future internet

MDPI

Article

Synonyms, Antonyms and Factual Knowledge in BERT Heads

Lorenzo Serina *, Luca Putelli *, Alfonso Emilio Gerevini and Ivan Serina

Department of Information Engineering, Università Degli Studi di Brescia, Via Branze 38, 25100 Brescia, Italy; alfonso.gerevini@unibs.it (A.E.G.); ivan.serina@unibs.it (I.S.)
* Correspondence: lorenzo.serina@unibs.it (L.S.); luca.putelli@unibs.it (L.P.)

Abstract: In recent years, many studies have been devoted to discovering the inner workings of Transformer-based models, such as BERT, for instance, attempting to identify what information is contained within them. However, little is known about how these models store this information in their millions of parameters and which parts of the architecture are the most important. In this work, we propose an approach to identify self-attention mechanisms, called heads, that contain semantic and real-world factual knowledge in BERT. Our approach includes a metric computed from attention weights and exploits a standard clustering algorithm for extracting the most relevant connections between tokens in a head. In our experimental analysis, we focus on how heads can connect synonyms, antonyms and several types of factual knowledge regarding subjects such as geography and medicine.

Keywords: BERT; attention heads; synonyms; antonyms; real-world knowledge

check for updates

Citation: Serina, L.; Putelli, L.; Gerevini, A.E.; Serina, I. Synonyms, Antonyms and Factual Knowledge in BERT Heads. *Future Internet* **2023**, *15*, 230. https://doi.org/10.3390/fi15070230

Academic Editor: Michael Sheng

Received: 31 May 2023
Revised: 23 June 2023
Accepted: 27 June 2023
Published: 29 June 2023

1. Introduction

Huge unsupervised textual corpora, such as Wikipedia or PubMed, have been frequently used for training Transformer-based models, such as BERT [1], BlueBERT [2] or RoBERTa [3], with the goal of learning how to predict words from the context or whether two sentences are consecutive or not. In order to do that, it has been established that these models need to acquire different forms of linguistic knowledge. In fact, they are able to identify verbs, nouns, direct objects and other similar concepts [4–6]. Moreover, especially considering the high performance in many natural language processing tasks, it has been debated whether Transformer-based models encode some forms of semantic knowledge or do not [7].

Another important line of work concerning these models regards their real-world knowledge. In fact, it was observed that the data used for training these models (which are usually taken from newspapers, Wikipedia or scientific papers) did not contain only linguistic information but also a large quantity of factual knowledge about real-world entities, such as the capital of a state, or where a famous person was born. Therefore, several studies [8–10] assess that this kind of knowledge is somehow captured by these large language models and that it is possible to retrieve it in a particular kind of classification task. This task can be structured as follows: given a sentence with a masked token such as "The capital of France is [MASK]", the model has to select a token among its dictionary and its prediction can be compared with a pre-determined label, which, in this case, would be the token "Paris". For performing such tasks, probing datasets, containing sentences like PARAREL [11] or T-REX [12], were introduced. These datasets are created by triples $\langle E_1, r, E_2 \rangle$, where E_1 and E_2 are entities (such as a state and its capital) and r is a relation that connects them. For the classification task, the entities are inserted into a sentence that describes their relationship, and then one entity is masked and has to be guessed by the model.

The studies regarding what knowledge (linguistical or not) is contained in BERT, its inner workings and its interpretability belong to the so-called field of "BERTology" [13]. In

particular, several studies focus on some specific components of BERT, its heads [14,15]. A head is a particular kind of a neural network layer, called a self-attention mechanism, with the following behaviour. Given a sequence of words, such as a sentence or a document, a head calculates how much a word is related to every other word (including itself) contained in the sequence. The overall architecture of BERT, which is made by several encoding layers (typically 12), can contain more than a hundred heads (typically 12 heads for each layer, for a total of 144 different heads), and each one of them can focus on different linguistic aspects or grammatical relations.

Since each head provides a numerical weight for every possible pair of tokens in a sentence, these weights can be visualized and manually inspected [16]. More interestingly, the study in [14] measures the performance of each head in different probing tasks. In the latter, they compare the pairs with the highest weights assigned by each head with some ground truth labels. These labels usually represent grammatical relations, such as the one between a noun and its modifier or a verb and its direct object, etc. Proposing different probing tasks, the study in [15] also evaluates what happens to the behaviour of the heads after adapting (or fine-tuning) the model for specific tasks, such as text classification or sentiment analysis. Finally, this study evaluates the effect that each head has on the performance in these tasks.

However, these studies focused only on linguistic capabilities, without taking into consideration the role of BERT heads in capturing semantic-related aspects, such as if two words are synonyms or antonyms. Moreover, despite it having been established that BERT models possess not only linguistic capabilities but also real-world factual knowledge, there are no available studies that assess whether this information is captured by the heads or is not, which heads are these, what their behaviour is and how they can be identified. This can be important in terms of model explainability, providing a more intuitive understanding of how BERT stores its knowledge and which parts of the architectures are involved in this process.

