Assessing the Asymmetric Behaviour of Italian Large Language Models across Different Syntactic Structures

Elena Sofia Ruzzetti[1,*], Federico Ranaldi[1], Dario Onorati[2], Davide Venditti[1], Leonardo Ranaldi[3], Tommaso Caselli[4] and Fabio Massimo Zanzotto[1]

[1]*University of Rome Tor Vergata, Italy*
[2]*Sapienza University of Rome, Italy*
[3]*School of Informatics, University of Edinburgh, UK*
[4]*University of Groningen, The Netherlands*

Abstract

While LLMs get more proficient at solving tasks and generating sentences, we aim to investigate the role that different syntactic structures have on models' performances on a battery of Natural Language Understanding tasks. We analyze the performance of five LLMs on semantically equivalent sentences that are characterized by different syntactic structures. To correctly solve the tasks, a model is implicitly required to correctly parse the sentence. We found out that LLMs struggle when there are more complex syntactic structures, with an average drop of $16.13(\pm 11.14)$ points in accuracy on Q&A task. Additionally, we propose a method based on token attribution to spot which area of the LLMs encode syntactic knowledge, by identifying model heads and layers responsible for the generation of a correct answer.

Keywords

LLMs, Natural Language Understanding, Syntax, Attributions, Localization

1. Introduction

Large Language Models (LLMs) excel at understanding and generating text that appears human-written. Thus, it is intriguing to determine whether the models' text comprehension aligns in some way with human cognitive processes. A peculiarity of natural languages is that the same meaning can be encoded by multiple syntactic constructions. In Italian, for instance, the unmarked sentence follows a subject-verb-object (SVO) word order. However, inversions of this ordering do not necessarily lead to ungrammatical sentences. A case in point is represented by cleft sentence, i.e., sentences where the unmarked SVO sequence is violated. This corresponds to specific communicative functions, namely emphasize a component, and it is obtained by putting one element in a separate clause. In particular, Object Relative Clauses – where the element that is emphasized is the object of the sentence – are difficult to understand [1, 2]. For example the sentence *"Sono i professori che i presidi hanno elogiato alla riunione d'istituto"* is more challenging for an Italian speaker than its semantically equivalent unmarked version *"I presidi hanno elogiato i professori alla riunione d'istituto"* where the SVO order is restored. Similarly, in Nominal Copular constructions, the inversion of subject and verb clause is documented to cause difficulties in understanding the meaning of the sentence [3].

Hence, syntax plays a crucial role not only in the general construction of language but also in the native speakers ability to comprehend sentences: in fact, a correct syntactic parsing of the sentences is necessary to understand their meaning, and some syntactic structures are preferred over others. To what extent this preference is replicated by LLMs needs to be further explored.

If the model shows some knowledge about syntax, there should be an area of the model responsible for that. We aim to detect the area of a model responsible for its syntactic knowledge. Extensive work has been devoted to understanding how Transformer-based architectures encode information and one main objective is to localize which area of the model is responsible for a certain behavior [4, 5]. Despite its usage as an explanation mechanism being debated [6, 7], the attention mechanism is an interesting starting point given its wide use in Transformer architecture. While the attention weights alone cannot be used as an explanation of a model's behavior [8, 9], an analysis that includes multiple components of the attention module is shown to be beneficial to obtain an interpretation of how a model processes an input sentence [10, 11].

Probing is a common method used to detect the presence of linguistic properties of language in models [12]. Probing consists of training an auxiliary classifier on top of a model's internal representation, which could be the output of a specific layer, to determine which linguistic property the model has learned and encoded. In particular, it has been proposed to probe Transformer-based models to reconstruct syntactic representations

CLiC-it 2024: Tenth Italian Conference on Computational Linguistics, Dec 04 — 06, 2024, Pisa, Italy

[*] Corresponding author.

elena.sofia.ruzzetti@uniroma2.it (E. S. Ruzzetti)

like dependency parse trees from their hidden states [13]. Probing tasks concluded that syntactic features are encoded in the middle layers [14]. Correlation analysis on the weights matrices of the monolingual BERT models confirmed the localization of syntactic information in the middle layers showing that the models trained on syntactically similar languages were similar on middle layers [15]. While an altered word order seems to play a crucial role in Transformer-based models' ability to process language [16, 17], the correlation between LLMs downstream performance and the encoding of syntax needs to be further explored.

In this paper, we initially examine how syntax influences the LLMs' capability of understanding language. To achieve this, we will analyze five open weights LLMs – trained on the Italian Language either from scratch or during a finetuning phase – and measure their performance in question-answering (Q&A) tasks that require an implicit parsing of the roles of words in the sentence to provide the correct answer. We use an available set of Q&A tasks designed for Italian speakers [1] and propose similar template-based questions for two other datasets of Italian sentences characterized by different syntactic structures (Section 2.1). The results show that the models are affected by the different syntactic structures in solving the proposed tasks (Section 3.1): LLMs struggle when more complex syntactic structures are present, with an average drop in accuracy of $16.13(\pm 11.14)$ points.

