Semantic Interpretation of BERT embeddings with Knowledge Graphs[*]

Discussion Paper

Alessandro De Bellis[1,*], Giovanni Maria Biancofiore[1,*], Vito Walter Anelli[1,*], Fedelucio Narducci[1], Tommaso Di Noia[1], Azzurra Ragone[2] and Eugenio Di Sciascio[1]

[1] *Polytechnic University of Bari, Bari, Italy*

[2] *University of Bari, Bari, Italy*

Abstract

Pretrained language models have transformed the way we process natural languages, enhancing the performance of related systems. BERT has played a pivotal role in revolutionizing the field of Natural Language Processing (NLP). However, the deep learning framework behind BERT lacks interpretability. Recent research has focused on explaining the knowledge BERT acquires from the textual sources used for pre-training its linguistic model. In this study, we analyze the latent vector space produced by BERT's context-aware word embeddings. Our aim is to determine whether certain areas of the BERT vector space have an explicit meaning related to a Knowledge Graph (KG). Using the Link Prediction (LP) task, we demonstrate the presence of explicit and meaningful regions of the BERT vector space. Moreover, we establish links between BERT's vector space and specific ontology concepts in the KG by learning classification patterns. To the best of our knowledge, this is the first attempt to interpret BERT's learned linguistic knowledge through a KG by relying on its pre-trained context-aware word embeddings.

Keywords

Natural Language Processing, Deep Learning, Knowledge Graphs

1. Introduction

There have been significant changes in the paradigms of Natural Language Processing (NLP). With the abundance of linguistic data available, deep learning models have become more prominent in learning textual data representations and have replaced hand-crafted feature engineering approaches. This shift has produced successful architecture designs for implementing language models, such as Bengio et al. [2] and Mikolov et al. [3]. The increasing attention this area is receiving, along with the progress in deep learning [4], has led to the development of a vast variety of NLP models capable of fulfilling various applications [5, 6, 7]. Vaswani et al. [8] achieved

SEBD 2023: 31st Symposium on Advanced Database System, July 02–05, 2023, Galzignano Terme, Padua, Italy

[*] Extended version [1] published in the 31st ACM International Conference on Information & Knowledge Management

[*] Corresponding author.

alessandro.debellis@poliba.it (A. De Bellis); giovannimaria.biancofiore@poliba.it (G. M. Biancofiore); vitowalter.anelli@poliba.it (V. W. Anelli); fedelucio.narducci@poliba.it (F. Narducci); tommaso.dinoia@poliba.it (T. Di Noia); azzurra.ragone@uniba.it (A. Ragone); eugenio.disciascio@poliba.it (E. Di Sciascio)

0000-0002-1220-9878 (A. De Bellis); 0000-0003-1317-8695 (G. M. Biancofiore); 0000-0002-5567-4307 (V. W. Anelli); 0000-0002-9255-3256 (F. Narducci); 0000-0002-0939-5462 (T. Di Noia); 0000-0002-3537-7663 (A. Ragone); 0000-0002-5484-9945 (E. Di Sciascio)

a remarkable performance in solving the sequence transduction task with Transformers, and Devlin et al. [9] built upon this architecture and the growing popularity of pre-training and fine-tuning formulas [10], resulting in the creation of Bidirectional Encoder Representations from Transformers (BERT). BERT marked a major shift in the state-of-the-art in NLP over the earlier proposed pre-trained language models [11, 12, 13], as it implemented a masked paradigm based on the Cloze task [14] and a next-sentence prediction assignment for pre-training. Once fine-tuned, BERT achieved competitive performances on various benchmarks (e.g., GLUE [15] and SQuAD [16, 17]). This success prompted the development of many BERT variants aimed at improving the resolution of the most popular NLP-based tasks.

Xia and colleagues [18] compiled all BERT implementations in five areas of advancement. The majority of published work in four of these areas revolves around modifying BERT to fulfill specific objectives such as improving the language model through pre-training objectives or data, enhancing model efficiency and multilingual capabilities. On the other hand, the fifth area regarding the interpretability of BERT has fewer published works, which highlights the challenges in interpreting such a sophisticated framework. However, current trends in published literature exhibit a growing interest in interpretability [19, 20], transparency [21] and fairness [22]. Many studies examine the model through attention heads [23, 24, 25, 26, 27], fine-tune it for interpretability tasks [28], or modify the pre-training procedure for the same objectives [29, 30].

