25% of them were treated as such.



Graph 12: Gemini Pro Individual & Cross-Validation Results (Testing)

Graph 12 shows the results of Gemini Pro individual testing and its cross-validation with GPT-4 prompts and settings. Looking at the graphs, the precision using Gemini Pro's own prompts and settings is consistently higher than the precision during cross-validation. The cross-validation recall compared to the recall with Gemini Pro's prompts and settings is higher in 4 out of 5 experiments. This indicates that Gemini Pro detects sentences containing correctly idiomatic expressions more often with GPT-4 prompts and settings, than with its own.

Concluding it can be said that the models behaved differently during development than during testing. GPT-4 scored a high precision, and low recall during development, however during testing the precision was slightly lower, and the recall was approximately similar as during development. Gemini Pro scored a relatively high precision and recall during development, however during testing the precision was lower, the recall was higher than during development. This difference could be due to the parameters used within their prompts in every experiment.

4.2.2 LSTMs

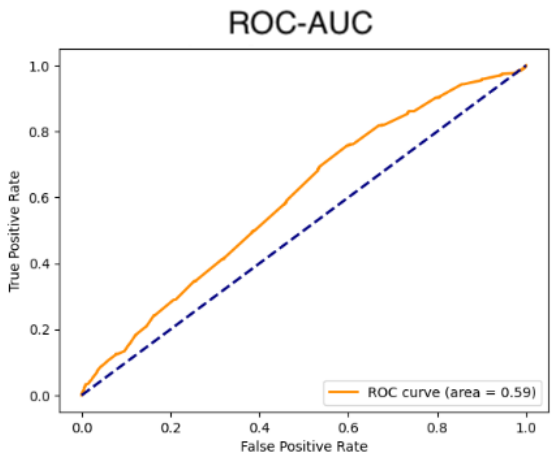
Between the comparison of the two RRN models, Table 3 containing the two BiLSTM-RNN & LSTM-RNN models, gives a good overview. Additional model results can be seen in Appendix D and E. These results of the bidirectional and unidirectional model are based on 5 experiments and the calculation of the mean result. The mean results based on all metrics are represented in the table.

	Model	
Metric	**Bi-LSTM-RNN**	**LSTM-RNN**
Recall	90.28%	25.24%
Precision	95.31%	79.27%
F 0.5	94	54
ROC-AUC	99.70%	59%
PR-AUC	97.71%	27%
Probability Threshold	80%	–
Dataset size	2409	2409

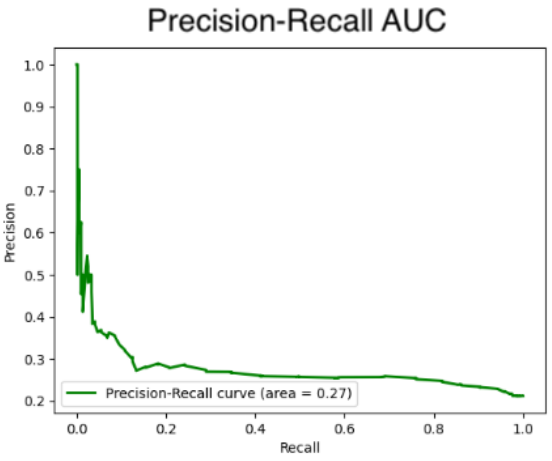
Table 3: BiLSTM-RNN & LSTM-RNN Performance Results

It can be clearly seen that the bidirectional model has higher values for all of the metrics. When looking at recall the bidirectional model scores 90.28% whereas the unidirectional model only scores 25.24% of recall. Additionally, the bidirectional model has a higher precision being 95.31%, whereas the unidirectional model has a precision of 79.27%. Based on F 0.5 the bidirectional model scores the value of 0.94, whereas the unidirectional model scores 0.54.

With regard to the probability threshold, its implementation was exclusive to the bidirectional model. During the analysis of the probabilities generated by the model's output, the threshold serves as a control mechanism in determining the appropriate label assignment to a processed sentence. If the probability generated by the model's output is below the threshold value, the label is set to 0; otherwise, the label is assigned a value of 1.

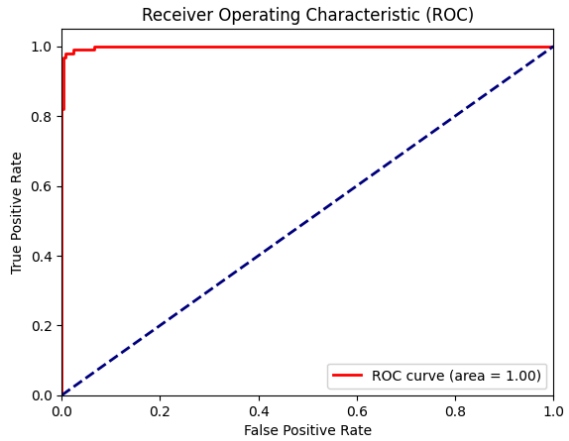


Graph 13: LSTM-RNN: ROC-AUC

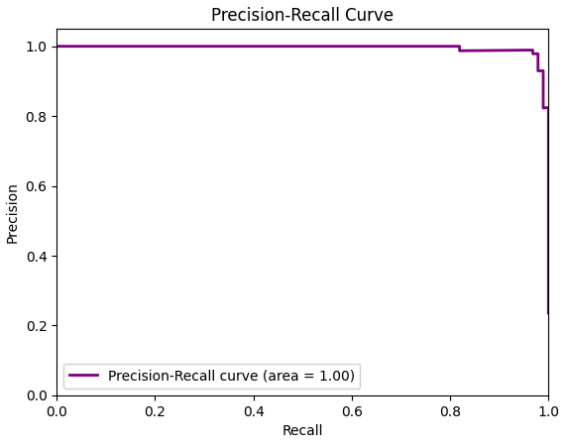


Graph 14: LSTM-RNN: Precision-Recall-AUC

Graph 13 and graph 14 show two curves: the ROC-AUC on the left, and the precision-recall curve on the right. The ROC-AUC shows the balance between recall and false positives. It shows how many sentences were identified as containing idiomatic expressions in them correctly, and how many sentences were falsely identified as containing idiomatic expressions in them; however, they do not contain idiomatic expressions. Furthermore, this relies on how large the dataset containing classifiable sentences is. The precision-recall curve on the right side shows how precision and recall change at different classification thresholds.

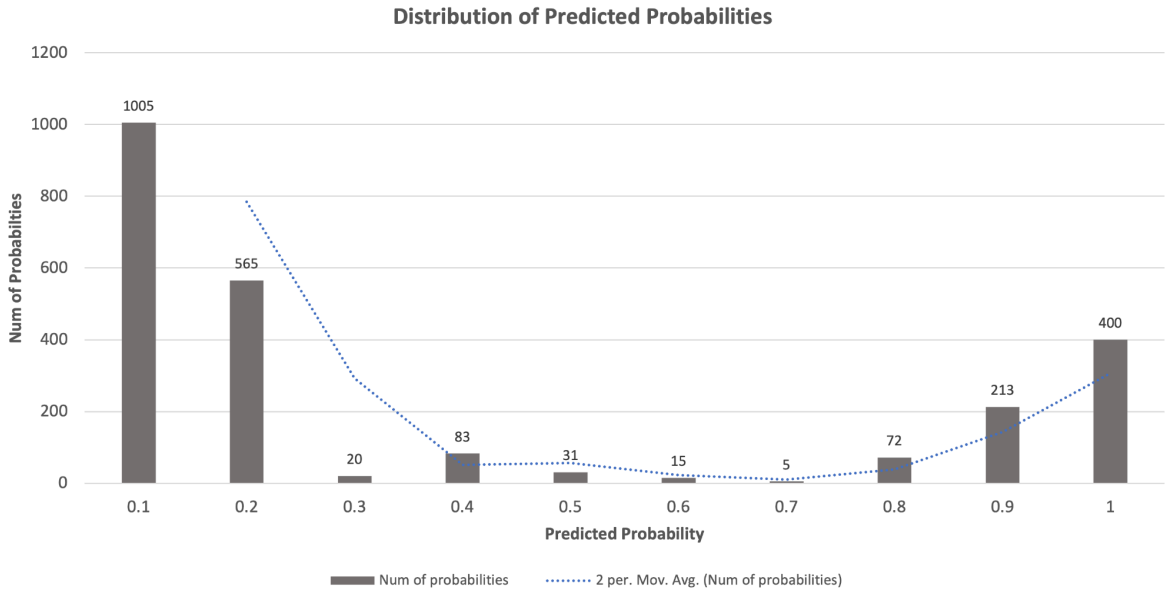


Graph 15: BiLSTM-RNN ROC-AUC



Graph 16: BiLSTM-RNN Precision-Recall-AUC

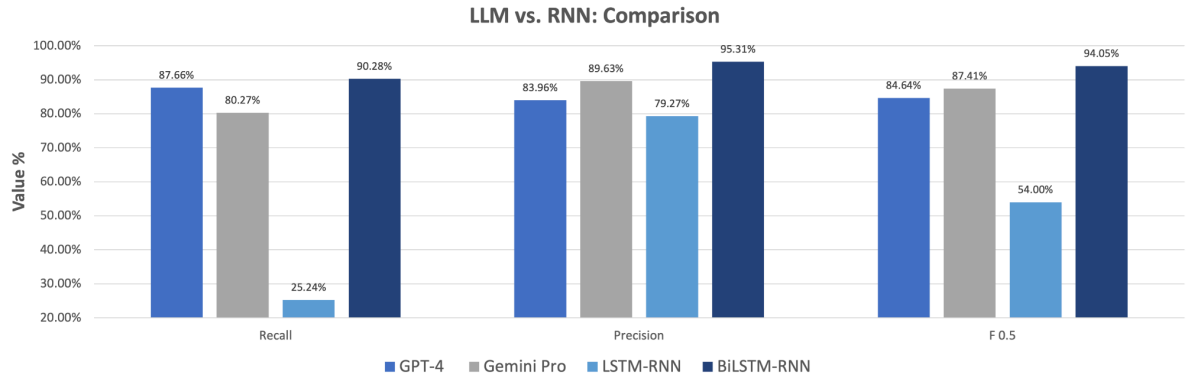
Graphs 15 and 16 represent the equivalent metrics as graphs 13 and 14, however for the BiLSTM-RNN approach.



