FOOL ME IF YOU CAN! An Adversarial Dataset to Investigate the Robustness of LMs in Word Sense Disambiguation

Anonymous ACL submission

Abstract

Word sense disambiguation (WSD) is a key task in natural language processing and lexical semantics. Pre-trained language models with contextualized word embeddings have significantly improved performance in regular WSD tasks. However, these models still struggle with recognizing semantic boundaries and often misclassify homonyms in adversarial context. Therefore, we propose **FOOL**: **FO**ur-fold **O**bscure **L**exical, a new coarse-grained WSD dataset, which includes four different test sets designed to assess the robustness of language models in WSD tasks. Two sets feature typical WSD scenarios, while the other two include sentences with opposing contexts to challenge the models further.

We tested two types of models on the proposed dataset: models with encoders, such as the BERT and T5 series of varying sizes by probing their embeddings, and state-of-the-art large decoder models like GPT-4o and the LlaMA3 family, using zero shot prompting. Across different state-of-the-art language models, we observed a decrease in performance in the latter two sets compared to the first two, with some models being affected more than others. We show interesting findings where small models like T5-large and BERT-large performed better than GPT-4o on Set 3 of the dataset. This indicates that, despite excelling in regular WSD tasks, these models still struggle to correctly disambiguate homonyms in artificial (Set 3) or realistic adversarial contexts (Set 4).

1 Introduction

The task of word sense disambiguation (WSD) is a fundamental challenge in natural language processing (NLP). Homonyms, which are formally identical words with completely independent meanings (Kempson, 1977, p. 80), present a challenge in tasks like machine translation, text annotation, and question answering (Agirre and Edmonds, 2007). In order to comprehend the intended meaning of homonyms, it is necessary to consider the context, in which they are used. Consequently, the accurate disambiguation of homonyms provides evidence of the model's comprehension of the context and, in turn, of language.

Contextualized language models, such as BERT (Devlin et al., 2019), produce word embeddings that reflect the word's meaning based on its context (Wiedemann et al., 2019). This has led to significant improvements in WSD performance in both fine-grained or coarse-grained WSD (Wiedemann et al., 2019; Reif et al., 2019; Loureiro et al., 2021). While fine-grained WSD addresses the nuanced senses a word can have, coarse-grained WSD focuses on broader, unrelated word meanings (Haber and Poesio, 2024). The emergence of context-based language models suggests that the challenge of regular WSD has largely been resolved. However, it is still unclear if these models can understand context well enough to disambiguate homonyms effectively. Let us consider the following sentence:

"I eat an apple while holding my iPhone."

For a human it is clear that "apple" refers to the fruit, and not the technology company. The question remains whether today's language models can differentiate these senses in this adversarial context.

Even though there are many existing WSD benchmarks, such as the Unified Evaluation Framework by Raganato et al. (2017) or CoarseWSD-20 by Loureiro et al. (2021), none of them considers the distinction between different types of context nor the use of opposing context in the sentences. For this purpose we introduce FOOL, a coarse-grained WSD dataset that differentiates between four distinct categories of context changes. The dataset includes one training set and four test sets

	Senses	Example Sentence for apple
Train Set	apple_apple_inc	"the ipod is first introduced by apple."
	apple_fruit	"the surrounding area produces 20% of patagonia's apple and 28% of its pear .
Set 1	apple_apple_inc	"I downloaded the latest app from the Apple App Store."
	apple_fruit	"An apple is a refreshing snack on a hot summer day."
Set 2	apple_apple_inc	"I downloaded the latest app from the innovative Apple App Store."
	apple_fruit	"A crisp apple is a refreshing snack on a hot summer day."
Set 3	apple_apple_inc	"I downloaded the latest app from the crisp Apple App Store."
	apple_fruit	"An innovative apple is a refreshing snack on a hot summer day."
Set 4	apple_apple_inc	"The cafeteria at Apple Headquarters serves delicious pie."
	apple_fruit	"Holding an apple, I scrolled through news about rival tech companies."

Table 1: Example sentences from the dataset for the word apple.

as illustrated in Table 1.[1] The first two test sets provide sentences for regular WSD, while the other two contain sentences with additional context that opposes the anticipated meaning of the homonym. This structure allows for the testing of state-of-the-art (SOTA) language models in both regular homonym disambiguation settings and adversarial context settings. Therefore, this dataset can be used to investigate the robustness of language models to different context changes.

We investigated two types of language models: models with encoders, from which we probed their embeddings using kNN algorithm, and state-of-the-art models that we prompted to classify the target word into one of two possible meanings. Our findings indicate that current SOTA models struggle to accurately disambiguate coarse-grained homonyms when adversarial contexts are added. We observed a performance decrease across all models when comparing results from Set 1 with those from Sets 3 and 4.

In models containing encoders, this effect is most significant in smaller models like BERT-base and T5-base, and less significant in larger models like T5-FLAN-xxl. Conversely, advanced and larger language models such as GPT-3.5 Turbo and LlaMA3-70b show a dramatic performance decline when faced with adversarial context changes, with performance drops of 25.6% and 10.4%, respectively in Set 4 compared to Set 1. However, models like GPT-4o exhibit more robustness against realistic opposing context examples (Set 4), with a performance drop of only around 4%, but more vulnerability for adding adversarial adjective (Set 3). Additionally, our findings suggest that models that contain encoder, such as those from the BERT or T5 family, tend to perform better in these tasks, specifically in Set 3. For instance, the BERT-large model with 340 million parameters outperformed the LlaMA3-8b model, which has around 8 billion parameters on both adversarial tasks (Set 3 and 4). In addition, T5-large and BERT-large performed better than GPT-4o on Set 3 of the dataset. To conclude, our contributions can be summarized as follows:

- We introduce FOOL, a new coarse-grained WSD dataset that features various test sets with added adversarial context to assess the robustness of pre-trained language models
- We perform an extensive analysis on various SOTA language models in WSD with experiments on our proposed dataset
- We show that current state-of-the-art language models are prone to misclassification when faced with adversarial and opposing realistic context

2 Related Work

Word Sense Disambiguation (WSD) is a well-studied task in natural language processing, focusing on fine-grained polysemy disambiguation. The majority of standard WSD benchmarks, such as the Unified Evaluation Framework by Raganato et al. (2017), heavily rely on WordNet (Miller, 1994). This dependence on WordNet, known for its fine-grained classification, poses a challenge even for humans to distinguish all possible senses. To tackle this issue, Loureiro et al. (2021) introduced the dataset "CoarseWSD-20", which extracts sentences from Wikipedia articles to create a coarse-grained sense inventory WSD dataset.

The performance of pre-trained language models has been tested on both fine-grained and coarse-grained datasets. Especially, BERT (Devlin et al., 2019) achieved overall good results with over 94%

[1] The full dataset can be downloaded from: https://drive.google.com/file/d/1WOUml_GGUrUXvKMt3ywBjK-lI_lsEydi/view?usp=sharing

accuracy in coarse-grained WSD (Loureiro et al., 2021). For example, Du et al. (2019) fine-tuned BERT on a WSD task and tested it on a variety of different fine-grained WSD benchmarks (Edmonds and Cotton, 2001; Moro and Navigli, 2015; Navigli et al., 2013; Pradhan et al., 2007; Snyder and Palmer, 2004), achieving promising results with accuracies ranging from 74% to 78%.