In this work, our goal is to understand the role of the heads in capturing semantic information and factual knowledge in BERT. Therefore, we analysed the behaviour of the heads and tested them with specific probing tasks, exploiting portions of the PARAREL and T-REX datasets and other corpora of sentences we specifically created for this analysis. We propose a new metric that identifies a group of heads that mainly focus on the identification of words in the same semantic field, such as synonyms and antonyms or words that have a relation based on some real-world factual knowledge, such as Jutland and Denmark. In our experimental analysis, we show how such heads can contain several forms of factual knowledge in subjects such as geography or medicine, for instance by linking a state to its capital, a drug to the disease treated by it, etc.

Another important aspect is that we show that, several times, these relationships based on factual knowledge are not strongly influenced by the overall context of the sentence or the document but are mostly based on the words themselves. For instance, in the sentences "The frigid temperature outside caused by the cold wind made my bones feel like they were about to break into pieces" and "I held onto her cold hand tightly as we braved through the frigid night without looking back", the head will give a high weight for the association between cold and frigid even if the context of the two sentences is completely different. Therefore, we claim that these kinds of relations are not based on grammatical properties but are based on actual semantic knowledge that the head has the duty to identify.

In this work, our contributions are the following:

- We propose a technique for identifying heads that find relationships among words that are in the same semantic field (synonyms or antonyms) or are related by some real-world knowledge;
- We experimentally verify that different types of relations (such as semantics, geography or medicine) are mostly identified by the same heads across different domains;

- We perform an experimental analysis that shows how the behaviour of these heads is correlated to the model performance in simple question-answering tasks (without access to external knowledge sources) using probing datasets;
- We show how semantic knowledge is not strongly influenced by the overall context of the sentence and is robust to different types of prompts and contexts.

The rest of the paper is organized as follows: in Section 2, we provide an overall description of the BERT architecture and we review the related work; in Section 3, we explain our methodology for studying heads; in Section 4, we present our case studies and datasets; in Section 5, we describe our experimental evaluation and the results we obtained and, finally, in Section 7, we provide our conclusions and discuss potential future work.

2. Background and Related Work

In this section, we provide a description of the architecture considered, BERT, and we review the most important related work regarding the study of which linguistic, grammatical or factual knowledge is contained in this type of large language model architecture and the techniques developed to provide explainability of the information stored in the model.

2.1. BERT

BERT (Bidirectional Encoder Representations from Transformer) [1] is an architecture based on Transformer [17] composed of several encoding layers, which progressively analyse a sequence of tokens (i.e., words or parts of a word) in order to capture their meaning.

An overall representation of the entire architecture can be seen in Figure 1. The document in input is divided into tokens and each token is represented as an array $x \in \mathbb{R}^d$. Then, the model adds a Positional Encoding to each token, allowing the model to understand the position of each word in a sentence and the distance between different words in the same sentence.

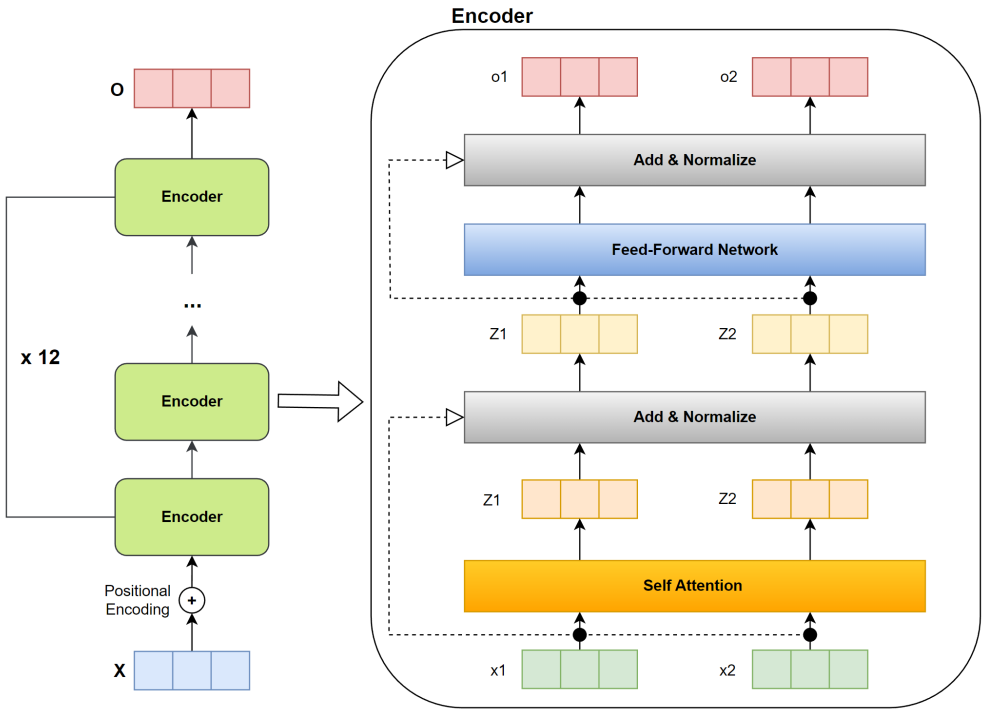


Figure 1. Architecture of the BERT model receiving two vectors representing two tokens (x_1 and x_2) as input. On the left, we show the overall stack of the encoder layers alongside the input, the output and the positional encoding. On the right, we show the components of each encoder layer (the Multi-Head Self-Attention Mechanism, the normalization layers and the Feed-Forward Neural Network). The dashed arrows represent skip connections.

