Exploring Italian sentence embeddings properties through multi-tasking

Vivi Nastase[1,*], Giuseppe Samo[1], Chunyang Jiang[1,2] and Paola Merlo[1,2]

[1]*Idiap Research Institute, Martigny, Switzerland*
[2]*University of Geneva, Geneva, Switzerland*

Abstract

We investigate to what degree existing LLMs encode abstract linguistic information in Italian in a multi-task setting. We exploit curated synthetic data on a large scale – several Blackbird Language Matrices (BLMs) problems in Italian – and use them to study how sentence representations built using pre-trained language models encode specific syntactic and semantic information. We use a two-level architecture to model separately a compression of the sentence embeddings into a representation that contains relevant information for a task, and a BLM task. We then investigate whether we can obtain compressed sentence representations that encode syntactic and semantic information relevant to several BLM tasks. While we expected that the sentence structure – in terms of sequence of phrases/chunks – and chunk properties could be shared across tasks, performance and error analysis show that the clues for the different tasks are encoded in different manners in the sentence embeddings, suggesting that abstract linguistic notions such as constituents or thematic roles does not seem to be present in the pretrained sentence embeddings.

L'obiettivo di questo lavoro è indagare fino a che punto gli attuali LLM apprendono rappresentazioni linguistiche astratte in configurazioni multitask. Utilizzando dati sintetici curati su larga scala di vari problemi BLM in italiano, studiamo come le rappresentazioni di frasi costruite da modelli di linguaggio pre-addestrati codifichino le informazioni semantiche e sintattiche. Abbiamo utilizzato un'architettura a due livelli per modellare separatamente, da un lato, la compressione degli embeddings delle frasi di input in una rappresentazione che contiene informazioni rilevanti per i tasks BLM e, dall'altro lato, il BLM stesso. Abbiamo poi verificato se fosse possibile ottenere rappresentazioni compresse di frasi che codificano informazioni sintattiche e semantiche rilevanti per i diversi tasks BLM. Contrariamente alla predizione che la struttura della frase - in termini di sequenza di frasi/chunks - e le proprietà dei chunk possano essere condivise tra i vari tasks, i risultati e l'analisi degli errori mostrano che gli indizi per i diversi task sono codificati in modo diverso negli embeddings delle frasi. Questo risultato suggerisce che nozioni linguistiche astratte come i costituenti o i ruoli tematici non vi sembrano essere presenti.

Keywords

synthetic structured data, multi-task, diagnostic studies of deep learning models

1. Introduction

Driven by increasing computational scale and progress in deep learning techniques, NLP models can rival human capabilities on established benchmarks. New benchmarks, then, that capture deeper levels of language understanding must be created and analysed [1].

Blackbird's Language Matrices (BLM) [2] is a recent task inspired by visual tests of analytic intelligence (Raven Progressive Matrices/RPMs, [3]). The BLM tasks have cast light on whether the correct predictions in previously studied linguistic problems, e.g. number agreement or verb alternations, stem from sentence embeddings that encode deeper linguistic information, such as syntactic structure and semantic properties of phrases [4, 5, 6]. We found that higher-level information – syntactic structure and argument structure – can be assembled from the information encoded in the sentence embeddings. This, however, may not be due to a deeper understanding of such information encoded by LLMs, but rather because of useful surface indicators [7].

In this paper, we adopt BLMs to investigate whether current pretrained models encode abstract linguistic notions, such as constituents, and are able to do so in a manner that comprises both functional elements, such as pronouns, demonstratives and lexical elements, such as nominal constituents.

We concentrate on Italian, and study several grammatical problems whose solutions can theoretically help each other, in a multi-task setting. We adopt a two-level architecture developed specifically to model what we know about how humans solve puzzles similar to BLMs [8]. Level 1 aims to obtain compressed sentence representations that capture information about constituents and their properties; level 2 uses the compressed sentence representations to solve a BLM problem. This architecture provides a tool to study how LLMs encode different types of syntactic and semantic information.

CLiC-it 2024: 10th Italian Conference on Computational Linguistics, Dec 04 — 06, 2024, Pisa, Italy
*Corresponding author.
vivi.a.nastase@gmail.com (V. Nastase); giuseppe.samo@idiap.ch (G. Samo); chunyang.jiang42@gmail.com (C. Jiang); Paola.Merlo@unige.ch (P. Merlo)

We make two contributions: (i) an initial core BLM dataset for Italian that covers linguistic problems of different nature; (ii) single and multi-task experiments that provide new insights into the information encoded by LLMs. The datasets are available at https://www.idiap.ch/dataset/(blm-agri|blm-causi|blm-odi) and the code at https://github.com/CLCL-Geneva/BLM-SNFDisentangling.

2. Related Work

Multi-task learning has been popular in improving NLP systems' performance by using knowledge shared across multiple tasks [9].