We then propose a method – based on norm-based attribution [10]– to localize where syntactic knowledge is encoded by identifying the models' attention heads and layers that are responsible for the generation of a correct answer (Section 2.2). Although some differences can be observed across the five LLMs, we notice that syntactic information is more widely included in the middle and top layers of the models.

2. Methods and Data

2.1. Question-answering Tasks to assess LLMs Syntactic Abilities

In this Section, we introduce the dataset we collected – largely extracted from the AcCompl-It task [18] in EVALITA 2020 [19] – to assess LLMs syntactic abilities. The dataset is split in three subdatasets. Each of the subdataset is composed of pairs of sentences that share the same meaning but a different word order. One of the sentences in each pair is characterized by a simpler structure, easier to understand also for humans, while the second is characterized by an alternative – but still correct – syntactic structure. We aim to understand whether a different structure can influence the model performance in processing those similar sentences. We define, for each subdataset, a Q&A task to assess the LLMs capabilities in understanding sentences when their syntactic structure makes them more complex. The Q&A task requires the model to implicitly parse the role of the words in the sentence to get the correct answer: for this reason, we identify some important words that the model should attend to while getting the correct answer.

Object Clefts constructions The first subset is derived from Chesi and Canal [1]: this dataset contains 128 sentences characterized by Object Clefts (OC) constructions. The OC sentences in this dataset all share the same structure (see Table 1): the object and subject are words indicating either a person or a group of people, the predicate describes an action that the subject performs towards the object. The object is always introduced as the first element of the sentence in a left-peripheral position. The displacement of the object in the left-peripheral position makes the OC harder to understand [2]. We will compare those sentences with semantically equivalent ones that preserve the unmarked SVO word order.

To assess whether the difficulty humans have in understanding Object Cleft sentences can also be registered in LLMs for the Italian language, we tested them on the same Q&A task that Chesi and Canal [1] proposed to human subjects. Given one OC sentence, the model is prompted with a yes or no question asking whether one of the participants (subject or object) was involved in the action described by the predicate (see Table 1 for an example). The ability of a model to comprehend cleft sentences can be measured as the accuracy it obtains on this Q&A task. Moreover, we perform the same Q&A task on SVO sentences that we directly derived from the OC clauses in Chesi and Canal [1]: in this case, we restored the SVO order and produced sentences that are semantically equivalent to the corresponding OC (see Table 1).

To correctly solve the task, the model must interpret the role of the nouns of the sentences playing the role of subject and object to answer the comprehension question. Hence, the model should implicitly parse the sentences and focus on those relevant words during the generation of the answer.

The Copular Constructions The second subdataset –which includes 64 pairs of sentences– is derived from a study involving Nominal Copular constructions (NC) from Greco et al. [20]. The NC sentences are composed of two main constituents: a Determiner Phrase (DP_{subj}) and a Verbal Phrase (VP). The verbal phrase contains a copula and another Determiner Phrase that acts as the nominal part of the predicate (DP_{pred}). In this dataset, the effect of the position of the subject with respect to the copular predicate is studied. Two semantically equivalent

OC	*Sono i professori*	*che i presidi*	*hanno elogiato*	*alla riunione d'istituto*	
	Copula + Obj	Subj	Predicate	PP	
SVO	*I presidi*	*hanno elogiato*	*i professori*	*alla riunione d'istituto*	
	Subj	Predicate	Obj	PP	
Question	*Qualcuno ha elogiato i professori alla riunione?* or *I presidi hanno elogiato qualcuno alla riunione?*				
NC inverse	*La causa*	*della rivolta*	*sono*	*le foto*	*del muro*
	noun of DP_{subj}	PP_{pred}	Copula	Subject	PP_{subj}
NC canonical	*Le foto*	*del muro*	*sono*	*la causa*	*della rivolta*
	Subject	PP_{subj}	Copula	noun of DP_{subj}	PP_{pred}
Question	*Di che cosa le foto sono la causa?*				
MVP post	*Hanno mangiato*	*le bambine*		*il dolce*	
	Predicate	Subj		Obj	
MVP pre	*Le bambine*	*hanno mangiato*		*il dolce*	
	Subj	Predicate		Obj	
Question	*Chi ha mangiato qualcosa?* or *Cosa è stato mangiato?*				

Table 1
Examples from the dataset under investigation. For each subdataset, an example is composed of two semantically equivalent sentences, that differ from the syntactic point of view, and a comprehension question on them.

sentences are presented for each example. In one case, the sentence presents a canonical structure (NC canonical), with the subject (DP_{subj}) preceding the copular predicate. In the second case, an inverse structure (NC inverse) –with the subject following the predicate and the DP_{pred} introduced as the first element of the sentence – is presented (see Table 1). NC inverse sentences are syntactically correct but are harder to understand for humans than the NC canonical [3].