This representation is then fed to the first Encoder layer, which contains a Multi-Head Self-Attention and a Feed-Forward Neural Network. This layer applies multiple self-attention mechanisms (called heads) in parallel. Considering a sequence of tokens S of length N, this mechanism produces a matrix $A_{i,j} \in \mathbb{R}^{N \times N}$, where i is the number of the encoding layer and j is the head number. For each token $w \in S$, the vector $a_w \in A_{i,j}$ contains the attention weights that represent how much w is related to the other tokens in S.

In order to calculate these weights, in each head, the input representation of the token sequence $X \in \mathbb{R}^{N \times d}$ (where d is the length of the input representation of each token) is projected into three new representations called key (K), query (Q) and value (V) with three matrices, W_k, W_q and W_v:

$$K = X \times W_k,\ Q = X \times W_q,\ V = X \times W_v \tag{1}$$

Then, the attention weights are calculated using a scaled dot product between Q and K and applying the softmax function. The new token representation Z is calculated by multiplying the attention weights for V.

$$A = softmax(\frac{Q \times K^{\intercal}}{\sqrt{d}}), \quad Z = A \times V \tag{2}$$

Given that in each encoding layer there are multiple heads, in order to create a single representation provided by the Multi-Head Attention Mechanism, the result of each head is concatenated and then passed to a feed-forward layer. The Multi-Head Attention Mechanism followed by a residual connection that adds its output to the original input before a layer normalization and a dropout are applied. The result of this operation is then passed to a Feed-Forward Neural Network composed of two layers, and to another residual connection with layer normalization. The output of an encoding layer is the input of the next one.

Exploiting a large collection of documents, BERT is trained for two tasks: language modeling, where BERT learns to predict a percentage (usually 15%) of masked tokens from context, and next sentence prediction, which is a binary classification task where BERT has to predict if a sequence of two sentences is correct or not. For the latter task, BERT introduces two special tokens: [CLS], whose representation is used for the binary classification task and represents the whole sequence, and [SEP], which separates the two sentences. Learning these two tasks allows BERT to create a meaningful representation of each token and also to summarise the most important information in a sentence. Once the model is trained, it can be adapted using smaller datasets for specific NLP tasks, like Named Entity Recognition, text classification [18], sentiment analysis, etc.

2.2. Related Work

Our work follows an active research field that studies the inner workings of BERT, its inner parts (such as its heads or its neurons) and how to explain its predictions. This field is often called "BERTology" [13]. In the last few years, many studies have been devoted to discovering the knowledge stored in large language models (LLM), such as Transformer, BERT or GPT, trying to identify the quantity of information stored in them. These studies are usually performed on pre-trained models, available online.

In particular, our work focuses on the study of the behaviour of the different heads in BERT models. Since the introduction of these models, several studies have been devoted to this subject. First of all, the work in [16] introduces BertViz, a visualization tool that can easily represent (especially for shorter sentences) the self-attention weights and the most important relations among words. Differently from this work, we do not focus on visualization or intuition but perform a more thorough analysis through several classification tasks. From a linguistic point of view, the work in [14] presents some interesting results. Firstly, the authors manually selected heads that give attention broadly to the

following token or to the end of sequence token. Next, they defined probing tasks in order to show that certain heads target specific linguistic information, like the most important dependency relations among words (preposition–noun, noun–adjective, etc.). A similar approach was adopted in [15]: similar patterns were presented and other tasks of the same kind related to linguistic properties were performed. Moreover, in this work, the authors study how fine-tuning BERT for specific tasks, for instance, sentiment analysis, influences the behaviour of the heads. They show that the last two layers of BERT are much more task-oriented with respect to the first ones, which usually focus on low-level linguistic information. Differently from these two works, we do not focus our study on linguistic or grammatical properties. Instead, we study how heads are involved in capturing semantic information and factual knowledge regarding real-world entities, without considering linguistic aspects.

Regarding factual knowledge, several works focused on assessing whether this kind of information is present in BERT or similar models or is not. One of the most important ones is presented in [10], where the authors compared Transformer-based models with traditional NLP methods that have access to oracle knowledge and test them on open-domain questions. They define the LAMA dataset and use it as a probe for verifying how different kinds of LLMs (with a special focus on BERT) can answer questions regarding well-known facts. Their results show that BERT actually has knowledge regarding basic semantics, geography, sports, famous people and other domains. Similar results are obtained also for multi-lingual models by the work in [19]. In this work, we do not try to assess whether Transformer-based models possess these types of knowledge: we take this result for granted. Instead, we aim to discover the relation between a specific part of their architecture, the heads, with basic semantic information and factual knowledge.