Multi-task learning architectures include parallel, hierarchical, and modular designs [10]. Parallel architectures share intermediate layers across tasks, conducive to efficient knowledge transfer [11]. Hierarchical architectures capture task dependencies by layering task-specific modules on shared bases. Modular approaches selectively share components among tasks to balance between generalisation and task-specific optimisation [12]. These training strategies are not mutually exclusive and can be combined.

Multi-task learning can be used efficiently in resource-constrained environments, to counter data scarcity and overfitting: aggregating training data and sharing parameters across related tasks acts as a form of data augmentation [13].

Effective multi-task learning depends on the relatedness of the tasks involved. Tasks that are similar or have similar objectives tend to benefit more from shared representations. This observation has been used in various NLP tasks, including named entity recognition [14], text generation[15], and machine translation [16], among others. Selecting related tasks that contribute positively to the shared model's training is important and remains an active area of research [9].

Pretrained large language models exhibit general-purpose abilities and knowledge, with high results with little or no fine-tuning on downstream tasks [17, 18]. We can then regard these language models as the results of "multi-task" learning, and our aim here is to test whether sentence embeddings obtained from these models encode syntactic and semantic information consistently, such that different BLM problems that rely on similar linguistic information draw on the same clues from these representations. In particular, we will use BLM tasks on subject-verb agreement – which relies on chunk structure and the chunks' grammatical number properties – and on verb alternations – which relies on chunk structure and the chunks' semantic role properties – to test whether chunk structure is encoded in a manner that allows for it to be shared by the two tasks.

BLM agreement problem (BLM-AgrI)

Context Template				
NP-sg	PP1-sg		VP-sg	
NP-pl	PP1-sg		VP-pl	
NP-sg	PP1-pl		VP-sg	
NP-pl	PP1-pl		VP-pl	
NP-sg	PP1-sg	PP2-sg	VP-sg	
NP-pl	PP1-sg	PP2-sg	VP-pl	
NP-sg	PP1-pl	PP2-sg	VP-sg	
Answer set				
NP-pl	PP1-pl	PP2-sg	VP-pl	Correct
NP-pl	PP1-pl	et PP2-sg	VP-pl	Coord
NP-pl	PP1-pl		VP-pl	WNA
NP-pl	PP1-sg	PP1-sg	VP-pl	WN1
NP-pl	PP1-pl	PP2-pl	VP-pl	WN2
NP-pl	PP1-pl	PP2-pl	VP-sg	AEV
NP-pl	PP1-sg	PP2-pl	VP-sg	AEN1
NP-pl	PP1-pl	PP2-sg	VP-sg	AEN2

Figure 1: BLM instances for verb-subject agreement, with two attractors. We build candidate answers displaying one of two types of errors: (i) sequence errors: WNA= wrong nr. of attractors; WN1= wrong gram. nr. for 1^{st} attractor noun (N1); WN2= wrong gram. nr. for 2^{nd} attractor noun (N2); (ii) grammatical errors: AEV=agreement error on the verb; AEN1=agreement error on N1; AEN2=agreement error on N2.

3. The BLM task and the BLM Italian datasets

Raven's progressive matrices are multiple-choice completion IQ tests, whose solution requires discovering underlying generative rules of a sequence of images [3].

A similar task has been developed for linguistic problems, called Blackbird Language Matrices (BLMs) [2], as given in Figure 1, which illustrates the template of a BLM agreement matrix. A BLM comprises a context and an answer set. The context is a sequence of sentences generated following the relevant rules of a given linguistic phenomenon under investigation and that this way implicitly illustrates these grammatical properties. This sequence also follows some extra-linguistic progression rules. Each context is paired with a set of candidate answers. The answer sets contain minimally contrastive examples built by corrupting some of the generating rules.

The BLM Italian datasets consists of BLMs focused on the property of subject-verb agreement and two transitive-intransitive alternations: the change-of-state alternation and the object-drop alternation.

3.1. BLM-AgrI – subject-verb agreement in Italian

The BLM-AgrI dataset is created by manually translating the seed French sentences [4] into Italian by a native speaker, one of the authors, and then generating the full dataset following the same process of lexical augmentation and sentence shuffling among instances described in [4]. The internal nominal structure in these languages is very similar, so translations are almost parallel. An illustrative, simplified example for Italian is provided in Figure 7, in the appendix. The dataset comprises three subsets of increasing lexical complexity (called Type I, Type II and Type III) to test the ability of the system to handle item novelty.

3.2. BLM-CausI and BLM-OdI

While BLM-AgrI tests information about a formal grammatical property, agreement, the Causative (*Caus*) and Object-drop (*Od*) alternation datasets test lexical semantic properties of verbs, their ability to enter or not a causative alternation. *Caus* represents the causative/inchoative alternation, where the object of the transitive verb bears the same semantic role (Patient) as the subject of the intransitive verb (*L'artista ha aperto la finestra/La finestra si è aperta* 'The artist opened the window'/'The window opened'). The transitive form of the verb has a causative meaning. In contrast, the subject in *Od* bears the same semantic role (Agent) in both the transitive and intransitive forms (*L'artista dipingeva la finestra/L'artista dipingeva* 'the artist painted the window'/'the artist painted') and the verb does not have a causative meaning [19, 20].

