

# A Game Interface to Study Semantic Grounding in Text-Based Models

Timothee Mickus  
ATILF  
CNRS, Université de Lorraine  
Nancy, France  
tmickus@atilf.fr

Mathieu Constant  
ATILF  
CNRS, Université de Lorraine  
Nancy, France  
mconstant@atilf.fr

Denis Paperno  
Linguistics Department  
Utrecht University  
Utrecht, Netherlands  
d.paperno@uu.nl

**Abstract**—Can language models learn grounded representations from text distribution alone? This question is both central and recurrent in natural language processing; authors generally agree that grounding requires more than textual distribution. We propose to experimentally test this claim: if any two words have different meanings and yet cannot be distinguished from distribution alone, then grounding is out of the reach of text-based models. To that end, we present early work on an online game for the collection of human judgments on the distributional similarity of word pairs in five languages. We further report early results of our data collection campaign.

**Index Terms**—Semantic Grounding, Distributional Semantics, Gamification

## I. INTRODUCTION

Semantic grounding is a central question in Natural Language Processing. Studies on grounding try to see how words and sentences can be linked to real-world objects. This question has recently gained prominence through Bender & Koller [1], who specifically set out to disprove that large language models like BERT [2] or GPT-3 [3] are capable of building semantically coherent representations. The thought experiment of Bender & Koller echoes previous works such as Harnad’s Symbol Grounding Problem [4] or Searle’s Chinese Room Argument [5].

The prominence of thought experiments, rather than experimental procedures, shows how elusive this question is. That is not to say that no experimental approach has been adopted: multimodal evaluation benchmarks exist [6], and authors have proposed semantically grounded models of language [7]–[9]. Yet most researchers seemingly adopt without discussion the position that something more than text is required to coherently describe the world. Here, we propose an experiment to test this claim. If there are pairs of words that humans cannot distinguish from distribution alone, then in principle text-based models cannot consistently and systematically use words in semantically appropriate contexts—because even perfectly grounded semantic representations as we expect to find in humans do not suffice to solve the task. This can also be seen

This work was supported by a public grant overseen by the French National Research Agency (ANR) as part of the “Investissements d’Avenir” program: Idex Lorraine Université d’Excellence (reference: ANR-15-IDEX-0004).

as an investigation of the limits of the distributional hypothesis [10]: we are looking into documenting cases where distribution is not enough to infer meaning.

We first study how to formulate a task that will be natural both for humans and distributional semantics models in Section II. We then detail our game-based data collection campaign in Section III. We present early results in the form of usage statistics in Section IV. Lastly, we give some concluding remarks in Section V.

## II. DISTRIBUTIONAL MODELS OF WORD MEANING

Most modern language models and embedding architectures used in NLP can be linked to Zellig Harris’ distributional hypothesis [10]. These models, while they all differ in architecture, share the common feature that they can exhibit which of two words they prefer in a specific context, or formally:

$$\Pr(t_1|c) > \Pr(t_2|c) \quad (1)$$

Some distributional models directly model  $P(t|c)$ . In particular, we can quote sequence denoising objectives [2], [11] or multinomial classification objectives like CBOW [12]. Other architectures like skip-gram [12] model  $P(c|\cdot)$ . In such cases, finding some equation corresponding to eq. (1) can be done by a straightforward application of Bayes’ rule:

$$\begin{aligned} P(t_1|c) > P(t_2|c) &= \frac{P(c|t_1)P(t_1)}{P(c)} > \frac{P(c|t_2)P(t_2)}{P(c)} \\ &= P(c|t_1)P(t_1) > P(c|t_2)P(t_2) \end{aligned} \quad (2)$$

A third family of architectures that can be linked to this framework concern negative sampling approaches [13], [14], which are trained to estimate  $P(t \in c)$ . To arrive exactly at the previous eq. (1), we can re-normalize this probability distribution with respect to the entire vocabulary:

$$\Pr(t_i|c) = \frac{1}{\sum_{t_j \in V} P(t_j \in c)} \cdot P(t_i \in c)$$

The above does sum to 1 over the full probability space of the vocabulary  $V$ , is defined with respect to the context  $c$ . As  $\frac{1}{\sum_{t_j \in V} P(t_j \in c)}$  is constant for a given context  $c$ , negative sampling models therefore yield the following formulation:

