Can a Large Language Model Replace Humans at Rating Lexical Semantic Relations Strength?

André Fernandes dos Santos* and José Paulo Leal

CRACS & INESC Tec LA / Faculty of Sciences
University of Porto, Portugal
afs@inesctec.pt, jpleal@fc.up.pt

This article investigates the ability of large language models (LLMs) to evaluate semantic relations between word pairs by examining their alignment with human-generated semantic ratings. Semantic relations represent the degree of connection (e.g., relatedness or similarity) between linguistic elements and are traditionally validated against human-annotated datasets. Due to the challenges of building such datasets and recent progress in LLMs' capacity to model human-like understanding, we explore whether LLMs can serve as reliable substitutes for traditional human ratings.

We conducted experiments using multiple LLMs from OpenAI, Google, Mistral, and Anthropic, evaluating their performance across diverse English and Portuguese semantic relations datasets. We included in the analysis PAP900, a recently published dataset of semantic relations in Portuguese, to examine the influence of prior exposure to the dataset on LLM training. The results show that the LLM predictions correlate strongly with human ratings. The findings reveal the potential of LLMs to supplement or replace traditional semantic measure algorithms and crowd-sourced human annotations in semantic tasks.

1. Introduction

Semantic measures (SMs) are algorithms that allow computers to mimic human ability to assess the strength of semantic relations (SRs) between units of language. Typically, these algorithms receive as input the pair of linguistic elements to rate and output their SR assessments as numeric values.

The evaluation of SM algorithms can be performed by comparing their output against values from gold standard SR datasets. These datasets are composed of pairs of elements, each matched with a numeric rating averaged from the individual responses of human annotators. The number of pairs annotated and the number of annotators for each pair are frequently limited by the difficulty of enlisting human annotators.

* Corresponding author.

Action Editor: Michael Roth. Submission received: 16 December 2024; revised version received: 30 October 2025; accepted for publication: 23 November 2025.

https://doi.org/10.1162/COLI.a.587

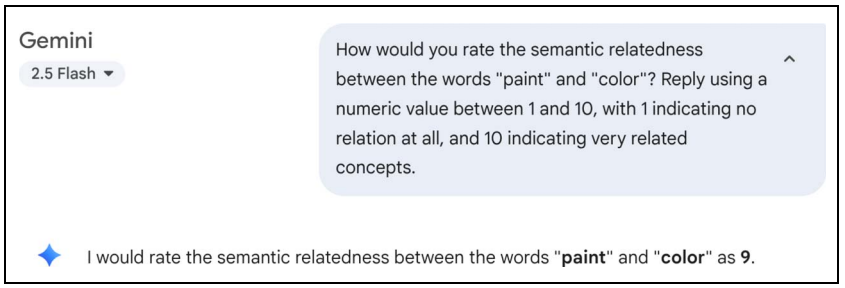

Figure 1
Gemini 2.5 Flash prompt and truncated response for the semantic relatedness rating for *paint* and *color*.

Platforms like Amazon Mechanical Turk are frequently utilized as a convenient source for human labor.

Large language models (LLMs) are computational models designed for tasks such as text generation. LLMs are usually built using artificial neural networks and trained on large corpora. These models have recently been the focus of much attention due to their ability to perform with high accuracy tasks that previously required human intervention.

One task where LLMs have shown promising potential is in predicting human ratings of SR strength. Anecdotal evidence suggests that LLMs may indeed be capable of accurately evaluating semantic relationships. For example, Figure 1 presents a query asking Google's Gemini (Google AI for Developers 2024a) to rate the semantic similarity between *paint* and *color*.[1] Anthropic's Claude (Anthropic 2024a) rated the semantic similarity between *car* and *truck* with an 8/10, and the semantic relatedness between *dog* and *leash* as 9/10.[2]

Although some evaluations for this kind of task have been described in the literature, they remain infrequent and focused only on a small number of LLMs and datasets. If LLMs' assessments of semantic relations prove to be strongly correlated with ratings attributed by humans, then LLMs could be used as an alternative to traditional corpus-based or knowledge-based SMs (Harispe et al. 2022), or even replace human annotators in the construction of SR gold standards.

In an earlier article (dos Santos and Leal 2024), we shared initial findings on LLMs in relation to lexical SR prediction. We focused on evaluating how LLM assessments of the MC30 dataset (Miller and Charles 1991) align with multiple instances of dataset annotations and examined the impact of dataset exposure on LLM outcomes. We also tested how different prompt verbosity levels affected LLM ratings.

The current study expands on that work by conducting tests with numerous LLMs from various providers, incorporating a wide range of English and Portuguese lexical SR datasets. The dataset collection includes a newly developed SR dataset that was unpublished at the time of analysis. We contrast the results obtained with existing SM evaluations and with datasets' reported inter-annotator agreement values.

1 Query performed using the Web interface at `https://gemini.google.com/app` with the `gemini-2.5-flash` model.
2 Queries performed using the Web interface at `https://claude.ai` with the `claude-3-haiku-20240307` model.

In this study, we focus on lexical SR datasets (i.e., those containing ratings for pairs of words or multi-word expressions). We tackle the following research questions:

RQ1. **How do LLMs compare against other semantic measure algorithms in assessing semantic relation strength?** When assessing SRs, LLMs may be considered a novel category of corpus-based SM algorithms. Is the correlation between LLM ratings and human ratings stronger than with other SM algorithms?

RQ2. **How closely do LLM evaluations of semantic relation strength align with ratings produced by humans?** Human annotators of SR datasets are not always in agreement with each other. This variability is measured as the inter-annotator agreement (IAA). The correlation of LLM ratings with the average human ratings can be compared with datasets' IAA values.

RQ3. **Are LLMs' evaluations of SRs influenced by exposure to datasets during training?** LLMs are generally trained on extensive collections of Web pages, books, and code, which probably encompass SR datasets. Do LLMs produce varying outcomes with older datasets compared to more recent ones? What is their performance like with unpublished data?

RQ4. **Can SR dataset characteristics influence the performance of LLMs?** SR datasets exhibit a wide range of characteristics, which can include differences in word language, annotated relation types, word part-of-speech categories, and the expertise level of the annotators involved. Which of these differences, if any, might affect the effectiveness of LLMs, leading to higher or lower average correlation values between LLM outputs and datasets' ratings?

This article is structured as follows. In Section 2 we provide more detailed information on LLMs, SR datasets, and the issue of dataset exposure. In Section 3 we list the datasets used for our work, we describe the creation of a new dataset to test LLMs with completely new pairs of words, and we describe the models and providers used in this research. Section 4 describes how we designed the prompt used for LLM evaluation, the evaluation trial procedure, and the whole experimental pipeline. Section 5 presents the results obtained, which are then discussed in Section 6. Section 7 points out conclusions and possible future work.

2. Background

In this section, we provide context and describe related work for large language models and their use in SR assessment; semantic relation datasets, their construction and evaluation processes; and the concern regarding LLM exposure to datasets during training, along with the attempts to detect it and evaluate its effects.

2.1 Large Language Models

Large language models are machine learning algorithms designed to process, understand, and generate both natural and formal language text (Hadi et al. 2024). They are

typically constructed using Transformers, a deep learning architecture that leverages a *self-attention* mechanism. This mechanism enables transformers to handle sequential data entirely in parallel, unlike earlier architectures such as recurrent or convolutional neural networks. This parallel processing approach makes transformers more efficient and allows them to capture long-range dependencies within text.

LLMs are trained on massive corpora, comprising millions of books, Web sites, and other text documents. LLMs have been used for diverse tasks such as language translation, question answering, summarization, and text generation (Hadi et al. 2024).

Several experiments have been carried out to evaluate semantic relations using LLMs. Di Caro et al. (2023) conducted a semantic similarity task using ChatGPT (OpenAI's conversational Web interface for GPT models) using pairs of words sampled from SimLex-999 (Hill, Reichart, and Korhonen 2015), reaching a Pearson correlation value between the model's output and the dataset ratings of 0.604. They also tested ChatGPT's ability to perform semantic relationship extraction, comparing the model's output against the corresponding relationships as defined in WordNet (Fellbaum 2010), reportedly achieving "a perfect accuracy rate". Liu, Melton, and Zhang (2024) used GPT-3.5 to predict semantic similarity and relatedness for pairs from MayoSRS and MiniMayoSRS (Pedersen et al. 2007), obtaining Pearson correlations of around 0.7. Trott (2024) used GPT-4 to collect multiple kinds of semantic judgments, including semantic similarity between pairs of words extracted from SimLex-999 and SimVerb3500 (Gerz et al. 2016). They report positive correlations between GPT-4 and human judgments, in some cases rivaling or exceeding the reported IAA.

De Deyne, Liu, and Frermann (2024) used GPT-4 to infer the types of semantic relations across multiple datasets, achieving strong results for broad semantic categories but encountering limitations with finer-grained distinctions, such as differentiating specific types of taxonomic relations. Musker et al. (2024) demonstrated that GPT-4 (Achiam et al. 2023) and Claude 3 Opus (Anthropic 2024a) perform at near-human levels on analogical reasoning tasks. These findings invite a deeper question: Can LLMs, beyond simply being advanced language tools, legitimately serve as cognitive models? Even more so for SR tasks: Because LLMs are not directly trained to assess semantic relationships, further testing is needed to settle whether their outputs truly capture semantic understanding or merely reflect statistical associations.

The extent to which these LLM evaluations have been affected by data contamination remains uncertain, as discussed in Section 2.3.

Another relevant question concerns how LLM tasks should be described. When composing a prompt (message) asking an LLM to complete a task, there are several details that should be taken into account. For example, the tone used, the level of detail of the instructions, and providing examples of the output wanted. The process of crafting and optimizing messages sent to LLMs with the goal of improving the results obtained has been dubbed **prompt engineering**. Several studies have been published on this topic (Grabb 2023; Knoth et al. 2024), despite some discussion on its validity as a scientific approach, as several of such articles are unpublished and non-peer-reviewed preprints (e.g., Sahoo et al. 2024; Shah 2024), press releases from companies providing commercial LLM-related services, or popular articles based on the former.

2.2 Semantic Relations Datasets

Quantitative semantic relation datasets are collections of mappings between pairs of units of language, such as words (e.g., Miller and Charles 1991; Vulić et al. 2021),

sentences (e.g., Dolan, Quirk, and Brockett 2004; Cer et al. 2017), or named entities (e.g., Pirró 2012; dos Santos and Leal 2023), and numeric values representing the strength of a given type of semantic relation between them (Harispe et al. 2022).

Most often, researchers focus on two main types of semantic relations: semantic similarity, which represents the likeness of items and takes into account only taxonomic relationships; and semantic relatedness, a more broad type of relation that encompasses all types of connections (Harispe et al. 2022). Consider a classic example: Although *dog* and *cat* share similarities (such as both being mammals and common household pets), *dog* and *flea* are connected (the former hosts the latter as a parasite) but are not as similar. The distinction between both relation types is important, as similarity is more useful in tasks such as lexical resource building, semantic parsing, and machine translation, while relatedness is better suited for word sense disambiguation and text classification (Hill, Reichart, and Korhonen 2015).

Lexical SR datasets can be composed of words-in-context, in which words are presented within sentences or annotated with a synset (e.g., SCWS2003 [Huang et al. 2012], SL7576 [Silberer and Lapata 2014]), or out-of-context, in which words are not tied down to one specific meaning or use case. The latter are the focus of this work.

SR datasets are often built by asking human annotators to numerically rate pairs of elements and averaging the values obtained for each pair. These annotators are frequently university students; more recently, platforms for crowd-sourcing, such as Amazon Mechanical Turk, have facilitated the creation of datasets by rewarding online participants with small monetary incentives for performing this task (Boyd-Graber et al. 2006; Radinsky et al. 2011; Halawi et al. 2012). However, this often requires additional verification steps to ensure quality results, such as testing the annotators' level of language comprehension or methods for detecting outliers.

Some existing SR datasets pose some ambiguity regarding the specific type of SR assessed (be it similarity or relatedness). This issue arises from the annotation stage, where the annotators were not clearly informed of the distinction, examples of both types of relationships were not provided, or the instructions were unclear or used the terms interchangeably. Hill, Reichart, and Korhonen (2015) and Banjade et al. (2015) have listed the issues arising from this ambiguity.

Other dataset construction methods have also been tried, with varying degrees of success. In Bruni, Tran, and Baroni (2014) annotators were asked to choose the most related pair among two candidate pairs. In dos Santos and Leal (2023), we adopted a gamification strategy to create a game where players were tasked with identifying, from a trio of entities, the one that was the least connected to the other two. This indirectly leads to the identification of the most connected pair among the three possible pairings.

SR datasets have multiple uses, including word-sense or named entity disambiguation, information retrieval, and semantic measure development and evaluation (AlMousa, Benlamri, and Khoury 2022; Sanderson 1994). Semantic measures are algorithms which allow computers to mimic the human ability to assess the strength of semantic relations. SMs are based on the analysis of information describing elements extracted from semantic sources. These sources can be unstructured (e.g., books, newspaper articles), semi-structured (e.g., dictionaries), or structured (e.g., semantic graphs) (Feng et al. 2017). Depending on the source type, different algorithms can be used. Semantic graphs are used for path-based SM algorithms, which can be based on techniques such as shortest path calculation, random walks, or distance to the least common subsumer (Harispe et al. 2015). Unstructured sources are used for algorithms based on the distributional hypothesis (i.e., words used together or in the same contexts are closer in meaning) (Sahlgren 2008). In more recent years, corpus-based measures

often take advantage of word embeddings (Kulmanov et al. 2021; Chandrasekaran and Mago 2022; Hadj Taieb, Zesch, and Ben Aouicha 2020).

SR datasets are used in SM evaluation by calculating the correlation between values from the datasets and the SM output (Harispe et al. 2015). Correlation is frequently calculated using Pearson r or Spearman's ρ formulas; Cohen's κ and Kendall's σ are occasionally used as well (Boslaugh 2012; Shaqiri et al. 2023). Krippendorff's alpha is a generalization of these and other inter-rater reliability metrics (Krippendorff 2018). Akila and Jayakumar (2014) provide an overview of existing semantic similarity measures. Extensive research has been published describing and evaluating SMs using SR datasets (Slimani 2013; Giabelli et al. 2022; Hussain et al. 2023; Zhang, Gentile, and Ciravegna 2013; Garla and Brandt 2012).

A thorough compilation of English and Portuguese lexical datasets has been assembled and is accessible online.[3] The data originally found in each dataset, typically in CSV format, but sometimes in HTML pages, PDF files, or other formats, has been extracted and restructured into JSON documents. In these documents, each word pair and its rating are represented as an object comprising two terms and a numerical value (e.g., {``term1'': ``boy'', ``term2'': ``lad'', ``value'': 3.84}).

In this repository, each dataset remains distinct, facilitating access to the specific datasets desired. In addition to the word pairs and corresponding numeric values, each dataset in this repository also includes (when available) metadata such as the number of annotators, the reported inter-annotator agreement, individualized ratings from all annotators for each pair, the scale used for rating, relation type classification, and more. Each dataset is also associated with authorship metadata, such as the names of its authors, its publication date, the title of associated publications, and home and download URLs.

2.3 Impact of Dataset Exposure

LLMs are typically trained with data collected from scraping pages, books, and other documents from the Web. Many datasets (SR and others) are available online, often made available through data repositories, attached as supplementary materials to the papers where they are introduced, hosted on the authors' personal or professional Web sites, mirrored on third-party platforms, or sometimes disseminated through multiple of these channels simultaneously.

This may lead to LLMs being unintentionally trained with corpora which include SR datasets (a scenario we refer to as **LLM dataset exposure**), and subsequently evaluated on the same data they were trained on. Researchers often call this "dataset leakage" or "dataset contamination" (Aiyappa et al. 2023), which we avoid in our own work as we find these expressions to carry an undue negative connotation. The capability of LLMs to regurgitate sequences of text included in their training is also referred to as "unintended memorization" (Carlini et al. 2019).

Several studies have raised concerns regarding the exposure of LLMs during training to data used in their evaluations. This is more relevant for proprietary or closed-source LLMs, in which little to no information is known regarding their training data or process. Furthermore, LLMs (especially general-purpose, large ones) are often trained using considerable chunks of the Internet. Sainz et al. (2023a) define different levels of data contamination on LLMs, leading to the creation of the LLM Contamination Index

3 https://github.com/andrefs/punuy-datasets.

(Sainz et al. 2023b, 2023c), where the contamination of each LLM-dataset pair is assessed by measuring how successfully the LLM is able to reproduce a sample of the dataset.

Another source of contamination described in the literature arises from *indirect data leaking*, which happens when LLM providers use the data submitted by the users to iteratively improve the models (Balloccu et al. 2024). This happens frequently when proprietary LLMs are accessed using their free, Web-based interfaces (e.g., ChatGPT, Mistral's Le Chat, Anthropic's Console). Most providers allow users to edit the interface settings to opt out of these data-collecting mechanisms. Furthermore, programmatic API access in all providers is generally free from data collection for model improvements (OpenAI 2025b; Anthropic 2025a; Gemini 2025; AI 2025b).

The SM-related LLM experiments referenced in Section 2.1 present varying degrees of potential LLM dataset exposure.

Musker et al. (2024) introduce a novel dataset (and therefore unfamiliar to the LLM), whereas other research uses publicly accessible datasets (e.g., on GitHub). Additionally, De Deyne, Liu, and Frermann (2024), Trott (2024), and Liu, Melton, and Zhang (2024) state they utilized OpenAI's API. While this does not ensure their evaluation is free from contamination, it offers reassurance that their own experiments did not allow for the data to be later reused for model development, preventing additional dataset exposure in future research. Musker et al. (2024), Liu, Melton, and Zhang (2024), and Di Caro et al. (2023) mention the use of ChatGPT, which poses a risk of indirect data leakage in later studies.