The structure of the sentences in this dataset is enriched by two Prepositional Phrases, one in each of the Determiner Phrases. The DP_{subj} includes a subject accompanied by an article and augmented with a Prepositional Phrase (PP_{subj}) that features a complement referring to the subject. Similarly, the DP_{pred} consists not only of a noun and an article but is instead further enriched with another Prepositional Phrase PP_{pred}. The PP_{pred} gives more information about the relation between the subject noun and the nominal part of the predicate.

We exploit the different role of the two Prepositional Phrases to design a Q&A task on NC canonical and NC inverse sentences and hence assess whether a more complex syntactic structure can influence LLMs capabilities. Given an NC sentence, the model is asked to correctly interpret the meaning of the sentence by examining its predicate: in particular, the model is asked to predict the additional information related to the nominal predicate – which is included in the PP_{pred} – by answering a "wh-" question (in Italian, "Di cosa", see the example in Table 1). While both Prepositional Phrase answer to a wh-question, only the PP_{pred} is related to the predicate of the sentence and hence the model should be able to predict the PP_{pred} and ignore the PP_{subj}.

To solve the proposed task and to properly understand NC sentences, humans and LLMs are required to implicitly parse the sentence and accurately identify the nominal part of the verbal phrase and, in particular, the Prepositional Phrase that it contains (PP_{pred}).

Minimal Verbal Structure with Inversion of Subject and Verb Finally, the last subdataset we investigate is derived from Greco et al. [20] and contains sentences characterized by minimal verbal structure (MVP). MVP sentences are composed of a subject, a predicate and – for sentences with transitive predicates – of an object (see Table 1). In this subdataset, the inversion of the subject and the verb is studied: the pairs of sentences under investigation have the same meaning (and lexicon) but in one cases the subject of the sentence follows the predicate (MVP post) while in the others the subject precedes the predicate (MVP pre). The latter configuration, in Italian, is more common that the former: we aim to investigate whether this syntactic variation can alter the performance of an LLM.

We define, for each pair of sentences, a question that asks the model to predict which element of the sentence is involved in a certain action, either as the subject entity or the object. In particular, for sentences that contain intransitive verbs, the model is always asked to predict the subject of the sentence, while in transitive cases (like the one in Table 1) the model is either asked to predict the subject or the object of the sentence. For this subdataset, while the original data included both declarative and interrogative sentences, we retained only the declarative ones: we test the model with a total of 192 sentence pairs.

To answer those questions, the relevant words – both for humans and LLMs – are the nouns that play the role of subject, or object if present, in sentences. In the next Section, we describe how it is possible to quantify whether a model is able to identify the role of those words during the generation of the answer.

	Qwen2-7B	LLaMAntino-3-ANITA-8B	Llama-2-7b	modello-italia-9b	Meta-Llama-3-8B
OC	75.78	76.56	57.81	56.25	64.84
SVO	89.06	83.59	66.41	71.09	80.4
NC inverse	62.50	78.12	15.62	82.81	81.25
NC canonical	81.25	84.38	62.50	93.75	87.50
MVP post	72.92	77.6	70.31	50.52	69.79
MVP pre	97.92	98.44	92.19	53.12	95.83

Table 2
Models accuracy on the different subdataset on the proposed Q&A tasks. Models tend to produce less accurate answers when exposed to more rare syntactic structures.

2.2. Localizing Syntactic Knowledge via Attributions

Knowing which sentence structures are easier or more difficult for a model to analyze is not enough. Considering the black-box nature of these models, it is essential to understand which layers are responsible for encoding syntax, thus making the models more interpretable.

We hypothesize that there is an area of the model responsible for correctly analyzing the sentence from the syntactic point of view in order to get the answer to the Q&A task. In fact, as discussed in the previous Section, to answer correctly, the model needs to implicitly parse the roles of the words in the sentence and identify the relevant words for the response (subjects and objects in the questions on OC, SVO and MVP sentences and the correct prepositional phrases in NC sentences). Hence, a knowledge of syntax is required to identify the relevant words and, consequentially, generate the correct answer.