Graph 17: BiLSTM-RNN: Probability Distribution

Graph 17 shows the predicted probability distribution of the BiLSTM model. An examination of the graph facilitates an assessment of the model's certainty in assigning a specific

probability to a sentence. It is evident that the model exhibits a tendency to assign probabilities that are either closer to the range of 0.1-0.2 or to the range of 0.8-1, meaning that the model has two peaks. This illustrates the model's clear separation of predictions rather than a uniform spread of probabilities. The model is confident classifying a particular sentence as idiomatic or non-idiomatic.



Graph 18: LLM vs. RNN: Comparison

Graph 18 shows a comparison between two methods for idiomatic detection by calculating mean scores from each approach: LLM using GPT-4 and Gemini Pro and RNN using LSTM-RNN and BiLSTM-RNN.

Having reviewed the differences between the two LLM models separately, it can be said that regarding the LLM models GPT-4 performs well on precision, however, not well on recall. Gemini Pro on the other hand, scores better in recall, however, not well on precision. When comparing the models during cross-validation, it can be seen that the highest values are miscellaneous across the experiments. Gemini Pro scores better using GPT-4 settings than vice versa. Precision is more balanced during cross validation of Gemini Pro, than for GPT-4.

When looking at the RNN models, it can be seen that the BiLSTM-RNN model achieves overall higher metrics than the LSTM-RNN model and both LLMs.

4.3 Analysis of Results

This section focuses on the analysis of results for both LLM models and both RNN models. It presents the LLM results and gives examples of sentences that were classified as idiomatic or not idiomatic by the LLM. The RNN results are presented and set in comparison to both RNN models. All the sentences that are presented as examples, are taken from the dataset. Due to restrictions the sentences have been paraphrased, however, the idiomaticity has been maintained showing in the examples in bold.

4.3.1 GPT-4

The quantitative results of GPT-4 will be presented again, before the qualitative examples are shown.

During development, GPT-4 is able to classify a high number of sentences containing idiomatic expressions, consistently achieving high precision. The recall score is not consistently as high as precision.

1. Precision: 86.36% - 100%
2. F 0.5: 83.33% - 90.91%
3. Recall: 63.64% - 73.08%

Regarding the testing GPT-4 precision was not consistently as high as in development. However, recall achieved higher scores during individual testing than during individual development. The results are the following:

1. Precision: 82.30% - 87.30%
2. F 0.5: 83.40% - 86.59%
3. Recall: 83.90% - 88.90%

In experiment 1, 2, and 3, zero shot prompts were used. In experiment 4, we instructed the model to act as a professional English translator, in experiment 5 the model mimicked a language and literature specialist. In experiment 3 the temperature was set to 0.8, in the other experiments the temperature was set to 1.0, and the max token was changed. The difference between experiment 1 and 2 is the difference in the usage of max tokens (max tokens=2 and max tokens=1, respectively). The same has been done in experiment 4 with 1 max token and in experiment 5 with 2 max token. Further information regarding prompts and settings can be found in Appendix B.

Since experiment 1 and 4 achieve the highest recall, it is best to examine several instances of its annotation output.

The experiment results show that the model is able to correctly identify the following cases of idiomatic expression:

1. Idiom: “This feels quite **nerve-racking** to me.” - *referring to something that is causing stress or nervousness*

2. Lexical Ambiguity: “It would **ease my pain** if we could postpone the meeting.” - *referring to it would be stress relieving if the meeting could be postponed*

3. Phrasal verb: “Relax and **shake off** the stress.” - *referring to letting go, relaxing*

4. Metaphor: “It’s **out there**, start searching for it.” - *referring to in the world outside*

5. Saying: “I’m **on my way**.” - *referring to a person moving/traveling*

However, GPT-4 also classifies sentences as idiomatic when it should not:

1. “-John Smith. -This is Australia’s best friend.”
2. “There could be a loss of species like sevengills.”
3. “You don’t have to worry about anything.”
4. “You gave that old house some character.”
5. “Don’t worry.”

It is unclear why GPT-4 classifies them as idiomatic. It could be that the model detects proper names (example 1) and species names (example 2) as idiomatic. *Character* in example 4 could be classified as idiomatic due to impersonation. In example 3 and 5 it is unclear which part of the sentence the model classified as idiomatic.

The following examples showcase when the model counted a sentence as non-idiomatic, whereas it should have been considered as one:

1. Idiom: “What **the future holds** for her, I can’t say.” - *referring to that the future is uncertain or unknown*

2. Lexical Ambiguity: “He’s in here **causing terror**.”- *referring to being literally responsible for creating problems/terror or figuratively causing terror in creating fear*

3. Phrasal verb: “During this season flowers **pop up** early.” - *referring to something appearing*

4. Metaphor: “Finally my **secrets** are **out**, and I don’t need to hide anything longer.” - *referring to something was spoken out loud*

5. Saying: “Take as much chocolate as you want, **don’t be shy**.” - *referring to that there is enough chocolate to be used*

Reviewing these examples we can see that they are being detected as non-idiomatic by the model. These examples should be classified as idiomatic, since they showcase typical examples based on the definition we made. Speculations can be that the model took these idiomatic expressions literally, like *don’t be shy*.

The results for cross-validation are the following:

1. Precision: 77.06% - 88.70%
2. F 0.5: 79.30% - 87.95%

3. Recall: 85.10% - 90.20%

This shows that the recall is higher when using cross-validation with Gemini Pro, than during normal testing.

The model marks the following sentences as correctly containing idiomatic expressions:

1. Idiom: “Sometimes, I don’t understand her, I need a bit more time to **get a read** of her.” - *referring to understanding her behavior*

2. Lexical Ambiguity: “If he’s **stretching for me** later, I would appreciate that.” - *referring to reaching out, getting in contact*

3. Phrasal verb: “Hopefully, this problem is solved soon, so I can **tick** it **off** and **move on**.” - *referring to finishing something and letting go of something*

4. Metaphor: “I’m short on money, so my **lady** is paying for everything.” - *referring to oneselves girlfriend*

5. Saying: “This could be **the key** to our problem.” - *referring to that something is crucial*

However, GPT-4 classifies also sentences as idiomatic which are not idiomatic:

1. “I didn’t expect such a place, I wouldn’t have thought that.”
2. “This tastes more like food served to animals than serving it to humans.”
3. “This feels really uncomfortable, they don’t make an effort to leave.”
4. “I don’t see that he’s ready to love her.”
5. “Was he okay when you left him?”

These examples are non-idiomatic, the model could have seen idiomaticity in the following examples: *okay* as colloquial speech in example 5, *food served to animals* in example 2 as metaphor for bad tasting food. It is unclear which idiomaticity the model saw in example 1 and example 3. In example 4 the model could have marked it as idiomatic based on *see that he’s ready to love*, this could be marked as metaphoric, since love can’t be seen through eyes.

The model has classified the following sentences as non-idiomatic, while, in fact, they are idiomatic:

1. Idiom: “I **got into this** course I wanted, now it’s **step by step** to finish it.” - *referring to being able to start a course and focussing on repeated small progress*

2. Lexical Ambiguity: “All this **attention being paid to her**.” - *referring to that she receives attention*

3. Phrasal verb: “We **put out** some pictures on the internet.” - *referring to providing access to the pictures*

4. Metaphor: “You can **see it in her**, that she’s romantic.” - *referring to somebody being romantic and its visibility*

5. Saying: “You scored a **ten out of ten** in that game, congratulations.” - *referring to that the person seems flawless*

Being classified as non-idiomatic could be based on the fact that the model had difficulty understanding them, and took them literally, like *attention being paid to her* or *step by step.*

In examining the impersonalization approach in depth, it has been noticed that the LLM is able to generate better results when impersonating a translator. One plausible explanation for this pattern is that translators are primarily concerned with conveying textual meaning from one language to another. This task demands a profound grasp of linguistic nuances, yet may not present a necessity for an exhaustive understanding of literary or grammatical rules. English language and literature specialists, in contrast, may concentrate more on stylistic elements of language, including grammar and vocabulary. It is possible that GPT-4 possesses greater proficiency in mimicking the core tasks of a translator than the more intricate role of a literature specialist.

Two approaches seem to be particularly effective when working with GPT-4:

1. No Impersonation, Low Temperature, Zero-Shot Settings: In this approach, the model is not required to impersonate any specific role, a low temperature setting is used to promote conservative behavior, and the prompt is presented in a zero-shot format without presenting specific figurative elements.

2. "Translator" Impersonation, Low Temperature, Zero-Shot Settings: This approach requires the LLM to impersonate a translator in order to complete the task. Similarly, the impersonation is accompanied by a low temperature and zero-shot prompt settings.

4.3.2 Gemini Pro

Speaking about the Gemini Pro’s results, the LLM can classify a high number of sentences containing idiomatic expression, consistently achieving high precision, and recall scores in

development demonstrating the model's preference to strike a balance between precision and recall with the following bands:

1. Precision: 71-84.10%
2. F 0.5: 73-84.25%
3. Recall: 81-84.84%

Regarding the testing part of the research, Gemini Pro still showcases even higher results as well, but this time, giving more preference towards precision rather than recall. The metric results are the following:

1. Precision: 85-95.97%
2. F 0.5: 84-91.53%
3. Recall: 79-81.54%

In experiments 1, 2, and 5, the effects of temperature settings were tested to determine how they affect the LLM in idiomatic expression detection tasks. All three experiments employed the same zero-shot prompt and impersonalization techniques.

In contrast, experiment 3 employed the same impersonalization techniques but used a one-shot prompt. This means that for each idiomatic expression notion, the model was provided with an example of how the idiom is used in context. This was done to investigate whether providing the model with an additional source of information would improve its performance.

Finally, experiment 4 utilized a distinct prompt, different impersonalization, and temperature configuration. The temperature setting was also adjusted to optimize the model's performance. The prompts and temperature settings are accessible in Appendix C.

The comparison between the development and the testing results reveals that, given more sentences, 3 out of 5 times that correspond to experiments 1, 2, and 5, Gemini Pro is inclined to give preference to precision rather than recall. Regarding experiments 3 and 4, the model strikes a balance between the two metrics, meaning that one-shot prompt in experiment 3 and moderate temperature might be more suitable for the idiomatic detection task. While experiment 4 also finds a balance, a high temperature utilized in the model’s settings makes the experiment less reproducible, meaning that further testing of the experiment might return different results.

Furthermore, the implementation of two distinct roles in the prompts, an English language and literature specialist and a lexicographer, had minimal impact on the results. This suggests that Gemini Pro primarily relies on its inherent linguistic knowledge and capabilities, rather than heavily relying on the contextual information provided by the assigned roles. This observation highlights the LLM's ability to leverage its internal resources effectively to generate consistent and accurate responses.