BLM-CausI context and answers The context set of the verb alternation varies depending on the presence of one or two arguments and their attributes (agents, Ag; patients, Pat) and the active (Akt) and passive (Pass) or passive voice of the verb. The non-linguistic factor that structures the sequence is an alternation every two items between a prepositional phrase introduced by any preposition (e.g., *in pochi secondi*, P-NP) and a PP introduced by the agentive da-NP (e.g., *dall'artista*, da-Ag/da-Pat).

The answer set is composed of one correct answer and contrastive wrong answers, all formed by the same four elements: a verb, two nominal constituents and a prepositional phrase. Figure 2 shows the template.[1]

BLM-OdI Context and Answers The BLM for *Od* is the same as for *Caus*, but here the passive voice serves as a confounding element and one of the contrastive answers for *Caus* is, in fact, the correct answer here.

[1]Following BLM formal specifications [2], we build the errors representing violations of internal (*I*), external (*E*) and relational (*R*) rules of the BLM, and their combination (e.g. *IE IER*, etc.). This information is used in the first part of the error acronym. The second part of the errors' label indicates the structure the sentence represent: intransitive (INT), passive (PASS), Transitive (TRANS) or, in some cases, the NP introduced by the *da* preposition (WRBY).

The template is also in Figure 2. Due to the asymmetry between the two classes of verbs, the contexts of the BLMs minimally differ in the intransitive followed by P-NP (sentence 7). The correct answer also varies across the two groups, although in both cases it is an intransitive form with a da-NP. Examples are shown in the Appendix.

CAUS CONTEXT					CAUS ANSWERS		
1	Ag	Akt	Pat	P-NP	1	Pat Akt da-NP	**CORRECT**
2	Ag	Akt	Pat	da-NP	2	Ag Akt da-NP	I-INT
3	Pat	Pass	da-Ag	P-NP	3	Pat Pass da-Ag	ER-PASS
4	Pat	Pass	da-Ag	da-NP	4	Ag Pass da-Pat	IER-PASS
5	Pat	Pass		P-NP	5	Pat Akt Ag	R-TRANS
6	Pat	Pass		da-NP	6	Ag Akt Pat	IR-TRANS
7	Pat	Akt		P-NP	7	Pat Akt da-Ag	E-WRBY
?	???				8	Ag Akt da-Pat	IE-WRBY
OD CONTEXT					**OD ANSWERS**		
1	Ag	Akt	Pat	P-NP	1	Pat Akt da-NP	I-INT
2	Ag	Akt	Pat	da-NP	2	Ag Akt da-NP	**CORRECT**
3	Pat	Pass	da-Ag	P-NP	3	Pat Pass da-Ag	IER-PASS
4	Pat	Pass	da-Ag	da-NP	4	Ag Pass da-Pat	ER-PASS
5	Pat	Pass		P-NP	5	Pat Akt Ag	IR-TRANS
6	Pat	Pass		da-NP	6	Ag Akt Pat	R-TRANS
7	Ag	Akt		P-NP	7	Pat Akt da-Ag	IE-WRBY
?	???				8	Ag Akt da-Pat	E-WRBY

Figure 2: BLM contexts answers and their location of errors (see text) for the Change of state group (*Caus*) and the object drop (*Od*) class.

We illustrate the data in Figure 8 in the appendix with the Italian Change-of-state verb *chiudere* 'close'.

Lexicalisation In line with previous work on BLMs, each dataset also contains a varying amount of lexicalisation. In type I the lexical material of the sentences within a single context does not change, in type II only the verb remains the same, in type III data all words can change (Figure 9, in the appendix).

3.3. Dataset statistics

Each subset is split 90:20:10 into train:dev:test subsets. The training and testing are disjoint (agreement data is split based on the correct answer, the alternations data based on the verb). Agreement has 230 test instances for type I, 4121 for types II and III. The verb alternations have 240 test instances for all subsets. We randomly sample a number of training instances, depending on the experimental set-up.

4. Multi-task representations

Sentence embeddings encode much information from the input sentence – lexical, syntactic, semantic, and possibly other types of information. Previous experiments have shown that sentence embeddings can be compressed into very small representations (vectors of size 5) that

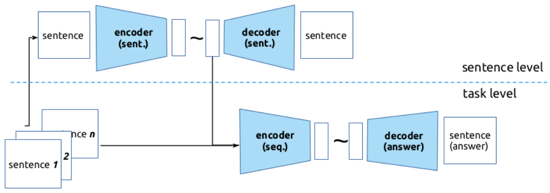

Figure 3: A two-level VAE: the sentence level learns to compress a sentence into a representation useful to solve the BLM problem on the task level.

encode information about the structure of the sentence in terms of chunks and their properties, such that they contribute to finding the sequence patterns in BLMs [6]. In this work, we investigate whether several BLM tasks can share the same structural information from a sentence embedding. Towards this end, we built a multi-task version of a two-level system, illustrated in Figure 3. In this system, one level processes individual sentences and learns to compress them into small vectors that retain information pertinent to a task and the other level uses the compressed sentence representation to find patterns across an input sequence to solve a BLM task. The multi-task variation consists in a single shared sentence-level component, and multiple task components, one for each of the BLM tasks.