In generating the answer, we expect the model to "focus" on those relevant words. We can identify to which token the model focuses during generation, measuring token-to-token attributions [8, 10]. In fact, token-to-token attribution methods quantify the influence of a token in the generation of the other. We argue that the part of the model architecture most aware of syntax is the one that systematically focuses on relevant words when the model is prompted to answer syntax-related questions. Kobayashi et al. [10] demonstrate that a mechanism – called the norm-based attribution – that it incorporates also the dense output layer of the Attention Mechanism is an accurate metric for token-to-token attribution. We will refer to the matrix $A^h(X)$ – computed for the attention head h for a sequence X – as an attribution matrix. Some examples and a more detailed description of norm-based attribution can be found in the Appendix (A.1). The attribution matrix $A^h(X)$, for each sequence of tokens X, describes where the model focuses during the generation of each token. By examining all the attention heads, some of them may focus more often on the subject, the object, or the prepositional phrase in the predicate while generating the answer for the task. In particular, for each attention head h, we consider the tokens to be attributed for the generated answer produced by the model: for each correct answer generated by the model, we count the number of times the tokens with the larger attribution value are the relevant ones. This measures the accuracy of the attention head h in recognizing the relevant words to generate the answer.

The more often the attention head focuses on the relevant words, the more syntactic knowledge the head encodes. For each downstream task presented in Section 2.1, we collect the accuracy of all heads at all levels. Then, we identify a head as "responsible" for generating the target word in a task if its score is higher than the average score for that task. Specifically, we assume a Gaussian distribution of scores for each task and identify a head as responsible if the probability of observing a value at least as extreme as the one observed is below a threshold $\alpha < 0.05$. We also consider responsible all heads that obtain an excellent accuracy score (greater than 0.9) in focusing on the relevant words. With this procedure, for each layer and task, we can localize the responsible heads and determine where the model encodes syntax the most.

2.3. Models and Prompting Method

We focus on Instruction-tuned LLMs, all of comparable size, and trained – either from scratch or only fine-tuned – on the Italian language. The models[1] under investigation are Qwen2-7B [22], LLaMAntino-3-ANITA-8B [23], Llama-2-7b [24], modello-italia [25], and Meta-Llama-3-8B [26]. To solve the Q&A task, we prompted each model with 4 different – but semantically equivalent – instructions. The complete list of the prompts is in Appendix A.2. All prompts ask the model to solve the task in zero-shot by answering only with one or two words. At most 128 tokens are generated, with greedy decoding. Once the generation is completed, a manual check of the responses is performed to obtain a simplified response to be compared with the gold. For the subsequent analysis, for each model and task, only the prompt for which the higher accuracy is obtained is considered.

[1] All models parameters are available on Huggingface's transformers library [21]

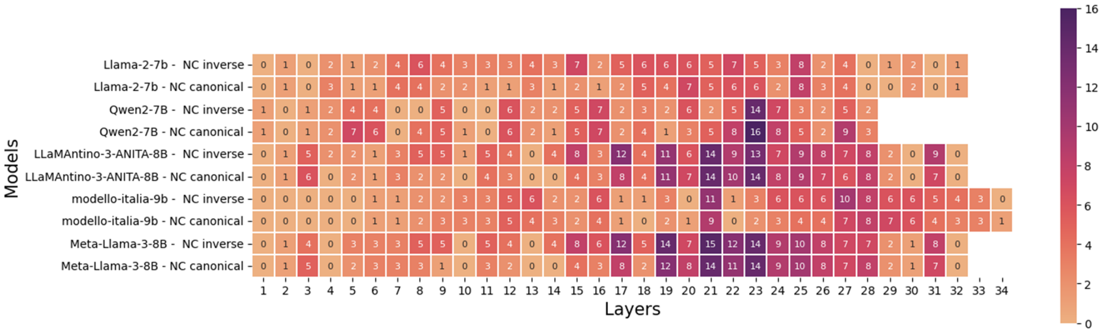


Figure 1: Number of responsible heads per layer in the Q&A task defined over NC sentences. The higher the number of responsible heads, the more the layer as a whole focus on syntax.

3. Experiments and Results

We initially revise model's accuracy on question comprehension task and assess models capabilities when different syntactic structures are involved (Section 3.1). Then, we aim to spot the layers responsible for the correct syntactic understanding of the sentences (Section 3.2).

3.1. Models accuracy on question-answering task

Results on each of the subdatasets show that the syntactic structure of a sentence influences the models' understanding of that sentence (see Table 2): across all tasks, LLMs tend to obtain larger accuracy on sentences characterized by a unmarked syntactic structure.

On the first task, on OC and SVO sentences, the models tend to struggle, especially in the OC sentences. On OC sentences, some models, in fact, do not perform far from the random baseline of 50% accuracy ("yes" and "no" answers are balanced). When comparing OC and SVO sentences, on average, the model accuracy drops by $11.88(\pm3.84)$ points when the sentence presents the object in the left-peripheral position. This result aligns with the difficulty that humans encounter in understanding those sentences. The model that achieves the highest accuracy in this task in OC sentences is LLaMAntino-3-ANITA-8B, with an accuracy of 76.56. It is important to note that the model performance increase of 11.72 points with respect to the corresponding Meta-LLama-3-8b (that achieves an accuracy of 64.84): these results stress the effectiveness of the finetuning for the Italian language. Across the LLaMa-based models the LLaMAntino-3-ANITA-8B is still the best performing model, followed by Meta-LLama-3-8b and with a larger gap by LLama-2-7b. The Qwen2-7B model is the best answering to the task on unmarked sentences.