Additionally, without fine-tuning Wiedemann et al. (2019) and Reif et al. (2019) showed that BERT can effectively perform fine-grained WSD by combining its contextualized word embeddings with a kNN classification algorithm. Moreover, Loureiro et al. (2021) employed a kNN BERT classifier and reported human-like performance on their coarse-grained noun WSD dataset, with over 94% accuracy. More recently, Proietti et al. (2024) tested different BERT-based models, including BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) on coarse-grained WSD. They clustered WordNet senses to match coarse-grained homonym sense distinction and found that BERT's accuracy is as high as 95%.

In our work, we conduct an extensive analysis on two types of models. On the one hand, we tested models that include encoder, such as the BERT and T5 family (Raffel et al., 2020), by probing their word embeddings. On the other hand, we analyzed state-of-the-art language models including GPT-3.5 Turbo (OpenAI, 2022), GPT-4 Turbo (OpenAI, 2023) , and GPT-4o (OpenAI, 2024), LlAMA3-8b, LlAMA3-70b (Meta, 2024) and Mixtral-8x7b (Jiang et al., 2024) using prompts in zero shot settings. While Kocoń et al. (2023) investigated GPT-3.5's performance on WSD among other tasks, our work significantly differs in that we have created a new coarse-grained adversarial dataset and tested various models from different families. To the best of our knowledge, this paper is the first to conduct an extensive analysis comparing models like BERT and T5 with models like GPT and Llama on adversarial WSD tasks.

Furthermore, it is evident that there is no existing dataset that aligns with the one proposed in this paper. Despite this, there have been some attempts to test models on adversarial sentences. For example, Emelin et al. (2020) considered adversarial attacks in WSD. They changed adjectives in sentences in front of homonyms and checked the performance in a machine translation task. These changes lead to translation errors in LSTM (Luong et al., 2015), Transformer (Vaswani et al., 2017) and ConvS2S (Gehring et al., 2017). Inspired by Emelin et al. (2020) approach, we adopted the idea of modifying adjectives in order to test the resilience of more recent pre-trained language models based on their contextualized word embeddings.

Moreover, Reif et al. (2019) incorporated opposing context words in their study. In their paper, the authors analyzed the performance of pre-trained language models, primarily BERT, on SemCor (Miller et al., 1993), a fine-grained sense dataset. While they succeeded in this task, they also combined two sentences with distinct meanings of a homonym to create sentences with opposing contexts. Thereby they found a higher number of classification errors than in normal conditions. This test was done using fine-grained senses of words. Although this represents a promising initial step, there is a need to further extend this idea. We analyze coarse-grained WSD performances of different state-of-the-art models beyond BERT and have developed an entire human-made test set to evaluate our approach.

3 Dataset

In this section, we introduce our dataset FOOL, a coarse-grained WSD dataset that is designed to differentiate between four different categories of context changes. This design allows us to test both regular homonym disambiguation settings and adversarial context settings. Therefore, this dataset serves as a tool to evaluate the robustness of large language models against different context changes and their ability to discern between various coarse-grained homonym senses.

3.1 Dataset Split

In order to assess the efficacy of distinct pre-trained models across different levels of contextual complexity, four different sets of sentences were created, with an additional set designated as the training set. Each set is associated with a specific context and serves a unique purpose. Examples from these sets are illustrated in Table 1.

- **Train Set:** The training dataset consists of sentences that use the homonym in its anticipated context. This ensures a solid foundation for fitting the classification algorithm.
- **Set 1:** Similar to the training set, the homonyms are used in its anticipated context. This set serves as the baseline for testing regular WSD performances.

- **Set 2:** This set extends the sentences from Set 1 by adding an adjective directly before the homonym, which aligns with the anticipated meaning of the homonym in that sentence.

- **Set 3:** This set modifies the sentences from Set 2 by changing the adjectives preceding the homonyms. The new adjectives are typically associated with the opposite meaning of the homonym, introducing an artificial adversarial context.

- **Set 4:** This set includes sentences that have been specifically crafted with realistic context that opposes the anticipated meaning of the homonym, further challenging the models' disambiguation capabilities.

While the context provided in Sets 1 and 2 is designed to facilitate the models' ability to distinguish between homonym senses, Sets 3 and 4 include adversarial examples to challenge the models. The dataset allows testing models in regular WSD with coarse-grained homonym senses and assessing their response to adversarial examples. This dual approach tests not only basic disambiguation capabilities but also the resilience of models under more complex and potentially confusing linguistic scenarios.

3.2 Statistics

Table 2 shows an overview of all words used in the dataset, which comprises 20 homonyms in total. Each homonym is confined to exactly two broad word senses that are unrelated to each other. It is crucial that in both senses, the word remains a noun, which is essential for the application of adjectives in Sets 2 and 3. The distribution of sentences per word sense is well balanced across each set. In Set 1 to 3 the number of sentences ranges from 40 to 60 sentences per word sense in each set. Set 4 consists of 25 to 30 sentences per word sense, reflecting the higher complexity and cost associated with creating these sentences. The training data includes 20 to 40 sentences per word sense. This structured approach ensures that each sense is adequately represented and tested throughout the dataset. Table 5 in the Appendix shows the full statistics of the dataset with the number of sentences for every word sense in each set is shown.

3.3 Data Collection

The construction of the dataset is mostly done by manually creating and revising sentences that are suitable for the desired sense of the homonym. Notably, Set 4 is entirely crafted by hand to include homonyms in their anticipated use along with opposing context—a task that cannot be automated using tools like ChatGPT or sourced from existing literature. This manual approach ensures that the sentences are fluent and meaningful, fulfilling their intended purpose in the dataset.

For our Training Set, we utilized the existing coarse-grained dataset "CoarseWSD-20" by Loureiro et al. (2021). Where there was an overlap of words between our dataset and CoarseWSD-20, we selected the most appropriate sentences for inclusion in our dataset. However, for words not covered by CoarseWSD-20, we sourced example sentences from platforms like Word Hippo (Kat IP Pty Ltd) and YourDictionary (LoveToKnow Media), which were then adapted to meet our criteria. Additionally, Set 1 was generated using both examples from these platforms and sentences created with ChatGPT (OpenAI, 2022) and GPT 4 (OpenAI, 2023) . Nevertheless, the adjectives in Set 2 and 3 are manually added by humans to ensure a diverse and contextually appropriate use of adjectives, tailored to our specific needs. Furthermore, all labels for the above mentioned sentences were generated by human annotators.

4 Word Embeddings Classification

4.1 Contextualized Language Models

For our evaluation we selected a variety of known language models that are proven to be efficient in WSD tasks. Besides well tested BERT-based models like BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), Distil-BERT and Distil-RoBERTa (Sanh et al., 2020), we included T5 (Raffel et al., 2020) and FLAN-T5 (Chung et al., 2022).

The T5-based models have an encoder-decoder architecture which proved to be useful in different benchmark tasks (Raffel et al., 2020). T5 models have been pre-trained on 750GB of cleaned data, significantly more than the 16GB and 160GB used for BERT and RoBERTa, respectively.