Balloccu et al. (2024) performed a systematic analysis of 255 papers reporting LLM usage to determine dataset contamination for GPT-3.5 and GPT-4. They conclude that 42% of the papers report access to ChatGPT using the Web interface, which resulted in GPT models being exposed to an estimated $\sim$ 4.7M samples from 263 benchmarks.

The studies we mentioned are focused on determining LLM exposure to datasets, i.e., whether (or how much) datasets were initially present or later added to LLM training corpora. Our own view is that trying to track this exposure is mostly a lost battle given the large percentage of the Internet that LLMs are trained on. We should try to avoid leaking unnecessary data. But when evaluating LLMs, their exposure to datasets should be taken as fact, except when the models are open and their training corpora are known, or when evaluated with brand new and yet unpublished datasets. This is also one of the conclusions pointed out by Aiyappa et al. (2023). A more critical examination involves determining the extent to which exposure to datasets influences the evaluation of LLMs, particularly in the context of SR assessment.

With the exception of PAP900, described in Section 3.2, the publication dates of all the datasets used in our study are prior to the data training cutoff dates of the LLM versions we evaluated (listed in Tables 1 and 4, respectively). Consequently, dataset exposure is very likely.

To determine whether such exposure affected LLM performance, in previous work (dos Santos and Leal 2024) we performed tests to determine whether LLMs have been exposed to publicly available SR datasets, and if they were capable of reproducing the datasets' contents. We did this by implementing three different tests:

T1: Given the name and date of publication of a dataset, ask the model to provide a sample of its word pairs. Measure the percentage of correct pairs. Listing 1 provides an example of the prompt used for the WS353 dataset.

T2: Given a sample of pairs included in a dataset, ask the model to provide a different sample of pairs from the same dataset. Measure the percentage of correct pairs.

Listing 1: Prompt asking for a sample of pairs from the WS353 dataset (T1).

```
WordSimilarity-353 is a gold standard dataset published in 2002. It is composed of pairs of
concepts and their semantic similarity score as reported by humans, and can be used to
evaluate semantic measures. Please list 5 pairs of concepts sampled from this dataset.
```

T3: Given a sample of pairs in a dataset, ask the model to rate their semantic relation strength. Measure the percentage of values matching *exactly* the values in the dataset.

The datasets used in that research were identical to those applied in this article, with the following exceptions: `bg100k` was excluded in the current study since it did not evaluate similarity or relatedness, and `pap900` was absent from the earlier work as it was developed later.

High values obtained in T1 and T2 for a given LLM-dataset pair would indicate that the LLM *has knowledge* of the dataset. High values obtained in T3 would indicate that the LLM, when prompted to assess semantic relations' strength, simply reproduced the dataset's values.

Most LLMs achieved positive scores on T1 for a selected group of datasets. Notably, the average accuracy values for all LLMs for `rg65`, `mc30`, and `ws353` were, respectively, 0.504, 0.553, and 0.577. Average values for the other datasets were all below 0.3. In almost half of the datasets, the values obtained across all LLMs were zero. The average LLM accuracy for all datasets and all LLMs in T1 was 0.098.

For the other two tests, almost all LLMs achieved very low scores with all datasets. Notable exceptions in T2 were the values obtained for `gpt-4-turbo-2024-04-09` with the `mc30` dataset (0.433), `claude-3-opus-20240229` with `mesh2` (0.600), and `claude-3-sonnet-20240229` with `ps65` (0.533). The overall average accuracy value obtained in T2 was 0.022. In T3, no dataset or LLM resulted in noticeable outliers, with an overall average accuracy of 0.020.

Despite the low accuracy values obtained for T1 and T2 for most LLM-dataset pairs, LLMs did prove to have been exposed to at least a few datasets. Nevertheless, prior exposure to a dataset did not influence the results obtained in T3, in which all LLMs failed to obtain positive results.

3. Materials

Section 3.1 offers a concise overview of the publicly accessible datasets utilized in this study. Section 3.2 describes the recently released PAP900, an SR dataset which we used to assess if LLMs' SR ratings rely on prior familiarity with the datasets. Section 3.3 outlines the criteria for selecting the LLMs being tested.

We named this LLM evaluation project `punuy-eval`, after the Inca divinity of dreams and sleeping. Most of the experiments took place from October to December 2024, with a few performed in April and July 2025. The code written for implementing these and more tests is publicly available.[4] LLMs were accessed using their APIs. The results obtained are presented and discussed in Section 5, and the raw files can also be found online.[5]

4 `https://github.com/andrefs/punuy-eval`.
5 `https://github.com/andrefs/punuy-results`.

Table 1
Number of SR datasets used in this study, categorized by relation type and language.

	Language		
Relation type	**English**	**Portuguese**	**Total**
Relatedness	21	3	24
Similarity	18	3	21
Total	39	5	45

3.1 Semantic Relations Datasets

For testing the LLMs, we used pairs extracted from previously published quantitative lexical SR datasets. We gathered English and Portuguese datasets reporting semantic similarity and semantic relatedness values between words, multi-word expressions, and named entities (NEs). In our study, each dataset was assigned a unique identifier based on either its name or the names of its creators. Some datasets are split into two or more sub-datasets (which we called **partitions**), which can differ on the set of word pairs rated, the group of annotators, annotation date, relation type, or strictness in outlier detection. Table C.1 in Appendix C presents mean, median, and standard deviation values for the ratings distributions for each dataset partition.

For the main part of this work, we tested LLMs with all the SR datasets we could obtain, with ratings for similarity and relatedness, in English and Portuguese. Table 1 presents the total datasets per language and relation type. Table 2 presents the full list of datasets, which are included in the `punuy-datasets` repository, version 7.0.0. It specifies each dataset's identifier, name, reference, publication year, the language involved in word pairs, and the size (number of pairs) and semantic relationship for each partition. Additionally, it provides a characterization of the datasets according to multiple categories: the level of expertise of the annotators, the part-of-speech of the words, and whether they have a specific focus such as NEs or rare words. Category values are provided in the legend at the end of the table.

Some datasets within the collection we examined overlap each other, meaning they share common pairs. This can happen when newer datasets fully include older datasets (e.g., `rg65` includes `mc30`), or when a new dataset is created by removing pairs deemed problematic for some reason (e.g., `umnsrs` and `umnsrsMod`). In some cases, this results from datasets being a re-annotation of older datasets, as is the case with `ps65` and `rg65`. Finally, overlaps can happen simply because two datasets include a few of the same pairs, by design or by chance.

Tables A.1 and A.2 in Appendix A present the number of word pairs in common for each pair of, respectively, similarity and relatedness datasets. The largest overlap happens between `umnsrs` and `umnsrsMod`. For the 65 overlaps we found, the average number of word pairs was 31.1 ± 82.1.

3.2 PAP900: A New SR Dataset

PAP900 (dos Santos et al. 2024, 2025) is a new gold standard SR dataset, focused on Portuguese affective words. The dataset includes raw and averaged ratings for both relatedness and similarity for 900 pairs of words, each rated by a minimum of 30

Table 2
SR datasets selected for testing the correlation of LLMs ratings of SRs with human judgments.

ID/Name/Ref.	Year	Language	Partition	Size	Rel. type	Categories
atlasify240 Atlasify240 Hecht et al. (2012)	2012	en	main	240	R	NE, O, CW
baker143 Baker, Reichart, and Korhonen (2014)	2014	en	main	144	S	No, O, V, CW
geresid50 GeReSiD Ballatore, Bertolotto, and Wilson (2014)	2014	en	rel sim	50 50	R S	No, Exp, CW
gm30 Gracia and Mena (2008)	2008	en	main	30	R	No, O, N, CW
gtrd GTRD Chen, Song, and Yang (2018)	2018	en	main	66	R	No, Exp, CW
lxrw2034 LX-Rare Word Similarity Dataset Querido et al. (2017)	2017	pt	main	2034	S	No, O, RW
lxsimlex999 LX-SimLex-999 Querido et al. (2017)	2017	pt	main	999	S	No, O, CW
lxws353 LX-WordSim-353 Querido et al. (2017)	2017	pt	main	352	R	No, O, CW
ma28 Martinez-Gil and Aldana-Montes (2013)	2013	en	main	28	S	No, O, CW
mayoSRS MayoSRS Pedersen et al. (2007)	2007	en	mean	101	R	No, Exp, CW
mc30 Miller and Charles (1991)	1991	en	table1	30	S	No, O, CW
men3000 MEN Bruni, Tran, and Baroni (2014)	2014	en	dev test full	2000 1000 3 000	R	No, CS, CW
mesh2 MeSH2 Petrakis et al. (2006)	2006	en	main	36	S	No, Exp, CW
miniMSRS MiniMayoSRS Pedersen et al. (2007)	2007	en	cod phy	29 29	R	No, Exp, CW
mt287 MTurk287 Radinsky et al. (2011)	2011	en	mturk	287	R	No, CS, CW
mturk771 MTurk771 Halawi et al. (2012)	2012	en	mturk	771	R	No, CS, N, CW
pap900 PAP900 dos Santos et al. (2024)	2024	pt	rel sim	900 900	R S	No, O, CW
ps65 Pirró and Seco (2008)	2008	en	main	65	S	No, O, CW
pt65 PT65 Granada, Trojahn, and Vieira (2014)	2014	pt	main	65	R	No, O, CW
rel122 Rel-122 Szumlanski, Gomez, and Sims (2013)	2013	en	main	122	R	No, O, CW

Table 2
Continued.

ID/Name/Ref.	Year	Language	Partition	Size	Rel. type	Categories
reword26 REWORdD Pirró (2012)	2012	en	g26	26	R	NE, O, CW
rg65 Rubenstein and Goodenough (1965)	1965	en	table1	65	S	No, O, N, CW
scws2003 SCWS Huang et al. (2012)	2012	en	main	2003	R	No, CS, CW
semeval2017 SemEval-2017 Camacho-Collados et al. (2017)	2017	en	main	500	S	NE, O, CW
simlex999 SimLex-999 Hill, Reichart, and Korhonen (2015)	2015	en	main	999	S	No, CS, CW
simverb3500 SimVerb-3500 Gerz et al. (2016)	2016	en	dev test all	500 3000 3500	S	No, CS, V, CW
sl7576 SL7576 Silberer and Lapata (2014)	2014	en	main	7576	S	No, CS, N, CW
srw2034 Stanford Rare Word Similarity Dataset Luong, Socher, and Manning (2013)	2013	en	rw	2034	S	No, CS, RW
tr9856 TR9856 Levy et al. (2015)	2015	en	main	9856	R	No, O, CW
umnsrs UMNSRS Pakhomov et al. (2010)	2010	en	rel sim	587 566	R S	No, Exp, CW
umnsrsMod UMNSRS (modified) Pakhomov et al. (2016)	2016	en	rel sim	458 449	R S	No, Exp, CW
word19k Wikipedia Oriented Relatedness Dataset Dor et al. (2018)	2018	en	test train	6307 12969	R	No, CS, CW
wp300 WP300 Li et al. (2013)	2013	en	wp	300	S	NE, O, CW
ws353 WordSimilarity-353 Finkelstein et al. (2001)	2002	en	set1 set2 combined	153 200 353	S	No, O, N, CW
ws353split WordSim353 - Sim. and Rel. Agirre et al. (2009)	2009	en	rel sim	252 203	R S	No, O, N, CW
yp130 YP-130 Yang and Powers (2006)	2005	en	verbpairs	130	S	No, CW
zie55 Ziegler, Simon, and Lausen (2006)	2006	en	B0 B1	25 30	R	NE, O, CW

Table legend:

Relation types: S – similarity, R – relatedness.

Categories: annotator expertise (Exp – experts, CS – crowd sourcing platforms, O – other), part-of-speech (V – verbs, N – nouns), containing entities (E – entities, No – non-entities), containing rare words (RW – rare words, CW – common words).

Table 3
Examples of word pairs and average relatedness and similarity ratings from the PAP900 dataset.

Term1	Term2	Relatedness	Similarity
alegre	alegre	4.000	4.000
aliviado	deslumbrado	0.974	1.024
angustiado	arrogante	1.132	1.263
angustiado	calmo	1.171	0.282
ansioso	frágil	2.316	2.070
apavorado	medroso	3.171	3.211

people. Table 3 provides examples of several word pairs from PAP900 along with their respective relatedness and similarity scores.

Annotators were gathered from students attending three different units from Psychology degrees at the University of Porto, Portugal. Students were mostly female (186 self-identified as women, 23 men, and 4 other), the average age was 22.3 years (standard deviation 6.3 years), and the vast majority (97.6%) were native Portuguese speakers.

One of the challenges of using existing datasets to evaluate LLMs is the possibility of data exposure, i.e., that the same datasets might have been included in the LLMs' training corpora.

The most recent versions (at the time of writing this article) of the LLMs used for this article all predate the PAP900 dataset publication. Furthermore, the queries we performed for our analysis used the LLMs' APIs. This guarantees that the LLMs were not directly or indirectly contaminated by PAP900 data at the time of the evaluation. Comparing the LLMs' results for this dataset with the results for other datasets provided additional insights on whether the LLMs' capability of predicting semantic relations is meaningfully impacted by data exposure during training.

3.3 Large Language Models APIs and Providers

For this task, we selected LLMs that complied with the following requirements:

1. **Accessible as an online service.** This allowed us to try several models without needing to have the hardware to run them and the extra work of provisioning them.
2. **Legally available in our jurisdiction.** Some model providers only operate on a few world regions (e.g., due to Europe's data use restrictions).
3. **JavaScript or HTTP API.** The code developed to implement our work was developed using Node.js.
4. **Supported *function calling*.** Also known as *tool choice*, *structured output*, or simply *JSON mode*, this feature allows requesting the response to be formatted according to a given JSON schema (OpenAI API 2024; Google AI for Developers 2024b; Mistral AI Large Language Models 2024; Anthropic 2024c).

Table 4
LLM breakdown by provider, pricing, the deadline for data training, and the number of internal parameters (when known), sorted by increasing cost of input tokens.

Provider	Model	Cost ($/million tokens) Input/Output	Cutoff date	Internal params.
		Very cheap		
Google	gemini-1.5-flash-8b	0.0375/0.15	Oct 2024	8B
Mistral	ministral-3b-2410	0.0400/0.04	Oct 2024	3B
Google	gemini-1.5-flash-002	0.0750/0.30	Sep 2024	
Mistral	ministral-8b-2410	0.1000/0.10	Oct 2024	8B
OpenAI	gpt-4o-mini-2024-07-18	0.1500/0.60	Oct 2023	
Mistral	open-mistral-nemo-2407	0.1500/0.15	Jul 2024	12B
Mistral	mistral-small-2409	0.2000/0.60	Sep 2024	24B
		Cheap		
Anthropic	claude-3-haiku-20240307	0.2500/1.25	Aug 2023	
OpenAI	gpt-3.5-turbo-0125	0.5000/1.50	Sep 2021	
		Medium		
Google	gemini-1.5-pro-002	1.2500/5.00	Sep 2024	
Mistral	mistral-large-2407	2.0000/6.00	Jul 2024	123B
		Expensive		
OpenAI	gpt-4o-2024-08-06	2.5000/10.00	Oct 2023	
Anthropic	claude-3-5-sonnet-20240620	3.0000/15.00	Apr 2024	
Anthropic	claude-3-sonnet-20240229	3.0000/15.00	Aug 2023	
OpenAI	gpt-4-turbo-2024-04-09	10.0000/30.00	Dec 2023	

After comparing models on leaderboards and benchmarking platforms (Chiang et al. 2024; Li et al. 2023; Artificial Analysis 2024), we chose four different model providers and several models from each: OpenAI, Google, Mistral, and Anthropic. Cohere's `command R` and `command R+` (Cohere 2024) were considered but eventually excluded as we were never able to obtain responses formatted according to the requested JSON schema. Anthropic's `claude-3-opus-20240229` and OpenAI's `gpt-4-0613` were excluded for cost reasons (15$/75$ per million input/output tokens and 30$/60$ per million input/output tokens, respectively). The full list of models used, categorized according to their cost per input token, can be found in Table 4.

Information about how these models were trained is limited and unclear. GPT-4's technical report (Achiam et al. 2023) mentions that the model was pre-trained and then fine-tuned using Reinforcement Learning from Human Feedback (RLHF) (Christiano et al. 2017), but explicitly omits "further details about the architecture (including model size), hardware, training compute, dataset construction, training method, or similar". Anthropic's Claude models were pre-trained on large diverse data, human feedback techniques (Anthropic 2024b), and Constitutional AI, a technique in which model training oversight is provided through a list of rules or principles (Bai et al. 2022). Gemini models were pre-trained using Web documents, books, and code. Post-training consisted of performing supervised fine-tuning on datasets of prompts, and collecting feedback on the generated responses using a reward model, which was

then used in a RLHF stage (Gemini Team et al. 2023). We were unable to find information regarding the training of Mistral's models.

Although the majority of the LLM providers we examined allow users the possibility to further fine-tune their models, our objective in this study was to assess the utility of these LLMs in their unaltered form. Thus, we approached these models as black boxes, refraining from any instruction tuning or modification of their default parameters. A different approach is briefly discussed in Section 7.