Subsequent works such as [8,9] studied the performance variation on the LAMA dataset using different sentences and how expressing facts in different ways influences the model predictions. This was observed also in [11], where the PARAREL dataset was defined. This dataset contains well-known facts represented as two entities and a relation, and the latter is paraphrased in different ways in order to express the same relationship with different words. They found that the BERT and RoBERTa outputs are not consistent with a paraphrased input. Similarly to what we discussed previously, these works focus on the entire architecture without looking in depth into the role of the heads. However, we found their results very interesting and we performed an experiment (please see Section 5.2) specifically designed for considering different types of sentences.

Although these studies prove that the Transformer-based models contain numerous information of different kinds, we found only one work regarding how this knowledge is stored inside the architecture. The work in [20] focuses on the neurons of Feed-Forward Networks in BERT encoding layers and found that different queries for specific facts mostly activate specific units, identifying them as the neurons containing that knowledge. In our work, we perform a parallel analysis, focusing on heads, the attention mechanisms of Transformer-based models, to identify which kinds of knowledge are contained in them.

Specifically for semantics, the work in [21] focuses on the word embedding representations produced by BERT models, retrieving the relations between them and using them to generate a Knowledge Graph. They also found that these word vectors contain basic semantic information by training a classifier that has the duty to predict the general category (extracted by Wikipedia) to which the word belongs. Differently from this work, we do not take into consideration the word embedding. Instead, we measure how heads are related to some basic semantic capabilities, such as identifying if two words are synonyms or antonyms, or verifying whether specific heads assign a high weight to a pair formed by a word and its category.

In a preliminary work published in [22], we manually observed the behaviour of BERT heads in the context of the classification of clinical reports in Italian [23,24]. We found that several heads could capture some related medical concepts and synonyms, which were manually annotated. In the present work, we conduct a more thorough study

defining a clearer methodology and evaluation procedure, with new and better-defined metrics. Moreover, in this work, we rely on standard datasets (based on the state-of-the-art techniques for assessing the factual knowledge of BERT) without the need of manual annotations. Finally, we perform a much more detailed analysis with four different BERT models (with respect to a single model for the Italian language) and seven different datasets.

3. Methodology

Differently from most works, which analyse heads in order to discover their grammatical capabilities, our goal is to discover the role of the heads in capturing semantic and factual knowledge in BERT architectures. In particular:

- Instead of analysing the entire architecture or relying on manual observations [14,15], we aim to identify which heads are the most promising ones to contain this kind of knowledge through calculating an evaluation metric (called **Self Metric**).
- For each head we have identified, we extract which pairs of tokens have the highest weights assigned by the head. This is conducted with a custom-made algorithm (the **Linker Algorithm**).
- We compare these pairs of tokens with some ground truth pairs. More specifically, we aim to verify whether a head (or a specific group of heads) is able to capture if two words are related by some semantic or factual knowledge, such as the one between the synonyms important and meaningful, or the state–capital relation between Paris and France. We claim that, if a head gives an high attention weight specifically to these pairs of words considering different examples, then it is able to capture such knowledge.

In the following, we provide a more detailed description of how our datasets are structured. Next, we describe the methods and techniques we designed (the Self Metric and the Linker Algorithm) and how we evaluated our experiments and which metrics were used.

3.1. Dataset Structure

Despite having considered different datasets built on different case studies (which we describe in Section 4), such as pairs of synonyms, antonyms or a disease and the drug that treats it, all our datasets have the same structure.

First, we consider a relation r, which describes a particular type of link between two words. For instance, a relation "Capital of" can represent the link between the name of a state (such as Italy) and a city (such as Rome). Next, we associate a prompt P to the relation r. The prompt is a sentence, and therefore a sequence of tokens, which express r in natural language. For instance, "The country called [X] has [Y] as its capital" can be associated to the "Capital of" relation. Finally, we define a series of Data Pairs. Each Data Pair $D = (X, Y)$ is a pair of words that are related by r, such as the previously mentioned Italy and Rome. Therefore, each instance of our dataset can be seen as a triple $\langle D, r, P \rangle$, into which a relation r connects a Data Pair D as expressed by a prompt P.

In Figure 2, we show a toy example of our Capitals dataset, into which we consider the "Capital of" relation, its prompt and three different real-world data pairs (Italy and Rome, Spain and Madrid, France and Paris). Therefore, we obtain three instances into which the prompt "The country called [X] has [Y] as its capital" is filled with the three data pairs.