$$\Pr(t_1|c) > \Pr(t_2|c) = P(t_1 \in c) > P(t_2 \in c) \quad (3)$$

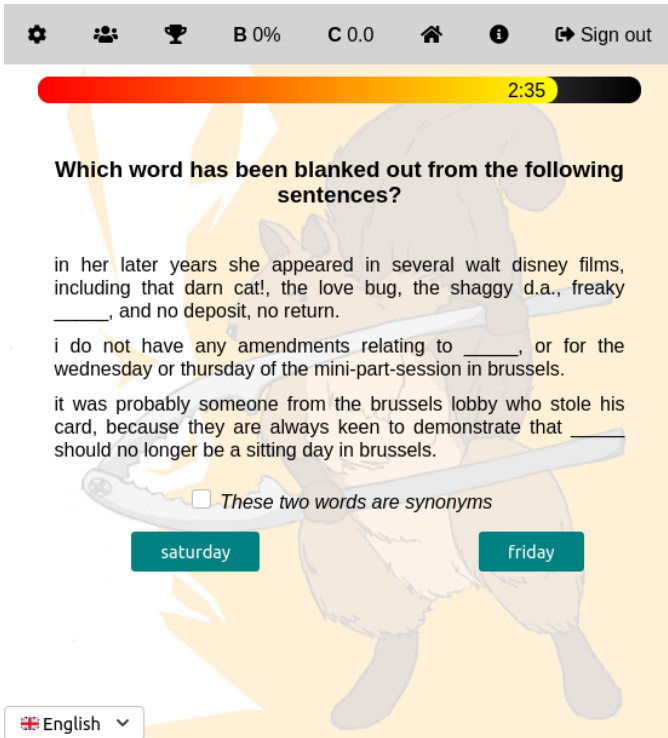


Fig. 1: Game Annotation Interface

Lastly, generative models like [3], [15] yield the probability of  $t_i$  being the next token:  $P(t_i|c_{1..i-1})$ . Hence, works like [16] derive the comparison given the full context  $c_{1..n}$  as:

$$P(c_n|c_{1..i-1}, t_1, c_{i+1..n-1}) > P(c_n|c_{1..i-1}, t_2, c_{i+1..n-1}) \quad (4)$$

This Context-Specific Term Preference (CSTP) therefore allows us to evaluate all these models using a united framework, specifically by answering which of two words is preferable in a given context. This entails that we can compare various architectures on this task. Crucially, the task is also easily understood by humans, and we therefore can collect human judgments to compare distributional models to.

### III. GAME INTERFACE

We propose to collect such human judgments through a game interface, as displayed in Figure 1. This game is available online at <https://blankcrack.atilf.fr/>.<sup>1</sup>

There are two aspects in which human knowledge can be useful to our enterprise:

- How hard is it to distinguish terms from contexts alone?
- Are there terms indistinguishable from context alone?

These two related questions lead to two distinct series of data to collect. To answer (a), we would need to run the CSTP task, but substituting distributional models with human annotators. To answer (b), we would instead require human participants to suggest word pairs that they expect to be

<sup>1</sup>Code for the interface will be made public at a later date.

difficult to distinguish. These two collection procedures lead us to an adversarial data collection project, where annotators can play either of two roles: proposing word pairs to answer (b) (henceforth (b)-annotators), or distinguishing word pairs proposed in (b) to answer (a) (likewise (a)-annotators).

Once the data are collected, we will be able to compare the performance of language models and distributional models on the very same task (as outlined in Section II). Pairs that humans find challenging should be difficult for neural models as well—as long as contexts are not attested in these models’ training data. We expect these collected data to constitute a strong human benchmark for distributional models.

#### A. Sentence Corpora

As we are interested in establishing a widely applicable benchmark, we collect data for multiple languages: English, French, Italian, Spanish and Russian.<sup>2</sup> The first element required for our game is a list of distributional contexts, or sentences. We further wish our data to be broadly comparable across languages: we therefore select a comparable number of sentences from comparable but varied corpora. We chose to select 4M sentences per language, equally drawn from four genres of corpora: Wikipedia dumps, books corpora (Gutenberg Project, WikiSource, LiberLiber.it), parliamentary debates (EuroParl [17] & UN Corpus [18]) and subtitles (OpenSubtitles [19]).