The BLM problems encode a linguistic phenomenon through data that has structure on multiple levels – within sentences, and across a sequence of sentences. We can exploit this structure to develop an indirectly supervised approach to discover and use these different levels of structure. We thus model the solving of a BLM task as a two-step process: (i) compress individual sentences into a representation that emphasizes the sentence structure relevant to the BLM problem (e.g. chunks and their grammatical number for the subject-verb agreement task) (ii) use the compressed representations to detect the sequence-level pattern and solve the BLM task. This two-step process has been shown to be used by people solving visual intelligence tests [21]. In our case, this setup allows us to investigate whether the sentence level can be guided to learn shared information, relevant to the different linguistic tasks described in section 3.

We implement this approach in the two-level intertwined architecture illustrated in Figure 3, and described in detail elsewhere [6]. The data is pre-encoded with Electra [18].[2] The sentence representations is provided by the embedding of the [CLS] token.[3]. We chose Electra because of its stronger sentence-level supervision signal, which leads to higher results when testing the encoding of structural information compared to BERT, RoBERTa, and models tuned by semantic similarity [6].

The two levels are learned together. The input is a BLM instance which is processed on the fly to produce training instances for the sentence level for each sentence in_k in the input sequence S. The compressed sentence representations on the latent layer z_{in_k} are stacked and passed as input to the task level, which produces a sentence representation $answ$ as output, which is compared to the answer set of the respective BLM instance A.

The sentence level uses a variational encode-decoder architecture to learn how to compress on the latent layer a representation that captures relevant structural information. We guide the system towards this representation by constructing a contrastive set of candidates for comparison with the reconstructed input. The correct output (out^+) is the same as the input (in), and a selection of other sentences from the input sequence will be the contrastive negative outputs ($Out^- = \{out_i^-, i = 1, N_{negs}\}$, $N_{negs} = 7$ (note that an input sequence consists of sentences with different patterns to each other – Figure 1 and 2). We use a max-margin loss function to take advantage of the contrastive answers, $\hat{in}$ is the reconstructed input sentence from the sampled latent vector z_{in}:

$$loss_{sent}(in) = maxM(\hat{in}, out^+, Out^-) \\ + KL(z_{in}||\mathcal{N}(0,1))$$

$$maxM(\hat{in}, out^+, Out^-) = \\ max(0, 1 - cos(\hat{in}, out^+) \\ + \frac{\sum_{out_i^- \in Out^-} cos(\hat{in}, out_i^-)}{N_{negs}})$$

The loss at the task level for input sequence S is computed in a similar manner for the constructed answer $answ$, but relative to the answer set $\mathcal{A}$ and the correct answer a_c of the task:

$$loss_{task}(S) = maxM(answ, a_c, A \setminus \{a_c\}) \\ + KL_{seq}(z_S|\mathcal{N}(0,1)).$$

The loss of the two-level systems is:

$$loss(S) = \sum_{in_k \in S} loss_{sent}(in_k) + loss_{task}(S)$$

The input batches are shuffled, to alternate between tasks during training, and avoid getting stuck in a local maximum for one of the tasks.

5. Multi-task results

Previous published work from our group and current ongoing work has benchmarked the problems generated

[2]Italian Electra (E-It) pretrained model: dbmdz/electra-base-italian-xxl-cased-discriminator. Multi-lingual Electra (E-M) model: google/electra-base-discriminator.

[3]To simplify the discussion of the method, we write "sentence" instead of "sentence embedding", when discussing the system.

by some of these datasets [4, 5]. This work has shown that information about the syntactic phrases in a sentence and their properties can be obtained from sentence embeddings, and this information is helpful in solving the BLM tasks. We had studied these tasks separately, and investigate here whether such structure is encoded in the sentence embeddings, or whether it is assembled based on shallower patterns within the sentence representations.

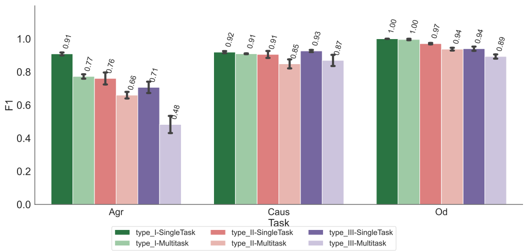

Figure 4: Performance comparison across single-task and multi-task training paradigms for the three subtasks (single task darker shade of each colour, multi-task lighter shade), trained on type-I data, tested on the three types, and averaged over three independent runs. Results obtained using the Italian Electra pretrained model.