Since experiment 3 manages to strike the most balance between precision and recall, it is best to examine several instances of its annotation output.

The experiment results showcase that the model is able to correctly identify the following cases of idiomatic expression:

1. Idiom: “She’s walking **into the lion’s den** full of enemies.” – *referring to entering a dangerous or hostile situation.*

2. Ambiguity (Lexical): “Why is she making **pissed chicken** again??”. – *it is unclear whether it is a certain type of recipe named “pissed chicken”, or the person is cooking and is drunk.*

3. Phrasal Verb: “Seriously? **Get outta here**, I’m not dealing with that.”. – referring to asking *to leave.*

4. Idiom: “Let’s be honest, that **kicked our asses**”. –referring *to defeating someone.*

5. Idiom: “We need to be **on the same page** about it.”. – referring *to having the same kind of understanding of a situation.*

However, Gemini Pro also identified the following instances as idiomatic, when, in fact, they were not:

1. “It’s cold enough to stay inside”.
2. “Let’s make them wait for a bit”.
3. “-I think you’re still angry. - I’m not. I’m smiling”.
4. “very nice. wow. -wow. look at the size of this.”

The first sentence could have been marked as idiomatic due to its colloquial structure. Whilst the sentence is grammatically correct and sends a clear message, it adopts a conversational structure commonly used in idiomatic expressions.

In the second sentence, while the structure is straightforward and grammatically sound as well, the inclusion of “for a bit”, which is commonly used in informal settings to suggest a short period of time, could have contributed to its classification as idiomatic.

The third sentence presents a conversational exchange that resembles idiomatic speech patterns. The brevity of the dialogue, alongside its use of contractions and informal language, mimic the structure of idiomatic expressions commonly presented in informal speech.

Moreover, while each sentence individually may not be idiomatic, the overall style of the conversation could have contributed to its misclassification.

The last sentence might have also been misclassified due to its conversational nature and interjections such as “wow”.

While it is unclear why Gemini Pro classified the aforementioned sentences as containing idiomatic expression, one of the possible speculations might be the fact the sentences exhibit characteristics of informal speech or conversational pattern, as well as the presence of informal speech elements such as contractions, colloquial phrases and direct dialogue , which Gemini Pro might have perceived as idiomatic in nature.

Lastly, there are a number of examples when the model counted a sentence as non-idiomatic, whereas it should have been considered as one:

1. Idiom: “He’s so in love with her, but she’s gonna **drop the bomb** that they’re breaking up”. *– it could refer to delivering shocking news.*

2. Idiom: “We’re sailing around, and sometimes what you can see in the water is a bit **hit or miss**”. *– referring to not being able to tell if something can be good or bad.*

3. Idiom: “I **put some things together** and realized she was cheating on me.” *– referring to understanding a topic or situation.*

4. Ambiguity (Syntactic): “I wish I **had Taylor Swift’s nose**”. *– referring to either obtaining Taylor Swift’s nose or having one’s nose done to make it look like one of Taylor Swift’s.*

5. Idiom: “Are you sure? She’s a **character**!”. *– referring to either mentioning a specific character in a story (e.g book) or describing a person as interesting or unusual.*

When comparing the individual testing results to the cross-validation, the outcomes are the following:

1. Precision: 81-83%
2. F 0.5: 81-84%
3. Recall: 79-95%

It is evident that when implementing prompts and settings tailored for GPT-4, the results shift, prioritizing recall over precision. Out of 5 experiments, only experiment 3 manages to find a balance between these two metrics, scoring 80.98% in precision and 83.30% in recall.

Investigating the prompt and the temperature setting of experiment 3 that obtained a balance between the metrics, it becomes possible to determine that the results have been obtained using a zero-shot prompt without mimicking any specialist or profession. However, a high temperature of 0.8 is utilized alongside the prompt, which might indicate that further

experimenting with the prompt and the setting might not bring the same results.

Referring to experiment 4, when precision is preferable over recall, scoring 83.33% and 79.94%, a zero-shot prompt with the "professional translator from English mimicking" and a temperature of 1 are used. However, since the experiment used a high temperature value, upon further testing, these results may not occur again.

These findings might imply that the prompts and corresponding settings generated for GPT-4 might be less suitable for Gemini Pro if the main goal of the research is to find a balance between the metrics to not only find actual idiomatic expression in the sentences that the system marks as idiomatic, but to detect all the possible idiomatic expression instances in the whole dataset as well.

Additionally, it is worth discerning one of the experiments to establish an understanding of what sentences have been classified correctly and where the model made mistakes. Since experiments 3 and 4 are less reproducible, it is better to pick one of the more robust experiments. For this task, experiment 1 is picked as it has been conducted using the lowest temperature, facilitating its reproducibility in the future.

Based on the output, it is visible that the model is able to classify many instances of idiomatic expression. Some of them are the following:

1. Idiom: "Age is just a number, I'm **50 going 20** and I feel so young at heart". – *referring to being physically 50 years old but feeling like a 20-year-old.*

2. Ambiguity (Lexical): "Gotta bring the **chick** back". – *referring to either a chicken or a vulgar name for a woman.*

3. Saying: "Hearing her say she had covid again, **my heart just sank**. Everything stopped". – *referring to experiencing a strong feeling of sadness or despair.*

4. Ambiguity (Lexical): "Why are you so **dry**?" – *referring to either experiencing a physical state of dryness (e.i. dry skin) or being unresponsive in a conversation.*

5. Idiom: "Your **underhand** mean casts suspicion on a man". – *referring to a person who is dishonest or unfair in a concealed way.*

In this experiment, Gemini Pro also identified the following instances as idiomatic, while they were not annotated as such:

1. "Feels like females need more persuading than men".
2. "-how are you? -All good here, thanks!"
3. "-Wanna watch a video? -Sure thing".
4. "She don't wanna pay no bills".

5. “Five years younger, ding, ding. Try 26 younger, haha”.

The first sentence presents a case of pronoun omission, which might have contributed to the model’s output results in having the sentence classified as idiomatic.

The second sentence reflects a standard conversational exchange marked by its informality. While each sentence separately may not be deemed as idiomatic, the pattern of short, direct responses as the use of contractions mimic conversational patterns usually presented in idiomatic expressions.

The third sentence resembles the same structure as the second sentence referring to its conversational structure and the use of contractions. Additionally, the omission of pronouns at the beginning of the dialogue might have also contributed to the misclassification.

The fourth sentence presents non-standard grammar and double negation, that are normally found in informal speech. While it is grammatically incorrect in standard English, the sentence reflects linguistic patterns usually found in idiomatic expressions tied to informal settings.

The last sentence might have also been characterized as idiomatic based on the use of the interjection like “ding, ding”, which could have contributed to Gemini’s inability to classify them as non-idiomatic.

While it might still be arduous to pinpoint why the LLM classified these sentences as idiomatic, one hypothesis could be that the sentences exhibit a certain state of conversational and informal nature, as well as the presence of specific linguistic features, such as double negation, which are commonly found in idiomatic expressions.

Finally, below are some of the sentences that the model has classified as non-idiomatic, while, in fact, they should have been classified as idiomatic:

1. Ambiguity (Lexical)/Idiom: “-Is this what I think it is? -It’s a **cracker**, isn’t it?” – *referring to either a piece of food or something that is considered as funny or excellent.*

2. Idiom: “European diseases **took their toll** on traditions.” – *referring to the destructive impact of disease to a cost. ‘*

3. Ambiguity (Lexical)*:* “Backyard chills by the pool are **my jam**, and **barbie** made it even better”. – *referring to a thing a person enjoys (my jam). Referring to either a Barbie toy or a slang word for barbecue (barbie).*

4. Idiom: “I’m excited about **what the future holds** for us”. – *referring to expressing*

uncertainty about future events.

5. Phrasal Verb: "I've gotta **take off**, but thanks for lunch again!" – *referring to leaving the place.*

To recapitulate, four main conclusions can be drawn:

1. If the main goal is to prioritize precision over recall, then Gemini Pro performs best using the "English language and literature specialist" impersonation paired with either a low temperature of 0.2 or a medium value of 0.5 and a zero-shot prompt.

2. If the main goal is to prioritize recall over precision, then any of the prompts generated for GPT-4, except experiments 3 and 4, are suitable for this task.

3. If the main goal is to strike a balance between precision and recall, then Gemini Pro performs best using the one-shot prompts paired with the "English language and literature specialist" impersonation and a medium value temperature of 0.4.

4. Gemini Pro is prone to classifying sentences as instances containing idiomatic expression if these sentences contain elements of colloquialism, as well as conversational and informal nature, which might not be a valid indicator for idiomaticity in a sentence.

4.3.3 RNNs

Speaking of the RNN model, there were two different models built: a BiLSTM (bidirectional) RNN and an LSTM (unidirectional) RNN. For the selection of the data, the same experiment was run 5 times, additionally, mean results for each metric were calculated. This was done for both models. The reason for calculating the mean results was to check the models' robustness.

The results for the bidirectional model are the following:

1. Precision: 95.31%
2. F 0.5: 0.94
3. Recall: 90.28%

Regarding these results it can be seen that the bidirectional model with almost full certainty mark sentences contain idiomatic expressions as correctly. In addition, it scores high at the precision, meaning it detects sentences containing idiomatic expression with 95% likelihood.

The results for the unidirectional model are the following:

1. Precision: 79.27%
2. F 0.5: 0.54
3. Recall: 25.24%

It can be seen that the results for the unidirectional are worse than the results from the bidirectional counterpart. This can be referred back to the different structure of the models and, in the case of the unidirectional model, to the simpler architecture.

By investigating some of the results produced by the BiLSTM-RNN, it becomes possible to assess whether or not the model is able to perform a binary classification task in the domain of idiomatic expressions.

The examples of correctly labeled sentences are the following:

1. Idiom: "That house of hers will be left **to my name**" – *referring to belonging to someone or inheriting something.*

2. Phrasal Verb: "Hey, stop, you're **wearing out** this car" – *referring to degrading the quality of something by excessively using it.*

3. Phrasal Verb: "Do you think you can **take her down** there?" – *referring to either bringing the person to a physical location or defeating her.*

4. Metaphor: "It's a **pot of adventure**" – *referring to comparing the abstract concept of adventure to a tangible object and implying a sense of abundance.*

5. Metaphor: "Zebras stay next to the banks of the river to avoid **jaws** lurking in the water". – *referring to predators.*

Additionally, BiLSTN-RNN identified the following examples as idiomatic despite the fact that they were not considered as such:

1. "Guess what we've got here!. -party!"
2. "...and they're talking and talking".
3. "-right now? - no, maybe later, though, ok?"
4. "This thing right there, like I don't understand it".
5. "oh no. she's getting mad".