On the NC sentences, similar patterns to the one observed in the previous subdataset emerge. In particular, the NC inverse sentences are harder than the corresponding NC canonical: the average model accuracy is $81.88(\pm11.78)$ on NC canonical sentences, while the accuracy on NC inverse sentences is much lower, with an average value of $64.06(\pm28.26)$. Also in this case, the results demonstrate that models are affected by different syntactic patterns. The model that better capture the right information to extract is modello-italia-9b on both NC inverse and NC-canonical sentences. Although the performance of Llama-2-7b is rather low on inverse NC sentences (the model tends to generate very often the PP_{subj}), the remaining LLaMA-base models achieve better performance on both tasks.

Finally, results on the MVP task further confirm the models' behavior observed on the previous two tasks: the inversion of the subject and verb positions causes the models to perform worst on MVP post sentences ($87.5(\pm19.38)$ average accuracy) with respect to MVP pre ($68.23(\pm10.37)$ average accuracy). The average drop in performance is larger than in previous subtasks: these results confirm that the inversion of the subject, even in basic sentences, can degrade models' understanding. Modello-italia-9b – probably due to the limited length of the input sentences – tends to replicate the input sentences. The other models solve the tasks with excellent accuracy in the MVP pre sentences.

3.2. Localizing Layers responsible for Syntax

After quantifying the impact of different syntactic structures on model performance, we can identify the attention heads and levels of the models that mostly encodes syntax. In Figure 1 the number of responsible head at each layer of the models is reported for the Q&A task on NC sentences, (the remaining tasks are in Appendix A.3).

The general trend is that the most active in identifying relevant words during response generation layers are comprised between layer 19 and 25. Moreover, for all models, the layers we identify as responsible often handle multiple syntactic structures. The most noticeable result is that for the same task, the same activation trend emerges across all sentences.

A large number of responsible attention heads appear around layer 19 to 27 in LLaMAntino-3-ANITA-8B and Meta-Llama-3-8B. Layer 21, in particular, is the layer with the most responsible heads both in NC and MVP tasks. This layer is predominant also in the OC task, concomitant with layers 19 and 22 (Figure 3a). For Llama-2, we observe the same pattern as the most active layers are between 18 and 25. On the Qwen2-7B model and modello-italia-9b active layers are higher in the architecture: from layer 18 to 24 for Qwen2-7B (with layer 23 being the more active in NC and MVP tasks) and from layer 21 to 31 on NC and MVP senteces for modello-italia-9b. This finding suggests a different interpretation of LLMs layers from that previously observed in BERT [27].

While we could expect some correlation between the accuracy of the task and the capability of the model to identify the correct word in the sentence, the responsible heads appear to be shared across different syntactic structures. Those results suggest that some layers, more than others, encode syntactic information about the role of a word in a sentence. Moreover, different models and architectures seem to share a rather similar organization.

4. Conclusions

In this paper, we have investigated how semantically equivalent sentences are processed by LLMs in Italian when their syntax differs. We tested LLMs trained on the Italian - or with Italian data in the pre-trainig material - and measured how their capabilities in a battery of Q&A tasks that rely on parsing the correct role of words in a sentence to be solved. Our findings confirm that cleft sentences and construction with an inversion of subject and verb are difficult to understand also for LLMs - similarly to what observed for humans. Furthermore, we have identified systematically using token-to-token attribution that syntactic information tends to be encoded in the middle and top layers of LLMs.

References

[1] C. Chesi, P. Canal, Person features and lexical restrictions in italian clefts, Frontiers in Psychology 10 (2019) 2105.

[2] J. King, M. A. Just, Individual differences in syntactic processing: The role of working memory, Journal of memory and language 30 (1991) 580–602.

[3] P. Lorusso, M. P. Greco, C. Chesi, A. Moro, et al., Asymmetries in extraction from nominal copular sentences: a challenging case study for nlp tools, in: Proceedings of the Sixth Italian Conference on Computational Linguistics CLiC-it 2019 (Bari, November 13-15, 2019), CEUR, 2019.

[4] A. Rogers, O. Kovaleva, A. Rumshisky, A primer in BERTology: What we know about how BERT works, Transactions of the Association for Computational Linguistics 8 (2020) 842–866. URL: https://aclanthology.org/2020.tacl-1.54. doi:10.1162/tacl_a_00349.

[5] J. Ferrando, G. Sarti, A. Bisazza, M. R. Costa-jussà, A primer on the inner workings of transformer-based language models, 2024. URL: https://arxiv.org/abs/2405.00208. arXiv:2405.00208.