To get a comprehensive overview of all models, we tested different sizes from small to xxl in T5 and FLAN-T5 and distil, base and large in BERT and RoBERTa. The parameters and embedding vector sizes are detailed in Table 3. All models are

Word	Senses	Word	Senses	Word	Senses	Word	Senses
apple	apple_inc	**date**	date_fruit	**match**	match_sports	**rock**	rock_music
	apple_fruit		date_romantic		match_lighter		rock_stone
bank	bank_bank	**digit**	digit_number	**nail**	nail_metal	**ruler**	ruler_governor
	bank_river		digit_anatomy		nail_finger		ruler_measure
bat	bat_mammal	**gum**	gum_bubblegum	**pitcher**	pitcher_jug	**seal**	seal_animal
	bat_equipment		gum_mouth		pitcher_sports		seal_close
cell	cell_prison	**java**	java_program	**pupil**	pupil_student	**spring**	spring_season
	cell_biology		java_island		pupil_eye		spring_device
crane	crane_machine	**letter**	letter_alphabet	**ring**	ring_arena	**trunk**	trunk_botany
	crane_bird		letter_mail		ring_jewelry		trunk_car

Table 2: All homonyms used in the dataset listed with their senses.

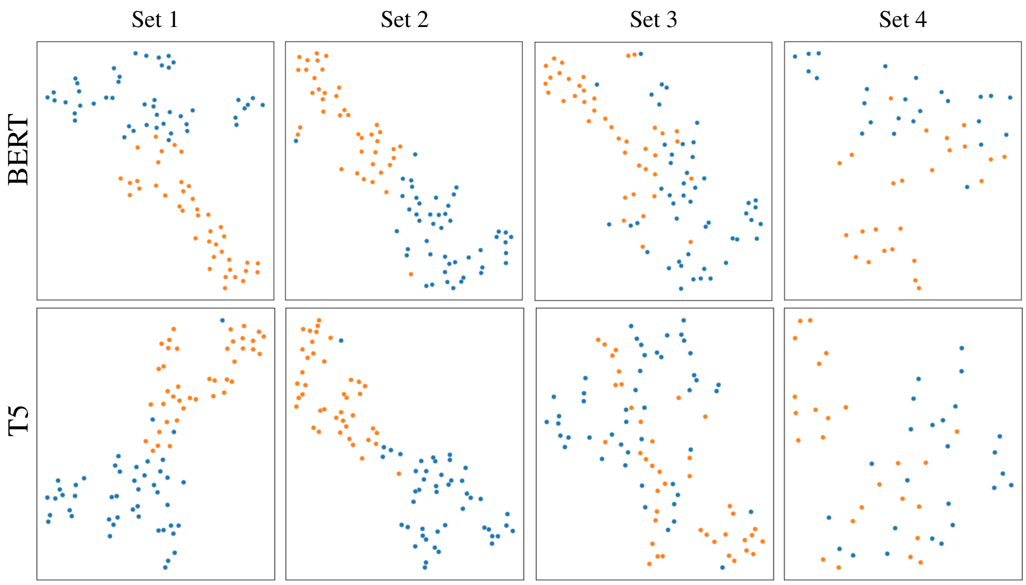

Figure 1: The visualization depicts the word embeddings of the word "crane" produced by BERT-base (first row) and T5-base (second row) for different sentences in each Set 1 to 4. Orange depicts all embeddings with the label "crane_bird" and blue all the ones labeled "crane_machine". We used tSNE (Van der Maaten and Hinton, 2008) for dimensionality reduction. One can see that the models are able to cluster the different senses in Set 1 and 2, while they struggle to differentiate them in Set 3 and 4.

utilized in their original, unmodified form from the HuggingFace library (Wolf et al., 2019) and were not specifically fine-tuned for this purpose.

4.2 Experimental Settings

To evaluate the performance of all models on the introduced dataset, a binary classification task is employed. All of the following is performed for each model in each set. For each sentence in a set, all words are converted to lower case, and the embedding vector for the homonym is extracted. To ensure the best results, it is recommended to sum and average the word embeddings from the final four layers of the encoder in BERT (Loureiro et al., 2021). This approach is also adopted for T5 and FLAN-T5 to ensure better comparability. We visualized the word embeddings of the homonym "crane" produced by BERT-base (first row) and T5-base (second row) in Figure 1 for every test set, with embeddings color-coded by their correct label. We include in the Appendix more visualizations of different words and models (figs. 2 to 7) The averaged word embeddings are categorized into one of the designated labels using k-nearest neighbor (kNN) algorithm (Cover and Hart, 1967), which uses our training data as a basis for classification. This algorithm takes a plurality vote of a sample's nearest labeled neighbors, in our case $k = 3$, and decide based on the 3 nearest neighbors which sense to assign the homonym to. Tests varying k showed no significant differences on the outcome, which is consistent with the findings of Wiedemann et al. (2019). Cosine similarity was used as the distance measure, and the macro F1-score as the performance measure. The kNN model

Models	#Parameter	VecSize	Set 1	Set 2	Set 3	Set 4
`distil-BERT`	66M	768	0.945	0.948	0.867	0.617
`BERT-base`	110M	768	0.962	0.976	0.869	0.662
`BERT-large`	340M	1024	0.97	0.978	0.874	0.689
`distil-RoBERTa`	82M	768	0.920	0.950	0.856	0.634
`RoBERTa-base`	125M	768	0.945	0.969	0.888	0.715
`T5-small`	60M	512	0.877	0.916	0.768	0.609
`T5-base`	220M	768	0.978	0.982	0.866	0.611
`T5-large`	770M	1024	0.984	0.987	0.896	0.691
`T5-xl`	3B	1024	0.991	0.992	0.907	0.71
`T5-xxl`	11B	1024	0.993	0.995	0.910	**0.786**
`FLAN-T5-small`	80M	512	0.906	0.938	0.803	0.575
`FLAN-T5-base`	250M	768	0.980	0.987	0.907	0.621
`FLAN-T5-large`	780M	1024	0.948	0.953	0.852	0.663
`FLAN-T5-xl`	3B	1024	0.955	0.958	0.881	0.718
`FLAN-T5-xxl`	11B	1024	**0.994**	**0.996**	**0.932**	0.778

Table 3: Results (F1-Scores) for all encoder models, including their parameters and embedding sizes, are presented

was trained on the averaged embeddings produced by the corresponding model for the Train Set of our dataset. Accordingly, the k-nearest neighbor (kNN) algorithm is employed to classify the data from the four test sets. For each word in a set, the F1-score is calculated and then averaged over all words in a set, resulting in four different F1-scores for each model

4.3 Results

All results are listed in the Table 3 together with the corresponding number of parameters and the embedding vector size of each model. In general, all models show good performances in Set 1 and Set 2. Almost all models score higher than 90% in the first two settings and some T5-based models even up to 99%. The T5-based model score in general higher than the BERT-based models with the same size.

Model Size In almost all cases, it is noticeable that as model size increases, so do the outcomes across all four sets. While T5-small achieves only 87.7%, T5-xxl shows results as good as 99.3%. This effect is seen in all models except in FLAN-T5-large and FLAN-T5-xl which show worse results in setting 1, and 3 than FLAN-T5-base.

Settings As previously stated, all models demonstrate a good performance on Sets 1 and 2. However, the performance of the models declines when evaluated on Sets 3 and 4. A comparison of the results observed in Set 1 with those in Set 3 reveals a decline in the F1-Score from 6% to up to 11%, even though only one additional adjective is introduced in this setting. Nevertheless, the performance drops from Set 1 to Set 4 are even more severe, with a decrease ranging from 20% to 33%. The most significant effect is observed in smaller model sizes, while in larger models, the difference between Set 1 and Set 4 is smaller, with approximately 20%. Overall the best performance is shown in FLAN-T5-xxl which has the best performance in all four settings and one of the smallest performance drop to Set 4.