Discussion We expect that if the multi-task setup succeeds in sharing information across tasks, then the results on the individual test data will be at least as good as when learning tasks individually, given that the multi-task setup uses a larger training set data – the union of the training sets of the individual tasks. But, overall, this does not seem to be the case.

As the results in Figure 4 show (and also the detailed results in Tables 1-2 for the Italian Electra pretrained model, and in Tables 3-4 for a multilingual Electra pretrained model), single-task training outperforms multi-tasking in the agreement and verb alternation subtasks. The drop suggests that the multi-task model is not able to learn shared properties for these tasks, and forcing it to do so leads to a model that is not optimal for either of them. Both tasks require information about the syntactic structure (or sequence of phrases), while each requires different phrase properties – grammatical number for the agreement task, and semantic properties for the verb alternation. While the system is able to distil all this information from sentence embeddings in the single-task setting, it is not able to compress it into a shared representation when learning the tasks together.

The *Od* single-task and multi-task have comparable performance, probably because the *Od* tasks involve a simpler alternation than the *Caus* task. They do not have a causative meaning and do not require a change in the semantic role of the subjects.

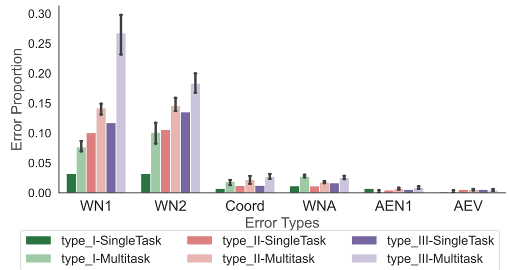

Figure 5: Error analysis for agreement: multi- vs. single task, training on type I data, testing on all.

The comparison of all the tasks suggests that some syntactic and semantic regularities –such as constituents, grammatical number and semantic roles– cannot be encoded together as they compete with each other when the system learns to distil them from the pretrained sentence embeddings.

Error Analysis For the agreement task, errors on the grammatical number of the attractor nouns (WN1, WN2) are high under both paradigms. These are "sequence errors", indicating that the system was not able to detect the patterns in the input sequence, possibly because individual sentence structures were not properly detected. Previous experiments have shown, though, that in the single-task setting, the sentence level does manage to compress the desired information [6]. The fact that both these errors increase in the multi-task setting indicates that the information compression on the sentence level is less successful than in the single-task setting.

For the alternation tasks, error patterns vary, although their distributions remain similar between single-task and multi-task environments. We observe an overall increase of error proportions in the multi-task environment. Specifically, mistakes of the type I-INT are frequent in type III data for the *Caus* task. These errors incorrectly map the thematic roles onto the syntax of the arguments (e.g. *L'artista si è chiuso* 'the artist closed' or *La carbonara mangiava* 'the carbonara was eating'). In the same dataset, we also note an increase of errors related to the last constituent in type I and type II data (errors of type E-WrBy, e.g. *La finestra si chiuse dall'artista* 'the window closed by the artist'). Finally, for the *Od* task, we remark that *R-trans* errors are not the most prominent —these are the errors resulting in standard transitive clauses (e.g., *L'artista dipinse un paesaggio* 'the artist painted a landscape')— and do not increase in multi-task environments, suggesting that the chosen answer is not derived from some forms of transitive bias [22].

An overall comparison shows that the error patterns vary across subtasks. This variety in error patterns confirms that the different dimensions (types of alternations, levels of lexicalisation and single and multi-task learning)

(a) *Caus* task error analysis

(b) *Od* task error analysis

Figure 6: Error analysis between single and multi-task training paradigms trained on type-I data, tested on the three types, as averages over three runs (single task darker shade of each colour, multi-task lighter shade). For the *Caus* and *Od* tasks, we report only three representative error types of *I*, *E* and *R*.

are separate uncorrelated dimensions. It also indicates that the differences in the F1 results shown in Figure 4 are real, despite the more homogeneous trends exhibited by these aggregated F1 numbers.

6. Conclusions

In this paper, we have presented curated synthetic datasets of Italian on two linguistic phenomena of an heterogeneous nature, such as agreement and verbal transitive/intransitive alternation, embedded in the BLM task.

The results on the performance and the error analysis of a tailored two-level architecture have shown that multitask environments do not help, suggesting that abstract linguistic notions, such as constituents or thematic roles do not seem to be present in the learning process.

Current work is developing new analyses and architectures to probe further in the encoding of information in sentence embeddings and creating new BLM problems across various languages and linguistic phenomena.

Acknowledgments

We gratefully acknowledge the partial support of this work by the Swiss National Science Foundation, through grant SNF Advanced grant TMAG-1_209426 to PM.