[6] S. Jain, B. C. Wallace, Attention is not Explanation, in: J. Burstein, C. Doran, T. Solorio (Eds.), 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 3543–3556. URL: https://aclanthology.org/N19-1357. doi:10.18653/v1/N19-1357.

[7] S. Wiegreffe, Y. Pinter, Attention is not not explanation, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 11–20. URL: https://aclanthology.org/D19-1002. doi:10.18653/v1/D19-1002.

[8] K. Clark, U. Khandelwal, O. Levy, C. D. Manning, What does BERT look at? an analysis of BERT's attention, in: T. Linzen, G. Chrupała, Y. Belinkov, D. Hupkes (Eds.), Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Florence, Italy, 2019, pp. 276–286. URL: https://aclanthology.org/W19-4828. doi:10.18653/v1/W19-4828.

[9] S. Serrano, N. A. Smith, Is attention interpretable?, in: A. Korhonen, D. Traum, L. Màrquez (Eds.), Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 2931–2951. URL: https://aclanthology.org/P19-1282. doi:10.18653/v1/P19-1282.

[10] G. Kobayashi, T. Kuribayashi, S. Yokoi, K. Inui, Attention is not only a weight: Analyzing transformers with vector norms, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Pro-

cessing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 7057–7075. URL: https://aclanthology.org/2020.emnlp-main.574. doi:10.18653/v1/2020.emnlp-main.574.

[11] G. Kobayashi, T. Kuribayashi, S. Yokoi, K. Inui, Incorporating Residual and Normalization Layers into Analysis of Masked Language Models, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 4547–4568. URL: https://aclanthology.org/2021.emnlp-main.373. doi:10.18653/v1/2021.emnlp-main.373.

[12] Y. Belinkov, J. Glass, Analysis methods in neural language processing: A survey, Transactions of the Association for Computational Linguistics 7 (2019) 49–72. URL: https://aclanthology.org/Q19-1004. doi:10.1162/tacl_a_00254.

[13] J. Hewitt, C. D. Manning, A structural probe for finding syntax in word representations, in: J. Burstein, C. Doran, T. Solorio (Eds.), 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4129–4138. URL: https://aclanthology.org/N19-1419. doi:10.18653/v1/N19-1419.

[14] G. Jawahar, B. Sagot, D. Seddah, What does BERT learn about the structure of language?, in: A. Korhonen, D. Traum, L. Màrquez (Eds.), Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 3651–3657. URL: https://aclanthology.org/P19-1356. doi:10.18653/v1/P19-1356.

[15] E. S. Ruzzetti, F. Ranaldi, F. Logozzo, M. Mastromattei, L. Ranaldi, F. M. Zanzotto, Exploring linguistic properties of monolingual BERTs with typological classification among languages, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 14447–14461. URL: https://aclanthology.org/2023.findings-emnlp.963. doi:10.18653/v1/2023.findings-emnlp.963.

[16] K. Sinha, R. Jia, D. Hupkes, J. Pineau, A. Williams, D. Kiela, Masked language modeling and the distributional hypothesis: Order word matters pretraining for little, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 2888–2913. URL: https://aclanthology.org/2021.emnlp-main.230. doi:10.18653/v1/2021.emnlp-main.230.

[17] M. Abdou, V. Ravishankar, A. Kulmizev, A. Søgaard, Word order does matter and shuffled language models know it, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 6907–6919. URL: https://aclanthology.org/2022.acl-long.476. doi:10.18653/v1/2022.acl-long.476.

[18] D. Brunato, C. Chesi, F. Dell'Orletta, S. Montemagni, G. Venturi, R. Zamparelli, et al., Accompl-it@ evalita2020: Overview of the acceptability & complexity evaluation task for italian, in: CEUR WORKSHOP PROCEEDINGS, CEUR Workshop Proceedings (CEUR-WS. org), 2020.

[19] EVALITA 2020 — evalita.it, https://www.evalita.it/campaigns/evalita-2020/, 2020.

[20] M. Greco, P. Lorusso, C. Chesi, A. Moro, Asymmetries in nominal copular sentences: Psycholinguistic evidence in favor of the raising analysis, Lingua 245 (2020) 102926.

[21] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Brew, HuggingFace's Transformers: State-of-the-art Natural Language Processing, ArXiv abs/1910.0 (2019).

[22] Qwen2 technical report (2024).

[23] M. Polignano, P. Basile, G. Semeraro, Advanced natural-based interaction for the italian language: Llamantino-3-anita, 2024. arXiv:2405.07101.

[24] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, T. Scialom, Llama 2: Open foundation and fine-tuned chat models, 2023. URL: https://arxiv.org/abs/2307.09288. arXiv:2307.09288.