5 Prompt-based Classification

5.1 Experimental Settings

We evaluate FOOL, using state-of-the-art large language models including GPT-3.5 Turbo (OpenAI, 2022), GPT-4 Turbo (OpenAI, 2023) , and GPT-4o (OpenAI, 2024), LlAMA3-8b, LlAMA3-70b (Meta, 2024) and Mixtral-8x7b (Jiang et al., 2024). Since these models are decoder models, we utilized prompt-based classification for testing. We input each sentence from the set and ask the model to classify the target word by providing two choices. For example, to classify the meaning of the word "apple" the prompt for GPT-4o would be:

"In this sentence: ‘She used iCloud to store photos from her visit to the apple orchard, ensuring she never lost a memory’, classify the occurrence of the word ‘apple’ for fruit or for a company. Answer only by one of these options: fruit or company."

The outputs were manually evaluated by humans because, although models like GPT-4o and GPT-4 Turbo strictly adhere to instructions by outputting only "fruit" or "company," other models such as GPT-3.5 Turbo occasionally respond with explanations that include both categories complicating the extraction of the correct answer. Such responses

Model	Set 1	Set 2	Set 3	Set 4
GPT-3.5 Turbo	0.981	0.990	0.786	0.725
GPT-4 Turbo	0.998	0.999	0.907	0.922
GPT-4o	0.998	0.999	0.860	0.956
Llama-3 8b	0.986	0.990	0.790	0.687
Llama-3 70b	0.994	0.998	0.907	0.890
Mixtral-7bx8	0.987	0.993	0.820	0.714

Table 4: F1-Scores showing the performance of large decoder models on FOOL using prompt-based classification

were considered correct if the classification was accurate. However, outputs that included both classes were marked as incorrect guesses. We conducted initial testing with multiple runs for the same sentences and observed little variance; therefore, the reported results are from a single run for each word.

5.2 Results

The results in Table 4 show that state-of-the-art models can distinguish perfectly between two homonyms in a regular context. All models score above 98%, indicating no difficulty in distinguishing homonyms. Adding an adjective to the homonym makes the performance even better for all models to score almost perfectly with an accuracy around 99.9% for models like GPT-4o. However, results from Set 3, where only one adversarial adjective is added to the sentences of Set 1, could fool the models and affect their performance. For example, the score of GPT4-o drops from 99.8% to around 87% showing vulnerability to a simple adversarial context change. However, GPT-4o shows more robustness to a realistic opposing context test in Set 4 with F1-score of 95.6%. In addition, models like GPT-3.5 Turbo, Llama3-8b and Mixtral-7bx8 experience significant performance drops in Set 4 with F1-score around 70%.

6 Discussion

In the following we discuss the main findings and open questions that remain after our analysis.

Set 1/2 vs. Set 3/4 One of the main findings from the analysis above is that there is a major performance gap between Set 1 and Set 2 compared to Set 3 or Set 4. The significant decline in performance observed between Set 1 and Set 3 in the WSD test, despite the only change being the replacement of one adjective, appears to be out of proportion. Also the performance decrease in Set 4 is disproportionate. Adding opposing yet realistic context while still remaining the overall meaning of the homonym can lead to a decrease in the F1-score to up to 30% even for advanced models like Llama3-8b and GPT-3.5 Turbo. One explanation for the changing results could be that contextualised language models do not pay attention to semantic boundaries like Reif et al. (2019) mentioned in his paper about BERT. This could be extended by the findings of Tang et al. (2018) who state that language models do not learn which context words are useful and pay attention mostly to the homonym itself. Unimportant context words, which humans can successfully sort out, have a major impact on the word embedding produced by language models. This could be one factor language models have to improve in order to achieve human-like results also in smaller model sizes.

Model Size Another finding is the correlation between model sizes and WSD performance in all four sets. The results indicate a positive correlation between model size and F1-score. Larger models with more parameters store more training data information and have bigger embedding vectors that capture extensive contextual details, improving disambiguation. Furthermore, a decline in performance is observed in Sets 3 and 4, with smaller models experiencing a larger drop than larger models. This supports the hypothesis that larger models are more robust to adversarial attacks. This robustness is likely due to larger models' ability to recall more information and recognize different contexts.

T5 vs. BERT The best overall performance is seen in the encoder of FLAN-T5-xxl. In general, the T5-based models show the best overall results not least because of the bigger model sizes. Even in the base size FLAN-T5 surpasses BERT-large which has more model parameters than FLAN-T5. This may suggest that T5-based language models are an optimal choice for the task of word sense disambiguation. One potential explanation for the enhanced performance is that T5 employs a distinct masking approach distinct from BERT. While BERT can only mask one word at a time, T5 masks multiple words at the same time. Additionally, T5 was trained on a larger data corpus than BERT which could also improve the performance in WSD since more knowledge about words in different usages is collected.

Embeddings vs. Prompt-based Classification In this paper, we tested the performance of two types of models: those that include an encoder, which provides bi-directional context of the sentence and thus reflects it in their embeddings, and large decoder models known for their ability when prompted. It is evident that having bi-directional context is an advantage, as reflected in the results when comparing models by size. We can see that even state-of-the-art models like LlaMA-3-8b, which is trained on around 15 trillion tokens, perform worse than T5-large, which is trained on around 1 trillion tokens and has approximately ten times fewer parameters than LlaMA-3-8b. Furthermore, we believe that the bi-directional context ability of T5 and BERT family models makes them less vulnerable to simple adversarial context changes, such as altering one adjective in a sentence. This is evidenced by the less significant performance drop in Set 3 compared to decoder models like LlaMA-3-8b or even GPT4-o. For example, GPT-4o's performance drops by about 12% from Set 1 to Set 3, whereas even a simple BERT-base model's performance drops only by about 9%. Additionally, the performance of GPT-4o in Set 3 is comparable to that of T5-base and lower than T5-large, which have approximately 220 million and 770 million parameters, respectively. While the types of models were tested differently, one could argue that encoder models are better suited to these types of tasks. On the other hand, both GPT4o and GPT-4 Turbo models show greater robustness in realistic opposing contexts when tested on Set 4. In this scenario, we believe that the set involves more reasoning abilities, which some claim these types of models possess, and smaller models like T5-base and BERT are less equipped for.

Error Analysis In this section, we analyze the mistakes made by the models and identify specific words that the models struggled to disambiguate in Set 4. There are many factors that affect model performance, but we will discuss a few key ones. Firstly, there are words that are predominantly used in one meaning and less so in another, such as "digit". We observed that performance for these types of words is generally lower. Another category of challenging words includes those that share similar contexts across different meanings, like "gum" and "letter". For instance, "gum" in both meanings involves the context of the mouth and chewing, making it more difficult for the model to distinguish between them. Similarly, "letter" involves writing in both contexts. Conversely, for words like "Java" where we intended two meanings—Java the programming language and Java the island—the models performed well. Even though "Java the island" is not widely used, the contexts of the two meanings are completely different, making it harder to create sentences that fool the models. Additionally, some models exhibit a bias towards a particular meaning; for example, Mixtral-7bx8 shows a bias towards interpreting "pitcher" as a container and "rock" as stone. The performance of the models on each word in Set 3, and 4 is detailed in figures 8 and 9 in Appendix A.3.