The number of internal parameters has been disclosed for all Mistral models used: 3B and 8B, respectively, for `ministral-3b-2410`, and `ministral-8b-2410` (Mistral AI 2024c), 12B for `open-mistral-nemo-2407` (Mistral AI 2024b), 24B for `mistral-small-2409` (Mistral AI 2025), and 123B for `mistral-large-2407` (Mistral AI 2024a). For Gemini 1.5 Flash-8B, it can be inferred from the model name. There are no disclosed numbers of parameters for any of the other models used from Gemini or for any of the models from OpenAI and Anthropic.

Some of the Mistral models can be freely downloaded from MistralAI's HuggingFace page.[6] While `ministral-8b-2410`, `ministral-3b-2410`, and `mistral-large-2407` are published under the Apache 2 License, `open-mistral-nemo-2407` and `mistral-small-2409` are published under the non-open-source Mistral Research License. The models we used from the other providers are not available for download and can only be used as a service.

All these providers allow modifying the model behavior and responses by sending additional parameters. While some of these are common across all providers (e.g., *temperature* or *max_tokens*), others are not: For example, Anthropic does not support setting a *presence_penalty* value. Furthermore, the configuration of these parameters can also vary (e.g., *top_k* must be a positive number in Anthropic, but is limited to the $[1, 40]$ interval in Gemini). The *function calling* feature is executed somewhat differently by the four providers. In particular, OpenAI necessitates configuring the *tool_choice* parameter to guarantee that the models do use a provided tool; otherwise, the models are allowed to decide not to call any tool and may return empty responses. The default values for the most relevant parameters of all LLMs can be found in Table 5.

All LLMs were accessed programmatically using their providers' APIs. This LLM-as-a-service approach presents some disadvantages (mainly regarding concerns with potential data leakage, monetary costs, and time spent in network request round trips). On the other hand, it allows access to state-of-the-art language models without requiring significant investments in powerful computing hardware and the time and work needed to train and deploy models locally. Furthermore, the data policies for all the APIs used (Google Gemini, Anthropic, OpenAI, and Mistral) assure that the data we submitted would not be reused for model development, preventing (further) data exposure.

Some LLM providers have recently introduced caching functionalities that could have been beneficial for our study. However, these features are implemented differently across providers, leading to significant heterogeneity. For instance, OpenAI automatically retrieves prompts from cache by redirecting identical queries to the same servers, whenever possible. Anthropic allows prompt caching but requires explicit instructions and enforces a minimum size for cached messages. In Gemini models, caching involves uploading content in advance, which can then be referenced in subsequent messages. In

6 `https://huggingface.co/mistralai`.

Table 5
Parameter default values for the LLM APIs used in this work.

Provider	Models	Parameters [range]: default
OpenAI (OpenAI 2025a)	gpt-4o-mini-2024-07-18 gpt-3.5-turbo-0125 gpt-4o-2024-08-06 gpt-4-turbo-2024-04-09	`frequency_penalty` $[-2, 2]$: 0 `present_penalty` $[-2, 2]$: 0 `temperature` $[0, 2]$: 1 `top_p` $[0, 1]$: 1
Anthropic (Anthropic 2025b)	claude-3-haiku-20240307 claude-3-5-sonnet-20240620 claude-3-sonnet-20240229	`temperature` $[0, 1]$: 1
Google (API 2025)	gemini-1.5-flash-8b gemini-1.5-flash-002 gemini-1.5-pro-002	`temperature` $[0, 2]$: 1 `top_p` $[0, 1]$: 0.95
Mistral (AI 2025a)	ministral-3b-2410 ministral-8b-2410 open-mistral-nemo-2407 mistral-small-2409 mistral-large-2407	`temperature` $[0, 2]$: 1 `top_p` $[0, 1]$: 1 `presence_penalty` $[-2, 2]$: 0 `frequency_penalty` $[-2, 2]$: 0

contrast, Mistral does not offer any built-in caching mechanism. Due to these differences in implementation, caching was not utilized in our experiments.

4. Experimental Design

This section details how we evaluated the ability of LLMs to produce SR ratings.

We began by designing a prompt and testing how varying the number of pairs to be rated in each query affected LLM results. We describe these tests in Section 4.1. Then, we devised a trial procedure to compare SR ratings produced by an LLM with the corresponding values from an SR dataset, explained in Section 4.2. Finally, Section 4.3 provides a high-level overview of the experimental setup.

4.1 Prompt Design

In this study, we sought to make the LLM querying process closely resemble the method used to build human-sourced SR datasets. To achieve this, we required a prompt that mirrored the kind of instructions generally given to human annotators. Although the focus of this study was to assess the similarity between LLM ratings of SRs and human evaluations (rather than identifying the optimal prompts for this task), we did implement some strategies to select a suitable prompt.

Repurposing a prompt from an existing published dataset might have given an unfair advantage to that specific dataset during evaluation, because one or more documents containing the dataset's prompt, pairs, and values could have been part of the training data for some LLMs.

In previous work (dos Santos and Leal 2024) we experimented on how different levels of detail in the prompt instructions affected LLMs' ability to rate semantic relations. This evaluation was performed using prompts in Portuguese and English, for similarity and relatedness, on several datasets (`pt65`, `lxws353`, `lxsimlex999`, `lxrw2034`, `ws353Rel`, `mturk287`, `ws353Sim`, `yp130`) and on multiple LLMs (`gpt-4-turbo`, `claude-3-opus`, `mistral-large`, and `open-mixtral-8×22b`).

We experimented with prompts with different verbosity levels: a basic prompt just asking to rate pairs on a 1–5 scale, another which provided an explanation for each value of the scale, and three prompts extracted from SR datasets: `ws353`, `sl999`, and a new prompt we crafted to be later used to build the `pap900` dataset (dos Santos et al. 2025). Generally, longer and more detailed prompts obtained the best results.

Although `pap900`'s prompt did not achieve the highest scores in every test, it delivered the most consistent results overall. This prompt was specifically designed to tackle deficiencies observed in others: It included example pair ratings, utilized clear language, and clarified the difference between similarity and relatedness. Because `pap900` was unpublished during our study, we adopted its prompt for the present research. Listings 2 and 3 present the English and Portuguese versions of this prompt.

Listing 2: Prompt in English asking for ratings of semantic relatedness between words.

```
In this survey you'll be asked to rate quantitatively, on a scale, the intensity of the
semantic relatedness between pairs of affective words. Please, before starting, read
carefully the instructions and the examples provided.

The question we're asking is: how much related are the two words? Vaguely related words
should be scored with lower values, and strongly related words with higher values. Please
note that opposite words frequently present high values of relatedness.

For example, the words 'modest' and 'smart' don't seem very related. 'Conceal' and 'mask'
seem very related. 'Confident' is highly related with itself. 'Violent' and 'pacific',
being opposite words, are frequently related, just like 'happiness' and 'sadness'.
Examples:
* modest, smart: 1,
* conceal, mask: 4
* confident, confident: 5
* violent, pacific: 4
* happiness, sadness: 5

Please rate from 0 to 4 the semantic relatedness of the following pairs of words, with 0
indicating words not related at all and 4 indicating very related words:
apple, galaxy
fault, system
champion, winner
town, city
[...]
```

Listing 3: Prompt in Portuguese asking for ratings of semantic similarity between words.

```
Neste questionário ser-lhe-á pedido que avalie quantitativamente, numa escala, a
intensidade da semelhança semântica entre pares de palavras afetivas. Por favor, antes de
começar, leia atentamente a instrução e exemplos abaixo.

A pergunta que lhe fazemos é: quão semelhantes são as duas palavras? Pares de palavras
pouco similares deverão ser pontuados com valores mais baixos, e pares de palavras muito
similares com valores mais altos.

Por exemplo, as palavras 'esperto' e 'inteligente' partilham muitas semelhanças, tal como
'alegria' e 'felicidade'. 'Confiante' partilha muitas semelhanças consigo mesmo. 'Feliz' e
'louco' partilham algumas semelhanças. Já 'triste' e 'divertido' não são nada semelhantes.
Exemplos:
* esperto, inteligente: 4
* alegria, felicidade: 5
* confiante, confiante: 5
* feliz, louco: 3
* triste, divertido: 1

Por favor avalie de 0 a 4 a semelhança semântica dos seguintes pares de palavras, sendo 0
palavras nada semelhantes e 4 palavras muito semelhantes:
maçã, galáxia
falha, sistema
campeão, vencedor
vila, cidade
[...]
```

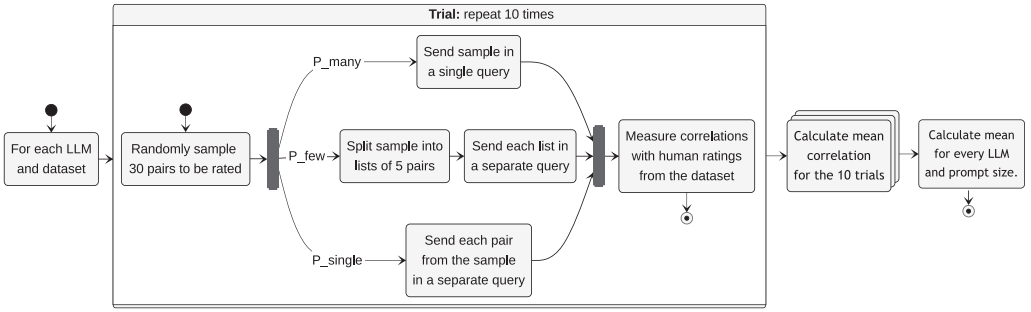

Figure 2
Activity diagram of the process of testing pair list size influence in the correlation of LLM and human ratings of semantic relations.

Within the current study, we also performed a test to determine if and how the size of the list of pairs to be rated sent in each query influences the quality of LLMs' results. Arguably, sending a single pair in each request (with no chat history) makes each task simpler to perform and evaluate, potentially leading to fewer trial repetitions due to invalid responses. On the other hand, sending multiple pairs in each request provides additional context to the LLM and more closely aligns with the human annotation process. Furthermore, sending the pairs in batches reduces input token costs (the instructions and the few-shot examples do not have to be replicated for every pair) and is faster (because it requires fewer network requests).

For this initial assessment, we chose a limited number of datasets[7] to reduce costs and time. These datasets encompass both SR types (similarity and relatedness) and languages (Portuguese and English). We also picked only a few LLMs (chosen to represent all providers while ensuring a diversity of performance levels).[8] For each dataset D and language model L we performed 10 trials, following the procedure described in Section 4.2, with one key distinction. In each trial, we used the sample of pairs to generate three different approaches, with different pair list sizes:

1. In P_{many} we sent the prompt and the sample of 30 pairs in one single query;
2. In P_{few} we split the sample into smaller lists containing 5 pairs and sent each on a different query;
3. In P_{single} we sent each pair from the sample to be rated on a different query.

Figure 2 presents an activity diagram for this workflow.[9] We measured the correlation between the ratings returned by each LLM for each dataset and prompt size. Then we averaged, for each LLM and each prompt size, the correlations obtained across all

7 Datasets: `mt287`, `pap900` (both partitions), `srw2034`, `ws353split` (both partitions), and `yp130`.
8 Models: `claude-3-haiku-20240307`, `gemini-1.5-flash-002`, `gemini-1.5-pro-002`, `gpt-4o-mini-2024-07-18` and `ministral-8b-2410`.
9 This evaluation was performed using the `batchVsSinglePair.ts` script from the `punuy-eval` platform, version 4.0.0.

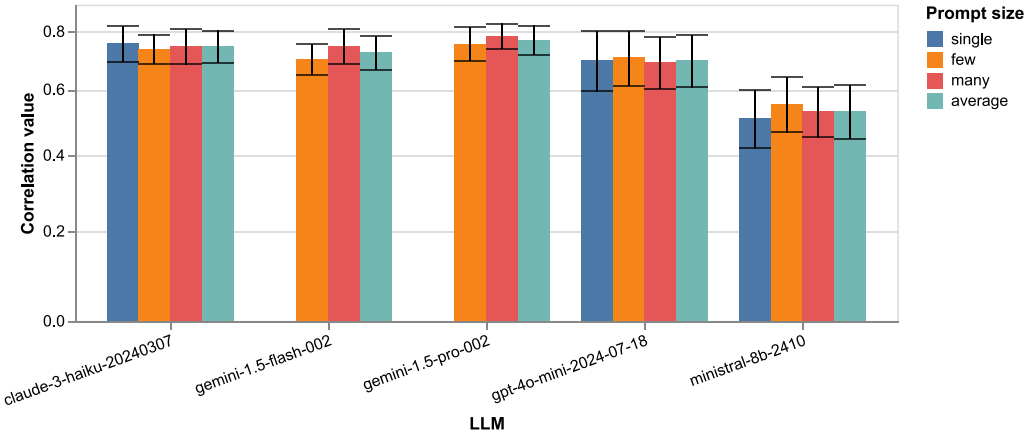

Figure 3
Correlations obtained using different prompt pair list sizes.

datasets, which are presented in Figure 3. For each LLM, the green bar represents the average results for P_{single}; the orange bar represents the average correlation obtained using the P_{few} prompt size across all datasets; and the red bar represents the average results obtained for the P_{many} prompt size. The dark-blue bars represent the average of all results obtained for each LLM. This figure shows no clear correlation between the number of pairs sent in each prompt and the quality of the results obtained, as the correlation values are close to each other and seem more dependent on the model used than on the prompt size.

A notable exception is the two Gemini models tested (`gemini-1.5-flash-002` and `gemini-1.5-pro-002`), which were unable to return valid results when sending a single pair in each query (P_{single}). Upon manually examining the responses of those models, we observed that, in addition to the pair meant to be evaluated, they also included the few-shot example pairs given in the prompts, thereby making the response invalid. This issue with invalid responses did not occur on the other models.

No specific prompt size consistently led to superior outcomes. Consequently, due to the benefits of minimizing time and cost by reducing queries, in each trial of the main LLM evaluation task, a random sample of 30 pairs was submitted in a single query to the LLMs.

4.2 Trial Evaluation Procedure

In this study, we sought to use LLMs in a manner closely resembling the protocol used with human evaluators for rating SRs. This involves explaining the task, instructing annotators to assign a numeric score to each element pair, supplying few-shot examples, and enumerating the pairs for assessment.

For the reasons described at the start of the previous section, we used the prompt from `pap900` (adapted to the language of each dataset) for all LLMs and datasets. Examples of prompts in each language can be found in Listings 2 and 3.

The evaluation mechanism for each LLM and dataset consisted of extracting a sample of word pairs from the dataset, building a prompt query, sending it to the LLM API, receiving the results, validating them, and comparing them with the values

from the dataset (converted to a common scale, when needed). We named each query-response cycle a **trial**.

LLMs occasionally returned invalid responses, such as:

- Server errors (e.g., HTTP 500).
- Responses with empty body.
- Malformed JSON outputs.
- JSON outputs that were well-formed but failed to meet the required schema.

We considered these issues (related to the *format* of the responses and not to their *content*) equivalent to issues found with human annotators: Individuals who are reached out to but fail to respond, or who respond with forms that are either incomplete or invalid. All responses that did not conform to the specified format were deemed invalid and triggered the repetition of the query.

Instead of using the entire datasets, we opted to test with a sample of 30 pairs per trial. This choice was primarily driven by the need to minimize data exposure risks and manage the expenses associated with querying the LLMs' APIs:

1. Splitting the pairs to be rated into multiple requests allows sending only a subset of pairs in each request. This minimizes the overlap between the prompt sent to the LLM and the original dataset and thus reducing the probability of the LLM answer being influenced by exposure to the dataset during training (Carlini et al. 2022).
2. Following a protocol that limits the total number of pairs from each dataset sent to the LLMs allows reducing the possibility of data leakage for future evaluations, which is especially relevant when evaluating new datasets, both for LLMs with no limitations on prompt data reuse for training, and for model providers not following their own restrictive data policies.
3. While the number of word pairs included in each dataset varies wildly, many include hundreds or thousands of pairs. Evaluating the LLMs on the full datasets would lead to a large increase in the number of requests made to the APIs, making the evaluation process too time consuming and resulting in considerable additional monetary costs.

Given a dataset D, a large language model L, a semantic relation R, and a number n, a trial consists of steps described next. Figure 4 presents a simplified version of the trial execution and evaluation protocols.

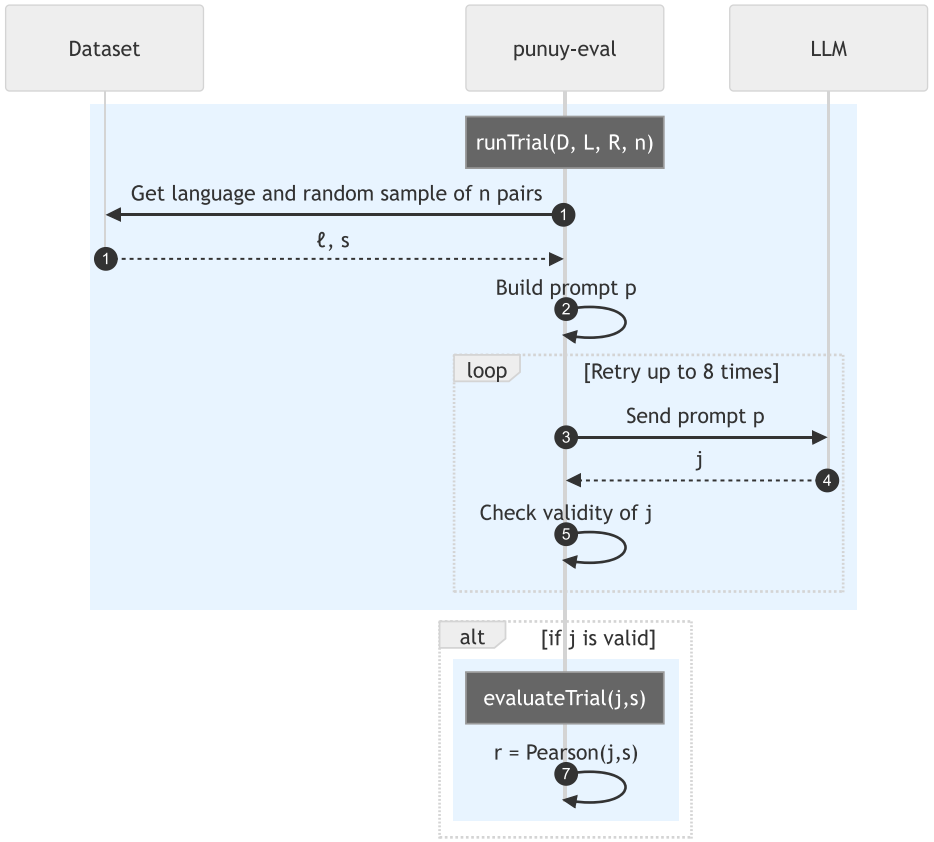

Figure 4
Simplified sequence diagram of the execution and evaluation of a single trial.