References

[1] S. Ruder, Challenges and Opportunities in NLP Benchmarking, http://www.ruder.io/nlp-benchmarking, 2021.

[2] P. Merlo, Blackbird language matrices (BLM), a new task for rule-like generalization in neural networks: Motivations and formal specifications, ArXiv cs.CL 2306.11444 (2023). URL: https://doi.org/10.48550/arXiv.2306.11444. doi:10.48550/arXiv.2306.11444.

[3] J. C. Raven, Standardization of progressive matrices, British Journal of Medical Psychology 19 (1938) 137–150.

[4] A. An, C. Jiang, M. A. Rodriguez, V. Nastase, P. Merlo, BLM-AgrF: A new French benchmark to investigate generalization of agreement in neural networks, in: Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Dubrovnik, Croatia, 2023, pp. 1363–1374. URL: https://aclanthology.org/2023.eacl-main.99.

[5] V. Nastase, P. Merlo, Grammatical information in BERT sentence embeddings as two-dimensional arrays, in: B. Can, M. Mozes, S. Cahyawijaya, N. Saphra, N. Kassner, S. Ravfogel, A. Ravichander, C. Zhao, I. Augenstein, A. Rogers, K. Cho, E. Grefenstette, L. Voita (Eds.), Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 22–39. URL: https://aclanthology.org/2023.repl4nlp-1.3. doi:10.18653/v1/2023.repl4nlp-1.3.

[6] V. Nastase, P. Merlo, Are there identifiable structural parts in the sentence embedding whole?, in: Y. Belinkov, N. Kim, J. Jumelet, H. Mohebbi, A. Mueller, H. Chen (Eds.), Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Miami, Florida, US, 2024, pp. 23–42. URL: https://aclanthology.org/2024.blackboxnlp-1.3.

[7] A. Lenci, Understanding natural language understanding systems, Sistemi intelligenti, Rivista quadrimestrale di scienze cognitive e di intelligenza artificiale (2023) 277–302. URL: https://www.rivisteweb.it/doi/10.1422/107438. doi:10.1422/107438.

[8] P. A. Carpenter, M. A. Just, P. Shell, What one intelligence test measures: A theoretical account of the processing in the Raven Progressive Matri-

ces Test., Psychological Review 97 (1990) 404–431. doi:10.1037/0033-295X.97.3.404.
[9] Z. Zhang, W. Yu, M. Yu, Z. Guo, M. Jiang, A survey of multi-task learning in natural language processing: Regarding task relatedness and training methods, in: A. Vlachos, I. Augenstein (Eds.), Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Dubrovnik, Croatia, 2023, pp. 943–956. URL: https://aclanthology.org/2023.eacl-main.66. doi:10.18653/v1/2023.eacl-main.66.
[10] S. Chen, Y. Zhang, Q. Yang, Multi-task learning in natural language processing: An overview, ACM Computing Surveys (2021).
[11] S. Ruder, An overview of multi-task learning in deep neural networks, arXiv preprint arXiv:1706.05098 (2017).
[12] J. Pfeiffer, S. Ruder, I. Vulić, E. M. Ponti, Modular deep learning, arXiv preprint arXiv:2302.11529 (2023).
[13] T. Standley, A. Zamir, D. Chen, L. Guibas, J. Malik, S. Savarese, Which tasks should be learned together in multi-task learning?, in: International conference on machine learning, PMLR, 2020, pp. 9120–9132.
[14] B. Zhou, X. Cai, Y. Zhang, X. Yuan, An end-to-end progressive multi-task learning framework for medical named entity recognition and normalization, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 6214–6224. URL: https://aclanthology.org/2021.acl-long.485. doi:10.18653/v1/2021.acl-long.485.
[15] Z. Hu, H. P. Chan, L. Huang, MOCHA: A multi-task training approach for coherent text generation from cognitive perspective, in: Y. Goldberg, Z. Kozareva, Y. Zhang (Eds.), Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 10324–10334. URL: https://aclanthology.org/2022.emnlp-main.705. doi:10.18653/v1/2022.emnlp-main.705.
[16] Y. Wang, C. Zhai, H. Hassan, Multi-task learning for multilingual neural machine translation, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 1022–1034. URL: https://aclanthology.org/2020.emnlp-main.75. doi:10.18653/v1/2020.emnlp-main.75.
[17] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.
[18] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, Electra: Pre- training text encoders as discriminators rather than generators, in: ICLR, 2020, pp. 1–18.
[19] B. Levin, English Verb Classes and Alternations A Preliminary Investigation, University of Chicago Press, Chicago and London, 1993.
[20] P. Merlo, S. Stevenson, Automatic verb classification based on statistical distributions of argument structure, Computational Linguistics 27 (2001) 373–408.
[21] P. A. Carpenter, M. A. Just, P. Shell, What one intelligence test measures: a theoretical account of the processing in the raven progressive matrices test., Psychological review 97 (1990) 404.
[22] K. Kann, A. Warstadt, A. Williams, S. R. Bowman, Verb argument structure alternations in word and sentence embeddings, in: Proceedings of the Society for Computation in Linguistics (SCiL) 2019, 2019, pp. 287–297. URL: https://aclanthology.org/W19-0129. doi:10.7275/q5js-4y86.