#### B. Initial Word Pairs Set

The second type of data we require is a set of word pairs to bootstrap our data collection process, so that both (a)-annotators and (b)-annotators can immediately start. To do so, we consider two strategies. The first “manual” strategy consists in manually constructing pairs that one initially expects to be challenging, such as months, days of the week, numbers (cardinal and ordinal) and colors. Any pair of terms from one of these series can constitute a word pair to annotate. The second strategy, which we call “w2v-based”, consists in automatically discovering distributionally similar items given our corpus of sentence. We train distinct word2vec models for each of our five language-specific corpora.<sup>3</sup> We randomly sample 1M word pairs, and narrow down to the 250 items whose vectors maximize cosine similarity.<sup>4</sup>

#### C. Annotation Items

From these word pairs and sentences, we can then automatically construct annotation items. We present each (a)-annotator with two words  $w_a$  and  $w_b$  from a given word pair  $\langle w_a, w_b \rangle$ , and  $k$  sentences<sup>5</sup> randomly selected such that all sentences contain the target  $w_a$  and none contains a word with the same

<sup>2</sup>These five languages were chosen on criteria of high data availability. We intend to facilitate adding new languages to the interface in future releases.

<sup>3</sup>We select hyperparameters with Bayesian optimization [20], using performance on a formal analogy dataset as the objective to maximize.

<sup>4</sup>We considered alternative strategies, such as using ontologies like WordNet to select initial word pairs, but distributional similarity proved the most efficient during a short pilot study.

<sup>5</sup>(a)-annotators can set  $k \in \{1, 3, 5\}$ ; by default,  $k = 5$ .

word stem as the foil  $w_b$ . We replace all occurrences of  $w_a$  by a blank token before presenting them to the (a)-annotator. The annotator is then tasked with retrieving which of the target  $w_a$  or the foil  $w_b$  corresponds to these blank tokens. Word pairs  $\langle w_a, w_b \rangle$  can correspond either to our initial set of word pairs, or to items proposed by (b)-annotators.

#### D. Player Engagement

At its core, our game is score-based, with two distinct scores per user corresponding to performances as (a)-annotators and (b)-annotators. The (b)-annotator score corresponds to the success rate (as a percentage) of the user’s proposed word pairs, i.e., how often (a)-annotators failed to solve riddles constructed using the (b)-annotator’s word pairs, and selected the foil  $w_b$  instead of the target  $w_a$ . The (a)-annotator score is a running tally of points: (a)-annotator get between 0.1 and 3 points per correctly solved annotation item (where the (a)-annotator selected the target  $w_a$ , rather than the foil  $w_b$ ), depending on whether the item was solved under 3 minutes, was based on a known difficult pair, or whether the (a)-annotator had set a lower number  $k$  of example sentences.

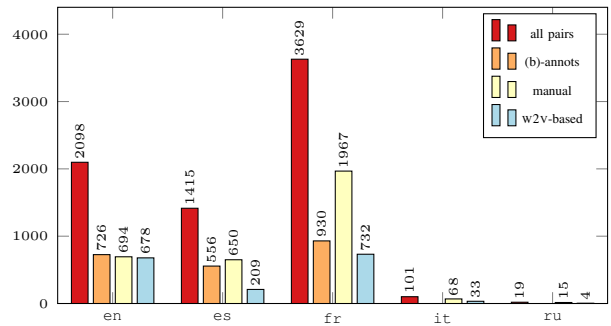
The possibility to set the number  $k$  of sentences per riddle is presented in-game as a difficulty level setting. Aside from this setting, we further implement several mechanisms to attempt to keep players engaged. First, we include a competition mode, whereby users compete against one another; this competition mode is based on a “friends list”. Second, we ensure that word pairs newly suggested by (b)-annotators get presented to (a)-annotators in priority, so that (b)-annotators receive feedback as early as possible. Third, we also include some materials to share on social media, e.g., when (a)-annotators successfully retrieve the blanked-out word in their annotations multiple times in a row, or at the end of a competition session. Thus far, sharing on social media and competitions have not been used much often by our users.

We also note that users tend to connect only once. One explanation may lie in that “manual” word pairs from our initial set (cf. Subsection III-B) are felt to be very hard to solve. We are currently investigating mechanisms to combat this trend such as high-score leader-boards displaying username, language and score for top players; our intuition is that it may motivate players to return to the platform to ensure they still appear on the leader-board. Another possibility would be to provide users with a way to opt-out of these word pairs.