The first sentence presents a conversational exchange marked by its informality. The use of contractions colloquial language and, possibly, the exclamation "party!" might have biased the model leading to the misclassification, which might imply that the model might present a degree of inability to differentiate certain informal conversational patterns and idiomatic expressions due to limited training data.

The second sentence features repetition and a casual tone. While grammatically correct, the mentioned features might have contributed to the misclassification, meaning that the model is not able to differentiate between repetitive language used for emphasis and genuine idiomatic expressions.

The third sentence includes a conversational pattern characterized by hesitancy and informality. The use of contraction, hesitation markers, such as “maybe later, though”, and the informal agreement “ok” contribute to its informal nature, resembling idiomatic expressions used in casual conversations. The misclassification by the model could be attributed to its inability to differentiate between informal conversations and idiomatic expressions.

The last sentence contains informal language “oh no” and a colloquial expression “getting mad”, both of which are features mostly found in idiomatic speech. The misclassification by the BiLSTM-RNN model could be tied to its failure to differentiate colloquial expressions and genuine idiomatic expressions.

When speculating about these results, one potential explanation for the observed misclassification may be attributed to the limited availability of training data, which may have skewed the model's ability to learn more nuanced linguistic patterns. Consequently, certain non-idiomatic sentences may have been mistakenly classified as idiomatic.

The following examples are the examples of sentences originally marked as containing idiomatic features, but they were misclassified by the model:

1. Idiom: “I’ve **been through hell and back**” – *referring to experiencing an extremely difficult event.*

2. Metaphor: “These connections and emotions - I’m **taking it home**” – *referring to withdrawing someone’s emotional investment in something/someone.*

3. Ambiguity (Lexical)/Idiom: “**Buckle up!**” – *referring to either physically putting a seatbelt on or preparing for something.*

4. Ambiguity (Lexical): “I didn’t **get it**” – *referring to either not obtaining an object or not understanding a certain concept.*

5. Idiom: “He’s **blowing** my **mind**. He’s done such a great job” – *referring to expressing intense positive surprise.*

By investigating the unidirectional LSTM, some sentences labeled as correctly idiomatic are the following:

1. Idiom: “I’m done, I’m not **standing for it** any longer.” - *referring to having stopped defending a position/an argument*

2. Phrasal Verb: “It feels that **something’s off** in here, so let’s **put our fingers on it**.” - *referring to something being strange and clarifying something*

3. Syntactic Ambiguity: “And my first thought was: “**is she fucking with us**”?” - *referring to sexual intercourse or that the person is intentionally causing confusion/understands what is going on*

4. Metaphor: “Why does he spend **face time** with me?” - *referring to meeting in person*

5. Sayings: “He is alone **in his corner**, nobody else is there.” - *referring to having nobody on your side*

Sentences that were originally marked as containing idiomatic features, but that were misclassified by the model as being non-idiomatic:

1. Idiom: “His behavior is all **up and down** all day.” - *referring to that there are feelings of unsteadiness*

2. Phrasal Verb: “Glad that you took the opportunity and **showed up**.” - *referring to that somebody arrived*

3. Lexical Ambiguity: “She always seemed to have a **business mind**.” - *referring to that the person is good at business*

4. Metaphor: “My feelings are all **through the roof**.” - *referring to that the anxiety is very high*

5. Sayings: “He’s a mess, and he’s feeling on the **wrong side of the tracks** most days.” - *referring to the fact that he feels poor/criminal most of the days*

These examples should be classified as idiomatic, since they showcase typical examples based on the definition we made. Speculations can be that the model took these idiomatic expressions literally, like *through the roof* or *wrong side of tracks*.

Sentences that were wrongly labeled as idiomatic, however, they are not idiomatic:

1. “Please contact us mr smith, that’s when we knew.”
2. “The white shark is an indolent hunter.”
3. “The tiny joey was more of an embryo than a baby, after being born it crawled up into the pouch of its mother’s belly.”

4. “This is unacceptable behavior.”
5. “Don’t hesitate to say hi.”

It is unclear why the model detects proper names like *mr. Smith* and *tiny joey* (example 1 and 3), species names like *white shark* (example 2) and colloquial speech like *hi* as in example 5 as idiomatic. In example 4 it is unclear which part of the sentence the model classified as idiomatic.

4.3.4 Commonalities

Despite having various answers, there are several general similarities and differences in LLMs’ and RNN’s responses; the following conclusions can be drawn regarding their outputs.

1. Both approaches are able to identify instances of idioms such as “into the lion’s den” or “on the same page” and others, which might imply that when dealing with instances that exhibit features that could only be discerned in a figurative way, both approaches are able to process such data successfully.

2. Some idiomatic instances that exhibit ambiguity tend to be misclassified more by Gemini Pro in the LLM and LSTM-RNN in the DNN approaches, indicating that Gemini Pro and LSTM-RNN might not have been trained on enough data that would represent identical or similar instances, allowing both models to process any other data that resembles the misclassified sentences correctly. Additionally, LSTM-RNN might exhibit additional challenges in identifying such instances successfully due to its unidirectional nature.

3. Both Gemini Pro and LSTM-RNN tend to misclassify sentences that exhibit conversational patterns, opening a question whether both models had enough data resembling such patterns during their training cycles.

4. GPT-4 and BiLSTM-RNN did not successfully classify instances that present incorrect grammatical structures, which also raises a question whether the training data for both models contained enough examples of similar patterns to allow the models to learn.

5. Apart from Gemini Pro, the other three models are able to successfully process long examples and mark them as either idiomatic or not.

6. None of the approaches managed to classify the following sentence correctly: “I’m excited about **what the future holds** for us”, which raises the question whether the transparency of the sentence’s meaning can lead to uncertainty in models’ classification of the sentence. “Future holds” might be considered a common

metaphorical phrase, but it does not deviate far from literal meaning as some idioms do (e.g. “kick the bucket” for dying).

5. Conclusion

This thesis has worked on automatically classifying idiomatic expression in sentences, using two different approaches - an LLM approach and an RNN approach - to establish how well the two different approaches detect idiomatic expression in sentences.

The first research question explored the extent to which LLMs (GPT-4 and Gemini Pro) and DNNs (LSTM-RNN and BiLSTM-RNN) could classify sentences based on the presence of idiomatic expressions. The findings demonstrate that both approaches hold promise for this task.

Both GPT-4 and Gemini Pro achieved comparable performance, particularly during cross-validation. They exhibited precision scores of 87% and 95%, respectively, and recall scores of 88% and 83%.

Regarding the DNN approach, the BiLSTM model significantly outperformed the LSTM model, achieving a precision of 95% and a recall of 90%. The LSTM model, on the other hand, exhibited lower performance with a precision of 79% and a meager recall of 25%.

The second research question aimed to determine if there were distinguishable differences between the chosen LLMs (GPT-4 and Gemini Pro) based on their performance in idiomatic expression detection. While both models achieved similar results, a slight edge was observed, where Gemini Pro demonstrated a slight advantage with a precision of 95% compared to GPT-4's 87%. However GPT-4 exhibited a modest advantage with a recall of 88% compared to Gemini Pro's 83%.

The third research question investigated how well the DNNs performed in idiomatic expression detection and if one architecture prevailed over the other. The analysis revealed a clear distinction showing that the BiLSTM model demonstrably outperformed the LSTM model across all metrics,achieving significantly higher precision and recall.

To recapitulate, the findings suggest that while both LLM models achieved comparable performance, particularly during cross-validation, further exploration of ensemble learning approaches might be beneficial to leverage their combined strengths. In contrast, the BiLSTM architecture demonstrably yielded superior results within the DNN framework.

Speculating why the BiLSTM model outperformed both LLMs, one could mention that the BiLSTM model is specialized meaning that it is specifically designed for the task of classifying sentences as idiomatic or non-idiomatic and trained on specific data for the mentioned task.

On the other hand, LLMs, such as GPT-4 or Gemini Pro, are generalists. They’re trained on massive amounts of various texts to understand language broadly. While scalable, the LLM approach might be less successful at classifying sentences as either idiomatic or not.

5.1 Future Research

To ameliorate the limitations of the current study, future research could explore several promising trajectories. One such trajectory is investigating the efficacy of both Gemini Pro and GPT-4 and other extensive language models in discovering idiomatic expressions across multiple languages. This would facilitate broader applicability of the findings, thereby providing insights into the cross-lingual capabilities of LLM models.

Furthermore, a crucial domain for future research is evaluating and comparing the capabilities of different Gemini models, including Gemini Pro, to ascertain the most suitable model for idiomatic expression detection. This would entail assessing factors such as model size, training data, and performance on idiomatic expression detection tasks.

Additionally, augmenting the amount of data used for experimentation is crucial to capturing a more diverse spectrum of idiomatic and non-idiomatic expression instances for both LLM and RNN models. This could involve collecting more data from various sources.

Lastly, exploring the utilization of supplementary techniques, such as transfer learning and fine-tuning, could further enhance the performance of both LLM and RNN models in capturing idiomatic expression.

5.2 Ethics

Between the company that represents a non-academic approach, and the university that represents an academic approach, conflicts arose in accessing the data. We are unable to make the original data available, due to reasons that could be negative for the company.

We have been working on a comparably small dataset. Working with this dataset we have not found any harmful speech or social biases. However, we cannot exclude the fact that the larger dataset may include biases or harmful speech.

When working with prompting the LLMs, explicitly GPT-4 and Gemini Pro, as well as, training and testing recurrent neural networks developed for this research, it is crucial to mention the topic of energy consumption.

Energy consumption refers to the amount of energy that is used to execute a certain process, which is measured in kilowatt-hours (kWh) (REPSOL, 2023).

Speaking about the training of Gemini Pro and its carbon footprint, Google does not provide any documentation on the environmental impact when developing the LLM.

The following passage offers speculation on how much CO2 Gemini Pro might produce when undergoing a training cycle based on its counterpart - ChatGPT, which is based on GPT-3.

In their research, several scientists reported that training GPT-3 produces approximately 552 metric tons of CO2e (Patterson et al., 2022). Consequently, taking into account Google's remarks mentioned in Chapter 3 stating that Gemini Pro is GPT-3's rival, it is safe to assume that Gemini Pro might be more computationally exhaustive producing a bigger amount of carbon emissions.