A. Appendix

A.1. An Italian example for the subject-verb agreement BLM

	Context			
1	Il vaso	con il fiore		si è rotto.
2	I vasi	con il fiore		si sono rotti.
3	Il vaso	con i fiori		si è rotto.
4	I vasi	con i fiori		si sono rotti.
5	Il vaso	con il fiore	del giardino	si è rotto.
6	I vasi	con il fiore	del giardino	si sono rotti.
7	Il vaso	con i fiori	del giardino	si è rotto.
8	???			

	Answer set	
1	Il vaso con i fiori e il giardino si è rotto.	coord
2	**I vasi con i fiori del giardino si sono rotti.**	correct
3	Il vaso con il fiore si è rotto.	WNA
4	I vasi con il fiore del giardino si sono rotti.	WN1
5	I vasi con i fiori dei giardini si sono rotti.	WN2
6	Il vaso con il fiore del giardino si sono rotti.	AEV
7	Il vaso con i fiori del giardino si sono rotti.	AEN1
8	Il vaso con il fiore dei giardini si sono rotti.	AEN2

Figure 7: An illustrative example for the BLM instances for verb-subject agreement, with 2 attractors (*fiore* 'flower', *giardino* 'garden'), with candidate answer set.

A.2. Verb alternation examples

	Caus - Context
1	Una stella del cinema chiuse la sua carriera con forza
2	Una stella del cinema chiuse la sua carriera da pochissimo tempo
3	La sua carriera fu chiusa da una stella del cinema con forza
4	La sua carriera fu chiusa da una stella del cinema da pochissimo tempo
5	La sua carriera fu chiusa con forza
6	La sua carriera fu chiusa da pochissimo tempo
7	La sua carriera si chiuse con forza
8	???

	Caus - Answers
1	**La sua carriera si chiuse da pochissimo tempo**
2	Una stella del cinema si chiuse da pochissimo tempo
3	La sua carriera fu chiusa da una stella del cinema
4	Una stella del cinema fu chiusa dalla sua carriera
5	La sua carriera chiuse una stella del cinema
6	Una stella del cinema chiuse la sua carriera
7	La sua carriera si chiuse da una stella del cinema
8	Una stella del cinema si chiuse dalla sua carriera

Figure 8: Examples for the *Caus* BLMs for the Italian verb *chiudere* 'close' belonging to *Caus* class

	Od, typeI - Context
1	La turista mangia una carbonara in un secondo
2	La turista mangia una carbonara da mezz'ora
3	Una carbonara è mangiata dalla turista in un secondo
4	Una carbonara è mangiata dalla turista da mezz'ora
5	Una carbonara è mangiata in un secondo
6	Una carbonara è mangiata da mezz'ora
7	La turista mangia in un secondo
8	???

	Od, typeI - Answers
1	Una carbonara mangia da mezz'ora
2	**La turista mangia da mezz'ora**
3	Una carbonara è mangiata dalla turista
4	La turista è mangiata da una carbonara
5	Una carbonara mangia la turista
6	La turista mangia una carbonara
7	Una carbonara mangia dalla turista
8	La turista mangia da una carbonara

	Od, typeII - Context
1	La zia mangia una bistecca nella sala grande
2	La presidente può mangiare una bistecca da programma
3	La specialità della casa deve essere mangiata dalla turista nella sala grande
4	Una bistecca fu mangiata dalla presidente da sola
5	La specialità della casa deve essere mangiata in un secondo
6	Una bistecca deve poter essere mangiata da sola
7	La turista deve mangiare con fame
8	???

	Od, typeII - Answers
1	La specialità della casa può mangiare da sola
2	**La squadra di calcio deve mangiare da mezz'ora**
3	Una bistecca è mangiata dalla turista
4	La squadra di calcio può essere mangiata da una carbonara
5	La pasta col pomodoro può mangiare la squadra di calcio
6	La squadra di calcio mangia una bistecca
7	La specialità della casa deve poter mangiare dalla turista
8	La presidente mangia da una bistecca

	Od, typeIII - Context
1	L'attore deve canticchiare un motivetto dopo il festival
2	L'amica di mia mamma deve cucire la tasca da qualche giorno
3	L'inno nazionale può essere cantato dal vincitore del festival con solo pianoforte
4	Una bistecca deve essere mangiata dalla turista da sola
5	Il manuale è insegnato nell'aula magna
6	Questi attrezzi devono essere intagliati da manuale
7	I due fratelli studiano con molta attenzione
8	???