However, only a small percentage of studies investigates the BERT space semantically. A few rely on classifiers to analyze information held about Entity-Linking [31], entity category clustering [32], and link prediction [33] tasks, which provide insights on the knowledge acquired by BERT. Nevertheless, these approaches require authors to adjust the embedding representations or target a multiclassification task. Thus, these methods are better suited to analyze what BERT learns to distinguish the embeddings rather than providing information on the properties of the BERT space. Conversely, Ethayarajh [34] was the first to investigate BERT's latent space properties by comparing cosine similarity between contextualized pre-trained BERT word embeddings. These word representations result from feeding BERT with words contained in contextual sentences without fine-tuning it. Hereinafter, we refer to this type of embeddings as BERT embeddings. Specifically, he observes that BERT vector representations are anisotropic within their direction, forming groups in narrow cones.
In this research, we seek to address the following research questions:

- R1: Does BERT generate a latent semantic space holding information about knowledge with explicit semantics?
- R2: Can we learn functions to automatically detect precise knowledge graph concepts from the BERT latent space?

The objective of the research is to explore the BERT embedding space and identify meaningful areas associated with explicit concepts and their boundaries. This is achieved by utilizing knowledge graphs (KGs) that connect entities through directed edges infused with explicit semantics. The semantic network is represented using a set of triples subject-predicate-object, where the subjects and objects denote specific entities with unique identifiers. Initially, we compare the behaviors of various KG embeddings to the BERT embeddings using Link Prediction

(LP). Our hypothesis is that similarities between the two embeddings illustrate that the BERT space showcases the inherent structure of the KG and reflects that BERT embeddings contain explicit semantic information that depends on their location, in other words, a topology. We then confirm that this property leads to exact KG concepts by implementing several classifiers, one for each KG ontological class, to learn patterns on BERT embeddings. Unlike a single classifier trained on a multi-classification task, binary classifiers extract feature patterns unique to each concept in the BERT space that refers to the ontological classes. In contrast, the multi-classifier learns the features that distinguish the word representations among a finite set of concepts, disregarding the existence of other categories or their simultaneous membership to several categories. The main contribution of this study is to demonstrate that BERT embeddings have information about the KG structure without requiring any fine-tuning or architectural changes. Lastly, we establish that the internal spatial properties of the BERT vector space enable us to deduce explicit KG concepts. Extensive experiments support our findings, which are available here[1] for reproducibility.

2. Methodology

Our first objective is to demonstrate the presence of discernible semantics within the latent space generated by BERT embeddings. Previous research has established that BERT representations contain hierarchical and syntactic information (i.e. parts of speech, syntactic functions, and subject-predicate agreements). In addition, they hold a linguistic knowledge that can identify semantic roles, entity types, and relationships. Nevertheless, the BERT vector space remains mainly unexplored. Recent findings [34] indicate that the BERT space is anisotropic, whereas Dalvi et al. [32] have established that BERT embeddings can be categorized based on interpretable, albeit implicit concepts. Therefore, the current literature does not permit definitive conclusions regarding the explicit semantic content within the BERT space.

We adopt an approach that examines the similarities between BERT embeddings and conventional KG embeddings in their ability to address the LP task. Specifically, KG embeddings are designed to capture the explicit semantics of a knowledge graph and represent them within a continuous vector space while preserving its underlying structure [35]. The LP, on the other hand, involves forecasting the exactness of novel subject-predicate-object triples and serves as a measure of embeddings' accuracy about the grasped KG's infrastructure [36]. By testing the effectiveness of BERT embeddings through the LP task, we aim to gain a deeper understanding of the semantic information and KG structure that they possess. To ensure the relevance of our study, we base our analysis on the following assumptions:

1. The BERT embeddings result from the BERT vanilla pre-trained model;
2. KG embeddings are the outcome of training wherein the subject and object entities, of a subject-predicate-object triple in a KG, share the same vector space;
3. All the embeddings share the same dimension.

By assuming (1), we can evaluate how well BERT can encode explicit semantic information in its pre-trained language model and make deductions about its derived latent space. Fine-

[1] https://bit.ly/3T0987I

tuning procedures specialize BERT embeddings' information and affect the generality of our study. Therefore, variants like KG-BERT [37] or those proposed by Petroni et al. [33] which modify or fine-tune BERT are not used. The comparisons between BERT embeddings and KG embeddings are made significant by the last two assumptions. With assumption (2), we focus only on the vector space resulting from BERT, so KG embeddings and LP methods that require space transformation or additional support spaces are beyond the scope of our investigation. Constraint (3) places the examined embeddings on equal grounds.