During the development of the BiLSTM-RNN module, the model has been re-trained 24 times to ensure that it achieves the best results possible. The model's training parameters as well as each version of the trained model have been stored locally to mitigate the need to excessively restart the program to obtain the same results should the need to use a specific version arise.

Additionally, most parts of the code have been automated and put into a pipeline, minimizing chances for human error and thus, the need to re-run the program.

The training and testing cycles of the BiLRM-RNN model were conducted using an Apple MacBook Pro 2020 with the M1 chip, 8 gigabytes of RAM, and 512 gigabytes of SSD. The full training cycle took approximately 20 minutes and taking into account that the program has been used 24 times, the total training time is approximately 8 hours, which results in ≈ 0.004kg of CO2 usage, which equates to ≈0.047 bananas.

When focusing on the carbon footprint of GPT-4, OpenAI does not provide any information on its website. However, they mention that GPT-4 has been trained on "Microsoft Azure AI supercomputers"[9]. Deng et. al (2023) who examine different LLMs on its Product Carbon Footprint qualifying the life-cycle emissions of products mention the training data of GPT-4 being 1 trillion parameters. Furthermore, they mention that GPT-4 and GPT-3 datasets consist of Wikipedia, book corpora, Webtext2 dataset, and lastly Common Crawl dataset. The last dataset, contributes 60% of the whole compounded dataset, the petabytes of the data resulting in 12 years of web crawling. Furthermore, GPT-4 contains 8192 maximum tokens, and its training data lasts up to September 2021.

As Patterson et al (2023) mention that GPT-3 the estimated carbon emission in training is 552 tCO2e, and the resulting energy consumption is 1287 MWh.14, based on 175B parameters. Based on that, and the knowledge that GPT-4 is the newest model by OpenAI, we can assume that GPT-4's energy consumption is even higher. Additionally, OpenAI mentions that they focus on "more data and more computation to create increasingly sophisticated and capable language models."[10]

[9] *OpenAI*, 2023. https://openai.com/gpt-4
[10] *OpenAI*, 2023. https://openai.com/gpt-4

During the development of the unidirectional LSTM model, the model has been retained 24 times to ensure that it achieves the best results possible. The model's training parameters, in addition to each version of the trained model, have been stored locally.

The training and testing cycles for the unidirectional LSTM were conducted using an Apple MacBook Air Retina 2020 with 8 gigabytes of RAM, and 256 gigabytes of SSD. The full training cycle took approximately maximum 5 minutes, the testing cycles took approximately 30 minutes. This results in ≈ 0.012 kg of CO2 usage. That equals to ≈ 0.155 bananas.

The aforementioned calculations have been conducted using the CO2 GU mltgpu tutorial presented by Simon Hengchen.[11]

The discussion of CO2 usage during the development of the neural network is paramount as not only it presents the possibility to shed light on how much the experiments conducted in the NLP community contribute to the total CO2 production but also paves a way to track, control, and possibly reduce CO2 emissions in the future.

[11] *GitHub*, 2021.
https://github.com/faustusdotbe/CO2_GU_mltgpu/blob/main/mltgpu_co2.ipynb

References

Adelnia, A., & Dastjerdi, H. V. (2011). Translation of idioms: a hard task for the translator. *Theory and Practice in Language Studies*, *1*(7). https://doi.org/10.4304/tpls.1.7.879-883

Alangari, M. A., Jaworska, S., & Laws, J. (2020). Who's afraid of phrasal verbs? The use of phrasal verbs in expert academic writing in the discipline of linguistics. *Journal of English for Academic Purposes*, *43*, 100814. https://doi.org/10.1016/j.jeap.2019.100814

Alberts, I., Mercolli, L., Pyka, T., Prenosil, G., Shi, K., Rominger, A., & Afshar‐Oromieh, A. (2023). Large language models (LLM) and ChatGPT: what will the impact on nuclear medicine be? *European Journal of Nuclear Medicine and Molecular Imaging*, *50*(6), 1549–1552. https://doi.org/10.1007/s00259-023-06172-w

Amin, M., Fankhauser, P., Kupietz, M., & Schneider, R. (2021). Data-driven Identification of Idioms in Song Lyrics. *Proceedings of the 17th Workshop on Multiword Expressions*. https://doi.org/10.18653/v1/2021.mwe-1.3

Anastasiou, D. (2010). *Idiom Treatment Experiments in Machine Translation*. http://scidok.sulb.uni-saarland.de/volltexte/2010/3381/pdf/Diss_FINAL.pdf

Baron, F. (2007). *Identifying non-compositional idioms in text using WordNet synsets* [MA Thesis, University of Toronto]. https://central.bac-lac.gc.ca/.item?id=MR40107&op=pdf&app=Library&oclc_number=653384469

Biggio, B., Nelson, B. D., & Laskov, P. (2012). Poisoning Attacks against Support Vector Machines. *arXiv (Cornell University)*. https://arxiv.org/pdf/1206.6389

Birke, J., & Sarkar, A. (2005). A Clustering Approach for Nearly Unsupervised Recognition of Nonliteral Language. *Aclanthology*. 11th Conference of the European Chapter of the Association for Computational Linguistics. https://www.aclweb.org/anthology/E06-1042.pdf

Biswas, K., Kumar, S., Banerjee, S., & Pandey, A. K. (2020). TanhSoft - a family of activation functions combining Tanh and Softplus. *arXiv (Cornell University)*. http://export.arxiv.org/pdf/2009.03863

Bonnell, J. A. (2011). *Implementation of a New Sigmoid Function in Backpropagation Neural Networks.* [East Tennessee State University]. https://dc.etsu.edu/cgi/viewcontent.cgi?article=2533&context=etd

Boyarskaya, E. L. (2019). Ambiguity matters in linguistics and translation. *Слово.Ру: Балтийский Акцент [Slovo.ru: Baltic Accent]*, *10*(3), 81–93. https://doi.org/10.5922/2225-5346-2019-3-6

Briskilal, J., Praneeth, C., Chaitanya, C. V. V., Karthik, M., & Reddy, P. C. (2023). An Ensemble Method to Classify Telugu Idiomatic Sentences using Deep Learning Models. *International Conference on Inventive Computational Technologies (ICICT 2023)*. https://doi.org/10.1109/icict57646.2023.10134038

Briskilal, J., & Subalalitha, C. N. (2021). Classification of Idioms and Literals Using Support Vector Machine and Naïve Bayes Classifier. In *Machine Vision and Augmented Intelligence – Theory and Applications* (Vol. 796, pp. 515–524). Springer Singapore. https://doi.org/10.1007/978-981-16-5078-9_42

Brown, P. F., deSouza, P., Mercer, R., Della Pietra, V. J., & Lai, J. C. (1992). Class-based n -gram models of natural language. *Computational Linguistics*, *18*(4), 467–479. https://doi.org/10.5555/176313.176316

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. *arXiv (Cornell University)*. https://arxiv.org/pdf/2005.14165.pdf

Cacciari, C., & Tabossi, P. (2014). *Idioms: Processing, structure, and interpretation*. http://ci.nii.ac.jp/ncid/BA21489707

Chang, M., Canseco, J. A., Nicholson, K., Patel, N. N., & Vaccaro, A. R. (2020). The role of machine learning in spine Surgery: The future is now. *Frontiers in Surgery*, *7*. https://doi.org/10.3389/fsurg.2020.00054

Cloudflare. (n.d.). *What is a Large Language model (LLM)?* Retrieved March 27, 2024, from https://www.cloudflare.com/learning/ai/what-is-large-language-model/#

Colombo, L. (1993). The comprehension of ambiguous idioms in context. *Idioms: Processing, Structure, and Interpretation*, 163–200. https://www.research.unipd.it/handle/11577/159158

Cook, P., Fazly, A., & Setevenson, S. (Eds.). (2007). *Pulling their weight: Exploiting syntactic forms for the automatic identification of idiomatic expressions in context.* Proceedings of the workshop on a broader perspective on multiword expressions.

Deng, Z., Liu, J., Luo, B., Yuan, C., Yang, Q., Xiao, L., Zhou, W., & Liu, Z. (2023). AutoPCF: Efficient Product Carbon Footprint Accounting with Large Language Models. *arXiv:2308.04241*. https://arxiv.org/pdf/2308.04241

Endalie, D., Haile, G., & Taye, W. (2023). Deep learning-based idiomatic expression recognition for the Amharic language. *PLOS ONE*, *18*(12), e0295339. https://doi.org/10.1371/journal.pone.0295339

FAQ about Google Trends data - Trends Help. (2024). https://support.google.com/trends/answer/4365533?hl=en#:~:text=Search%20results%20are%20normalized%20to,would%20always%20be%20ranked%20highest.

Fazly, A., Cook, P. F., & Stevenson, S. (2009). Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics*, *35*(1), 61–103. https://doi.org/10.1162/coli.08-010-r1-07-048

Feldman, A., & Peng, J. (2013). Automatic detection of idiomatic clauses. In *Lecture Notes in Computer Science* (pp. 435–446). https://doi.org/10.1007/978-3-642-37247-6_35

Gamage, G., De Silva, D., Adikari, A., & Alahakoon, D. (2022). A BERT-based Idiom Detection Model. *2022 15th International Conference on Human System Interaction (HSI)*. https://doi.org/10.1109/hsi55341.2022.9869485

Gantar, P., Colman, L., Escartín, C. P., & Alonso, H. M. (2018). Multiword Expressions: between lexicography and NLP. *International Journal of Lexicography*, *32*(2), 138–162. https://doi.org/10.1093/ijl/ecy012

Gishkheteliani, I. (2013). Idioms in cross-cultural communication. In *Research on Phraseology Across Continents 2* (2nd ed., pp. 19–36). https://d1wqtxts1xzle7.cloudfront.net/46725000/Dialog-2-libre.pdf?1466665974=&response-content-disposition=inline%3B+filename%3DIntercontinental_Dialogue_on_Phraseology.pdf&Expires=1715084824&Signature=Fw5M-2caV6F2hfERk0vK2GvA-k0fHGHcdkVrfCEjKOvpbttUehMH63IZHRiP3QBWLvIsxvkbHwjYXImRwglX0yR7KRvePLhP1HX4BEHXpguCUeVfs0NduSd3M~XVLA4r1kN6QaCUmfaHEezJDFp8lN9bDGgV107fkLgrAh~WwD7lu~C1BQpQE6V3~Oi0WvlDtuZv9pT~h9r5Gity2ibjiwwhS4M5UamASo7607jNNiSYSjSlS3fJsfXJZH7omFIxJOhvnmnRDukDGxBRx090l5TjaUCdGwo29o0d3wwiyMo4CjG6ZtNiYmwqDKCgcGV30V5SsJRSHCJsOLaQaPzSUw__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA#page=19