3.2. Self Metric

The intuition behind the use of this metric for identifying the most promising heads, in terms of semantic or factual knowledge, comes from manual observation and from some analyses performed in a previous preliminary work presented in [22]. However, analysing the behaviour of these heads over a dataset of medical reports, some of them did not assign high attention value only to equal tokens but also to words that are semantically related. These words could be basic synonyms (such as segments and portions), antonyms (left and right); however, they also could be more complicated medical concepts, such as artery and

aorta, nodule and lesion or texture and parenchyma. These preliminary results lead us to continue such analysis in a more structured way. Therefore, in the first step of the analysis, we designed the Self Metric in order to identify the ones showing this particular behaviour.

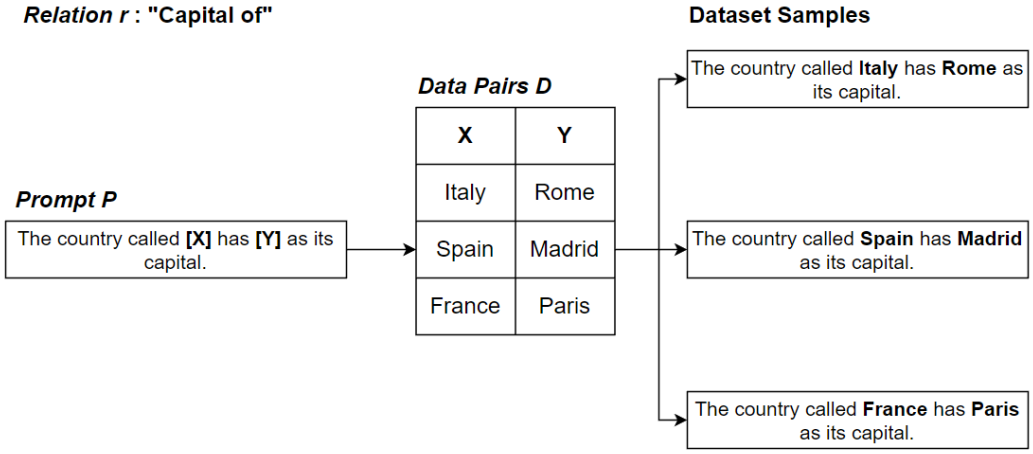


Figure 2. Example of our dataset structure. Starting from the "Capital of" relation, we associate it with a general prompt, which is filled by three real-world pairs made by a state and its capital. On the right, we show the actual dataset instances, which are formed by filling the prompt with the data pairs considered.

The Self Metric σ_w is defined by the Jensen–Shannon distance between a_w, the vector of the attention values of a head on a token w in relation to the other tokens in the sentence or document, and a binary vector B_w, where the 1s are in the position of w and where this token repeats itself.

For instance, in the sentence "The earth revolves around the sun. The moon revolves around the earth", analysing the token w = "the" (case-insensitive), we will have a binary vector $V_{the} = [1,0,0,0,1,0,0,1,0,0,0,1,0,0]$, and we will compute the distance between it and a_w. If we analyse the token w = moon, instead, the vector is $V_{moon} = [0,0,0,0,0,0,0,0,1,0,0,0,0,0]$.

More formally,

$$\sigma_w = JSD(a_w||V_w),\ V_w[k] = \begin{cases} 0 & V_w[k] \neq w \\ 1 & V_w[k] = w \end{cases} \quad (3)$$

where k are the positions of the tokens in the document. Considering an entire document, the Self Metric of a document is calculated as the average of the Self Metrics of all the tokens in a document.

3.3. Linker Algorithm

In [14,16], the connections between pairs of tokens are simply shown with visualization techniques, with lines of different thickness on the basis of the attention weights, as can be seen in the left part of Figure 3. However, with relatively long sentences or documents, the amount of connections increases drastically, making the visualization less understandable and very complex to compute. Therefore, our approach is to directly extract the most important connections among tokens from a specific head and verify if they are related to some specific semantic or factual knowledge comparing them to a ground truth.

However, extracting these connections is not a trivial task. In fact, the attention weights distribution can vary across different weights. For instance, as is shown in [16], some heads basically distribute their attention evenly across the whole sentence, while other heads connect each token with just another token. Moreover, this behaviour can differ (as can be seen on the left of Figure 3) across different tokens even in the same sentence. Therefore,

we cannot simply extract the pairs of tokens with the highest attention weights and we need a more refined procedure.