Trial Execution and Evaluation.

Step 1. Get the language ℓ and a random sample s of n pairs from D.

Step 2. Build a prompt p consisting of a set of instructions matching R and the language ℓ, a fixed list of few-shot examples, and the list s of n pairs to be rated (Listings 2 and 3 present truncated examples of prompts in English and Portuguese, respectively).

Step 3. Send a request with p to L.

Step 4. Get back the response j from L.

Step 5. Check the validity of the JSON object in j (Listing 4 presents a truncated example of the JSON response obtained).

Step 6. If the format of the response is invalid, repeat the query until a maximum of eight retries are performed. If all responses are invalid, the trial is considered invalid and discarded.

Step 7. Measure the Pearson correlation r between the values reported in j for s and the original values for s on D.

Listing 4: Excerpt of Claude 3 Opus evaluation of pairs from the SemEval17 dataset.

```
{ "scores": [
  { "words": [ "apple"   , "galaxy" ], "score": 0 },
  { "words": [ "fault"   , "system" ], "score": 1 },
  { "words": [ "champion", "winner" ], "score": 4 },
  { "words": [ "town"    , "city"   ], "score": 4 },
  [...]
]}
```

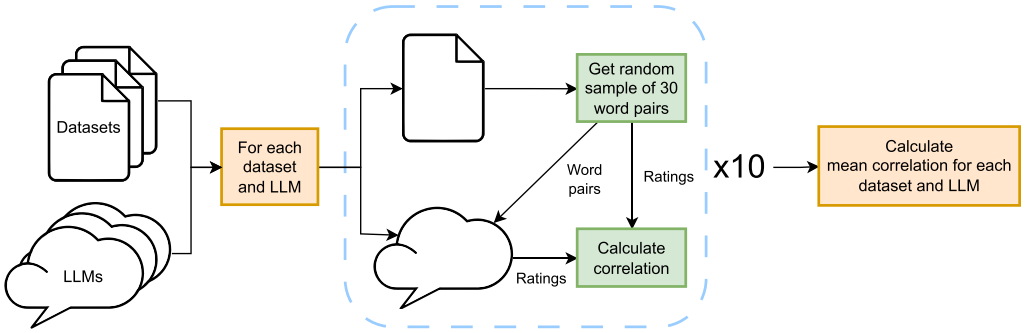

Figure 5
Pipeline for the evaluation of LLM ratings of SRs.

We used the *function calling* feature of the LLM APIs to ensure that the responses would match a given JSON schema. Some responses would be well-formed and match the required schema, but have missing or extraneous pairs, deviating from the requested set. These were also considered invalid, and the whole trial was repeated.

4.3 Rating Semantic Relations Strength Using LLMs: Evaluation Pipeline

As previously discussed, LLMs have shown promising results in evaluating semantic relations. However, comprehensive testing with multiple LLMs across various SR datasets had not yet been conducted. This testing, along with subsequent analysis, forms the core of our study.

For this experiment, we used all the datasets and LLMs listed in Section 3.[10] In each trial, we sampled 30 random pairs of words from a dataset and asked an LLM to rate them, using the procedure described before. In the end of each trial, we measured the correlation between the ratings obtained and the values reported in the datasets. Given the non-deterministic nature of LLMs, we repeated this process, performing 10 trials for each LLM and dataset. For each LLM and dataset pair, after the 10 trials, we calculated the mean correlation value obtained over all valid trials. Figure 5 presents a diagram for this workflow. Figure 6 represents graphically the mean correlation values obtained, which are also presented in Table B.1.

To establish a comparative baseline, we also calculated the cosine between FastText's word embeddings for both English and Portuguese languages (Mikolov et al. 2018). This approach is identified as `ftwe-cosine` throughout this article. FastText functions as a word embedding method akin to Word2Vec's SkipGram (Mikolov et al.

10 This evaluation was performed using the `predictionCorrelation.ts` script from the `punuy-eval` platform, version 4.0.0.

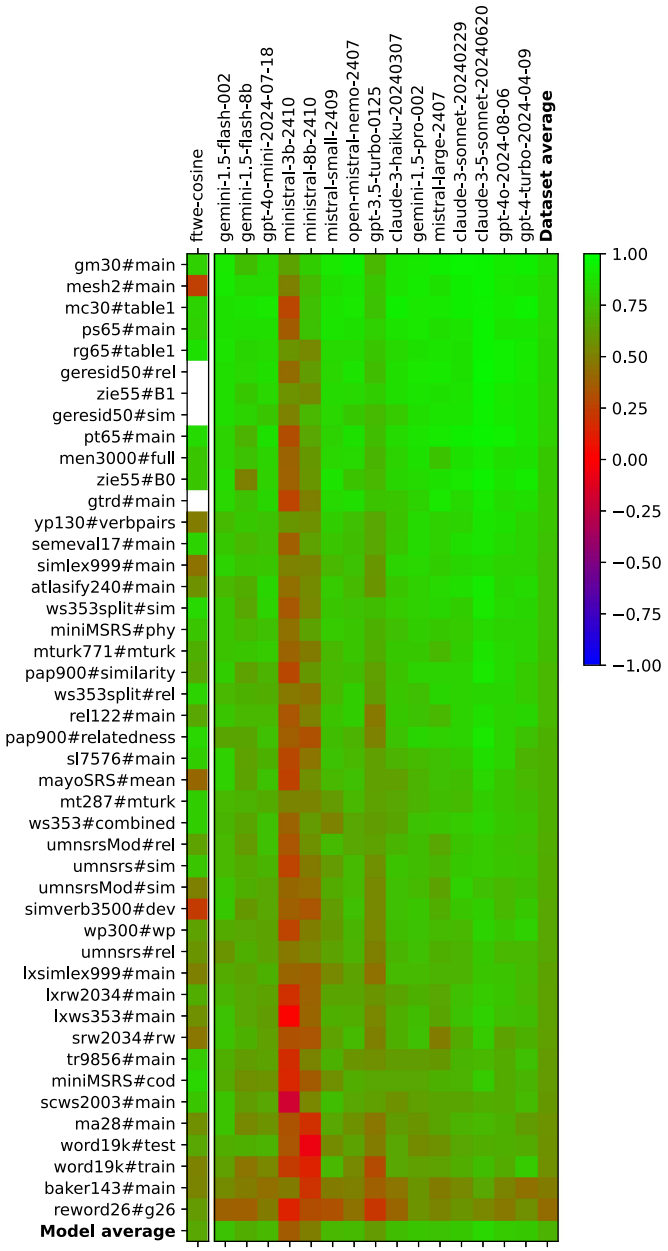

Figure 6
Correlation between LLM results and sampled dataset values, calculated using the Pearson correlation coefficient.

2013), but it processes at the character *n*-gram level rather than the word level. This allows it to achieve better results on word similarity assessment for out-of-vocabulary words and words sharing the same lemma (Bojanowski et al. 2017).

These embeddings were used to, following the same procedures used with LLMs, rate the strength of the SR relation between pairs of words from each dataset by

calculating the cosine similarity between their vectors.[11] Due to the deterministic nature of these calculations, there was no need to perform multiple trials.

For a few datasets and LLMs we also performed evaluations on the full list of pairs, in addition to the 30-pair samples. For these full-dataset evaluations the methodology used was mostly the same as in the sampled evaluation. However, instead of asking the LLM to score a sample of the datasets' pairs, in each trial the order of the pairs in the dataset was randomized and then split into 30-pair chunks, which were then all sent sequentially to the LLM to be scored. This allows direct comparison of the performances against other semantic measures and datasets' inter-annotator agreements. The results of these full evaluations are presented in Tables 7 and 8.

5. Results

This section showcases the findings derived from having each LLM evaluate a sample of pairs from each dataset and then comparing their ratings to the original values to determine correlation. It also encompasses the usage metrics associated with the models and providers used. Lastly, it details the outcomes of statistical hypothesis testing, examining how the attributes of datasets might affect LLMs' performance.

5.1 Evaluation of SRs by LLMs

Figure 6 displays a heat matrix representing the average correlation between the values reported by each LLM and the values sampled from each dataset. Datasets are sorted by decreasing average correlation value obtained. A table with the values used for this figure can be found in Appendix B.

Two datasets in particular achieve low correlation values for all LLMs: `reword26#g26` and `baker143#main`. A distinctive feature of `reword26#g26` is its focus on NEs, but there are several other datasets that also contain entities (`zie55`, `wp300`, `atlasify240`, and `semeval17`) and whose correlation values are higher. The `baker143#main` is a verb similarity dataset; however, `simverb3500#dev` is also dedicated to verbs and achieves better results. The two largest datasets also have low scores: Both partitions of `word19k` and `tr9856` present relatively low correlation values.

Two LLMs, `ministral-3b-2410` and `ministral-8b-2410`, achieved low correlation scores across all datasets. These are small LLMs (<10B parameters), devised mainly to be used for on-device computing and at-the-edge use cases (e.g., on-device translation, Internet-less smart assistants), so these results are not surprising. However, `gemini-1.5-flash-8b`, containing the same number of parameters as `ministral-8b-2410`, obtains much better results (0.679 and 0.493, respectively). Without additional information about their architecture and training methodologies, one might surmise that the variance in outcomes could stem from distinct differences in the training datasets or the post-training and fine-tuning processes for the two models.

The baseline model `ftwe-cosine` did not return values for pairs of words which were missing from the pretrained FastText vectors we used. The `ftwe-cosine` model was able to rate 85% of the pairs. We excluded datasets for which less than 10 values were obtained: `geresid50#rel` and `geresid50#sim` (only 7 pairs rated in each), `gtrd` (2 pairs rated), and `zie55#B1` (7). In the end, `ftwe-cosine` obtained an average correlation of 0.663, below the average for all LLMs, which was 0.701, and far lower than the best

11 The code implementing this can be found at `https://github.com/andrefs/we-cos-sim`.

performing LLM, `claude-3-5-sonnet-20240620`, which achieved a correlation of 0.839. Overall, the LLMs obtained high correlation values with almost all datasets.

Notably, PAP900, the dataset we created recently and that was unpublished at the time of LLM evaluation, obtained above-average correlations (in both similarity and relatedness partitions, `pap900#sim` and `pap900#rel`). This once again disproves the hypothesis that LLMs' capability of predicting human ratings of SRs was dependent on the inclusion of the test datasets in their training corpora.

Figure 7 displays average correlation with dataset values for the results of the best performing LLMs, revealing a high degree of correlation in their performance. Also included are the results obtained with the `ftwe-cosine` baseline model. The standard deviations, also presented, appear to be inversely proportional to the averages. Conversely, in Figure 8 (the lowest-performing LLMs and the baseline model) the lines are farther apart, as the variability between the ratings of each model is higher. The standard

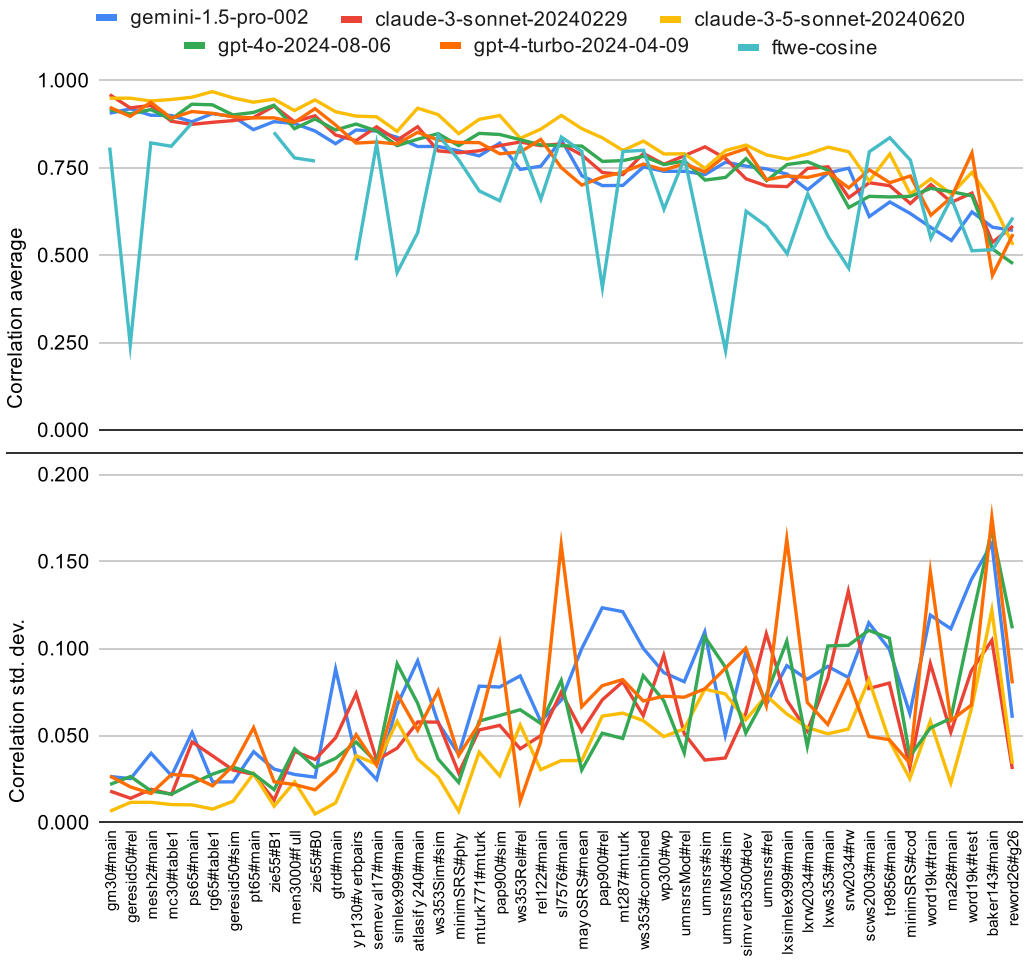

Figure 7
Correlation average and standard deviation for the best performing LLMs, sorted by decreasing average correlation.

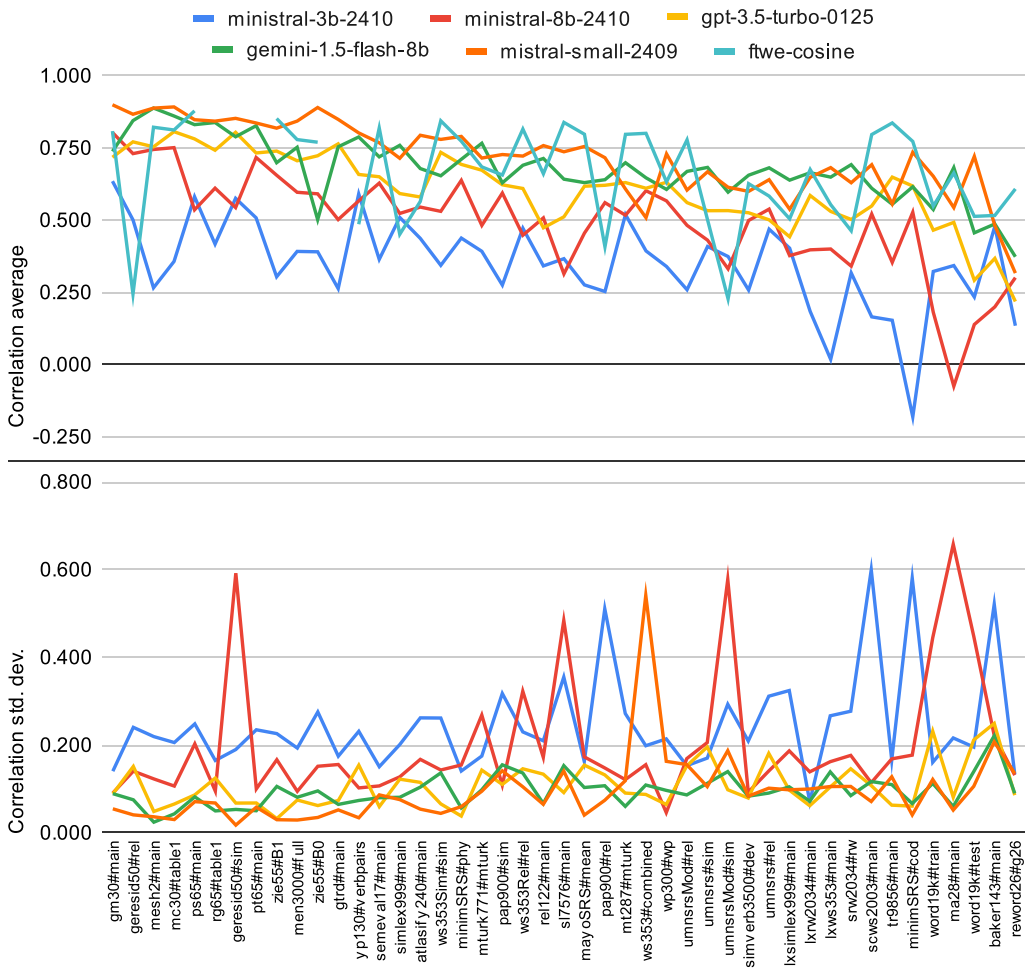

Figure 8
Correlation average and standard deviation for the worst performing LLMs, sorted by decreasing average correlation.

deviations for the results of these models are higher. In both figures, datasets are sorted by decreasing average LLM correlation value.