Assumed a KG, BERT embeddings for its entities are computed. In order to obtain optimal BERT representations for a semantic task, we provide the BERT model with the entity label and its corresponding context. To accomplish this, we form sentences that follow the template "label + *be* + abstract" for each KG entity. Contextualized BERT representations are obtained from the final hidden units that pertain to the label. It is important to note that we avoid making assumptions about entity label granularity. BERT operates on WordPiece [38] segmentation on a sub-word level to prevent any issues with mismatched vocabulary. Thus, each BERT hidden unit corresponds to a single sub-word. The BERT embedding for the label is created by aggregating its sub-word hidden units that may belong to more than one word.

To be more specific, assume w to be a word consisting of a collection of WordPiece tokens t s.t. $w = [t_1, t_2, ..., t_n]$ and l to be the entity label that may contain one or multiple words s.t. $l = [w_1, w_2, ...w_m]$. Therefore, for $1 \leq i \leq n$ and $1 \leq j \leq m$, t_{ji} refers to the i-th WordPiece token of the j-th word. The BERT embedding of a WordPiece token t is derived from the BERT hidden unit $h_s(t)$, which includes contextual information from the complete input sentence s. Hence, $h_s(t_{ji})$ represents the BERT embedding of the i-th WordPiece token of the j-th word. Our method employs b to denote the singular word "be" and c represents the group of d words making up the entity abstract. Consequently, we utilize $s = l + b + c$ to refer to the input that we use to encode each entity with BERT. We denote the words and WordPiece tokens belonging to the entity label l with w^l and t^l respectively. Suppose we have the aggregate function f, the BERT embedding of an entity $\mathbf{e}$ is created by aggregating the embeddings of the WordPiece tokens of each word in its label, resulting in $\mathbf{e} = f(h_s(t_{ji}^l))$.

After encoding all KG entities, relation embeddings are learned over their entity BERT embeddings using existing LP models. BERT embeddings for relations are not used as they differ from entities in terms of semantic components. In an anisotropic space, such embeddings depend on both entity and relation directions. This prevents the assumption that relationships and entities have similar properties or are analogous elements. To meet requirement (2), three well-established LP models (TransE [39], TransH [40], and DistMult [41]) are used. TransE is a translational distance model that defines both entities and relations as vectors in the same space. Given a KG fact (h, r, t), the relation is interpreted as a translation vector $\mathbf{r}$ so that the embedded entities $\mathbf{h}$ and $\mathbf{t}$ can be connected with $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$. TransH introduces relation-specific hyperplanes defined by the normal vector $\mathbf{w}_r$, but it still maintain the intuition of TransE. In detail, we have to first project the entity representations $\mathbf{h}$ and $\mathbf{t}$ onto the hyperplane to have their connection: $\mathbf{h}_\perp = \mathbf{h} - \mathbf{w}_r^T \mathbf{h} \mathbf{w}_r$; $\mathbf{t}_\perp = \mathbf{t} - \mathbf{w}_r^T \mathbf{t} \mathbf{w}_r$. Conversely, DistMult represents each relation as a diagonal matrix $\mathbf{M}_r = \text{diag}(\mathbf{r})$ that models pairwise interactions between components of entities along the same dimensions with a scoring function: $f_r(h, t) = \mathbf{h}^T \text{diag}(\mathbf{r}) \mathbf{t}$. These models are directly applied to BERT embeddings of KG entities to infer relations based on the KG structure. The BERT embeddings for entities are fixed during the entire training process.

As LP models encode the KG explicit semantics in resulting KG embeddings, similarities of performances on the LP task for KG embeddings and BERT embeddings provide an answer to research question R1 in Section 1.

The second goal is to evaluate whether the BERT space holds distinguishing features that facilitate the identification of specific KG concepts. To identify these properties, we train multiple binary classifiers to identify whether the BERT embedding of an entity is categorized under an ontological class. Each classifier is tailored to recognize a particular concept, with a separate classifier for each KG class. These classifiers comprehend only their own class and extract patterns from the BERT representations that are relevant to the respective ontological concept. This enables to concentrate solely on the input features, i.e., BERT embeddings, that are relevant to derive their class. These feature properties can subsequently be applied to the entire embedding space. Achieving outstanding performance on each classifier addresses research question R2 in Section 1.