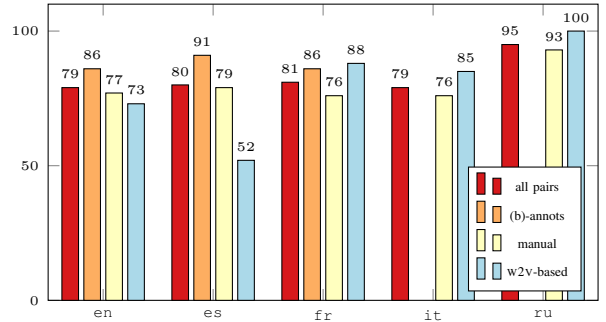
## IV. USAGE STATISTICS

Thus far, we have collected slightly over 7000 annotations in the first month of widespread advertisement. We take this as an encouraging response, although improvements in the interface and game engagement mechanisms can be made.

A more precise breakdown is displayed in Figure 2. Subfigure 2a displays the number of annotations we collected, whereas Subfigure 2b provides an estimate of the success rate of (a)-annotators. Each subfigure is broken down further according to the type of word pairs: “manually constructed”



(a) Counts of (a)-annotations by language.



(b) Overall success rate of (a)-annotators (in %)<sup>6</sup>

Fig. 2: Overview of collected (a)-annotations

and “w2v-based” both come from our initial set (see Subsection III-B) whereas “(b)-annots” have been proposed by (b)-annotators. The first obvious trend we can observe from Subfigure 2a is that the number of Italian and Russian annotations is severely lacking, with respect to English, French and Russian. For these two languages, word pairs from (b)-annotators have yet to be annotated. This low number of annotations indicates that percentages from Subfigure 2b are most likely off. This is a consequence of our current advertisement: we have not yet reached out to Italian and Russian audiences.

On average, 80% of all items are correctly annotated by (a)-annotators. This suggests the task is in itself rather simple, which would entail that in principle, most words can be distinguished from distribution alone. On the other hand, this also entails that humans do not invariably solve the task with perfect accuracy, and we intend to strongly focus on incorrect (a)-annotations in future research. Likewise, note that a random baseline would already entail a rate of 50%.

If we compare (a)-annotator success rates on word pairs from (b)-annotators and on our two strategies for constructing of word pairs (Subfigure 2b), we see that the latter are generally found more difficult than the former, corroborating users’ feedback (see Subsection III-D). Another interesting trend to note is that Spanish, French and English annotators seem to differ from one another: word pairs from Spanish (b)-annotators are found to be generally easier than what we

<sup>6</sup>No data collected for (b)-annotators word pairs in Italian and Russian. Russian and Italian percentages are unreliable, owing to the too few items collected.

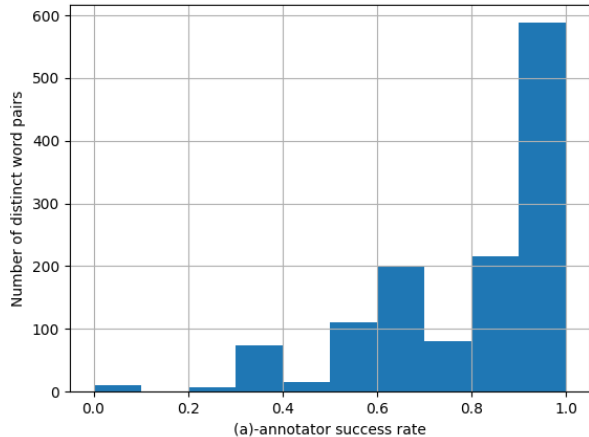


Fig. 3: Number of distinct word pairs per (a)-annotators success rate, all languages combined.

observe in other languages, whereas “manual” word pairs are found to be harder across the board, but with varying degrees: Spanish pairs are found to be easier than English pairs, which in turn are easier than French pairs, while our “w2v-based” strategy yields the opposite ordering. At this stage of our experiments, we cannot rule out an effect of sample selection, especially given the surprisingly low, barely above chance (a)-annotator success rate for Spanish “w2v-based” word pairs.

Lastly, Figure 3 summarizes the number of distinct word pairs for various bins of (a)-annotators success rate. To mitigate noise, we ignore word pairs that have been annotated less than three times: for such pairs, the associated average success rate can only be either 0, 1 or 0.5 if it has been seen twice, which would inflate values at these specific peaks. We can see that the vast majority of our 1656 distinct word pairs is found to be rather simple to distinguish: 588 pairs are solved correctly 9 times out of ten or more, and 804 are solved correctly 8 times out of ten or more. This leaves us with a significant number of word pairs with low scores, some around chance level, which we plan to focus on in future experiments. It is important to note that this illustration is not entirely reliable: the average number of (a)-annotations per word pair is 4.6, meaning that (a)-annotators success rates are often not statistically reliable. In any event, more data collection and curating is necessary.