Google. (2024, January 9). *Gemini models*. Google AI for Developers. Retrieved February 6, 2024, from https://ai.google.dev/models/gemini

Google Cloud. (2024, February 3). *Strengths and limitations*. Retrieved February 6, 2024, from https://cloud.google.com/vertex-ai/docs/generative-ai/multimodal/strengths-limits?hl=en

Grant, L. (2004). Criteria for Re-defining Idioms: Are we Barking up the Wrong Tree? *Applied Linguistics*, *25*(1), 38–61. https://doi.org/10.1093/applin/25.1.38

Graves, A., Fernández, S., & Schmidhuber, J. (2005). Bidirectional LSTM networks for improved phoneme classification and recognition. In *Lecture Notes in Computer Science* (pp. 799–804). https://doi.org/10.1007/11550907_126

Hale, S., & Campbell, S. J. (2002). The interaction between text difficulty and translation accuracy. *Babel*, *48*(1), 14–33. https://doi.org/10.1075/babel.48.1.02hal

Hashimoto, C., & Kawahara, D. (2008). Construction of an idiom corpus and its application to idiom identification based on WSD incorporating idiom-specific features. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.3115/1613715.1613844

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hsiao, S. (2024, February 8). Bard becomes Gemini: Try Ultra 1.0 and a new mobile app today. *Google*. https://blog.google/products/gemini/bard-gemini-advanced-app/

Hu, Y., Huber, A., Anumula, J., & Liu, S. (2018). Overcoming the vanishing gradient problem in plain recurrent networks. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1801.06105

Huang, S. H., Papernot, N., Goodfellow, I. J., Duan, Y., & Abbeel, P. (2017). Adversarial attacks on neural network policies. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1702.02284

IBM. (n.d.). *What are Recurrent Neural Networks?* Retrieved May 9, 2024, from https://www.ibm.com/topics/recurrent-neural-networks

IBM documentation. (2024, March 21). Retrieved March 27, 2024, from https://www.ibm.com/docs/en/watsonx-as-a-service?topic=atlas-evasion-attack

Idiom, N. (2023, September). Oxford English Dictionary. Retrieved January 18, 2024, from https://www.oed.com/dictionary/idiom_n?tab=meaning_and_use#909611

Imran, M. M., Chatterjee, P., & Damevski, K. (2023). Shedding light on software engineering-specific metaphors and idioms. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2312.10297

Jouppi, N., & Patterson, D. (2023, April 6). *https://cloud.google.com/blog/topics/systems/tpu-v4-enables-performance-energy-and-co2e-efficiency-gains*. Google Cloud. Retrieved February 6, 2024, from https://cloud.google.com/blog/topics/systems/tpu-v4-enables-performance-energy-and-co2e-efficiency-gains

Kaddour, J., Harris, J. S., Mozes, M., Bradley, H., Raileanu, R., & McHardy, R. (2023). Challenges and applications of large language models. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2307.10169

Kanagavalli, V. R., & Raja, K. (2013). Detecting and resolving spatial ambiguity in text using named entity extraction and self learning fuzzy logic techniques. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1303.0445

Katz, G., & Giesbrecht, E. (2006). Automatic identification of non-compositional multi-word expressions using latent semantic analysis. *In Proceedings of the Workshop on Multiword Expressions: Identifying and Exploring Underlying Properties*. https://doi.org/10.3115/1613692.1613696

Keren, G., & Schuller, B. (2016). Convolutional RNN: An enhanced model for extracting features from sequential data. In *International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/ijcnn.2016.7727636

Khanna, T. (2021). *Rule-based pre-processing of idioms and non-compositional constructions to simplify them and improve black-box machine translation.* [MA Thesis, International Institute of Information Technology]. https://d1wqtxts1xzle7.cloudfront.net/101522430/ForUpload-libre.pdf?1682520815=&response-content-disposition=inline%3B+filename%3DRule_based_pre_processing_of_idioms_and.pdf&Expires=1706525444&Signature=WI7Lnx~8~WbBT~OvNFteTaGQj7QF2kZ-lGPktvTExIUvx43WafByq4YeM3~Tx3BCCNrcNDzdTdJHydLWw5Gzyu6tUMp4Ohjq1hQb8VGYqbNwmfT213T4CAWqxUI7PHZSxa4AtdwBHU39N4MaO4ftsZYZU2TQYMv8WHnQzxd1oDkX9iFrpIOGlsOgnSJGpZ0uCjNGSMeprLziYd4B7IrKqtLfRsaLUqaANsP0lZsrKjGRL6GTb-YfWt2KE7xR76mxke-l9YGi-Wh5wJ9lKE4KJo7y7H3BOFsWFaru8LRG4beyB0aF528n7uSoRjgHJih~wA0rurbgu18JnptbNDa8vQ__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

Klein, S., & Simmons, R. F. (1963). A computational approach to grammatical coding of English Words. *Journal of the ACM*, *10*(3), 334–347. https://doi.org/10.1145/321172.321180

Kurniasy, D., & Sonia, E. (2020). AN IDIOMATIC EXPRESSION ANALYSIS ON AN AUTHENTIC MATERIAL "PRIDE AND PREJUDICE MOVIE" a MOVIE FROM JANE AUSTEN BOOK. *JL3T (Journal of Linguistics, Literature and Language Teaching)*, *6*(1), 55–65. https://doi.org/10.32505/jl3t.v6i1.1883

Li, S., Chen, J., Yuan, S., Wu, X., Hao, Y., Tao, S., & Xiao, Y. (2024). Translate Meanings, Not Just Words: IdiomKB's Role in Optimizing Idiomatic Translation with Language Models. *Proceedings of the . . . AAAI Conference on Artificial Intelligence*, *38*(17), 18554–18563. https://doi.org/10.1609/aaai.v38i17.29817

Li, S., Chen, J., Yuan, S., Wu, X., Yang, H., Tao, S., & Xiao, Y. (2023). Translate Meanings, Not Just Words: IdiomKB's Role in Optimizing Idiomatic Translation with Language Models. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2308.13961

Liontas, J. I. (2017). Why teach idioms? a challenge to the profession. *Iranian Journal of Language Teaching Research*, *5*(3), 5–25. https://doi.org/10.30466/ijltr.2017.20302

Liu, C. C., Koto, F., Baldwin, T., & Gurevych, I. (2023). Are Multilingual LLMs Culturally-Diverse Reasoners? An Investigation into Multicultural Proverbs and Sayings. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2309.08591

Liu, G., & Guo, J. (2019). Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing*, *337*, 325–338. https://doi.org/10.1016/j.neucom.2019.01.078

Loftsson, H. (2008). Tagging Icelandic text: A linguistic rule-based approach. *Nordic Journal of Linguistics*, *31*(1), 47–72. https://doi.org/10.1017/s0332586508001820

M. Espinal, T., & Jaume Mateu, M. (2019). Idioms and Phraseology. *Oxford Research Encyclopedia of Linguistics*. https://oxfordre.com/linguistics/display/10.1093/acrefore/9780199384655.001.0001/acrefore-9780199384655-e-51

Makal, J. (2017). *A comparative study of British, American and Australian idioms expressing emotions* [Diploma Thesis, Masaryk University]. https://is.muni.cz/th/ct4cu/Diploma_Thesis_Makal_385473.pdf

Manning, M., Wong, G. T. W., Graham, T., Ranbaduge, T., Christen, P., Taylor, K., Wortley, R., Makkai, T., & Skorich, P. (2018). Towards a 'smart' cost–benefit tool: using machine learning to predict the costs of criminal justice policy interventions. *Crime Science*, *7*(1). https://doi.org/10.1186/s40163-018-0086-4

McCloskey, M. A. (1964). VI.—METAPHORS. *Mind*, *LXXIII*(290), 215–233. https://doi.org/10.1093/mind/lxxiii.290.215

McIntosh, T. R., Sušnjak, T., Liu, T., Watters, P. A., & Halgamuge, M. N. (2023). From Google Gemini to OpenAI Q* (Q-Star): a survey of reshaping the generative Artificial Intelligence (AI) research landscape. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2312.10868

Meskó, B. (2023). Prompt engineering as an important emerging skill for medical professionals: tutorial. *Journal of Medical Internet Research*, *25*, e50638. https://doi.org/10.2196/50638

METAPHORS. (1964). *Mind*, *LXXIII*(290), 215–233. https://academic.oup.com/mind/article-abstract/LXXIII/290/215/948038?redirectedFrom=fulltext

Milmo, D. (Ed.). (2023, December 6). Google says new AI model Gemini outperforms ChatGPT in most tests. *The Guardian*. Retrieved February 6, 2024, from https://www.theguardian.com/technology/2023/dec/06/google-new-ai-model-gemini-bard-upgrade

Morrison, R. (2024, February 5). Google may be rolling out Gemini Ultra this week and renaming Bard at the same time. *Tom's Guide*. Retrieved February 6, 2024, from https://www.tomsguide.com/ai/google-may-be-rolling-out-gemini-ultra-this-week-and-renaming-bard-at-the-same-time

Muzny, G., & Zettlemoyer, L. (2013). Automatic Idiom Identification in Wiktionary. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. https://aclanthology.org/D13-1145/

Nammous, M. K., & Saeed, K. (2019). Natural Language Processing: Speaker, Language, and Gender Identification with LSTM. In *Advances in intelligent systems and computing* (pp. 143–156). https://doi.org/10.1007/978-981-13-3702-4_9

Nichols, J. A., Chan, H. W. H., & Baker, M. a. B. (2018). Machine learning: applications of artificial intelligence to imaging and diagnosis. *Biophysical Reviews*, *11*(1), 111–118. https://doi.org/10.1007/s12551-018-0449-9

Nicholson, K., Collins, G. S., Waterman, B. R., & Bullock, G. S. (2021). Machine Learning and Statistical Prediction of Pitching Arm Kinetics. *The American Journal of Sports Medicine*, *50*(1), 238–247. https://doi.org/10.1177/03635465211054506

Pascanu, R., Gülçehre, Ç., Cho, K., & Bengio, Y. (2014). How to Construct Deep Recurrent Neural Networks. *arXiv:1312.6026v*. http://lib-arxiv-008.serverfarm.cornell.edu/pdf/1312.6026