3. Semantic Analysis

This section provides the detailed configuration of the experiments that led to our analysis of the BERT space. We explore its inner properties through the embeddings computed by the pre-trained language model of the BERT-Base-cased. We choose to investigate the cased version since we believe that cased words enclose a different semantic than uncased ones.

We use Freebase (FB15k-237) [42] as the benchmark dataset to implement LP over the BERT embeddings. The main intuition here is that BERT already contains Freebase explicit semantics since it was pre-trained on the English Wikipedia corpus where Freebase was built. Since we need to feed BERT with sentences formed concatenating the label, the verb *be*, and the abstract for each entity, we discard from the FB15k-237 all the entities with no label or abstract in their Wikidata mapping[2]. Thus, we obtain an FB15K-237 subset by removing the facts related to the discarded entities, and we call it FB15K-237-Desc. It contains 266,263 facts over 13,667 entities compared to FB15K-237, which has 310,116 facts over 14,541. For completeness, we also compute separated entities BERT representations by feeding the BERT model with only the entities labels. These embeddings will benchmark the utility of the context for BERT in positioning them into the most appropriate space region. In both cases, the aggregation of the hidden units of the WordPiece tokens referring to the entities takes place through the arithmetic mean function.

Link Prediction and Semantics. Once we have computed the two BERT representations for all the FB15K-237-Desc entities, respectively BL (i.e., BERT Label) and BD (i.e., BERT Desc), we start the LP task in three configurations. The first one computes the standard KG embedding of TransE, TransH, and DistMult to compose our baselines. Their performance needed to be recalculated to accomplish the requirement (3) since their embeddings size is 768. The second setting learns the relation embeddings over the BL entities representations, which enable the evaluation of the LP over the BL entity embeddings. The last scenario differs in the computation of the BERT entity embeddings through contextualized sentences (BD). All the configurations have their models trained on FB15K-237-Desc, which is split into 80%-10%-10% to generate the

[2] https://developers.google.com/freebase/

train, evaluation, and test sets. The training procedure follows the minibatch mode over the raw and filtered negative sampling proposed by Bordes et al. [39].

	Raw			Filtered		
	MR	hit@10	hit@5	MR	hit@10	hit@5
TransE	**305.32**	**33.15**	**24.48**	**177.86**	**45.94**	**37.19**
TransH	412.73	29.74	21.39	271.88	40.98	32.39
DistMult	395.48	25.46	17.30	281.00	33.79	24.84
TransE$_{BL}$	866.47	21.30	15.88	773.43	25.03	19.56
TransH$_{BL}$	968.67	20.98	15.88	875.44	24.54	19.62
DistMult$_{BL}$	847.86*	19.84*	14.61*	753.46*	23.76*	18.09*
TransE$_{BD}$	604.83	23.26	17.04	508.40	28.68	22.15
TransH$_{BD}$	702.62	20.62	15.49	609.64	24.84	19.01
DistMult$_{BD}$	560.64*	21.31*	15.60*	467.01*	26.11*	19.83*

Table 1
Evaluation results of the LP task through the standard, BERT label (BL) and BERT description (BD) embeddings of TransE, TransH and DistMult. In bold, the best achieved results over the three configurations. The asterisks mark the BL and BD results closest to those of the standard model. The underlined outcomes highlight those values most relative to the best results.

We used the grid-search with early stopping to select the hyperparameters leading to the best performance in each configuration. The batch size has a value fixed to 1200 while the margin value γ can assume values among (1, 2, 10) for TransE, (0.25, 0.5, 1, 2) for TransH and (0.001, 0.005, 0.01) for DistMult. In addition, TransH has its soft constraint weight selected among (0.015625, 0.0625, 0.25, 1), and TransE has its distance function tested between L1 and L2. Both TransE and TransH were trained with stochastic gradient descent (SGD) in the first LP configuration, while in the remaining two settings they exploit the Adam optimizer. DitsMult instead uses the AdaGrad optimizer in each LP scenario. We adopt the Mean Rank (MR) and the hit@n (i.e., hit@10 and hit@5) metrics to assess the LP performances. Table 1 resumes the results we achieved.