7 Conclusion

In this paper, we introduce FOOL, a new coarse-grained WSD dataset featuring various types of contexts, which serves as both a benchmark for assessing model performance on WSD tasks and a tool for evaluating context comprehension by models. Our experiments using this dataset demonstrate that SOTA language models still struggle to understand context and disambiguate homonyms in the presence of opposing contexts, compared to their performance in regular WSD tasks. This effect is most prominent not only in smaller models like BERT-base and T5-base but also in larger models like Llama-3 and GPT-3.5 Turbo. Among the series of models that include an encoder, our results show that T5, especially FLAN-T5 is a better alternative to BERT-based models. With more than 99% score in Set 1, FLAN-T5-xxl shows human-like disambiguation skills. Furthermore, we showed that models incorporating an encoder are less vulnerable to adversarial addition of context (Set 3) with the best performing model being FLAN-T5-xxl, which outperforms GPT-4o and GPT-4 Turbo. Interestingly, small models like BERT-large and FLAN-T5-base outperform GPT-4o on the same set. However, these small models struggle with Set 4, which includes realistic opposing context usage of words, which we believe requires a deeper understanding of language and some degree of reasoning abilities. In the future, we plan to extend the FOOL dataset to include sentences with fine-grained homonyms to investigate how language models perform on them. Additional adversarial settings could also be added to further challenge the models, potentially exposing new weaknesses in their contextual understanding and disambiguation capabilities. This will provide further insights into the limitations of current language models and guide the development of more robust systems.

8 Limitations

While our study presents significant findings in the field of Natural Language Processing, several limitations should be acknowledged to contextualize the results.

Our approach deals with homonymous nouns in a coarse-grained manner, which may oversimplify the complexities of word sense disambiguation. Our coarse-grained homonym resolution does not consider the nuanced differences between the various meanings of a word that are closely related to each other; instead, it focuses on only two distinct senses. This limitation might affect the precision of our models' understanding and processing of the context. Moreover, the exclusive focus on nouns, while ignoring other word types, such as verbs, adjectives, or adverbs, may result in limited generalizability.

Furthermore, it would have been beneficial to extend the study by testing additional languages, models, and a larger dataset.

References

Eneko Agirre and Philip Edmonds. 2007. *Word Sense Disambiguation: Algorithms and Applications*. Springer Science & Business Media.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *Preprint*, arxiv:2210.11416 [cs].

T. Cover and P. Hart. 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Jiaju Du, Fanchao Qi, and Maosong Sun. 2019. Using BERT for word sense disambiguation. *Preprint*, arxiv:1909.08358 [cs].

Philip Edmonds and Scott Cotton. 2001. SENSEVAL-2: Overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5. Association for Computational Linguistics.

Denis Emelin, Ivan Titov, and Rico Sennrich. 2020. Detecting word sense disambiguation biases in machine translation for model-agnostic adversarial attacks. *Preprint*, arxiv:2011.01846 [cs].

Jonas Gehring, Michael Auli, David Grangier, and Yann N. Dauphin. 2017. A convolutional encoder model for neural machine translation. *Preprint*, arxiv:1611.02344 [cs].

Janosch Haber and Massimo Poesio. 2024. Polysemy–evidence from linguistics, behavioral science, and contextualized language models. *Computational Linguistics*, pages 1–67.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Kat IP Pty Ltd. 2008. WordHippo!

Ruth M. Kempson. 1977. *Semantic Theory*. Cambridge University Press.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *Preprint*, arxiv:1907.11692 [cs].

Daniel Loureiro, Kiamehr Rezaee, Mohammad Taher Pilehvar, and Jose Camacho-Collados. 2021. Analysis and evaluation of language models for word sense disambiguation. *Computational Linguistics*, 47(2):387–443.

LoveToKnow Media. 2024. Sentence examples | Examples of words used in a sentence.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. *Preprint*, arxiv:1508.04025 [cs].

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date.

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

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Andrea Moro and Roberto Navigli. 2015. SemEval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 288–297. Association for Computational Linguistics.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. SemEval-2013 task 12: Multilingual word sense disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231. Association for Computational Linguistics.

OpenAI. 2022. ChatGPT.

OpenAI. 2023. GPT-4 Technical Report.

OpenAI. 2024. GPT-4o.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. SemEval-2007 task-17: English lexical sample, SRL and all words. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92. Association for Computational Linguistics.

Lorenzo Proietti, Stefano Perrella, Simone Tedeschi, Giulia Vulpis, Leonardo Lavalle, Andrea Sanchietti, Andrea Ferrari, and Roberto Navigli. 2024. Analyzing homonymy disambiguation capabilities of pretrained language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 924–938. ELRA and ICCL.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110. Association for Computational Linguistics.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Preprint*, arxiv:1910.01108 [cs].

Benjamin Snyder and Martha Palmer. 2004. The english all-words task. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43. Association for Computational Linguistics.

Gongbo Tang, Rico Sennrich, and Joakim Nivre. 2018. An analysis of attention mechanisms: The case of word sense disambiguation in neural machine translation. Publication Title: arXiv e-prints ADS Bibcode: 2018arXiv181007595T.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does BERT make any sense? interpretable word sense disambiguation with contextualized embeddings. *Preprint*, arxiv:1909.10430 [cs].

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. HuggingFace's transformers: State-of-the-art natural language processing. Publication Title: arXiv e-prints ADS Bibcode: 2019arXiv191003771W.

A Appendix

A.1 Word Definitions and Dataset Statistics

Table 5 lists the number of examples in each subset and Table 6 shows definitions for the 20 homonyms in the FOOL dataset (cmp. Table 2 in the main text).

Words	**Senses**	Set Train	Set 1 - 3	Set 4
apple	apple_apple_inc	40	55	25
	apple_fruit	40	51	25
bank	bank_bank	40	57	25
	bank_river	41	54	25
bat	bat_mammal	30	56	25
	bat_equipment	30	55	25
cell	cell_prison	40	40	25
	cell_biology	40	40	25
crane	crane_machine	40	47	25
	crane_bird	40	41	25
date	date_fruit	40	40	25
	date_romantic	40	40	25
digit	digit_number	40	45	30
	digit_anatomy	29	45	30
gum	gum_bubblegum	40	40	25
	gum_mouth	40	40	25
java	java_program	40	40	30
	java_island	40	41	29
letter	letter_alphabet	40	40	25
	letter_mail	40	40	25
match	match_sports	40	40	25
	match_lighter	40	40	25
nail	nail_metal	40	40	25
	nail_finger	40	40	25
pitcher	pitcher_jug	40	40	25
	pitcher_sports	41	40	25
pupil	pupil_student	40	52	25
	pupil_eye	40	52	25
ring	ring_arena	40	40	25
	ring_jewelry	40	40	25
rock	rock_music	20	60	25
	rock_stone	30	60	25
ruler	ruler_governor	40	40	25
	ruler_measure	40	40	25
seal	seal_animal	40	50	25
	seal_close	40	50	25
spring	spring_season	40	56	25
	spring_device	40	42	25
trunk	trunk_botany	40	40	25
	trunk_car	40	41	25

Table 5: Number of sentences for every word sense in each set.