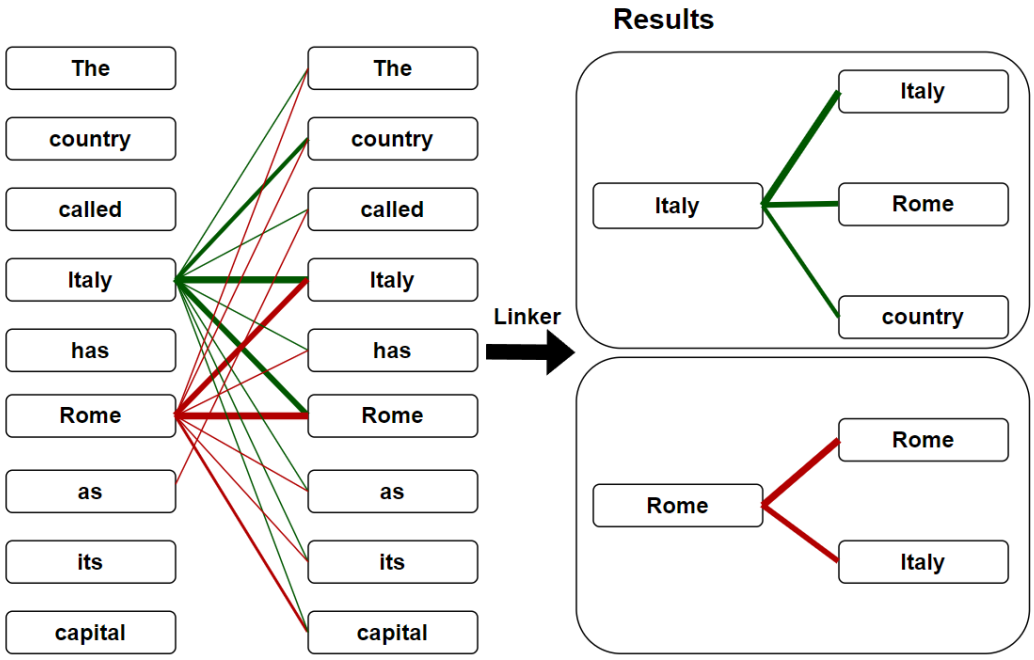


Figure 3. Example of how the Linker Algorithm works on the words Italy and Rome. On the left, we show a visualization of the attention weights of these two words, into which the thickest lines indicate a higher value of attention between the word pairs. On the right, we show the word pairs returned the algorithm, which is the cluster of words with the attention weights extracted with the Mean Shift algorithm [25]. These results can be noisy, and, in this example, we can see that our algorithm can return pairs that are not the ones we want to identify, such as (Italy, country). We use different colours for different tokens.

Hence, our algorithm for automatically finding these connections is based on Mean Shift [25], a clustering algorithm suitable for density functions and one-dimensional clustering [26]. Given a distribution of attention weights $a_w = [\alpha_1, \alpha_2 \dots \alpha_N]$ for a token w, the different $\alpha_i \in a_w$ are grouped into several clusters depending on their value. Then, the Linker Algorithm returns the cluster with the highest average weight, containing the most valuable pairs of words. Please note that this algorithm works only at the numeric level of the attention weights and it has no information regarding the tokens.

A simplified representation of the results of this algorithm is presented in Figure 3. On the left, we have a typical visualization of the attention weights very similar to the one provided by [16], into which the pair of words with the highest attention weights are connected by a thick line. On the right, we have the output of the algorithm for the tokens Italy and Rome, showing which words are mostly related to them according to the head. Please note that the algorithm can provide a variable number of relations with a specified token. For instance, Italy is mostly related to 3 tokens (itself, Rome and country), while Rome is connected to itself and Italy.

3.4. Evaluation Metrics

Once we have extracted the word pair pairs with the Linker Algorithm, we can compare them to some ground truth relations. Thus, we can verify whether a head possesses some specific knowledge and quantify it through specific evaluation metrics.

For this purpose, we defined an evaluation metric with the same idea of the *F*1-Score. We compute this metric as the harmonic mean between precision and recall, which we define this way:

$$precision = \frac{\#Identified}{\#Pairs} \tag{4}$$

$$recall = \frac{\#Identified}{\#Dataset} \tag{5}$$

where $\#Identified$ is the number of word pairs retrieved by the Linker Algorithm for the head that are correctly identified; $\#Pairs$ is the total number of pairs retrieved by the Linker Algorithm and $\#Dataset$ is the number of ground truth pairs present in the dataset. With these metrics, we can compute the *F*1-Score as:

$$F1\ score = \frac{2 * precision * recall}{precision + recall} \tag{6}$$

Another evaluation metric we considered is accuracy, which is the number of word pairs correctly identified (by a head or a group of heads) over the total number of pairs in the dataset. In particular, we considered two different types of accuracy, acc_N and acc_{Tot}, which are computed this way:

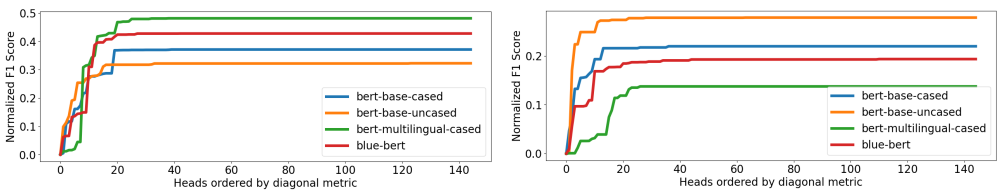
$$acc_N = \frac{\#Identified_N}{\#Dataset} \tag{7}$$

$$acc_{Tot} = \frac{\#Identified_{Tot}}{\#Dataset} \tag{8}$$

into which $\#Identified_N$ is the number of pairs correctly identified by N heads, and $\#Identified_{Tot}$ is the number of pairs correctly identified by all the heads in the BERT model.