Surprisingly, the standard TransE reaches the best results in each configuration. This outcome derives from its known ability to catch the KG geometrical properties. In contrast, TransH and DistMult collect more semantic information, which can behave like noises given the different embedding dimensionality. The BL embeddings instead obtain the worst performances as we expected, further proving the importance of the context in correctly positioning the BERT representations in its vector space. The last configuration gives the most exciting results. Albeit maintaining the same behaviors of the models, TransE over the BD embeddings gets comparable results with the standard DistMult model. This finding shows that we can infer meaningful explicit semantics by referring to the BERT embeddings' relevant properties (i.e., specific space dimensions). Therefore, we can say that the BERT space intrinsically contains a KG structure and precise semantics.

Ontological Analysis. The second experiment trains binary classifiers for each KG ontological class to detect whether a contextualized BERT embedding belongs to a specific category. Thus, we first retrieve from the Wikidata mapping all the entities' ontological classes through their *instanceOf* property, discarding those having no label or category. We use the FB15K [39] since it contains more entities than its smaller counterpart, and we limit our observations to those classes with enough entities to train meaningful classifiers. Hence, we obtain 20 categories with 9,258 entities distributed as in Figure 1.

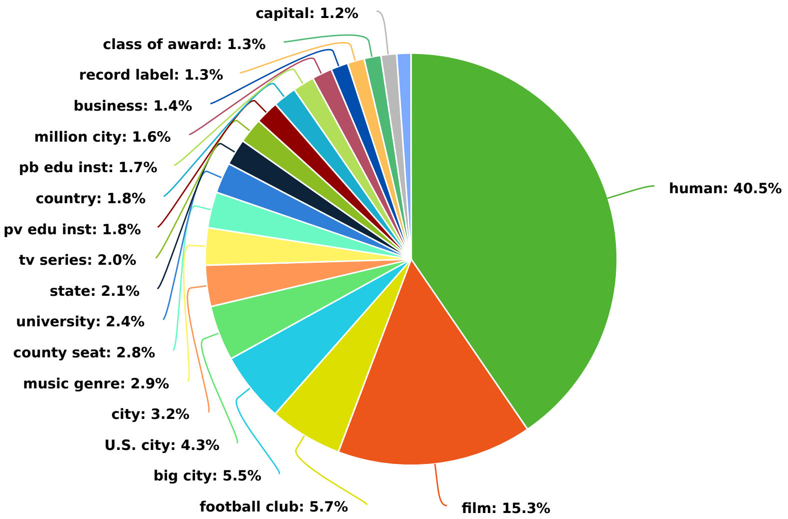

Figure 1: Distribution of the FB15K entities among the Wikidata ontological classes.

It is worth noting how classes like *city*, *big_city* and *US_city* can group into a single category. However, we first split the overall dataset into 80%-10%-10% to generate the train, validation, and test set. In this manner, we avoid classifiers knowing the entity class from the training set during their test. In addition, we perform this partition maintaining the same distribution of classes. Then, for each category, we select from the train set all those samples which belong to the related classifier category, and we added an equal amount of entities from other classes to have the train set balanced. We model each classifier as a feedforward neural network with a single hidden layer of 300 units that uses the ReLU activation function and the Adam optimizer. We evaluate the classifiers' performances through their accuracy, precision, recall, and F1 measure. Table 2 resumes the results of each model and gives data about the positive output for each class through the support.

As we can see, all the classifiers reach a high value of accuracy, among which the performance of the *human* classifier emerges. We explain these outcomes since the *human* classifier possesses most of the KG data supporting its modeling. Moreover, the *human* category identifies the most unambiguous class of the dataset, making it easy for its classifier to recognize its entities. Conversely, classes like *city*, *big_city*, and *US_city* have their classifier reaching low precision values despite their high recall. This result mainly depends on how these categories actually identify a single one. Indeed, the high recall highlights that the classifiers recognize all the positive samples of that class. At the same time, the low precision shows that they also identify other entities as belonging to that group. Therefore, We can infer that the BERT embeddings contain precise information that leads to explicit ontological classes, which can be extended to the spatial properties of the BERT vector space.