5.2 LLM Usage Metrics

Table 6 presents data relative to the usage of the models and providers on the sampled evaluation. For each model, we show the overall count of executed requests, the count of invalid responses received, and specifically those due to HTTP errors. Additionally, we provide the average number of input and output tokens for each request, along with the total incurred cost. For each provider, we present the average number of requests per model, along with the totals for the remaining categories.

The numerous provider errors encountered by `gpt-4o-mini-2024-07-18` and `gpt-3.5-turbo-0125` align with a timeframe when the OpenAI API was experiencing issues. The other invalid responses from the Google models, and from the models of the other

Table 6
Statistics on the utilization of models and providers.

Models/providers	Requests	Invalid responses[1]	Provider errors[2]	Input tokens/ req.	Output tokens/ req.	Cost (USD)
Anthropic	613/model	366		910	650	12.96
claude-3-haiku-20240307	773	307		1,002	740	0.91
claude-3-sonnet-20240229	596	59		772	563	6.42
claude-3-5-sonnet-20240620	470			931	613	5.64
Google	534/model	108	6	540	157	0.85
gemini-1.5-flash-002	561	1	1	545	88	0.04
gemini-1.5-flash-8b	574	101		538	187	0.03
gemini-1.5-pro-002	466	6	5	536	203	0.78
Mistral	839/model	2,025	8	632	512	2.76[3]
ministral-3b-2410	925	453	2	629	567	0.05[3]
ministral-8b-2410	595	141		620	561	0.08[3]
mistral-small-2409	489	29	4	683	585	0.24[3]
open-mistral-nemo-2407	1,735	1,402	2	611	430	0.26[3]
mistral-large-2407	450			683	573	2.14[3]
OpenAI	745/model	999	900	405	382	27.61
gpt-4o-mini-2024-07-18	847	378	360	308	272	0.18
gpt-3.5-turbo-0125	1,062	594	540	276	291	0.61
gpt-4o-2024-08-06	454			541	472	3.61
gpt-4-turbo-2024-04-09	618	27		495	494	23.21

[1] Any response that did produce an evaluatable result: including empty responses, invalid JSON syntax, invalid JSON Schema format, responses with mismatched pairs, and *provider errors*.
[2] Subset of *invalid responses* caused by HTTP client/server errors.
[3] Originally charged in EUR, converted to USD for consistency.

providers, resulted mostly from the returned JSON data not matching the requested schema. Most of the invalid responses obtained with the `ministral-*` models resulted from the returned JSON list missing the `score` property on the last item, or from issues related to multi-word expressions in some datasets. A more forgiving approach than the one we followed could lead to a lower number of invalid responses, e.g., by accepting responses with missing pairs, or simply discarding responses including scores for pairs not sent in the prompt query.

The input and output tokens and the costs incurred were gathered from the metadata included in the models' responses. While some variation in the number of input and output tokens per request was expected (as evaluated pairs often contain multi-word expressions), other factors are likely responsible for the differences observed. The methods used for counting tokens can differ between providers and even between models of the same provider. The number of output tokens can also vary wildly on responses whose JSON did not respect the requested schema. The `claude-3-haiku-20240307` model has an unusually large count of input tokens, whereas Google's models, such as `gemini-1.5-flash-002`, show notably low output token counts.

The model that achieved the top performance, `claude-3-5-sonnet-20240620`, turned out to be significantly cheaper than the priciest model, `gpt-4-turbo-2024-04-09`. Google's models provide the most optimal cost/benefit balance, with `gemini-1.5-`

`flash-8b` and `gemini-1.5-flash-002` delivering commendable results, minimal rates of invalid responses, and low costs, especially in the case of the latter model.

The sampled evaluation required 10 trials for 45 dataset partitions, each composed of 30 pairs to be rated. This resulted in the assessment of 13,500 pairs of words, which (including retries due to invalid responses) cost around 44 USD (see Table 6). Performing the same evaluation on the full datasets would require assessing more than 555k pairs of words, which would likely raise the costs to almost 2,000 USD.

5.3 Impact of Dataset Features on LLMs

We conducted statistical hypothesis tests to assess if certain dataset attributes might result in notable differences in correlation values derived from the LLMs. To achieve this, we outlined a group of attributes such as language, relation type, and part-of-speech, along with potential values for each attribute, including options like Portuguese/English, relatedness/similarity, and verbs/nouns. This characterization was performed based on the description of each dataset provided by the authors in the literature and has been summarized in the columns *Language*, *Relation type*, and *Categories* in Table 2.

For every attribute, we created subsets for each category containing the average correlations obtained with each of the 15 LLMs for each dataset, and examined if the distribution within each sample remained consistent. According to the Shapiro-Wilk test (Shapiro and Wilk 1965), none of the samples exhibited a normal distribution. Therefore, we applied the two-tailed Mann-Whitney U test (Fay and Proschan 2010) to compare the distributions of each pair of samples.

Portuguese vs. English: The English datasets group contained 585 correlation values (39 English datasets multiplied by 15 models), with a median correlation score of $Md = 0.74$, while the Portuguese datasets group contained 90 values (6 Portuguese datasets multiplied by 15 models), with a median correlation score of $Md = 0.72$. The Mann-Whitney test showed no significant difference: $U = 28822.50$, $p = 0.147$, $r = -0.09$.

Relatedness vs. Similarity: The relatedness datasets group contained 360 values, with a median correlation score of $Md = 0.72$, while the similarity datasets group contained 315 values, with a median correlation score of $Md = 0.74$. The Mann-Whitney test showed no significant difference: $U = 54353.50$, $p = 0.353$, $r = 0.04$.

Verbs vs. Nouns: A few of the datasets we analyzed reportedly are focused on verbs, while others focus on nouns instead. We put the former and the latter into two different groups. The verbs datasets group contained 30 values, with a median correlation score of $Md = 0.54$, while the nouns datasets group contained 105 values, with a median correlation score of $Md = 0.78$. The Mann-Whitney test showed a significant difference: $U = 575.00$, $p = 0.000$, $r = 0.63$.

Entities vs. Non-entities: Some of the datasets we analyzed reportedly have a special focus on named entities. We put those into one group, and all others into another group. The entities datasets group contained 90 values, with a median correlation score of $Md = 0.74$, while the non-entities datasets group contained 585 values, with a median correlation score of $Md = 0.73$. The Mann-Whitney test showed no significant difference: $U = 27248.50$, $p = 0.592$, $r = -0.04$.

Rare words vs. Common words: The rare words datasets group contained 30 values, with a median correlation score of $Md = 0.66$, while the common words datasets group contained 645 values, with a median correlation score of $Md = 0.74$. The

Mann-Whitney test showed a significant difference: $U = 6308.00$, $p = 0.001$, $r = 0.35$.

Annotator expertise: For this analysis, instead of two groups we divided the datasets into three groups:

(a) datasets annotated by domain experts (medical and geographical datasets),

(b) datasets annotated using crowd-sourcing platforms (such as Amazon Mechanical Turk), and

(c) all other datasets.

The assumption here is that crowd-sourcing platform workers are, on average, less specialized and arguably less motivated, than other groups of annotators. Experts, on the other hand, not only possess in-depth knowledge of the relevant field, but are often intrinsically motivated, due to their professional interests, to perform a quality job.

The (a) group contained 165 values, with a median correlation score of $Md = 0.75$; (b) contained 150 values, with a median correlation score of $Md = 0.70$, and (c) contained 360 values, with a median correlation score of $Md = 0.74$.

A Kruskal-Wallis test (Kruskal and Wallis 1952) revealed a significant difference among groups ($H = 15.98$, $p < 0.001$). Post-hoc pairwise comparisons using Dunn's test (Dunn 1964) with Holm correction showed that (b) were significantly different ($p < 0.05$) from both (a) ($p = 0.000480$) and (c) ($p = 0.001682$), and no significant difference between (a) and (c) ($p = 0.28089$).

Ratings distribution: We normalized the ratings in all datasets to use the same 1–5 scale, and then we calculated the mean (μ), median (M), and standard deviation (σ) of the ratings distribution for each dataset (the values obtained are presented in Table C.1). For each measure μ, M, and σ, we split the correlation values from the 45 datasets evenly into three groups:

(a) containing the correlation values from the 15 datasets with the lowest values,

(b) containing the correlation values from the 15 datasets with intermediate values,

(c) containing the correlation values from the 15 datasets with the highest values.

When analyzing the ratings means (μ), the $(a)_\mu$ group contained 225 samples, with a median correlation score of $Md = 0.70$; $(b)_\mu$ contained 225 samples, with a median correlation score of $Md = 0.78$; and $(c)_\mu$ contained 225 samples, with a median correlation score of $Md = 0.72$.

A Kruskal-Wallis test revealed a significant difference among groups ($H = 35.54$, $p < 0.001$). Post-hoc pairwise comparisons using Dunn's test with Holm correction showed that $(b)_\mu$ were significantly different ($p < 0.05$) from both $(a)_\mu$ ($p = 3.800166e{-}10$) and $(c)_\mu$ ($p = 0.000023$); with no significant difference between

$(a)_{\mu}$ and $(c)_{\mu}$ ($p = 0.1915798$). The analysis of the ratings medians (M) yielded comparable results.

When analyzing the ratings standard deviations (σ), the $(a)_{\sigma}$ group contained 225 samples, with a median correlation score of $Md = 0.70$; $(b)_{\sigma}$ contained 225 samples, with a median correlation score of $Md = 0.73$; and $(c)_{\sigma}$ contained 225 samples, with a median correlation score of $Md = 0.79$.

A Kruskal-Wallis test revealed a significant difference among groups ($H = 53.34$, $p < 0.001$). Post-hoc pairwise comparisons using Dunn's test with Holm correction showed that all group differences were statistically significant ($p < 0.05$): $p = 0.001576$ for $(a)_{\sigma}$ and $(b)_{\sigma}$; $p = 7.5029e{-}05$ for $(b)_{\sigma}$ and $(c)_{\sigma}$; and $p = 9.8291e{-}13$ for $(a)_{\sigma}$ and $(c)_{\sigma}$.

6. Discussion

In this section we compare the results obtained by evaluating LLMs on the full datasets (no sampling) with existing semantic measure algorithms (Section 6.1) and with the datasets' reported IAAs (Section 6.2). We also observe how the results of the sampled evaluation vary with the datasets' publication date (Section 6.3) and with other characteristics of the datasets (Section 6.4).

6.1 RQ1. How do LLMs compare against other semantic measure algorithms in assessing semantic relation strength?

Despite all their (often surprising) capabilities, LLMs are machine learning algorithms. When used to assess the strength of semantic relations between elements, they can be viewed as semantic measure algorithms. As such, the correlation values obtained rating SRs with LLMs can be compared with the correlation values obtained in the evaluations of other SMs.

For this, we picked four datasets commonly used for SM evaluation. We selected SM algorithms from the literature based on their evaluations' coverage of several of these datasets, or for achieving exceptional scores on at least one of them, allowing us to compare LLMs against the state-of-the-art SMs. To make results directly comparable, we asked a few LLMs to rate the full list of pairs in each of these datasets. Table 7 contains, for these datasets, the LLM correlation values for the five best-performing LLMs. It also includes, for the same datasets, the correlation values obtained for the selected SMs evaluated with the same four SR datasets, gathered from the literature, and the values obtained using a baseline approach based on measuring the cosine of FastText embeddings (`ftwe-cosine`). In all cases, correlations were calculated using Pearson's r.

The table provides details on each SM algorithm, specifying the approach type (whether it is a structured approach relying on knowledge bases, an unstructured/semi-structured method utilizing text corpora, vocabularies, or dictionaries, or a hybrid technique that integrates elements from both approaches) and the semantic proxies it was applied on (i.e., the source of information from which the algorithms extract the semantic evidence to compare the linguistic elements). The highest value obtained on each dataset for both LLMs and SM algorithms is highlighted in bold.

`claude-3-5-sonnet-20240620` achieved the best results on all datasets. The `gpt` models achieved the second and third best results for RG65, followed by Do19-hybrid and ADW. For SimLex999, all the evaluated LLMs performed better than any other approach. Both `claude` models achieved the best results on MEN3000, with the third and fourth places going to DC20-hybrid and DC19-hybrid, respectively. For WS353,

Table 7
Comparison of the best-performing LLMs with existing SMs, evaluated on the full datasets.

SM algorithm	Type	Semantic proxies	RG65	SimLex999	MEN3000	WS353
ADW[1]	KB	Wiktionary	0.910			
DC20-hybrid[2]	H	GNC, BNC			**0.865**	
Do19-corpus[3]	CB	BNC	0.737	0.401	0.707	0.577
Do19-hybrid[3]	H	GNC, BNC	**0.914**	0.481	0.859	0.276
Pe14[4,3]	CB	Wikipedia	0.770	0.433	0.803	**0.705**
Sa18[5,3]	CB	Wikipedia	0.792	0.426	0.803	0.704
Sp17[6,3]	H	ConceptNet, CN-NB	0.896	0.634	0.846	
SVR4[7]	H	Wikipedia, WordNet, other		**0.642**		
SRel[8]	H	Wikipedia, WordNet	0.894	0.621	0.622	0.650
SimFirNon[9]	H	Wikipedia, WordNet	0.903			
Word embeddings cosine			**RG65**	**SimLex999**	**MEN3000**	**WS353**
`ftwe-cosine`			0.833	0.485	0.824	0.748
LLM			**RG65**	**SimLex999**	**MEN3000**	**WS353**
`claude-3-5-sonnet-20240620`			**0.959**±0.006	**0.866**±0.006	**0.910**±0.002	**0.825**±0.010
`gpt-4o-2024-08-06`			0.923±0.021	0.841±0.002	0.858±0.003	0.760±0.012
`claude-3-sonnet-20240229`			0.885±0.020	0.810±0.006	0.869±0.003	0.791±0.011
`gemini-1.5-pro-002`			0.869±0.026	0.846±0.004	0.855±0.003	0.721±0.018
`gpt-4-turbo-2024-04-09`			0.916±0.016	0.837±0.006	0.853±0.003	0.719±0.048

The references adjacent to each algorithm link to the papers describing the algorithms and (if different) to the implementation the results were gathered from:

[1] Pilehvar and Navigli (2015)
[2] Dobó and Csirik (2020)
[3] Dobó (2019)
[4] Pennington, Socher, and Manning (2014)
[5] Salle, Idiart, and Villavicencio (2016)
[6] Speer, Chin, and Havasi (2017)
[7] Banjade et al. (2015)
[8] Hussain et al. (2023)
[9] Huang et al. (2023)

Algorithm types:

KB: Knowledge-based
CB: Corpus-based
H: Hybrid

Semantic proxies:

GNC: Google News Corpus (Mikolov et al. 2013)
BNC: British National Corpus (Aston and Burnard 2020)
CN-NB: ConceptNet Numberbatch (Speer, Chin, and Havasi 2017)

`ftwe-cosine` achieved the third best result, and LLMs performed better than any other SM.

A notable point is that some of the SM algorithms used in this analysis incorporate word embeddings, which are a fundamental method crucial to large language models, within their computations—e.g., Sp17 (Speer, Chin, and Havasi 2017) and Do19-hybrid (Dobó 2019).

In the past, SMs typically returned better results at predicting relatedness when utilizing approaches with large context windows; on the other hand, SMs showed improved accuracy in predicting similarity when they employed narrower context windows (Hill, Reichart, and Korhonen 2015). One defining characteristic of LLMs is being able to operate with very large context windows (due to their architecture), often several orders of magnitude above older approaches. If the same trend persisted, LLMs ought to have shown better performance in predicting relatedness. However, based on the sample-based evaluation, whose results are presented in Figure 6 and Appendix B,

LLMs performed comparably on both relatedness and similarity, with an average score of 0.696 (standard deviation 0.105) for relatedness datasets and 0.708 (standard deviation 0.097) for similarity datasets.

6.2 RQ2. How closely do LLM evaluations of semantic relation strength align with ratings produced by humans?

LLM correlation values can also be compared with the datasets' IAA values. The IAA is widely assumed to represent the ceiling of what automated approaches can achieve in NLP tasks (Hill, Reichart, and Korhonen 2015), a belief arising from the idea that, if human annotations are considered the source of truth, the instances where annotators disagree represent cases where the truth is not known, or is at least debatable. Automated systems have shown, however, to be able to achieve results that show a correlation with the dataset values (averaged from all annotators) higher than the dataset IAA. Other authors consider this to be proof of IAA not being a ceiling for automated approaches (Boguslav and Cohen 2017; Richie, Grover, and Tsui 2022).