Words	Senses	Definitions
apple	apple_apple_inc	"Apple Inc. (formerly Apple Computer, Inc.) is an American multinational corporation and technology company headquartered in Cupertino, California, in Silicon Valley."
	apple_fruit	“the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.”
bank	bank_bank	"a financial establishment that uses money deposited by customers for investment, pays it out when required, makes loans at interest, and exchanges currency.."
	bank_river	"the land alongside or sloping down to a river or lake.."
bat	bat_mammal	"a mainly nocturnal mammal capable of sustained flight, with membranous wings that extend between the fingers and limbs.."
	bat_equipment	"an implement with a handle and a solid surface, typically of wood, used for hitting the ball in games such as cricket, baseball, and table tennis.."
cell	cell_prison	"a small room in which a prisoner is locked up or in which a monk or nun sleeps."
	cell_biology	"the smallest structural and functional unit of an organism, which is typically microscopic and consists of cytoplasm and a nucleus enclosed in a membrane.
crane	crane_machine	"a large machine that moves heavy things by lifting them in the air"
	crane_bird	"a kind of large bird with a long neck and long legs."
date	date_fruit	"the sweet fruit of various types of palm tree"
	date_romantic	"a social meeting planned before it happens, especially one between two people who have or might have a romantic relationship"
digit	digit_number	"any one of the numbers 0 through 9"
	digit_anatomy	"one of the fingers or toes"
gum	gum_bubblegum	"short for chewing gum or bubblegum."
	gum_mouth	"the firm area of flesh around the roots of the teeth in the upper or lower jaw."
java	java_program	"a general-purpose computer programming language designed to produce programs that will run on any computer system."
	java_island	"a large island that forms part of Indonesia"
letter	letter_alphabet	"a character representing one or more of the sounds used in speech; any of the symbols of an alphabet."
	letter_mail	"a written, typed, or printed communication, sent in an envelope by post or messenger."
match	match_sports	"a contest in which people or teams compete against each other in a particular sport."
	match_lighter	"a short, thin piece of wood or cardboard used to light a fire, being tipped with a composition that ignites when rubbed against a rough surface."
nail	nail_metal	"a small metal spike with a broadened flat head, driven into wood to join things together or to serve as a hook."
	nail_finger	"a horny covering on the upper surface of the tip of the finger and toe in humans and other primates."
pitcher	pitcher_jug	"a large, round container for liquids that has a flat base, a handle, and a very narrow raised opening at the top for pouring"
	pitcher_sports	"the player who delivers the ball to the batter."
pupil	pupil_student	"a person who is taught by another, especially a schoolchild or student in relation to a teacher."
	pupil_eye	"the dark circular opening in the centre of the iris of the eye, which varies in size to regulate the amount of light reaching the retina."
ring	ring_arena	"an enclosed space, surrounded by seating for spectators, in which a sport, performance, or show takes place."
	ring_jewelry	"a small circular band, typically of precious metal and often set with one or more gemstones, worn on a finger as an ornament or a token of marriage, engagement, or authority."
rock	rock_music	"a type of popular music with a strong, loud beat that is usually played with electric guitars and drums"
	rock_stone	"the dry solid part of the earth’s surface, or any large piece of this that sticks up out of the ground or the sea"
ruler	ruler_governor	"the leader of a country; a person who is in charge of a country"
	ruler_measure	"a straight strip or cylinder of plastic, wood, metal, or other rigid material, typically marked at regular intervals and used to draw straight lines or measure distances."
seal	seal_animal	"a large mammal that eats fish and lives partly in the sea and partly on land or ice"
	seal_close	"something fixed around the edge of an opening to prevent liquid or gas flowing through it"
spring	spring_season	"the season after winter and before summer, in which vegetation begins to appear, in the northern hemisphere from March to May and in the southern hemisphere from September to November."
	spring_device	"an elastic device, typically a helical metal coil, that can be pressed or pulled but returns to its former shape when released, used chiefly to exert constant tension or absorb movement."
trunk	trunk_botany	"the main woody stem of a tree as distinct from its branches and roots."
	trunk_car	"an enclosed space at the back of a car for carrying luggage and other goods; a boot."

Table 6: Definitions for all word senses used in our dataset. The definitions are adopted from the Oxford Dictionary.

A.2 Words Embeddings

Figures 2, 3, and 4 complement Figure 1 from the main text by showing the distribution of embeddings for the word “crane” for the other models studied in our experiments. Additionally, Figure 5, 6 and 7 show the same distribution for the word ”bank” to supplement our findings.

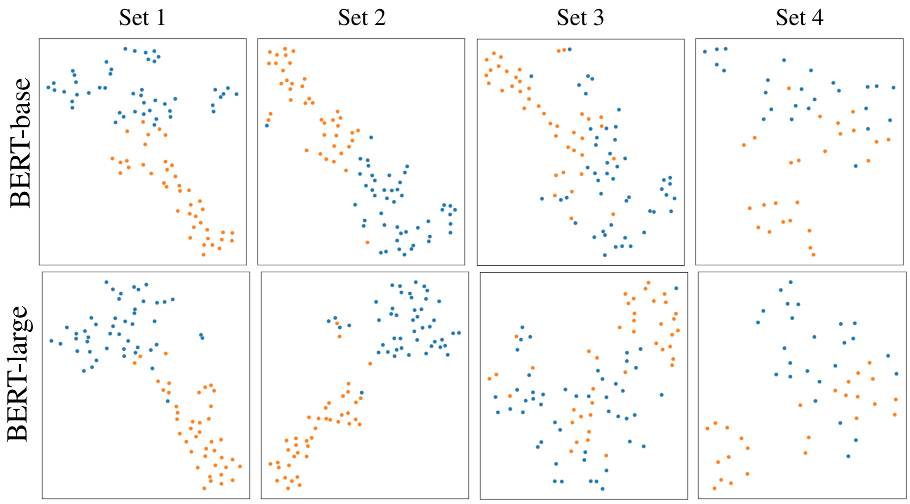

Figure 2: The embeddings for the word "crane" from BERT in size base and large.

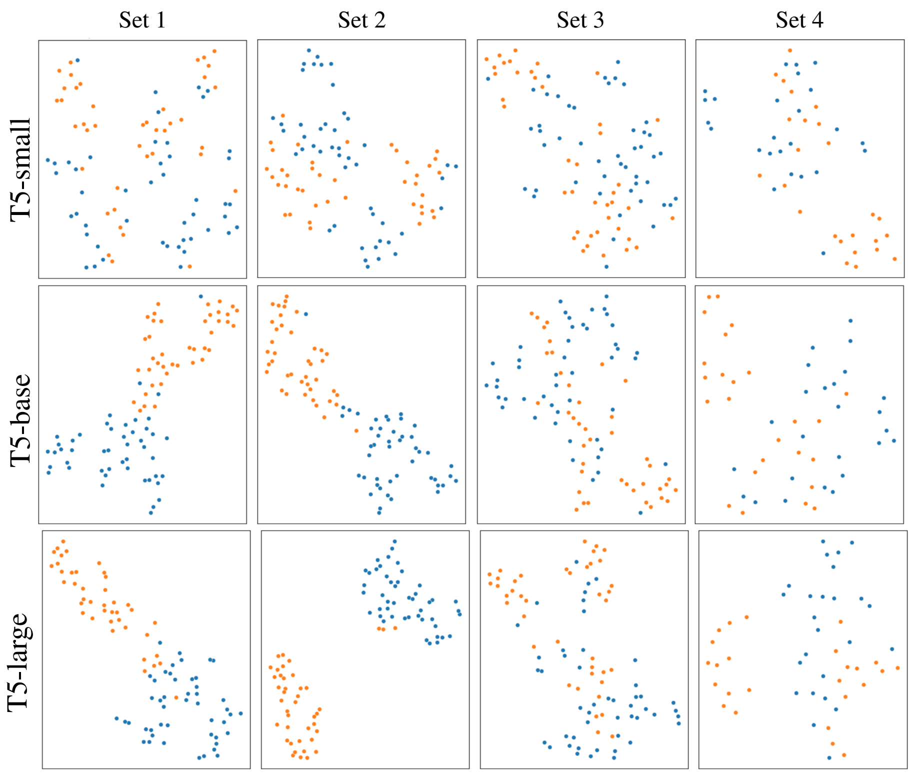

Figure 3: The embeddings for the word "crane" from T5 in size small, base and large.