	Acc. (%)	P (%)	R (%)	F1 (%)	Supp. (#)
human	**99.95**	100	99.88	**99.94**	866
film	99.14	95.33	100	97.61	327
football club	99.78	96.82	100	98.39	122
big city	94.06	50.46	97.32	66.46	112
U.S. city	98.49	76.47	100	86.67	91
city	95.30	43.05	98.48	59.91	66
music genre	99.84	95.38	100	97.64	62
county seat	96.76	50	100	66.67	60
university	95.65	45.04	98.04	61.73	51
state	99.19	75.43	97.73	85.15	44
tv series	93.84	26.62	97.62	41.83	42
pv edu Inst	95.79	33.34	100	50	39
country	99.46	80	100	88.89	40
pb edu Inst	97.35	42.23	100	59.50	36
million city	94.60	23.08	100	37.50	30
business	95.63	26.36	100	41.73	23
record label	97.03	33.73	100	50.45	28
class of award	99.62	80	100	88.89	28
capital	94.55	18.70	95.83	31.29	24
o.a. publisher	95.46	21.70	95.84	35.38	24

Table 2
Evaluation results of the binary classifiers trained on each Wikidata category. In bold, the classifier's best results.

4. Conclusion

The latent vector space derived from BERT context-aware word embeddings was examined to determine if explicit semantic content from Knowledge Graphs (KGs) is present in the BERT vector space regions. By utilizing the characteristics of the Link Prediction task, we have verified the existence of meaningful regions that correspond to explicit ontology concepts of a KG through learning classification patterns on the BERT embeddings. To our knowledge, there have been no previous attempts to interpret BERT's linguistic knowledge through a KG. In the future, we aim to broaden our analysis to other KGs and uncover common semantic patterns among different graphs.

Acknowledgments

The authors acknowledge partial support from the projects PASSEPARTOUT, ServiziLocali2.0, Smart Rights Management Platform, BIO-D, ERP4.0, Casa delle Tecnologie Emergenti della Città di Matera, and Fincons Smart Rights Management Platform II.

References

[1] V. W. Anelli, G. M. Biancofiore, A. D. Bellis, T. D. Noia, E. D. Sciascio, Interpretability of BERT latent space through knowledge graphs, in: M. A. Hasan, L. Xiong (Eds.),

Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022, ACM, 2022, pp. 3806–3810. URL: https://doi.org/10.1145/3511808.3557617. doi:10.1145/3511808.3557617.

[2] Y. Bengio, R. Ducharme, P. Vincent, A neural probabilistic language model, Advances in Neural Information Processing Systems 13 (2000).

[3] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781 (2013).

[4] V. W. Anelli, A. Bellogín, T. D. Noia, C. Pomo, Reenvisioning the comparison between neural collaborative filtering and matrix factorization, in: RecSys, ACM, 2021, pp. 521–529.

[5] J. Hirschberg, C. D. Manning, Advances in natural language processing, Science 349 (2015) 261–266.

[6] G. M. Biancofiore, T. D. Noia, E. D. Sciascio, F. Narducci, P. Pastore, Aspect based sentiment analysis in music: a case study with spotify, in: SAC, ACM, 2022, pp. 696–703.

[7] G. M. Biancofiore, T. D. Noia, E. D. Sciascio, F. Narducci, P. Pastore, Guapp: a knowledge-aware conversational agent for job recommendation (short paper), in: KaRS/ComplexRec@RecSys, volume 2960 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[9] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[10] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heinz, D. Roth, Recent advances in natural language processing via large pre-trained language models: A survey, arXiv preprint arXiv:2111.01243 (2021).

[11] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: NAACL-HLT, Association for Computational Linguistics, 2018, pp. 2227–2237.

[12] J. Howard, S. Ruder, Universal language model fine-tuning for text classification, in: ACL (1), Association for Computational Linguistics, 2018, pp. 328–339.

[13] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training, OpenAI blog (2018) 12.

[14] W. L. Taylor, "cloze procedure": A new tool for measuring readability, Journalism quarterly 30 (1953) 415–433.

[15] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. R. Bowman, GLUE: A multi-task benchmark and analysis platform for natural language understanding, in: BlackboxNLP@EMNLP, Association for Computational Linguistics, 2018, pp. 353–355.

[16] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, Squad: 100, 000+ questions for machine comprehension of text, in: EMNLP, The Association for Computational Linguistics, 2016, pp. 2383–2392.

[17] P. Rajpurkar, R. Jia, P. Liang, Know what you don't know: Unanswerable questions for squad, in: ACL (2), Association for Computational Linguistics, 2018, pp. 784–789.

[18] P. Xia, S. Wu, B. V. Durme, Which *bert? A survey organizing contextualized encoders, in: EMNLP (1), Association for Computational Linguistics, 2020, pp. 7516–7533.