[25] iGenius | Large Language Model — igenius.ai, https://www.igenius.ai/it/language-models, 2024.

[26] AI@Meta, Llama 3 model card (2024). URL:

https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

[27] I. Tenney, D. Das, E. Pavlick, BERT rediscovers the classical NLP pipeline, in: A. Korhonen, D. Traum, L. Màrquez (Eds.), Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 4593–4601. URL: https://aclanthology.org/P19-1452. doi:`10.18653/v1/P19-1452`.

A. Appendix

A.1. Token-to-token norm-based attribution

As described in Section 2.2, we adopt norm-based token-to-token attribution to spot what is the most relevant word during the generation of the answer in LLMs on our task. The norm based approach is proposed in Kobayashi et al. [10]. Given the query weight matrix W_Q^h, key weight matrix W_K^h, value weight matrix W_V and the attention output weight matrix W_O^h of an attention head h, the norm-based attribution for each token of a sequence X is calculated as the product of the attention weights and the norm of the projected token representation $XW_V^hW_O^h$ (see the original work Kobayashi et al. [10] for a detailed discussion).

$$\mathrm{A}^h(X) := softmax\left(\frac{XW_Q^h\cdot(XW_K^h)^\top}{\sqrt{d_v}}\right)\cdot\|XW_V^hW_O^h\|$$

For our analysis, we consider all rows relative to a token in the answer generated by the model. To assess whether a model understands the syntactic relationship between words, it must focus on relevant words during the generation. In particular, the token with the highest attribution should be one belonging to the relevant word. For example, in Figure 2, the attribution of Meta-Llama-3-8B on one NC sentence is presented. During the generation of the answer (the tokens of the answer index rows in the figure), the most attributed tokens belong to the relevant words in the input (the tokens of the input index columns).

A.2. Prompts to Instruction-Tuned LLMs for the Italian Laguage

Each model has been prompted with four different prompts for each Q&A task (as described in Section 2.1). Here is a complete list of the prompts template used in our experiments: in the template the `{Item}` is the sentence to be analyzed and `{Question}` is replaced with the corresponding comprehension question.

OC and SVO senteces:

- Data la frase `"{Item}"`, rispondi alla seguente domanda:`"{Question}"` Rispondi SOLAMENTE con SI o NO.
- Considera la frase: `"{Item}"`. Rispondi con 'SI' o 'NO' alla seguente domanda:`"{Question}"`
- Considera la frase: `"{Item}"`. `{Question}` Rispondi brevemente, SOLAMENTE con con 'SI' o 'NO'.
- Considera la frase: `"{Item}"`. Rispondi con 'SI' o 'NO'. `{Question}`

NC sentences:

- Data la frase `"{Item}"`, rispondi alla seguente domanda:`"{Question}"` Rispondi in due parole.
- Considera la frase: `"{Item}"`. Rispondi solo con le due parole che rispondono alla seguente domanda:`"{Question}"`
- Considera la frase: `"{Item}"`. `{Question}` Rispondi SOLO con le due parole che rispondono alla seguente domanda.
- Considera la frase: `"{Item}"`. Rispondi solo con due parole. `{Question}`

MVP sentences:

- Data la frase `"{Item}"`, rispondi alla seguente domanda:`"{Question}"` Rispondi solo con un nome.
- Considera la frase: `"{Item}"`. Rispondi solo con il nome che risponde alla seguente domanda:`"{Question}"`
- Considera la frase: `"{Item}"`. `{Question}` Rispondi SOLO con il nome che risponde alla domanda.
- Considera la frase: `"{Item}"`. Rispondi solo con un nome. `{Question}`

A.3. Responsible Attention Heads per Layer in each subtask

In Figure 3, the responsible attention heads per layer is depicted. As described in Section 3.2, some layers tend to demonstrate a high number of attention heads responsible for the generation. In particular, layers around layer 20 seem to focus more on relevant words for the correct generation of the answer than the other. Since the correct generation implies the capability of understanding the role of different words by a model, we claim that those level encodes some kind of syntactic information. It is worth noticing that similar layers are responsible for the different sub tasks, in particular for the LLaMa-base models and for Qwen-2-7b model.