Not all datasets used in our analysis have reported inter-annotator agreement metrics. Additionally, IAA is not consistently measured with the same metrics: While APIAA and AMIAA (Vulić et al. 2021) are commonly referenced in the literature, other methods are also used.

While IAA values are computed based on the raw annotator values, SMs are evaluated by measuring the correlation against dataset-averaged values. We followed the same approach to evaluate the LLMs. Consequently, AMIAA (also known as *leave-one-out*) is the method most suitable for comparison with LLM correlation values, as it is calculated by averaging the correlations between each annotator and the mean values from the other annotators. In AMIAA, for each annotator i, a mean μ_i is calculated for the scores of all other annotators. Then the correlation $c(s_i, \mu_i)$ is calculated. The final agreement value is the average of all those correlations.

$$\text{AMIAA}_c = \frac{\sum_i c(s_i, \mu_i)}{N}, \text{ where: } \mu_i = \frac{\sum_{j,j\neq i} s_j}{N-1} \tag{1}$$

IAA values, together (when available) with the metric and method used to calculate them, are included in the metadata linked to each dataset in our repository. For datasets that provide raw annotator values, we calculated the AMIAA values (using Pearson's r) if they were not already reported.

Table 8 presents the AMIAA value obtained for several datasets, whether reported by the authors or calculated by us. It also presents correlation values, obtained on the full datasets, for three different LLMs: a high-performing one (`claude-3-5-sonnet-20240620`), a low-performing one (`ministral-3b-2410`), and a cost-effective one (`gemini-1.5-flash-002`). These LLMs were selected to illustrate variations in LLM performance and to demonstrate the existing trade-offs between cost and performance.

Our analysis reveals that `ministral-3b-2410`, which is the least effective LLM in our study, consistently produced correlation values lower than the IAA across all datasets. The `gemini-1.5-flash-002` model showed correlation values above the IAA in two datasets, and below the IAA in the other two. `claude-3-5-sonnet-20240620`, our top-performing model, exhibited correlation values far surpassing the datasets' stated IAA. Consequently, it is possible to conclude that LLMs are, in fact, capable of producing SR assessments at least as good as those produced by groups of human annotators.

Table 8
Comparison between LLMs' correlation values and datasets' IAAs, evaluated on the full datasets.

Datasets	gtrd#main	pap900#rel	pap900#sim	baker143#main
AMIAA	0.852*	0.757†	0.675†	0.517*
claude-3-5-sonnet-20240620	**0.903** ± 0.020	**0.813** ± 0.010	**0.899** ± 0.003	**0.666** ± 0.032
ministral-3b-2410	0.287 ± 0.156	0.368 ± 0.060	0.429 ± 0.035	0.044 ± 0.110
gemini-1.5-flash-002	0.883 ± 0.016	0.723 ± 0.018	0.790±0.010	0.512 ± 0.025
ftwe-cosine		0.726	0.661	0.499

∗ AMIAA value recalculated for the present article.
† AMIAA value reported by the authors of the dataset.

6.3 RQ3. Are LLMs' evaluations of SRs influenced by data exposure during training?

Datasets are often duplicated and accessible online from various origins. Older datasets have been around longer, potentially leading to more replications. For instance, a simple search on Google and GitHub for the mc30 dataset (Miller and Charles 1991) showed that it has been reproduced over ten times on GitHub, and it can also be found on platforms like Kaggle and ResearchGate. Many copies of the original article it was reported on (Miller and Charles 1991), which includes the whole dataset, can also be found online.

These numerous replications on popular sites are likely to be incorporated into the training data of LLMs. Carlini et al. (2019) have demonstrated that the *unintended memorization* of a word sequence is more likely when the number of copies of the sequence in the training corpus is higher. If LLMs performed better with older datasets, it might be attributable to this difference in representation in the training corpora and be a sign of data exposure impacting LLM results on SR rating.

Figure 9 illustrates the change in correlation with dataset values over time for the top-performing LLMs and all datasets. The colored lines represent linear regression trends for each LLM, all of which exhibit a small negative slope. The coefficients of determination (R^2) for these trends are relatively low (around 0.1). We did F-tests to determine the statistical significance of the results of these linear regressions. The data can be found in Table 9. In all five cases, the p-value is below the 0.05 threshold, suggesting that the linear regressions are statistically significant.

This allows us to conclude that, while LLMs tend to predict human ratings slightly more effectively on older datasets, the correlation is rather small, with the dataset year accounting for 9% to 11% of the correlations' variance.

Combining these findings with the above average values obtained with the PAP900 dataset and the earlier experiment demonstrating that LLM results do not improve even in cases where they can demonstrate exposure to existing SR datasets (as detailed in Section 2.3), we conclude that the high correlation values LLMs obtained evaluating lexical semantic relations are not determined by dataset exposure during training.

6.4 RQ4. Do SR dataset characteristics influence the performance of LLMs?

Based on the hypothesis testing findings presented in Section 5.3, it can be concluded that some features of the datasets can significantly affect the LLMs' capability to evaluate semantic relations.

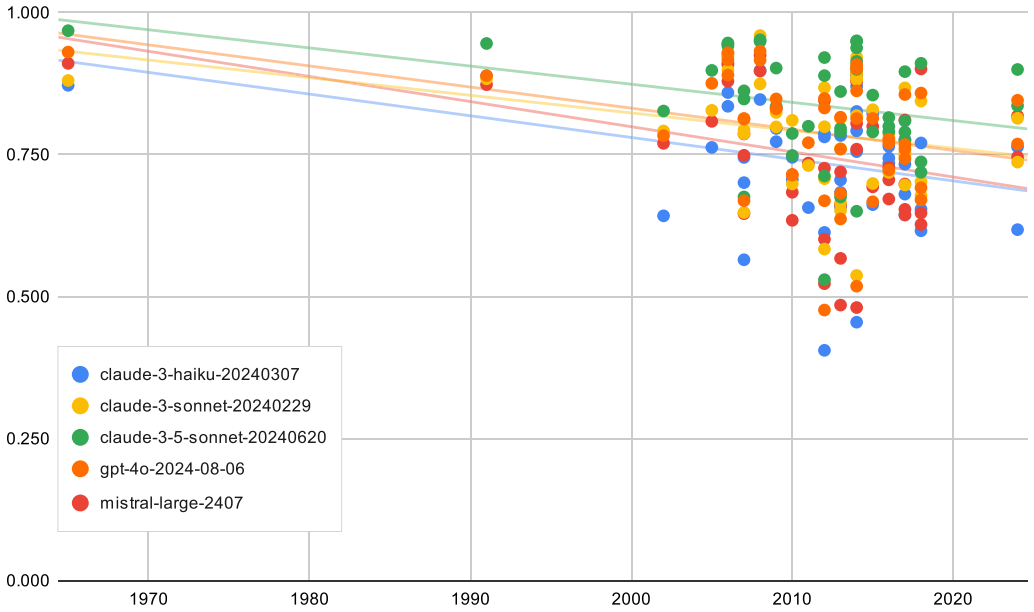

Figure 9
Evolution of the correlation between LLM results and dataset values over time for the best performing LLMs.

Table 9
Simple linear regression models between the LLM correlations and datasets publication year.

Model	Regression equation		R^2	p-value
	Slope	Intercept		
claude-3-haiku-20240307	−0.0038	8.4041	0.099	0.0356
claude-3-sonnet-20240229	−0.0031	7.0503	0.087	0.0489
claude-3-5-sonnet-20240620	−0.0032	7.2611	0.088	0.0479
gpt-4o-2024-08-06	−0.0037	8.2620	0.102	0.0326
mistral-large-2407	−0.0044	9.6578	0.112	0.0244

Concerning the word language, the p-value observed, which exceeds the established threshold, indicates no significant difference in LLM performance across the two groups. This outcome might be surprising: English resources are estimated to constitute approximately 50% of Web sites, compared with Portuguese, which makes up only around 4% (W3Techs 2025). As such, one would expect English to be overrepresented in the LLMs' training corpora. These results imply that LLMs may achieve similar performances in various languages, even if those languages are less prevalent in the training data compared with English.

The resulting p-value indicates that there is likewise no notable difference in LLM performance when comparing relatedness and similarity datasets. These findings were anticipated: Understanding the difference between types of relations is frequently challenging for human individuals; the creation methods for some datasets did not clearly highlight this distinction; and even conventional SM algorithms often overlook this differentiation (Hill, Reichart, and Korhonen 2015; Banjade et al. 2015; Costa and Leal 2016).

The low p-value of the test between noun and verb datasets suggested that the former tend to have higher scores than the latter. One possible explanation is that verbs, in both Portuguese and English, are morphologically richer: Their form can change based on tense, person, number, voice, and so on, while noun forms vary only by number and (in Portuguese) by gender. This may have been a confounding factor for LLMs. Nevertheless, LLM tokenizers are able to split texts at the sub-word level, reducing the ambiguity, and one of the verb datasets (`baker143`) includes verb pairs in different inflections and conjugations.

There was no significant difference found in LLM performance between datasets reportedly focusing on NEs, and all others. We found that in `semeval17` and `wp300`, both described in the literature as including NEs, these make up only a small fraction of the total words. We tried repeating the test after removing these datasets from the *named entities* group but obtained similar results.

The test between datasets containing rare words and all the other datasets suggested the latter tend to lead to higher LLM correlation values. This can be caused by an under-representation of the rare words in the LLMs' training corpora. Although the medical datasets do not specifically aim at rare words, they do include domain-specific terms that are infrequent in everyday conversation. We tried repeating the test while also including these datasets in the *rare words* group and obtained similar results.

The analysis of the different levels of annotator expertise found significant differences between the results obtained by the crowd-sourcing workers and both the experts and the other annotators. LLMs obtained the worst correlations against the ratings from crowd-sourcing workers, and the best correlations against the experts and the other annotators.

The analysis of the means and medians of the ratings distributions demonstrated that LLMs performed best with datasets whose ratings were centered around moderate values, and obtained worse results with datasets with the lowest and highest ratings. When analyzing the standard deviations of the ratings distributions, LLMs performed the best with high standard deviation values, and the worst with low standard deviation values.

One possible explanation for these variations in LLM correlations with datasets' ratings distributions is human annotators getting more tired evaluating pairs whose ratings are very close to each other. Based on our experience, assessing multiple pairs consecutively is simpler when they exhibit clearly different relation values. In contrast, evaluation in pairs with similar relation values is less obvious, demands greater mental effort, and therefore tends to be more tiring. Because LLMs do not suffer from this effect, their ratings are more distant from humans in these datasets. Verifying the validity of this hypothesis would require further experiments which are out of scope for the present work.

In summary, from the dataset characteristics tested, the words' part-of-speech, the rarity of the words, the expertise of the annotators, and the distribution of word pair ratings seem to have a significant influence on LLM performance.

7. Conclusions and Future Work

The findings of this study demonstrate that LLMs can effectively assess the strength of semantic relations, as their predictions show a strong correlation with human-reported values. Furthermore, some models achieved better correlation values with human ratings than previous SM algorithms; in some cases, even surpassing datasets' inter-annotator agreement values. This highlights the potential for future research in semantic

measures to leverage LLMs, either as proxies for human ratings in evaluating semantic measures or as components within SM algorithms.

Importantly, the high correlation of LLM ratings with human assessments of SMs is not attributable to LLMs being trained on the specific datasets used for testing. In previous research, described in Section 2.3, we determined, using the same datasets and a subset of the LLMs featured in the present study, that LLMs are generally incapable of reproducing the contents of previously published SR datasets. Even in the few cases where LLMs were capable of demonstrating knowledge of datasets, this had no effect on their results in SR assessment. Building on this, in this research we show that the correlation between LLM predictions and dataset ratings is not considerably higher for older datasets. Furthermore, the results obtained with PAP900, a dataset created after the LLMs were trained, were even above average, further supporting the robustness of LLMs in this context.

Some of the datasets used in this study have reported inter-annotator agreements. LLMs' correlation values compare favorably against these human IAAs. This supports the claim that LLMs are at least as good as groups of humans in assessing lexical semantic relation strength.

LLMs did not perform better when we changed the size of the list of pairs to be rated sent with each prompt. Sending multiple pairs to be rated in each query allows reducing costs by minimizing the queries made to the LLMs and thus the repetition of prompt instructions. While some LLMs now support caching parts of requests, batching pairs in prompts still offers the advantage of saving time on network request round trips.

LLMs occasionally return invalid responses, from server errors to badly formatted JSON objects. The error rates can differ significantly depending on the model and provider. Therefore, it is essential to implement data validation mechanisms instead of trusting LLM responses to be well-formatted without verification.

This work can be extended in several directions. The same analysis can be applied to other languages, newer versions of the same LLMs, or models from different LLM providers. Although this study was focused on lexical semantic relations, the same methodology could be used to assess LLMs' capabilities in evaluating quantitative semantic relations involving other elements, such as words in context, sentence pairs, or entire documents.

In this study, we also explored the impact of dataset features on the ability of LLMs to evaluate semantic relations. A comparable evaluation could be extended to LLMs to identify how aspects of their design, training methodology, or training data contribute to variations in outcomes. Due to the limited availability of information about the closed-source LLMs we utilized, we were unable to conduct such an analysis. Additionally, future work could examine word pair-level properties such as difficulty and impactfulness, factors that may not be captured by agreement rates alone but could meaningfully affect LLM behavior and the quality of downstream annotations. Operationalizing these concepts in a measurable and reproducible way remains a methodological challenge, but doing so could shed light on qualitative differences between human and model judgments, particularly in edge cases.

This research aimed to explore the feasibility of employing LLMs as substitutes for a panel of annotators in the creation of SR datasets. We strived to closely mimic, with LLMs, the surveying techniques typically applied with human annotators. Future research could delve into alternative prompt engineering strategies, like 0-shot or chain-of-thought techniques, which are less common in traditional SR dataset construction processes. These approaches could be attempted not just with LLMs but also with human annotators, aiming to improve correlation outcomes for the former and to enhance inter-annotator agreement metrics for the latter.

LLMs have demonstrated that their layers encode rich knowledge about word semantics. This latent information can potentially be harnessed for more specialized applications involving semantic relations and measures. While this study utilized general-purpose LLMs without any customization, future research could explore fine-tuning LLMs specifically for semantic relation assessment. The datasets provided in the `punuy-datasets` repository could prove valuable for this fine-tuning process.

In Section 6.1, we analyzed the outcomes from certain models across several datasets, comparing them to the reported evaluation scores for SMs on those same datasets. The research community would benefit from a large-scale evaluation of SM algorithms, applied to different semantic proxies, using multiple datasets (again taking advantage of the data collected in `punuy-datasets`).

Lastly, LLMs offer a promising avenue for generating new datasets. Creating quantitative SR datasets traditionally involves asking human subjects to rate word pairs, a labor-intensive process that limits dataset size. Additionally, such datasets are often sparse, as including the same word in multiple pairs can lead to annotator fatigue. In contrast, LLMs do not get tired, they do not get bored, and they can be leveraged to construct large, dense SR datasets efficiently.

The magnitude and density of these datasets make them applicable in situations where existing datasets fall short. For example, these LLM-generated large datasets could be used to comprehensively analyze the structure of a given domain, and together with clustering algorithms, automatically build domain taxonomies, or infer non-taxonomical relationships.

Appendix A. Overlaps Between Datasets

Table A.1
Total overlapping word pairs for each pair of datasets with relatedness ratings.

Dataset	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.
1. atlasify240					3	1			1					
2. gm30													1	
3. lxws353								14					1	
4. mayoSRS										1	1			
5. men300	3					3	9					1	1	
6. mt287	1				3							1		
7. mturk771					9				1					
8. pt65			14											
9. tr9856	1						1					30		1
10. umnsrs				1							420			
11. umnsrsMod				1						420				
12. word19k					1	1			30				1	

Table A.2
Total overlapping word pairs for each pair of datasets with similarity ratings.

Dataset	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.
1. lxrw2034		4		1								5						
2. lxsimlex999	4			3					1									
3. mc30					29	29	19		1							30	27	
4. pap900	1	3																
5. ps65			29			65	19		1							29	26	
6. rg65			29		65		19		1							29	26	
7. scws2003			19		19	19		5	9	2	2	4	1	1	3	269	149	
8. semeval17							5		8		4					4	4	
9. simlex999		1	1		1	1	9	8		28	21	1				9	7	1
10. simverb3500							2		28									1
11. sl7576							2	4	21						2	1		
12. srw2034	5						4		1									
13. umnsrs							1							417				
14. umnsrsMod							1						417					
15. wp300							3				2					5	5	
16. ws353			30		29	29	269	4	9		1				5		203	
17. ws353split			27		26	26	149	4	7						5	203		
18. yp130									1	1								

Appendix B. Correlation Values Between LLM Results and Dataset Values

Table B.1
Correlation between LLM results and sampled dataset values, calculated using Pearson's *r*.