The idea behind these two metrics is the following. We consider acc_{Tot} as a baseline, which represents all the possible knowledge contained in all the heads across the entire BERT architecture. If a relationship between two terms is not identified by any of the heads, that information is contained in other parts of the architecture or not known by the model. Nonetheless, we are not interested in just verifying if that information is present or not but aim to create a selection mechanism into which the Self Metric identifies the most promising heads in terms of semantics and factual knowledge. Therefore, we design also the acc_N metric, into which we consider only the first N heads identified by the Self Metric. If acc_N is very similar or equal to acc_{Tot}, we can say that our method for selecting the most promising heads performs well because we select those heads that contain all (or almost all) the knowledge of the entire model. Instead, if acc_N is much lower than acc_{Tot}, our Self Metric is not able to identify the most promising heads.

After some preliminary experiments, as described in Section 5 (and, in particular, in Figure 4), we selected $N = 20$.



(**a**) Synonyms (**b**) Capitals

Figure 4. Cumulative Normalized *F*-Score progressively considering more heads, which are ordered on the basis of the Self Metric value. These results are obtained considering the Synonyms (**a**) and Capitals (**b**) datasets for all the models we considered.

4. Case Studies

We tested our approach on different types of knowledge: Semantic Knowledge, which is related to the words and their meaning, and Factual Knowledge, with three different domains: Geography, General Knowledge and Medicine. For each of these relations, we selected pairs entities (which are related by some ground truth factual knowledge) and a prompt to compose the sentences, as explained in Section 3.

Our datasets follow (and, sometimes, are based on) the general scheme defined by PARAREL [11] and T-REX [12], which are the state-of-the-art examples for verifying the factual knowledge of Transformer-based models [8,9,19,20]. Please note that these datasets only consider words made by one token. Although this can be a limitation, this configuration is also the most suitable for our study since we extract pairs of tokens.

As observed in [14,22], in the first layers of BERT, there are heads that split attention evenly across all word pairs. In order to avoid that such heads obtained high $F1$-Scores, we chose prompts long enough to penalise lucky guesses. With respect to PARALEL, we use longer prompts. In fact, the average length of our instances is 14 words, while PARAREL has an average length of five words. Although it could be interesting to study the performance of our method considering longer sentences (or even entire documents taken directly from real-world applications), this is left as a future work.

In the following, we describe in detail the datasets we considered.

4.1. Semantic Knowledge

For testing the semantic capabilities of BERT, we chose two different domains: Synonyms and Antonyms and created two datasets with pairs belonging to these categories:

- **Synonyms:** we took the dataset containing synonyms from Kaggle (https://www.kaggle.com/datasets/duketemon/wordnet-synonyms accessed on 26 June 2023), which is based on WordNet [27]. We randomly selected 250 synonym pairs and filled the prompt: *"If you would ask me to describe it I could say that it is [X], or, in other words, it is [Y]"*, replacing the [X] and [Y] with a word and its synonym. For example, a pair of synonyms is (frigid, cold).
- **Antonyms:** Similarly to the procedure we used for the Synonyms, we took the WordNet-based antonyms dataset from Kaggle (https://www.kaggle.com/datasets/duketemon/antonyms-wordnet accessed on 26 June 2023) and we randomly selected 250 antonym pairs considering the prompt *"You described it as [Y], but I would say that it is the opposite, I would describe it as [X]"*, replacing the [X] and [Y] with a word and its antonym. For example, a pair of antonyms is (hot, cold).

4.2. Real-World Factual Knowledge

While Semantic knowledge is based on common words that can be identified because they belong to the same semantic field, this type of knowledge is typically based on nouns such as the name of cities, drugs or other entities. Here we aim to verify if the model is able to identify pairs of words related by some known facts, such as the name of a state and the name of its capital. Despite these differences, the datasets we designed followed the same scheme we showed in the previous sections. The domains we considered for testing the capabilities of our model are Geography, General Knowledge and Medicine (Drugs). In detail, we created these datasets:

- **States and Capitals:** we created a dataset using state–capital pairs that can be commonly found on the Internet. In order to simplify the analysis of the relationship based on attention, we took only the pairs where both capital and nation names were one word long in order to have precise word pairs to compare with the ground truth labels. Thus, we created a dataset of 159 sentences based on the state–capital pairs and the prompt *"The country called [X] has the city of [Y] as its capital"*. For example, a pair state–capital is (Italy, Rome).
- **Locations:** we used some entities included in the T-REX dataset [12], containing a state and a well-known place belonging to that state. As we did for the States and

Capitals, we consider one-word entities. The dataset is created from 95 sentences and exploits the prompt *"[X] is a place of great fame and it is located in the country of [Y]"*. For example, a relation location nation is (Catalonia, Spain).