Patterson, D. S., Gonzalez, J. E., Hölzle, U., Le, Q. V., Chen, L., Munguía, L., Rothchild, D., So, D. R., Texier, M., & Dean, J. (2022). The carbon footprint of machine learning training will plateau, then shrink. *Computer*, *55*(7), 18–28. https://doi.org/10.1109/mc.2022.3148714

Peng, J., & Feldman, A. (2017). Automatic Idiom Recognition with Word Embeddings. In *Communications in computer and information science* (pp. 17–29). https://doi.org/10.1007/978-3-319-55209-5_2

Pere, J. (1079). Ambiguity in linguistic theory. In *Metalogicon* (23rd ed., Vol. 2, pp. 65–102). http://web.mclink.it/MI2701/rivista/2010ld/Julia2010ld.pdf

Pichai, S., & Hassabls, D. (2023, December 6). *Introducing Gemini: our largest and most capable AI model*. Google The Keyword. Retrieved February 6, 2024, from https://blog.google/technology/ai/google-gemini-ai/#sundar-note

Plisson, J., Lavrač, N., & Mladenić, D. (2004). A Rule based Approach to Word Lemmatization. *IS*, *3*. http://ailab.ijs.si/dunja/SiKDD2004/Papers/Pillson-Lematization.pdf

Pokharel, R., & Agrawal, A. (2023). Generating continuations in multilingual idiomatic contexts. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2310.20195

Ptiček, M., & Dobša, J. (2023). Methods of annotating and identifying metaphors in the field of natural language processing. *Future Internet*, *15*(6), 201. https://doi.org/10.3390/fi15060201

REPSOL. (2023, September 11). *What is energy consumption and why is it important? | Repsol*. Retrieved May 7, 2024, from https://www.repsol.com/en/energy-and-the-future/future-of-the-world/energy-consumption/index.cshtml

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., & Flickinger, D. (2002). Multiword Expressions: a pain in the neck for NLP. In *Lecture Notes in Computer Science* (pp. 1–15). https://doi.org/10.1007/3-540-45715-1_1

Sennet, A. (2023). Ambiguity. In E. N. Zalta & U. Nodelman (Eds.), *The Stanford Encyclopedia of Philosophy* (Summer 2023). Metaphysics Research Lab, Stanford University. https://plato.stanford.edu/archives/sum2023/entries/ambiguity/

Shirin, A., & Raseek, C. (2018). Replacing idioms based on their figurative usage. *2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR)*. https://doi.org/10.1109/icetietr.2018.8529042

Shojaei, A. (2012). Translation of Idioms and Fixed Expressions: Strategies and Difficulties. *Theory and Practice in Language Studies*, *2*(6). https://doi.org/10.4304/tpls.2.6.1220-1229

Shuang, L., Jiangjie, C., Siyu, Y., Xinyi, W., Hao, Y., Shimin, T., & Yanghua, X. (2023). Translate Meanings, Not Words: IdiomKB's Role in Optimizing Idiomatic Translation with Language Models. *arXiv Preprint arXiv:2308.13961*. https://arxiv.org/pdf/2308.13961.pdf

Škobo, M., & Petričević, V. (2023). Navigating the challenges and opportunities of literary translation in the age of AI: striking a balance between human expertise and machine power. *Društvene I Humanističke Studije*, *8*(2(23)), 317–336. https://doi.org/10.51558/2490-3647.2023.8.2.317

Škvorc, T., Gantar, P., & Robnik–Šikonja, M. (2021). MICE: Mining Idioms with Contextual Embeddings. *Knowledge-Based Systems*, *235*, 107606. https://doi.org/10.1016/j.knosys.2021.107606

Sulaymonovna, T. I. (2023). PROVERBS AND SAYINGS AS COMMUNICATIVE PHRASEOLOGICAL UNITS. *Новости Образования: Исследование В XXI Веке [Education News: Research in the 21st Century]*, *2*(16). http://nauchniyimpuls.ru/index.php/noiv/article/download/13252/9221

Sun, H. (2023). Reinforcement Learning in the Era of LLMs: What is Essential? What is needed? An RL Perspective on RLHF, Prompting, and Beyond. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2310.06147

Tahayna, B., & Ayyasamy, R. K. (2023). Applying English idiomatic expressions to classify deep sentiments in COVID-19 tweets. *Computer Systems Science and Engineering*, *47*(1), 37–54. https://doi.org/10.32604/csse.2023.036648

Team, G. H. C., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Petrov, S., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., . . . Pillai, T. S. (2023). Gemini: a family of highly capable multimodal models. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2312.11805

Tedeschi, S., Martelli, F., & Navigli, R. (2022). ID10M: Idiom Identification in 10 languages. *Findings of the Association for Computational Linguistics: NAACL 2022*. https://doi.org/10.18653/v1/2022.findings-naacl.208

Tedeschi, S., & Navigli, R. (2022). NER4ID at SEMEVAL-2022 Task 2: Named Entity Recognition for Idiomaticity Detection. *Proceedings of the 16th International*

Workshop on Semantic Evaluation (SemEval-2022). https://doi.org/10.18653/v1/2022.semeval-1.25

Tien-Ping, T. E. (2021). Translating IdiomsusingParaphrasing, machine translation and rescoring. *Türk Bilgisayar Ve Matematik Eğitimi Dergisi*, *12*(3), 1942–1946. https://doi.org/10.17762/turcomat.v12i3.1027

Tuor, A., Kaplan, S., Hutchinson, B., Nichols, N., & Robinson, S. M. (2017). Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. *arXiv (Cornell University)*. https://arxiv.org/pdf/1710.00811.pdf

Tyasrinestu, P., & Ardi, P. (2020). Idiomatic expressions and their Indonesian subtitles in the good doctor TV series. *A Journal on Language and Language Learning*, *23*(1), 37–57. https://doi.org/10.24071/llt.v23i1.2360.g1784

Verma, R., & Vuppuluri, V. (2015). A New Approach for Idiom Identification Using Meanings and the Web. In *Proceedings of Recent Advances in Natural Language Processing*. https://www.aclweb.org/anthology/R15-1087.pdf

Vespignani, F., Canal, P., Molinaro, N., Fonda, S., & Cacciari, C. (2010). Predictive mechanisms in idiom comprehension. *Journal of Cognitive Neuroscience*, *22*(8), 1682–1700. https://doi.org/10.1162/jocn.2009.21293

Wang, S., & Jiang, J. (2016). Learning Natural Language Inference with LSTM. *arXiv:1512.08849v2*. https://doi.org/10.18653/v1/n16-1170

What is a Transformer Model? | IBM. (n.d.). https://www.ibm.com/topics/transformer-model

Ye, Q., Axmed, M., Pryzant, R., & Khan, F. (2023). Prompt Engineering a Prompt Engineer. *arXiv Preprint arXiv:2311.05661*. https://arxiv.org/pdf/2311.05661.pdf

Zeng, Z., & Bhat, S. (2021). Idiomatic Expression Identification using Semantic Compatibility. *Transactions of the Association for Computational Linguistics*, *9*, 1546–1562. https://doi.org/10.1162/tacl_a_00442

Appendices

Appendix A: Separate & Cross-Validation Results

GPT-4: Individual & Cross-Validation Results (Testing)						
Experiment Nr	**Precision**	**Precision (CV)**	**F0.5**	**F0.5 (CV)**	**Recall**	**Recall (CV)**
Experiment 1	82.50%	82.50%	83.70%	83.60%	88.90%	88.80%
Experiment 2	84.80%	81.60%	85.49%	83.06%	88.30%	89.50%
Experiment 3	87.30%	88.70%	86.59%	87.95%	83.90%	85.10%
Experiment 4	82.90%	77.06%	84.03%	79.30%	88.90%	90.20%
Experiment 5	82.30%	82.20%	83.40%	83.50%	88.30%	89.60%
Experiment (mean)	83.96%	82.41%	84.64%	83.48%	87.66%	88.64%

Table 1: GPT-4: Individual & Cross-Validation Results

Gemini Pro: Individual & Cross-Validation Results (Testing)						
Experiment Nr	Precision	Precision (CV)	F0.5	F0.5 (CV)	Recall	Recall (CV)
Experiment 1	91.71%	80.92%	88.98%	82.48%	79.50%	89.39%
Experiment 2	95.97%	81.82%	91.53%	82.82%	79.65%	87.14%
Experiment 3	84.94%	80.98%	84.02%	81.43%	80.54%	83.30%
Experiment 4	85.94%	83.33%	85.02%	82.63%	81.54%	79.94%
Experiment 5	89.59%	81.50%	87.53%	83.89%	80.15%	95.03%
Experiment (mean)	89.63%	81.71%	87.41%	82.65%	80.27%	86.96%

Table 2: Gemini Pro: Individual & Cross-Validation Results

Gemini Pro: Individual Results (Development)			
Experiment Nr	Precision	F0.5	Recall
Experiment 1	84.10%	84.25%	84.84%
Experiment 2	83.24%	83.06%	82.35%
Experiment 3	80.28%	80.73%	82.60%
Experiment 4	87.63%	86.52%	82.32%
Experiment 5	71.25%	73.07%	81.42%
Experiment (Mean)	81.30%	81.53%	82.71%

Table 3: Gemini Pro: Individual Results (Development)

Appendix B: GPT-4: Prompt and Settings

Experiment	Prompt	Temperature
Experiment 1	Classify the sentence and answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following: Phrasal verbs, Idioms, Ambiguity: a sentence that can be understood literally and figuratively, Metaphors, Sayings. Work out your own solution on idiomatic expression first by reading the sentences. Then reason with your solution. Response in the following JSON format: {"idiomatic": }. Do not include ``` .	0.1
Experiment 2	I give you sentences, and I would ask you to please classify them. I will give you instructions on how to do so. Please classify the sentence and answer with 1 for idiomatic and 0 for not idiomatic. A sentence can be idiomatic if it contains one of the following: Phrasal verbs, Idioms, Ambiguity: a sentence that can be understood literally and figuratively, Metaphors, Sayings, Work out your own solution on idiomatic expression first by reading the sentence. Then reason with your solution. Answer in the following JSON format: {"idiomatic": }. Do not include ``` .	0.1
Experiment 3	Please classify the sentence and answer with 1 for idiomatic and 0 for not idiomatic. A sentence can be idiomatic if it contains one of the following: Phrasal verbs, Idioms, Ambiguity: a sentence that can be understood literally and figuratively, Metaphors, Sayings, Work out your own solution on idiomatic expression first by reading the sentence. Then reason with your solution. Answer in the following JSON format: {"idiomatic": }. Do not include ``` .	0.8
Experiment 4	You are a professional translator from English. Your job is to classify a sentence as either idiomatic or not. Answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following. Phrasal verbs. Idioms.	1