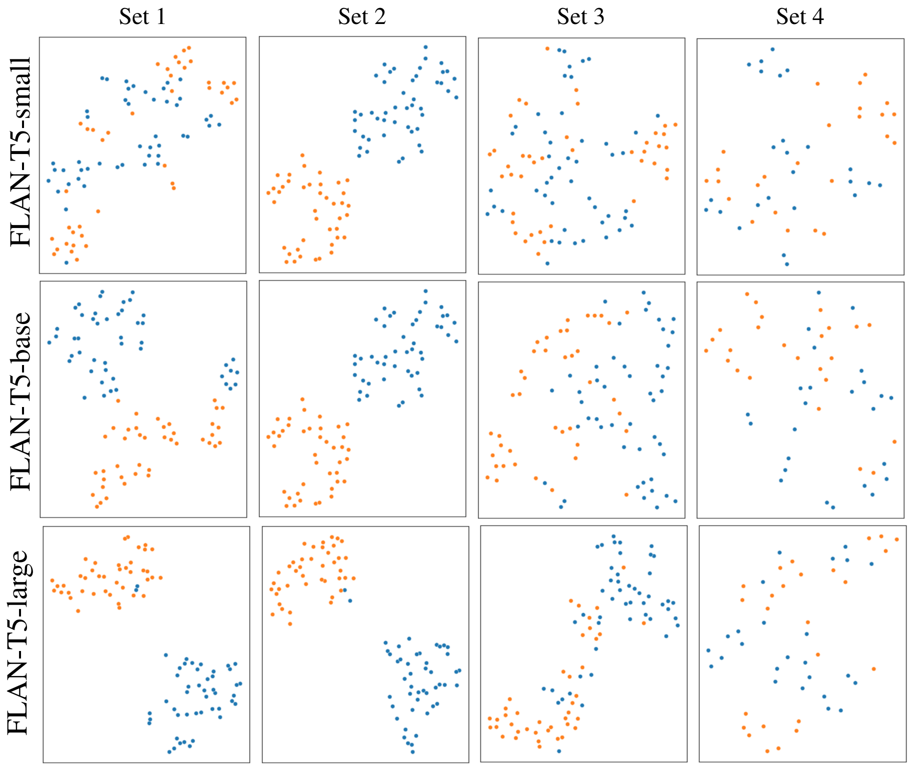

Figure 4: The embeddings for the word "crane" from FLAN-T5 in size small, base and large.

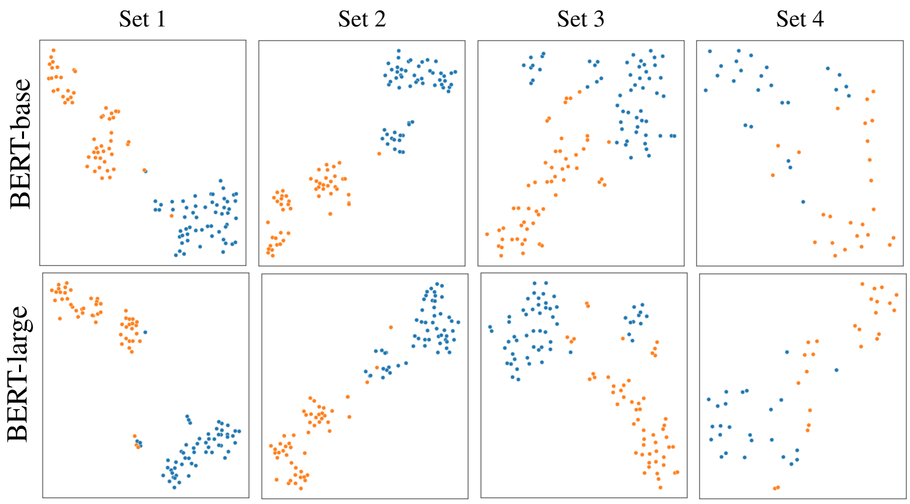

Figure 5: The embeddings for the word "bank" from BERT in size base and large.

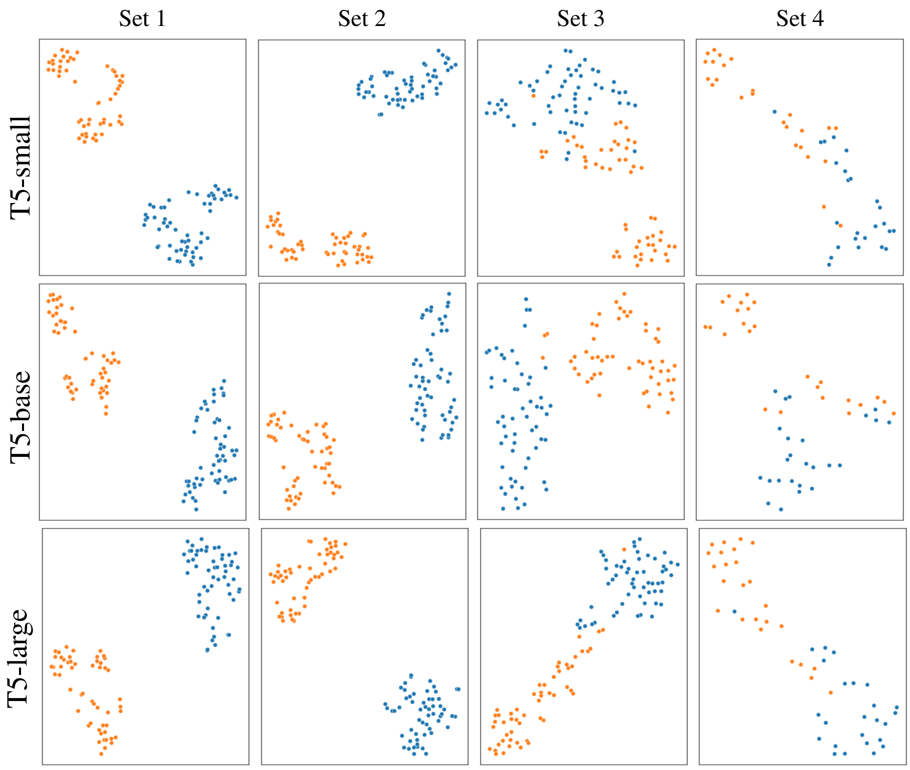

Figure 6: The embeddings for the word "bank" from T5 in size small, base and large.

Figure 7: The embeddings for the word "bank" from FLAN-T5 in size small, base and large.

A.3 Performance on Individual Words

In this section, the performance of different LLMs is shown. Figure 8 shows the perfomance of the LLMs on each word in Test Set 3, while Figure 9 shows the performance of the same LLMs on each word in Test Set 4.