Dataset ID	ftwe-cosine	gemini-1.5-flash-002	gemini-1.5-flash-8b	gpt-4o-mini-2024-07-18	ministral-3b-2410	ministral-8b-2410	mistral-small-2409	open-mistral-nemo-2407	gpt-3.5-turbo-0125	claude-3-haiku-20240307	gemini-1.5-pro-002	mistral-large-2407	claude-3-sonnet-20240229	claude-3-5-sonnet-20240620	gpt-4o-2024-08-06	gpt-4-turbo-2024-04-09	Dataset average
gm30#main	0.808	0.902	0.738	0.841	0.635	0.803	0.899	0.927	0.718	0.898	0.907	0.925	0.960	0.949	0.917	0.924	0.863
mesh2#main	0.244	0.878	0.888	0.900	0.266	0.745	0.888	0.903	0.754	0.923	0.901	0.908	0.929	0.942	0.917	0.938	0.845
mc30#table1	0.822	0.879	0.861	0.869	0.357	0.751	0.892	0.877	0.806	0.888	0.900	0.873	0.884	0.946	0.889	0.892	0.838
ps65#main	0.812	0.887	0.831	0.846	0.581	0.536	0.848	0.844	0.781	0.847	0.882	0.897	0.875	0.952	0.932	0.911	0.830
rg65#table1	0.878	0.865	0.838	0.862	0.417	0.611	0.843	0.884	0.742	0.872	0.905	0.911	0.881	0.968	0.931	0.906	0.829
geresid50#rel		0.909	0.845	0.844	0.501	0.730	0.867	0.893	0.771	0.889	0.919	0.879	0.922	0.950	0.905	0.898	0.848
zie55#B1		0.824	0.699	0.877	0.304	0.656	0.819	0.875	0.739	0.859	0.883	0.911	0.927	0.947	0.930	0.893	0.809
geresid50#sim		0.874	0.789	0.841	0.576	0.544	0.853	0.846	0.804	0.871	0.897	0.881	0.885	0.951	0.902	0.896	0.827
pt65#main	0.852	0.865	0.827	0.781	0.508	0.717	0.836	0.788	0.733	0.792	0.860	0.893	0.894	0.938	0.909	0.893	0.816
men3000#full	0.779	0.825	0.752	0.821	0.392	0.597	0.843	0.844	0.706	0.826	0.877	0.760	0.882	0.914	0.862	0.881	0.785
zie55#B0	0.486	0.853	0.502	0.809	0.390	0.591	0.890	0.758	0.723	0.835	0.856	0.879	0.899	0.945	0.890	0.920	0.783
gtrd#main		0.842	0.754	0.834	0.263	0.502	0.849	0.865	0.764	0.771	0.819	0.901	0.844	0.911	0.859	0.873	0.777
yp130#verbpairs	0.486	0.731	0.788	0.751	0.591	0.567	0.802	0.743	0.658	0.763	0.859	0.809	0.828	0.898	0.876	0.821	0.766
semeval17#main	0.819	0.782	0.719	0.748	0.364	0.628	0.768	0.735	0.650	0.777	0.856	0.811	0.868	0.896	0.856	0.824	0.752
simlex999#main	0.451	0.821	0.759	0.776	0.509	0.523	0.714	0.754	0.592	0.822	0.838	0.800	0.829	0.855	0.814	0.819	0.748
atlasify240#main	0.565	0.716	0.679	0.829	0.437	0.545	0.794	0.729	0.580	0.781	0.811	0.845	0.868	0.921	0.832	0.852	0.748
ws353split#sim	0.844	0.778	0.654	0.817	0.344	0.531	0.780	0.755	0.735	0.796	0.812	0.830	0.799	0.903	0.848	0.830	0.747
miniMSRS#phy	0.773	0.768	0.711	0.748	0.438	0.638	0.790	0.773	0.693	0.745	0.799	0.787	0.793	0.848	0.814	0.823	0.745
mturk771#mturk	0.684	0.765	0.766	0.793	0.392	0.482	0.715	0.774	0.673	0.787	0.785	0.726	0.799	0.889	0.849	0.823	0.735
pap900#sim	0.656	0.799	0.629	0.701	0.275	0.594	0.727	0.735	0.622	0.764	0.821	0.818	0.814	0.900	0.846	0.791	0.722
ws353split#rel	0.815	0.712	0.690	0.678	0.472	0.448	0.722	0.800	0.610	0.773	0.746	0.833	0.824	0.835	0.831	0.796	0.718

Table B.1
Continued.

Dataset ID	ftwe-cosine	gemini-1.5-flash-002	gemini-1.5-flash-8b	gpt-4o-mini-2024-07-18	ministral-3b-2410	ministral-8b-2410	mistral-small-2409	open-mistral-nemo-2407	gpt-3.5-turbo-0125	claude-3-haiku-20240307	gemini-1.5-pro-002	mistral-large-2407	claude-3-sonnet-20240229	claude-3-5-sonnet-20240620	gpt-4o-2024-08-06	gpt-4-turbo-2024-04-09	Dataset average
rel122#main	0.661	0.776	0.714	0.722	0.342	0.508	0.758	0.797	0.474	0.784	0.755	0.720	0.814	0.861	0.816	0.832	0.711
pap900#rel	0.839	0.810	0.640	0.755	0.253	0.561	0.716	0.745	0.621	0.618	0.700	0.746	0.737	0.836	0.769	0.724	0.682
sl7576#main	0.798	0.647	0.642	0.743	0.367	0.314	0.737	0.701	0.512	0.755	0.831	0.805	0.819	0.900	0.814	0.751	0.689
mayoSRS#mean	0.408	0.806	0.630	0.688	0.276	0.455	0.755	0.724	0.618	0.701	0.728	0.749	0.787	0.862	0.813	0.701	0.686
mt287#mturk	0.797	0.717	0.699	0.667	0.519	0.517	0.608	0.718	0.630	0.657	0.700	0.735	0.731	0.800	0.771	0.738	0.681
ws353#combined	0.801	0.707	0.647	0.743	0.394	0.602	0.509	0.654	0.610	0.642	0.753	0.770	0.792	0.827	0.784	0.761	0.680
umnsrsMod#rel	0.632	0.726	0.669	0.708	0.259	0.482	0.604	0.728	0.561	0.765	0.741	0.706	0.785	0.791	0.770	0.762	0.670
umnsrs#sim	0.778	0.762	0.683	0.652	0.409	0.431	0.668	0.724	0.533	0.745	0.731	0.634	0.810	0.749	0.715	0.738	0.666
umnsrsMod#sim	0.501	0.789	0.597	0.653	0.374	0.331	0.614	0.749	0.533	0.744	0.767	0.672	0.777	0.800	0.723	0.786	0.661
simverb3500#dev	0.227	0.665	0.656	0.637	0.258	0.499	0.600	0.708	0.526	0.735	0.755	0.728	0.719	0.815	0.777	0.805	0.659
wp300#wp	0.626	0.724	0.606	0.744	0.340	0.567	0.730	0.656	0.631	0.706	0.740	0.662	0.761	0.790	0.760	0.744	0.677
umnsrs#rel	0.583	0.584	0.681	0.640	0.469	0.538	0.640	0.701	0.501	0.706	0.747	0.684	0.699	0.787	0.714	0.716	0.654
lxsimlex999#main	0.505	0.669	0.639	0.692	0.403	0.377	0.538	0.640	0.442	0.733	0.732	0.698	0.697	0.776	0.760	0.727	0.635
lxrw2034#main	0.675	0.707	0.664	0.634	0.184	0.397	0.648	0.646	0.585	0.644	0.688	0.654	0.749	0.790	0.768	0.723	0.632
lxws353#main	0.554	0.756	0.649	0.611	0.018	0.400	0.682	0.675	0.530	0.681	0.736	0.644	0.753	0.810	0.743	0.737	0.628
srw2034#rw	0.463	0.762	0.692	0.624	0.317	0.341	0.630	0.728	0.502	0.684	0.749	0.485	0.665	0.796	0.637	0.693	0.620
tr9856#main	0.796	0.667	0.555	0.572	0.153	0.354	0.557	0.677	0.649	0.662	0.653	0.693	0.699	0.790	0.667	0.708	0.604
miniMSRS#cod	0.837	0.762	0.616	0.662	−0.181	0.527	0.735	0.660	0.618	0.565	0.620	0.646	0.648	0.676	0.669	0.727	0.597
scws2003#main	0.773	0.685	0.610	0.633	0.165	0.521	0.692	0.582	0.549	0.613	0.611	0.601	0.708	0.712	0.669	0.745	0.607
ma28#main	0.549	0.678	0.682	0.700	0.343	−0.075	0.543	0.638	0.492	0.658	0.543	0.567	0.652	0.676	0.682	0.666	0.563
word19k#test	0.663	0.622	0.456	0.527	0.234	0.139	0.721	0.534	0.292	0.654	0.624	0.627	0.678	0.737	0.671	0.793	0.554
word19k#train	0.513	0.759	0.537	0.573	0.322	0.181	0.652	0.570	0.465	0.616	0.580	0.647	0.703	0.719	0.692	0.615	0.575
baker143#main	0.516	0.541	0.487	0.435	0.479	0.200	0.481	0.507	0.368	0.455	0.581	0.481	0.537	0.650	0.519	0.443	0.477
reword26#g26	0.608	0.377	0.373	0.482	0.135	0.301	0.317	0.447	0.218	0.406	0.572	0.523	0.584	0.530	0.476	0.561	0.420
Model average	0.663	0.755	0.679	0.721	0.352	0.493	0.717	0.734	0.607	0.738	0.768	0.750	0.789	0.839	0.791	0.787	0.701

Appendix C. Summary Statistics of the Ratings Distributions in SR Datasets

Table C.1
Mean, median, and standard deviation values for the ratings distributions in SR datasets.

Dataset partition	Mean	Median	Std. dev.	Dataset partition	Mean	Median	Std. dev.
atlasify240#main	3.0337	2.9000	1.3476	**reword26#g26**	3.0969	3.0120	1.0001
baker143#main	2.0343	1.8800	0.5571	**rg65#table1**	2.8757	2.4800	1.3377
geresid50#rel	2.7614	2.9624	1.1265	**scws2003#main**	2.5409	2.3600	0.9729
geresid50#sim	2.5603	2.5572	1.0829	**semeval17#main**	2.9908	3.0800	1.2313
gm30#main	3.2453	3.8700	1.2219	**simlex999#main**	2.8246	2.8680	1.0453
gtrd#main	2.7864	2.6584	1.0665	**simverb3500#dev**	2.7256	2.7520	1.0630
lxrw2034#main	2.9362	3.0000	0.7914	**sl7576#main**	1.6872	1.2000	0.9730
lxsimlex999#main	2.4811	2.6680	1.0707	**srw2034#rw**	3.4840	3.7440	1.0043
lxws353#main	3.3387	3.5240	0.8675	**tr9856#main**	2.9821	3.0000	1.4086
ma28#main	3.0813	3.1040	0.8567	**umnsrs#rel**	3.2380	3.1825	0.8991
mayoSRS#mean	1.9361	1.8222	0.8307	**umnsrs#sim**	2.6770	2.7050	0.8267
mc30#table1	2.9713	2.6800	1.4071	**umnsrsMod#rel**	3.2651	3.3244	0.9270
men3000#full	3.0011	3.0000	0.9847	**umnsrsMod#sim**	2.7106	2.7612	0.8212
mesh2#main	2.9583	3.0000	1.1997	**word19k#test**	1.7859	1.4000	1.1464
miniMSRS#cod	1.6621	1.1333	1.0548	**word19k#train**	1.8109	1.4000	1.1421
miniMSRS#phy	2.2736	2.3333	1.1109	**wp300#wp**	3.3300	3.8000	1.1707
mt287#mturk	2.7633	2.5625	0.8015	**ws353#combined**	3.3423	3.5240	0.8690
mturk771#mturk	2.9630	3.0000	1.0001	**ws353split#rel**	3.1179	3.3480	0.8565
pap900#rel	2.2005	1.9210	0.8626	**ws353split#sim**	3.0530	2.9000	0.9991
pap900#sim	2.086	1.7500	0.9333	**yp130#verbpairs**	2.7051	2.3330	1.2541
ps65#main	2.5451	2.1660	1.0047	**zie55#B0**	3.0786	2.9824	0.8818
pt65#main	2.9105	2.5200	1.3019	**zie55#B1**	3.4189	2.9020	1.1779
rel122#main	3.1880	3.2571	1.0545				

Acknowledgments

André Fernandes dos Santos: Ph.D. grant SFRH/BD/129225/2017 from Fundação para a Ciência e Tecnologia (FCT), Portugal. This work is funded by national funds through FCT – Fundação para a Ciência e a Tecnologia, I. P., under the support UID/50014/2025 (https://doi.org/10.54499/UID/50014/2025).

References

Achiam, Josh, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023, GPT-4 technical report. Technical report, OpenAI.

Agirre, Eneko, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Pasca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27. https://doi.org/10.3115/1620754.1620758

AI, Mistral. 2025a. Mistral AI API. https://docs.mistral.ai/api/

AI, Mistral. 2025b. Terms of use. https://mistral.ai/terms#terms-of-service

Aiyappa, Rachith, Jisun An, Haewoon Kwak, and Yong-yeol Ahn. 2023. Can we trust the evaluation on ChatGPT? In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 47–54. https://doi.org/10.18653/v1/2023.trustnlp-1.5

Akila, D. and C. Jayakumar. 2014. Semantic similarity- a review of approaches and metrics. *International Journal of Applied Engineering Research*, 9(24):27581–27600.

AlMousa, Mohannad, Rachid Benlamri, and Richard Khoury. 2022. A novel word sense disambiguation approach using WordNet knowledge graph. *Computer Speech & Language*, 74:101337. https://doi.org/10.1016/j.csl.2021.101337

Anthropic. 2024a. Anthropic models overview. https://docs.anthropic.com/claude/docs/models-overview. Accessed: May 8, 2024.

Anthropic. 2024b. The Claude 3 model family: Opus, Sonnet, Haiku. Technical report, Anthropic.

Anthropic. 2024c. Tool use (function calling). https://docs.anthropic.com/claude/docs/tool-use. Accessed: May 9, 2024.
Anthropic. 2025a. Is my data used for model training? | Anthropic Privacy Center. https://privacy.anthropic.com/en/articles/7996868-is-my-data-used-for-model-training
Anthropic. 2025b. Messages. https://docs.anthropic.com/en/api/messages
API, Gemini. 2025. Generating content. https://ai.google.dev/api/generate-content
Artificial Analysis. 2024. AI Model & API Providers Analysis | Artificial Analysis. https://artificialanalysis.ai. Accessed: December 20, 2024.
Aston, Guy and Lou Burnard. 2020. *The BNC Handbook: Exploring the British National Corpus with SARA*. Edinburgh University Press. https://doi.org/10.1515/9780748628889
Bai, Yuntao, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022, Constitutional AI: Harmlessness from AI feedback. Anthropic.
Baker, Simon, Roi Reichart, and Anna Korhonen. 2014. An unsupervised model for instance level subcategorization acquisition. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 278–289. https://doi.org/10.3115/v1/D14-1034
Ballatore, Andrea, Michela Bertolotto, and David C. Wilson. 2014. An evaluative baseline for geo-semantic relatedness and similarity. *GeoInformatica*, 18(4):747–767. https://doi.org/10.1007/s10707-013-0197-8
Balloccu, Simone, Patrícia Schmidtová, Mateusz Lango, and Ondřej Dušek. 2024. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source LLMs. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*. https://doi.org/10.18653/v1/2024.eacl-long.5
Banjade, Rajendra, Nabin Maharjan, Nobal B. Niraula, Vasile Rus, and Dipesh Gautam. 2015. Lemon and tea are not similar: Measuring word-to-word similarity by combining different methods. In *Computational Linguistics and Intelligent Text Processing: 16th International Conference, Cicling 2015, Proceedings, Part I 16*, pages 335–346. https://doi.org/10.1007/978-3-319-18111-0_25
Boguslav, Mayla and Kevin Bretonnel Cohen. 2017. Inter-annotator agreement and the upper limit on machine performance: Evidence from biomedical natural language processing. In *MEDINFO 2017: Precision Healthcare through Informatics*, pages 298–302. https://doi.org/10.3233/978-1-61499-830-3-298
Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146. https://doi.org/10.1162/tacl_a_00051
Boslaugh, Sarah. 2012. *Statistics in a Nutshell*. O'Reilly Media, Inc.
Boyd-Graber, Jordan, Christiane Fellbaum, Daniel Osherson, and Robert Schapire. 2006. Adding dense, weighted connections to WordNet. In *Proceedings of the Third International WordNet Conference*, pages 29–36.
Bruni, Elia, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49:1–47. https://doi.org/10.1613/jair.4135
Camacho-Collados, Jose, Mohammad Taher Pilehvar, Nigel Collier, and Roberto Navigli. 2017. SemEval-2017 Task 2: Multilingual and cross-lingual semantic word similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 15–26. https://doi.org/10.18653/v1/S17-2002
Carlini, Nicholas, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*. https://doi.org/10.52202/075280-1708
Carlini, Nicholas, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 267–284.
Cer, Daniel, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity – Multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*. https://doi.org/10.18653/v1/S17-2001

Chandrasekaran, Dhivya and Vijay Mago. 2022. Evolution of semantic similarity – A survey. *ACM Computing Surveys*, 54(2):1–37. https://doi.org/10.1145/3440755

Chen, Zugang, Jia Song, and Yaping Yang. 2018. An approach to measuring semantic relatedness of geographic terminologies using a thesaurus and lexical database sources. *ISPRS International Journal of Geo-Information*, 7(3):98. https://doi.org/10.3390/ijgi7030098

Chiang, Wei Lin, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, et al. 2024. Chatbot arena: An open platform for evaluating LLMs by human preference. *Forty-first International Conference on Machine Learning*.

Christiano, Paul F., Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems*, 30.

Cohere. 2024. Models. https://docs.cohere.com/docs/models. Accessed: May 8, 2024.