	Ambiguity: a sentence that can be understood literally and figuratively. Metaphors. Sayings. Develop your own solution and reason with it. Answer only with 1 for idiomatic and 0 for not idiomatic when you see idiomatic expression. Answer in the following JSON format: {"idiomatic": }. Do not include ``` .	
Experiment 5	You are an English language and literature specialist. Your job is to classify a sentence as either idiomatic or not. Answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following. Phrasal verbs: example: put down (idiomatic: to kill; literal: to put something down). Idioms: example: (A dog's breakfast: something that is disorganized). Ambiguity: a sentence that can be understood literally and figuratively: example: (A good life depends on a liver: liver can be considered either to be an organ or a person). Metaphors: example: (curtain of the night: metaphor that expresses the way the night came at that area). Sayings: example: (Don't count your chickens before they're hatched). Answer in the following JSON format: {"idiomatic": }. Don't include the prompt in the answer. Do not include ``` .	1

Appendix C: Gemini Pro: Prompts and Settings

Experiment	Prompt	Temperature
Experiment 1	You are an English language and literature specialist. Your job is to classify a sentence as either idiomatic or not. Answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following. Phrasal verbs. Idioms. Ambiguity: a sentence that can be understood literally and figuratively. Metaphors. Sayings. Answer in the JSON format that looks like this: {"prediction": }. Do not include ``` .	0.5
Experiment 2	You are an English language and literature specialist. Your job is to classify a sentence as either idiomatic or not. Answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following. Phrasal verbs. Idioms. Ambiguity: a sentence that can be understood literally and figuratively. Metaphors. Sayings. Answer in the JSON format that looks like this: {"prediction": }. Do not include ``` .	0.2
Experiment 3	You are an English language and literature specialist. Your job is to classify a sentence as either idiomatic or not. Answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following: Phrasal verbs (e.g., kick the bucket, spill the beans). Idioms (e.g., raining cats and dogs, see eye to eye). Lexical ambiguity with figurative meaning: A sentence with a word that has multiple meanings, and one of those meanings contributes to a non-literal interpretation (e.g., "He spilled the beans"). Syntactic or semantic ambiguity with figurative meaning: A sentence with an ambiguous structure or meaning that leads to a non-literal interpretation (e.g., "She's seeing someone"). Metaphors (e.g., life is a journey). Sayings (e.g., a stitch in time saves nine). Important: Sentences with other types of ambiguity, such as spatial ambiguity ("The box is on the table, next to the lamp"), are not considered idiomatic for this task. Answer in the JSON format for each sentence that looks like this:	0.4

	{"prediction": }. Do not include ``` . Do not include the prompt in the output. examples: but disease brought by european settlers took its toll on tradition. output: {"prediction": 1} examples: -he didn't give her, her space. -he’s gonna go annoy her. output: {"prediction": 0} examples: and i was like, 'is she fucking with us right now?' output: {"prediction": 1} examples: 0.0lb i'm straight up terrified to see the results right now. output: {"prediction": 1} examples: he detects that she's close to estrous and wherever she goes, he follows. output: {"prediction": 0}	
Experiment 4	You are a lexicographer that works with the English language. Your task is to classify a sentence whether it contains idiomatic expressions or not. A sentence can be considered to contain idiomatic expression if it has one of the following: phrasal verbs, idioms, ambiguity, metaphors, sayings. When classifying sentences, answer with 1 for yes and 0 for no. Work out your own solution and reason with it. Answer in the JSON format that looks like this: {"prediction": }. Add nothing else to the response. Do not include ``` .	0.9
Experiment 5	You are an English language and literature specialist. Your job is to classify a sentence as either idiomatic or not. Answer with 1 for yes and 0 for no. A sentence can be idiomatic if it contains one of the following. Phrasal verbs. Idioms. Ambiguity: a sentence that can be understood literally and figuratively. Metaphors. Sayings. Answer in the JSON format that looks like this: {"prediction": }. Do not include ``` .	0.3

Appendix D: LSTM-RNN

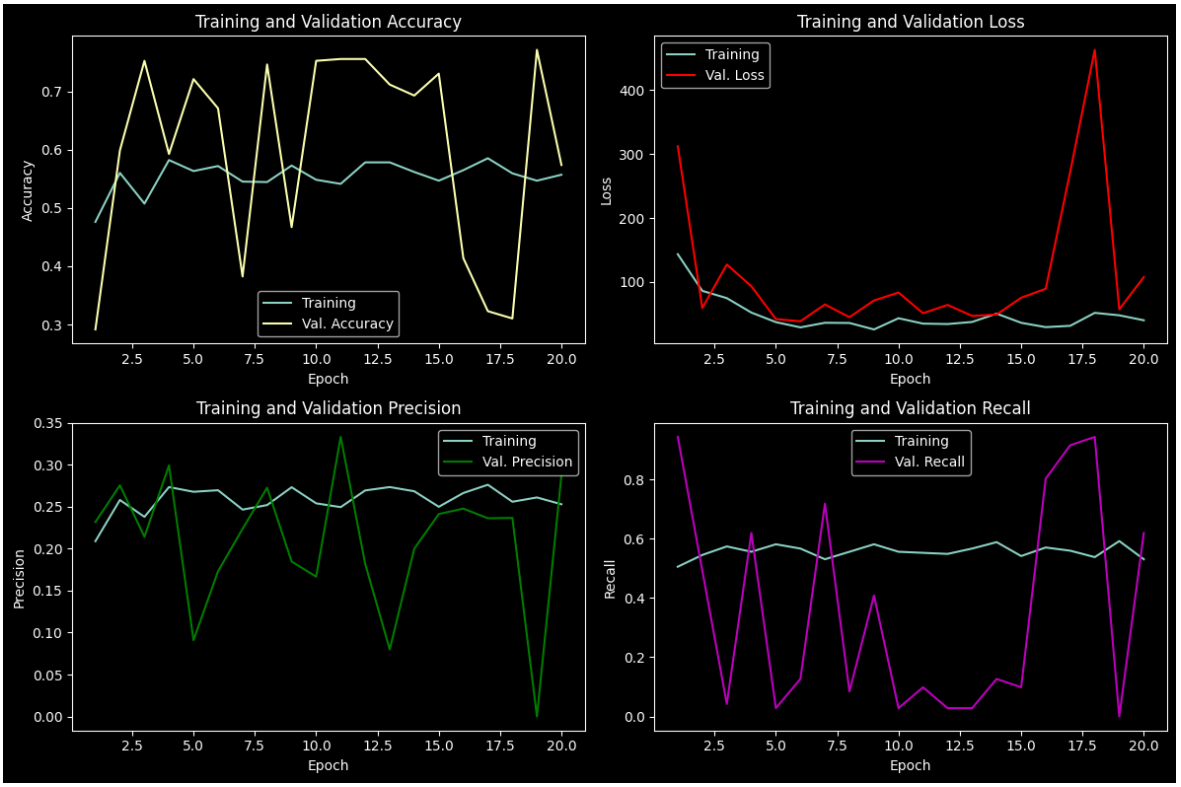


Image 1: LSTM-RNN Training & Validation Results

Appendix E: BiLSTM-RNN Results

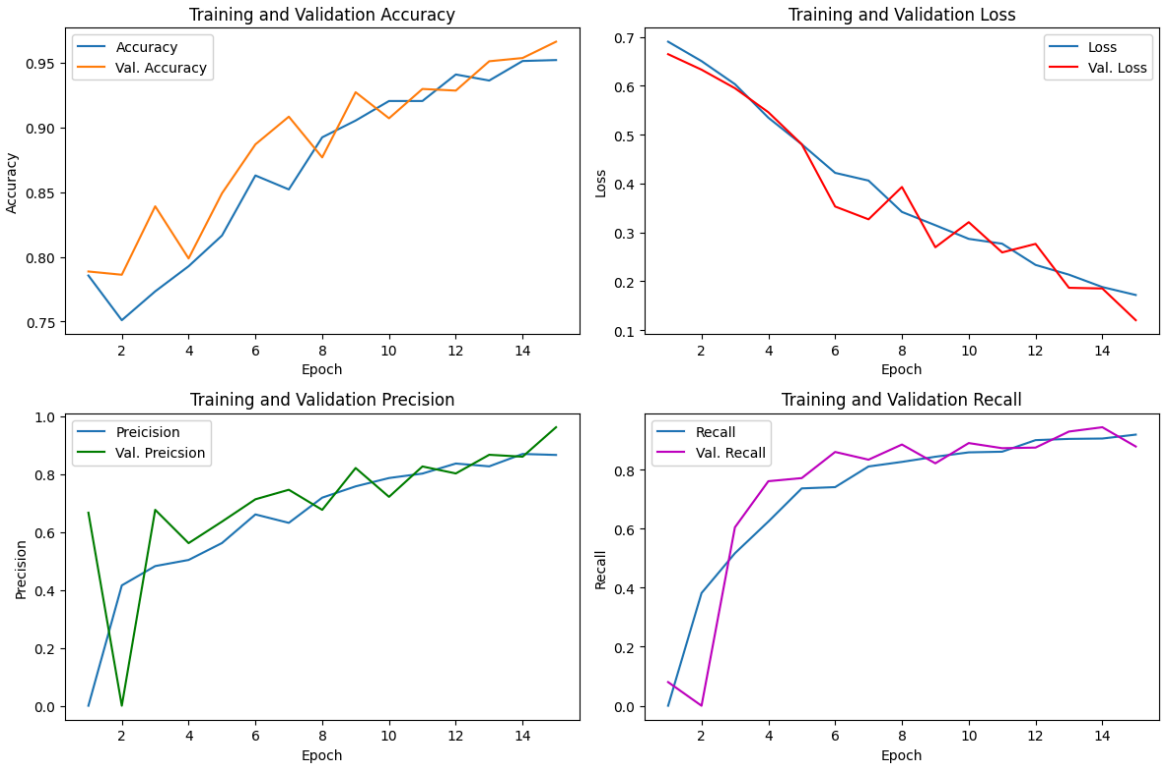


Image 5: BiLSTM-RNN Training & Validation Results