	Od, typeIII - Answers
1	La pasta frolla deve impastare da sola
2	**L'autrice deve poter scrivere da qualche giorno**
3	I libri di testo devono poter essere studiati dai candidati
4	Questi stilisti devono poter essere tessuti dai vestiti per la parata
5	Questi motivi greci possono tessere questi stilisti
6	L'idraulico saldò i cavi del lampadario
7	La stanza pulisce da una delle propretarie dell'albergo
8	Le sommozzatrici pescarono da delle trote

Figure 9: Examples of *Od* BLMs for type I, type II and type III

B. Results

B.1. Results with the Italian Electra pretrained model: dbmdz/electra-base- italian-xxl-cased-discriminator

train on	test on	task		
		agreement	*Caus*	*Od*
type I	type I	**0.772 (0.011)**	0.910 (0.002)	**0.996 (0.003)**
	type II	**0.660 (0.016)**	0.849 (0.022)	0.938 (0.007)
	type III	**0.483 (0.042)**	0.870 (0.027)	0.893 (0.010)
type II	type I	0.504 (0.056)	0.917 (0.012)	0.993 (0.004)
	type II	0.519 (0.027)	0.872 (0.007)	0.981 (0.007)
	type III	0.406 (0.018)	**0.907 (0.004)**	0.950 (0.009)
type III	type I	0.274 (0.012)	**0.946 (0.003)**	0.994 (0.002)
	type II	0.330 (0.004)	**0.929 (0.003)**	**0.983 (0.003)**
	type III	0.325 (0.008)	0.889 (0.014)	**0.967 (0.007)**

Table 1
Multi-task learning results as F1 averages over three runs (and standard deviation). Training with 3000 instances – 1000 from each task.

train on	test on	task		
		agreement	*Caus*	*Od*
type I	type I	**0.909 (0.007)**	0.919 (0.005)	**1.000 (0.000)**
	type II	**0.760 (0.030)**	0.906 (0.017)	0.971 (0.003)
	type III	**0.707 (0.028)**	0.926 (0.005)	0.940 (0.010)
type II	type I	0.881 (0.013)	0.932 (0.007)	1.000 (0.000)
	type II	0.784 (0.007)	0.903 (0.010)	0.983 (0.003)
	type III	0.714 (0.005)	**0.956 (0.005)**	0.975 (0.009)
type III	type I	0.296 (0.011)	**0.960 (0.005)**	0.998 (0.002)
	type II	0.345 (0.002)	**0.950 (0.007)**	**0.993 (0.004)**
	type III	0.336 (0.005)	0.918 (0.010)	**0.994 (0.004)**

Table 2
Single task learning results as F1 averages over three runs (and standard deviation). Training with 2160 instances for *Caus* and *Od* for all types, and for agreement 2052 instances for type I (maximum available), and 3000 instances for type II and type III.

B.2. Results with the multilingual Electra pretrained model: google/electra-base-discriminator

train on	**test on**	**task**		
		agreement	*Caus*	*Od*
type I	type I	**0.664 (0.053)**	0.543 (0.011)	0.714 (0.012)
	type II	**0.733 (0.018)**	0.407 (0.023)	0.561 (0.002)
	type III	**0.586 (0.022)**	0.483 (0.016)	0.656 (0.016)
type II	type I	0.599 (0.025)	**0.610 (0.035)**	0.646 (0.010)
	type II	0.660 (0.019)	**0.536 (0.004)**	0.601 (0.004)
	type III	0.518 (0.025)	**0.601 (0.011)**	**0.686 (0.019)**
type III	type I	0.320 (0.047)	0.551 (0.014)	**0.729 (0.015)**
	type II	0.401 (0.058)	0.450 (0.021)	**0.661 (0.020)**
	type III	0.378 (0.052)	0.413 (0.012)	0.618 (0.005)

Table 3
Multi-task learning results as F1 averages over three runs (and standard deviation). Training with 3000 instances – 1000 from each task.

train on	**test on**	**task**		
		agreement	*Caus*	*Od*
type I	type I	**0.875 (0.031)**	0.599 (0.040)	0.749 (0.030)
	type II	**0.886 (0.005)**	0.425 (0.019)	0.579 (0.037)
	type III	0.815 (0.016)	0.529 (0.020)	0.660 (0.014)
type II	type I	0.841 (0.024)	0.543 (0.027)	0.651 (0.007)
	type II	0.881 (0.003)	0.486 (0.005)	0.596 (0.010)
	type III	0.814 (0.008)	**0.582 (0.026)**	**0.685 (0.013)**
type III	type I	0.826 (0.022)	**0.632 (0.023)**	**0.761 (0.023)**
	type II	0.878 (0.005)	**0.557 (0.013)**	**0.697 (0.009)**
	type III	**0.874 (0.006)**	0.475 (0.010)	0.592 (0.024)

Table 4
Single task learning results as F1 averages over three runs (and standard deviation). Training with 2160 instances for *Caus* and *Od* for all types, and for agreement 2052 instances for type I (maximum available), and 3000 instances for type II and type III.