Costa, Teresa and José Paulo Leal. 2016. Semantic measures: How similar? How related? In *International Conference on Web Engineering*, pages 431–438. https://doi.org/10.1007/978-3-319-38791-8_29

De Deyne, Simon, Chunhua Liu, and Lea Frermann. 2024. Can GPT-4 recover latent semantic relational information from word associations? A detailed analysis of agreement with human-annotated semantic ontologies. In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon @ LREC-COLING 2024*, pages 68–78. https://doi.org/10.63317/22u6z3oumx2t

Di Caro, Luigi, Laura Ventrice, Stefano Locci, and Rachele Mignone. 2023. Semantic doppelgängers: How LLMs replicate lexical knowledge. In *GENERAL@ Chitaly*, pages 12–18.

Dobó, András. 2019. *A Comprehensive Analysis of the Parameters in the Creation and Comparison of Feature Vectors in Distributional Semantic Models for Multiple Languages*. Ph.D. thesis, University of Szeged.

Dobó, András and János Csirik. 2020. A comprehensive study of the parameters in the creation and comparison of feature vectors in distributional semantic models. *Journal of Quantitative Linguistics*, 27(3):244–271. https://doi.org/10.1080/09296174.2019.1570897

Dolan, William B., Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 350–356. https://doi.org/10.3115/1220355.1220406

Dor, Liat Ein, Alon Halfon, Yoav Kantor, Ran Levy, Yosi Mass, Ruty Rinott, Eyal Shnarch, and Noam Slonim. 2018. Semantic relatedness of Wikipedia concepts – Benchmark data and a working solution. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

dos Santos, André Fernandes and José Paulo Leal. 2023. A game with a purpose for building crowdsourced semantic relations datasets for named entities. In *Science and Information Conference*, pages 422–439. https://doi.org/10.1007/978-3-031-37963-5_30

dos Santos, André Fernandes and José Paulo Leal. 2024. Early findings in using LLMs to assess semantic relations strength. In *13th Symposium on Languages, Applications and Technologies (SLATE 2024)*, volume 120 of *Open Access Series in Informatics (Oasics)*, pages 4:1–4:9.

dos Santos, André Fernandes, José Paulo Leal, Rui Alves, and Teresa Jacques. 2024. PAP900. https://doi.org/10.17632/5mhxtv8pn2.3

dos Santos, André Fernandes, José Paulo Leal, Rui Alexandre Alves, and Teresa Jacques. 2025. PAP900: A dataset of semantic relationships between affective words in Portuguese. *Data in Brief*, 61:111726. https://doi.org/10.1016/j.dib.2025.111726, PubMed: 40534920

Dunn, Olive Jean. 1964. Multiple comparisons using rank sums. *Technometrics*, 6(3):241–252. https://doi.org/10.1080/00401706.1964.10490181

Fay, Michael P. and Michael A. Proschan. 2010. Wilcoxon-Mann-Whitney or t-test? On assumptions for hypothesis tests and multiple interpretations of decision rules. *Statistics Surveys*, 4:1. https://doi.org/10.1214/09-SS051, PubMed: 20414472

Fellbaum, Christiane. 2010. WordNet. In *Theory and Applications of Ontology: Computer Applications*. pages 231–243.

https://doi.org/10.1007/978-90-481-8847-5_10
Feng, Yue, Ebrahim Bagheri, Faezeh Ensan, and Jelena Jovanovic. 2017. The state of the art in semantic relatedness: A framework for comparison. *Knowledge Engineering Review*, 32:e10. https://doi.org/10.1017/S0269888917000029
Finkelstein, Lev, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1). https://doi.org/10.1145/371920.372094
Garla, Vijay N. and Cynthia Brandt. 2012. Semantic similarity in the biomedical domain: An evaluation across knowledge sources. *BMC Bioinformatics*, 13:1–13. https://doi.org/10.1186/1471-2105-13-261, PubMed: 23046094
Gemini. 2025. How Gemini for Google Cloud uses your data. https://cloud.google.com/gemini/docs/discover/data-governance#submit-receive-data
Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: A family of highly capable multimodal models. (preprint).
Gerz, Daniela, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A large-scale evaluation set of verb similarity. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2173–2182. https://doi.org/10.18653/v1/D16-1235
Giabelli, Anna, Lorenzo Malandri, Fabio Mercorio, Mario Mezzanzanica, and Navid Nobani. 2022. Embeddings evaluation using a novel measure of semantic similarity. *Cognitive Computation*, 14(2):749–763. https://doi.org/10.1007/s12559-021-09987-7
Google AI for Developers. 2024a. Gemini models. https://ai.google.dev/gemini-api/docs/models/gemini. Accessed: May 8, 2024.
Google AI for Developers. 2024b. Intro to function calling with the Gemini API. https://ai.google.dev/gemini-api/docs/function-calling. Accessed: May 8, 2024.
Grabb, Declan. 2023. The impact of prompt engineering in large language model performance: A psychiatric example. *Journal of Medical Artificial Intelligence*, 6. https://doi.org/10.21037/jmai-23-71
Gracia, Jorge and Eduardo Mena. 2008. Web-based measure of semantic relatedness. In *Web Information Systems Engineering - WISE 2008: 9th International Conference, Auckland, New Zealand, September 1–3, 2008. Proceedings 9*, pages 136–150. https://doi.org/10.1007/978-3-540-85481-4_12
Granada, Roger, Cassia Trojahn, and Renata Vieira. 2014. Comparing semantic relatedness between word pairs in Portuguese using Wikipedia. In *Computational Processing of the Portuguese Language: 11th International Conference, PROPOR 2014, Proceedings 11*, pages 170–175. https://doi.org/10.1007/978-3-319-09761-9_17
Hadi, Muhammad Usman, Qasem Al Tashi, Abbas Shah, Rizwan Qureshi, Amgad Muneer, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, Seyedali Mirjalili, and Mubarak Shah. 2024. Large language models: A comprehensive survey of its applications, challenges, limitations, and future prospects. https://doi.org/10.36227/techrxiv.23589741.v6
Hadj Taieb, Mohamed Ali, Torsten Zesch, and Mohamed Ben Aouicha. 2020. A survey of semantic relatedness evaluation datasets and procedures. *Artificial Intelligence Review*, 53. https://doi.org/10.1007/s10462-019-09796-3
Halawi, Guy, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1406–1414. https://doi.org/10.1145/2339530.2339751
Harispe, Sébastien, Sylvie Ranwez, Stefan Janaqi, and Jacky Montmain. 2015. Semantic similarity from natural language and ontology analysis. *Synthesis Lectures on Human Language Technologies*, 8(1):1–254. https://doi.org/10.1007/978-3-031-02156-5
Harispe, Sébastien, Sylvie Ranwez, Jacky Montmain, et al. 2022. *Semantic Similarity from Natural Language and Ontology Analysis*. Springer Nature.
Hecht, Brent, Samuel H. Carton, Mahmood Quaderi, Johannes Schöning, Martin Raubal, Darren Gergle, and Doug Downey. 2012. Explanatory semantic relatedness and explicit spatialization for exploratory

search. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 415–424. https://doi.org/10.1145/2348283.2348341

Hill, Felix, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695. https://doi.org/10.1162/COLI_a_00237

Huang, Eric H., Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 873–882.

Huang, Guangjian, Xingtu Zhu, Shahbaz Hassan Wasti, and Yuncheng Jiang. 2023. Multi-knowledge resources-based semantic similarity models with application for movie recommender system. *Artificial Intelligence Review*, 56(2):2151–2182. https://doi.org/10.1007/s10462-023-10573-6

Hussain, Muhammad Jawad, Heming Bai, Shahbaz Hassan Wasti, Guangjian Huang, and Yuncheng Jiang. 2023. Evaluating semantic similarity and relatedness between concepts by combining taxonomic and non-taxonomic semantic features of WordNet and Wikipedia. *Information Sciences*, 625673–699. https://doi.org/10.1016/j.ins.2023.01.007

Knoth, Nils, Antonia Tolzin, Andreas Janson, and Jan Marco Leimeister. 2024. AI literacy and its implications for prompt engineering strategies. *Computers and Education: Artificial Intelligence*, 6:100225. https://doi.org/10.1016/j.caeai.2024.100225

Krippendorff, Klaus. 2018. *Content Analysis: An Introduction to Its Methodology*. Sage Publications. https://doi.org/10.4135/9781071878781

Kruskal, William H. and W. Allen Wallis. 1952. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47(260):583–621. https://doi.org/10.1080/01621459.1952.10483441

Kulmanov, Maxat, Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. 2021. Semantic similarity and machine learning with ontologies. *Briefings in Bioinformatics*, 22(4):bbaa199. https://doi.org/10.1093/bib/bbaa199, PubMed: 33049044

Levy, Ran, Liat Ein Dor, Shay Hummel, Ruty Rinott, and Noam Slonim. 2015. Tr9856: A multi-word term relatedness benchmark. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 419–424. https://doi.org/10.3115/v1/P15-2069

Li, Peipei, Haixun Wang, Kenny Q. Zhu, Zhongyuan Wang, and Xindong Wu. 2013. Computing term similarity by large probabilistic isA knowledge. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, pages 1401–1410. https://doi.org/10.1145/2505515.2505567

Li, Xuechen, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. AlpacaEval: An automatic evaluator of instruction-following models.

Liu, Ying, Genevieve B. Melton, and Rui Zhang. 2024. Exploring large language models for acronym, symbol sense disambiguation, and semantic similarity and relatedness assessment. *AMIA Summits on Translational Science Proceedings*, 2024:324.

Luong, Minh Thang, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113.

Martinez-Gil, Jorge and José F. Aldana-Montes. 2013. Semantic similarity measurement using historical Google search patterns. *Information Systems Frontiers*, 15:399–410. https://doi.org/10.1007/s10796-012-9404-7

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, Tomas, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Miller, George A. and Walter G. Charles. 1991. Contextual correlates of semantic

similarity. *Language and Cognitive Processes*, 6(1):1–28. https://doi.org/10.1080/01690969108406936

Mistral AI. 2024a. Large enough. https://mistral.ai/news/mistral-large-2407. Accessed: March 10, 2025.

Mistral AI. 2024b. Mistral NeMo. https://mistral.ai/news/mistral-nemo. Accessed: May 8, 2024.

Mistral AI. 2024c. Un Ministral, des Ministraux. https://mistral.ai/news/ministraux. Accessed: May 8, 2024.

Mistral AI. 2025. Mistral Small 3. https://mistral.ai/news/mistral-small-3. Accessed: May 8, 2024.

Mistral AI large language models. 2024. Function calling. https://docs.mistral.ai/capabilities/function_calling/. Accessed: May 9, 2024.

Musker, Sam, Alex Duchnowski, Raphaël Millière, and Ellie Pavlick. 2024. Semantic structure-mapping in LLM and human analogical reasoning. https://doi.org/10.1016/j.jml.2025.104676

OpenAI. 2025a. API reference. https://platform.openai.com

OpenAI. 2025b. Enterprise privacy at OpenAI. https://openai.com/enterprise-privacy/

OpenAI API. 2024. Function calling. https://platform.openai.com/docs/guides/function-calling. Accessed: May 9, 2024.

Pakhomov, Serguei, Bridget McInnes, Terrence Adam, Ying Liu, Ted Pedersen, and Genevieve B. Melton. 2010. Semantic similarity and relatedness between clinical terms: An experimental study. In *AMIA Annual Symposium Proceedings*, volume 2010, page 572, American Medical Informatics Association.

Pakhomov, Serguei V. S., Greg Finley, Reed McEwan, Yan Wang, and Genevieve B. Melton. 2016. Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics (Oxford, England)*, 32(23):3635–3644. https://doi.org/10.1093/bioinformatics/btw529, PubMed: 27531100

Pedersen, Ted, Serguei V. S. Pakhomov, Siddharth Patwardhan, and Christopher G. Chute. 2007. Measures of semantic similarity and relatedness in the biomedical domain. *Journal of Biomedical Informatics*, 40(3):288–299. https://doi.org/10.1016/j.jbi.2006.06.004, PubMed: 16875881

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. https://doi.org/10.3115/v1/D14-1162

Petrakis, Euripides G. M., Giannis Varelas, Angelos Hliaoutakis, and Paraskevi Raftopoulou. 2006. Design and evaluation of semantic similarity measures for concepts stemming from the same or different ontologies. In *4th Workshop on Multimedia Semantics (WMS'06)*, pages 44–52.

Pilehvar, Mohammad Taher and Roberto Navigli. 2015. From senses to texts: An all-in-one graph-based approach for measuring semantic similarity. *Artificial Intelligence*, 228:95–128. https://doi.org/10.1016/j.artint.2015.07.005

Pirró, Giuseppe. 2012. Reword: Semantic relatedness in the Web of data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 129–135. https://doi.org/10.1609/aaai.v26i1.8107

Pirró, Giuseppe and Nuno Seco. 2008. Design, implementation and evaluation of a new semantic similarity metric combining features and intrinsic information content. In *On the Move to Meaningful Internet Systems: OTM 2008: OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008, Proceedings, Part II*, pages 1271–1288. https://doi.org/10.1007/978-3-540-88873-4_25

Querido, Andreia, Rita Carvalho, João Rodrigues, Marcos Garcia, João Silva, Catarina Correia, Nuno Rendeiro, Rita Valadas Pereira, Marisa Campos, and António Branco. 2017. LX-LR4DistSemEval: A collection of language resources for the evaluation of distributional semantic models of Portuguese. *Revista da Associação Portuguesa de Linguística*, 1(3):265–283. https://doi.org/10.26334/2183-9077/rapln3ano2017a15

Radinsky, Kira, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: Computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th International Conference on World Wide Web*, pages 337–346. https://doi.org/10.1145/1963405.1963455

Richie, Russell, Sachin Grover, and Fuchiang Rich Tsui. 2022. Inter-annotator agreement is not the ceiling of machine learning performance: Evidence from a comprehensive set of simulations. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 275–284. https://doi.org/10.18653/v1/2022.bionlp-1.26

Rubenstein, Herbert and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633. https://doi.org/10.1145/365628.365657

Sahlgren, Magnus. 2008. The distributional hypothesis. *Italian Journal of Linguistics*, 20:33–53.

Sahoo, Pranab, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*. https://doi.org/10.48550/arXiv.2402.07927

Sainz, Oscar, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023a. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787. https://doi.org/10.18653/v1/2023.findings-emnlp.722

Sainz, Oscar, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, and Eneko Agirre. 2023b. Did ChatGPT cheat on your test? https://hitz-zentroa.github.io/lm-contamination/blog/

Sainz, Oscar, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, and Eneko Agirre. 2023c. LM Contamination Index. https://hitz-zentroa.github.io/lm-contamination/. Accessed: April 7, 2025.

Salle, Alexandre, Marco Idiart, and Aline Villavicencio. 2016. Matrix factorization using window sampling and negative sampling for improved word representations. In *The 54th Annual Meeting of the Association for Computational Linguistics*, page 419. https://doi.org/10.18653/v1/P16-2068

Sanderson, Mark. 1994. Word sense disambiguation and information retrieval. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 142–151. https://doi.org/10.1007/978-1-4471-2099-5_15

Shah, Chirag. 2024. From prompt engineering to prompt science with human in the loop. *arXiv preprint arXiv:2401.04122*. https://doi.org/10.1145/3709599

Shapiro, Samuel Sanford and Martin B. Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3–4):591–611. https://doi.org/10.1093/biomet/52.3-4.591

Shaqiri, Mirlinda, Teuta Iljazi, Lazim Kamberi, and Rushadije Ramani-Halili. 2023. Differences between the correlation coefficients Pearson, Kendall and Spearman. *Journal of Natural Sciences and Mathematics of UT*, 8(15–16):392–397.

Silberer, Carina and Mirella Lapata. 2014. Learning grounded meaning representations with autoencoders. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 721–732. https://doi.org/10.3115/v1/P14-1068

Slimani, Thabet. 2013. Description and evaluation of semantic similarity measures approaches. *International Journal of Computer Applications*, 80(10):25–33. https://doi.org/10.5120/13897-1851

Speer, Robyn, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31. https://doi.org/10.1609/aaai.v31i1.11164

Szumlanski, Sean, Fernando Gomez, and Valerie K. Sims. 2013. A new set of norms for semantic relatedness measures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 890–895.

Trott, Sean. 2024. Can large language models help augment English psycholinguistic datasets? *Behavior Research Methods*, 56(6):6082–6100. https://doi.org/10.3758/s13428-024-02337-z, PubMed: 38261264

Vulić, Ivan, Simon Baker, Edoardo Maria Ponti, Ulla Petti, Ira Leviant, Kelly Wing, Olga Majewska, Eden Bar, Matt Malone, Thierry Poibeau, Roi Reichart, and Anna Korhonen. 2021. Multi-SimLex: A large-scale evaluation of multilingual and crosslingual lexical semantic similarity. *Computational Linguistics*, 46(4):847–897. https://doi.org/10.1162/coli_a_00391

W3Techs. 2025. Usage statistics and market share of content languages for Websites, April 2025. https://w3techs.com/technologies/overview/content_language. Accessed: April 7, 2025.

Yang, Dongqiang and David Powers. 2006. Verb similarity on the taxonomy of WordNet. In *The Third International WordNet Conference: GWC 2006*.

Zhang, Ziqi, Anna Lisa Gentile, and Fabio Ciravegna. 2013. Recent advances in methods of lexical semantic relatedness—A survey. *Natural Language Engineering*, 19(4):411–479. https://doi.org/10.1017/S1351324912000125

Ziegler, C., K. Simon, and G. Lausen. 2006. Automatic computation of semantic proximity using taxonomic knowledge. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, pages 465–474. https://doi.org/10.1145/1183614.1183682