## V. CONCLUSIONS

We have presented an online game platform for collecting human judgments on the distributional hypothesis. We have outlined theoretical motivations based on a formal approach subsuming many different distributional models (Section II). We have discussed the implementation of our games and the mechanism we set in place to attract users (Section III). We detailed early results in the form usage statistics (Section IV).

While we are still in early stages of development and advertisement, the data we have already collected suggest that

while the majority of word pairs can be reliably distinguished, some pairs may indeed prove to be challenging. To give an anecdotal overview of what such pairs could be, we can highlight that in all the following word pairs the target has been correctly identified only 2 out of 6 times: Spanish “cilantro” and “cebollino”, French “chenapan” and “polisson”, English “hyena” and “jackal”. These encouraging results strongly suggest our experimental design has intrinsic worth: for instance, it has already reliably demonstrated how user-provided word pairs are generally less challenging than manually constructed ones.

On the other hand, further data collection is required before any strong conclusion can be reached. In future work, we intend to focus more keenly on the data we collect to identify hard-to-distinguish word pairs, as well as study the impact of the various strategies used to create word pairs. We also plan to compare human behavior to that of large language models and distributional models, both semantically grounded and ungrounded.

## ACKNOWLEDGMENT

We thank Maria Copot, Hermès Martinez, Elena del Olmo, Tomara Gotkova and Aurore Coince for helping translating and advertising our game, Cyril Pestel for his help in deploying the game online, and Karèn Fort, Mathieu Lafourcade and three anonymous reviewers for suggestions for improvement. This work was supported by a public grant overseen by the French National Research Agency (ANR) as part of the “Investissements d’Avenir” program: *I dex Lorraine Université d’Excellence* (reference: ANR-15-IDEX-0004).

## REFERENCES

- [1] E. Bender and A. Koller, “Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data,” *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5185–5198, July 2020.
- [2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1 (Long and Short Papers), pp. 4171–4186, June 2019.
- [3] T. Brown et al., “Language Models are Few-Shot Learners,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020.
- [4] S. Harnad, “The symbol grounding problem,” *Physica D*, vol. 42, pp. 335–346, 1990.
- [5] J. Searle, “Minds, Brains and Programs,” *Behavioral and Brain Sciences*, vol. 3, pp. 417–57, 1980.
- [6] S. Antol, A. Agrawal et al. “VQA: Visual Question Answering,” *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2425–2433, 2015.
- [7] E. Bruni, G. Boleda, M. Baroni, and N.-K. Tran, “Distributional Semantics in Technicolor,” *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, vol. 1: Long Papers, pp. 136–145, July 2012.
- [8] D. Kiela and S. Clark “Learning Neural Audio Embeddings for Grounding Semantics in Auditory Perception,” *J. Artif. Int. Res.*, vol. 60.1, September 2017.
- [9] D. Kiela, L. Bulat, and S. Clark, “Grounding Semantics in Olfactory Perception”, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, July 2015.
- [10] Z. Harris, “Distributional structure,” *Word*, vol. 10, pp. 146–162, 1954.

- [11] M. Lewis et al., “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension,” Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7871–7880, June 2020.
- [12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” unpublished, 2013.
- [13] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching Word Vectors with Subword Information,” Transactions of the Association for Computational Linguistics, vol. 5, pp. 135–146, 2017.
- [14] K. Clark, M.-T. Luong, Q. Le, and C. Manning, “Pre-Training Transformers as Energy-Based Cloze Models,” Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 285–294, November 2020.
- [15] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, “A neural probabilistic language model,” J. Mach. Learn. Res., vol. 3, pp. 1137–1155, 2003.
- [16] T. Linzen, E. Dupoux, and Y. Goldberg, “Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies,” Transactions of the Association for Computational Linguistics, vol. 4, pp. 521–535, 2016.
- [17] P. Koehn, “Europarl: A Parallel Corpus for Statistical Machine Translation,” Proceedings of the tenth Machine Translation Summit, pp. 79–86, 2005.
- [18] M. Ziemski, M. Junczys-Dowmunt, and B. Pouliquen, “The United Nations Parallel Corpus”, Proceedings of the 10th International Conference on Language Resources and Evaluation, May 2016.
- [19] P. Lison and J. Tiedemann, “OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles,” Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC), May 2016.
- [20] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” Advances in Neural Information Processing Systems, pp. 2951–2959, 2012.