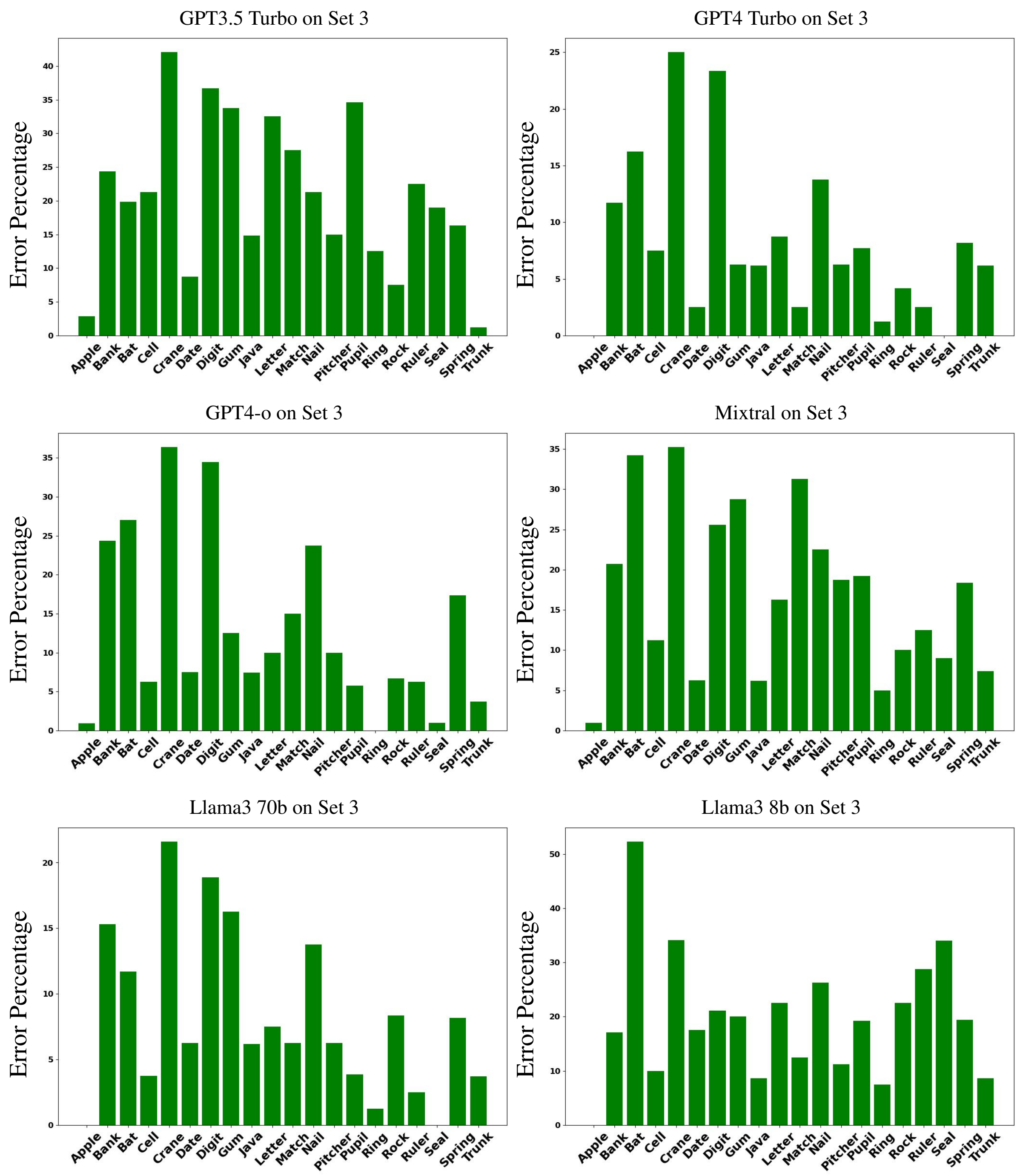

Figure 8: The figures show the error percentages of the different LLMs on each word in Test Set 3.

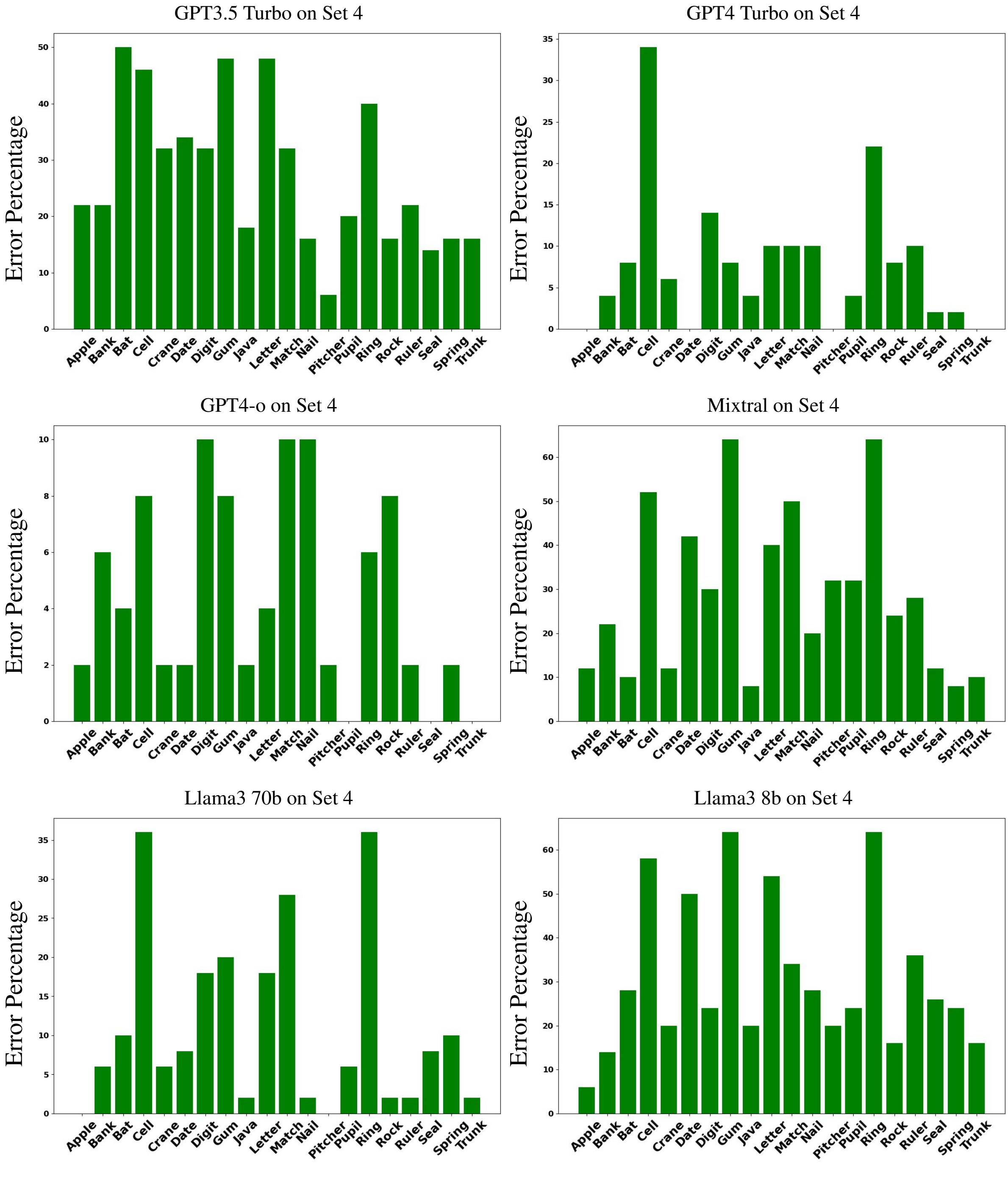

Figure 9: The figures show the error percentages of the different LLMs on each word in Test Set 4.