[19] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ferrara, A. C. M. Mancino, Sparse feature

factorization for recommender systems with knowledge graphs, in: RecSys, ACM, 2021, pp. 154–165.

[20] V. W. Anelli, T. D. Noia, P. Lops, E. D. Sciascio, Feature factorization for top-n recommendation: From item rating to features relevance, in: RecSysKTL, volume 1887 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 16–21.

[21] V. W. Anelli, Y. Deldjoo, T. D. Noia, A. Ferrara, F. Narducci, How to put users in control of their data in federated top-n recommendation with learning to rank, in: SAC, ACM, 2021, pp. 1359–1362.

[22] G. Cornacchia, V. W. Anelli, G. M. Biancofiore, F. Narducci, C. Pomo, A. Ragone, E. D. Sciascio, Auditing fairness under unawareness through counterfactual reasoning, Inf. Process. Manag. 60 (2023) 103224.

[23] J. Vig, A multiscale visualization of attention in the transformer model, in: ACL (3), Association for Computational Linguistics, 2019, pp. 37–42.

[24] K. Clark, U. Khandelwal, O. Levy, C. D. Manning, What does BERT look at? an analysis of bert's attention, in: BlackboxNLP@ACL, Association for Computational Linguistics, 2019, pp. 276–286.

[25] S. Serrano, N. A. Smith, Is attention interpretable?, in: ACL (1), Association for Computational Linguistics, 2019, pp. 2931–2951.

[26] S. Jain, B. C. Wallace, Attention is not explanation, in: NAACL-HLT (1), Association for Computational Linguistics, 2019, pp. 3543–3556.

[27] S. Wiegreffe, Y. Pinter, Attention is not not explanation, in: EMNLP/IJCNLP (1), Association for Computational Linguistics, 2019, pp. 11–20.

[28] D. Li, S. Yang, K. Xu, Y. H. Ming Yi, H. Wang, Multi-task pre-training language model for semantic network completion, arXiv preprint arXiv:2201.04843 (2022).

[29] I. Tenney, P. Xia, B. Chen, A. Wang, A. Poliak, R. T. McCoy, N. Kim, B. V. Durme, S. R. Bowman, D. Das, E. Pavlick, What do you learn from context? probing for sentence structure in contextualized word representations, in: ICLR (Poster), OpenReview.net, 2019.

[30] Z. Wu, Y. Chen, B. Kao, Q. Liu, Perturbed masking: Parameter-free probing for analyzing and interpreting BERT, in: ACL, Association for Computational Linguistics, 2020, pp. 4166–4176.

[31] S. Broscheit, Investigating entity knowledge in BERT with simple neural end-to-end entity linking, in: CoNLL, Association for Computational Linguistics, 2019, pp. 677–685.

[32] F. Dalvi, A. R. Khan, F. Alam, N. Durrani, J. Xu, H. Sajjad, Discovering latent concepts learned in BERT, CoRR abs/2205.07237 (2022).

[33] F. Petroni, T. Rocktäschel, S. Riedel, P. S. H. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, Language models as knowledge bases?, in: EMNLP/IJCNLP (1), Association for Computational Linguistics, 2019, pp. 2463–2473.

[34] K. Ethayarajh, How contextual are contextualized word representations? comparing the geometry of bert, elmo, and GPT-2 embeddings, in: EMNLP/IJCNLP (1), Association for Computational Linguistics, 2019, pp. 55–65.

[35] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, IEEE Trans. Knowl. Data Eng. 29 (2017) 2724–2743.

[36] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, G. Bouchard, Complex embeddings for simple link prediction, in: ICML, volume 48 of *JMLR Workshop and Conference Proceedings*,

JMLR.org, 2016, pp. 2071–2080.

[37] L. Yao, C. Mao, Y. Luo, KG-BERT: BERT for knowledge graph completion, CoRR abs/1909.03193 (2019).

[38] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, J. Dean, Google's neural machine translation system: Bridging the gap between human and machine translation, CoRR abs/1609.08144 (2016).

[39] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: NIPS, 2013, pp. 2787–2795.

[40] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: AAAI, AAAI Press, 2014, pp. 1112–1119.

[41] B. Yang, W. Yih, X. He, J. Gao, L. Deng, Embedding entities and relations for learning and inference in knowledge bases, in: ICLR (Poster), 2015.

[42] K. Toutanova, D. Chen, Observed versus latent features for knowledge base and text inference, Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality (2015) 57–66.