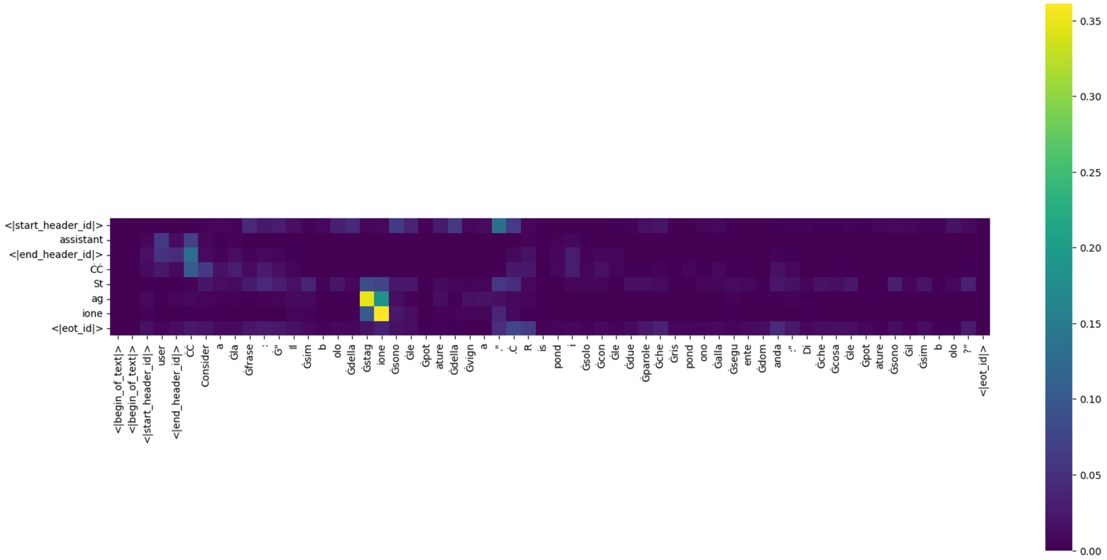
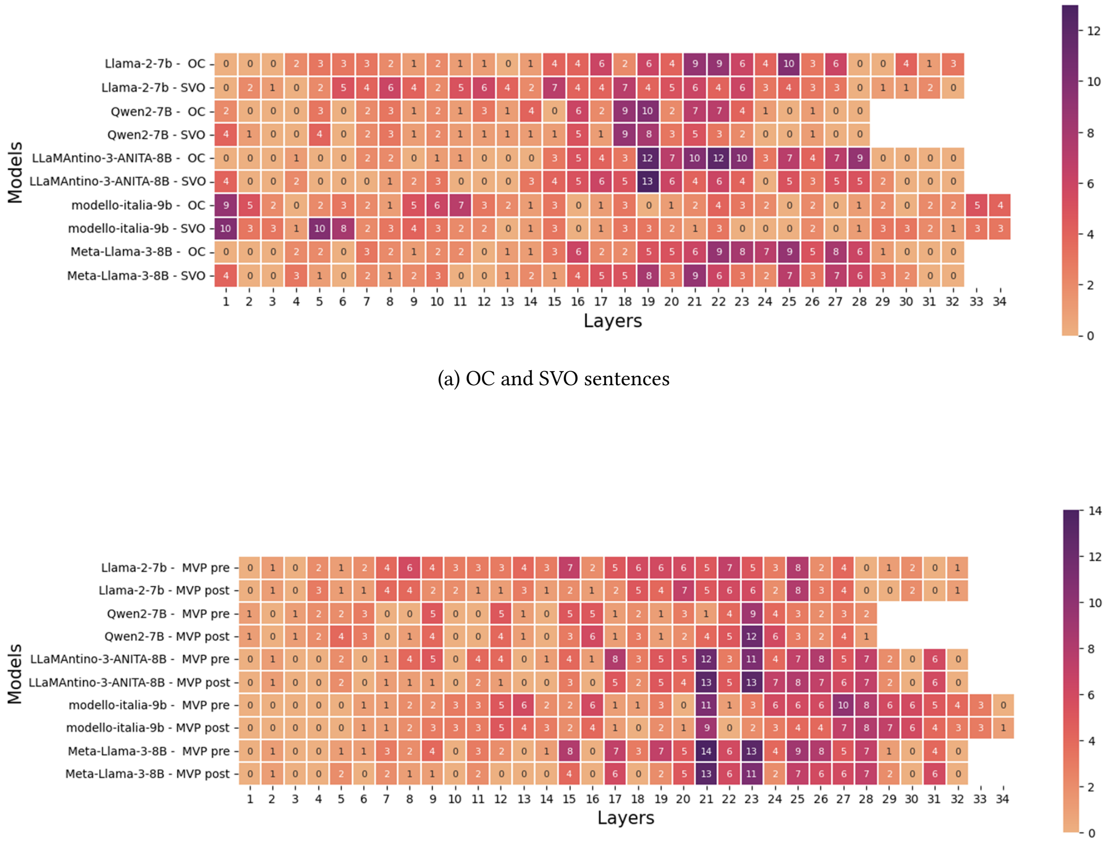


Figure 2: Norm-based attribution matrix of Meta-Llama-3-8B on one example of the task presented in Section 2.1 on NC sentences.



(a) OC and SVO sentences

(b) MVP sentences

Figure 3: Number of responsible heads per layer in the Q&A task defined over two task: OC and SVO sentences (3a) and MVP sentences (3b).