- **Belongs to:** this dataset is based on the T-REX dataset and contains pairs made by a concept and a more general category to which the concept belongs. It is made by 220 sentences with the prompt *"As we all know [X] belongs to the bigger category of [Y]"*. For instance, it contains the relation (Champagne, Wine).
- **Part of:** this dataset contains pairs of sets, into which one is a subset of the other. It contains 260 sentences with the prompt *"It has been proven that [X] is a specific part of [Y]"*. For example, it contains the relation (Torah, Bible). Although this dataset is quite similar to the previous one (also based on T-REX), please note that we used a completely different prompt for better generalization.
- **Medicine:** this dataset contains pairs (which are commonly available online) of drugs and medical conditions treated by them. It contains 100 sentences and it exploits the prompt *"The medicine [X] is commonly used for the treatment of [Y] and other medical conditions"*. For example, it contains the relation (Aspirine, Inflammation).

5. Experimental Settings and Evaluation

In this section, we describe how we conducted the experimental evaluation of our approach and we present our results. We applied these techniques to different BERT models. All these models have the same architecture (12 layers and 12 heads for each layer), and they are available on HuggingFace. They mostly differ in case-sensitivity, training process and training data. However, our approach is completely general and it can be applied also to different architectures based on Multi-Head Attention Mechanisms. For instance, we tested it on a BERT model with 8 layers and 8 heads for each layer, obtaining similar results.

The models considered are the following ones:

- bert-base-uncased (https://huggingface.co/bert-base-uncased accessed on 26 June 2023), a BERT English model insensitive to capital letters [1];
- bert-base-cased (https://huggingface.co/bert-base-cased accessed on 26 June 2023), a BERT English model sensitive to capital letters [1];
- bert-base-multilingual-cased (https://huggingface.co/bert-base-multilingual-cased accessed on 26 June 2023), a multilingual BERT model [1];
- bluebert (https://huggingface.co/bionlp/bluebert_pubmed_uncased_L-12_H-768_A-12 accessed on 26 June 2023), an uncased BERT model training on a generic corpus and on medical documents [2].

5.1. Experimental Results

First of all, we computed Self Metric, as defined in Section 3.2, on each head of the model for a subset of sentences belonging to the Synonyms dataset, and we identified the most promising heads according to the metric. In Table 1, we show the five most promising heads according to the Self Metric across all the considered domains for the bert-base English models (cased and uncased). We can notice that the heads are almost entirely the same across all the domains. This means that the behaviour of the heads is mostly independent from the data we considered and the prompt we designed. However, we can also see that the heads identified by the Self Metric are completely different between bert-base-cased and bert-base-uncased. For the other models, we obtained very similar results, which are not reported for brevity's sake. A more thorough analysis of this phenomenon is reported in Section 6.

After computing the Self Metric for each token, we can select the more promising heads and verify if they connect words belonging to the same semantic field or if they identify relations based on some factual real-world knowledge, exploiting the Linker Algorithm described in Section 3.3. In order to select an adequate number of heads for our study, we conducted a preliminary experiment on the Synonyms dataset, considering the bert-base

English models. In this experiment, we evaluate the Cumulative Normalized *F*-Score, which is calculated as follows:

$$\text{Cumulative Normalized } F\text{-Score}_{(N)} = \sum_{i=1}^{N} \frac{F\text{-Score}_i}{L * H} \quad (9)$$

We progressively considered N heads (from 1 to 144) from the most promising to the least promising according to the Self Metric. For each head, we computed its *F*-Score and divided, as a normalization process, by the number of layers (L) multiplied by the number of heads per layer (H). Then, we calculated the sum of the *F*-Score of the N heads.

Table 1. Most important heads identified by the Self Metric and ordered by their value for the bert-base-uncased and bert-base-cased models. Each head is expressed by a pair (l, h), where l is the layer number and h is the head number of the l-th layer.

(a) Bert-Base-Uncased						
Synonyms	Antonyms	Capitals	Locations	Part of	Belogns to	Medical
(3,7)	(3,7)	(3,7)	(3,7)	(3,7)	(12,9)	(3,7)
(12,9)	(12,9)	(12,9)	(12,9)	(12,9)	(3,7)	(12,9)
(4,1)	(11,11)	(2,12)	(2,12)	(2,12)	(2,12)	(2,12)
(2,12)	(11,10)	(11,10)	(11,10)	(11,10)	(11,11)	(4,1)
(11,10)	(2,12)	(11,11)	(4,1)	(11,10)	(11,10)	(11,10)
(b) Bert-Base-Cased						
Synonyms	Antonyms	Capitals	Locations	Part of	Belogns to	Medical
(12,4)	(12,4)	(12,4)	(12,4)	(12,2)	(12,4)	(12,4)
(12,12)	(12,12)	(12,12)	(12,12)	(12,4)	(12,12)	(12,12)
(4,8)	(4,8)	(12,3)	(12,3)	(12,3)	(12,3)	(12,3)
(12,3)	(12,3)	(4,8)	(4,8)	(4,8)	(11,3)	(4,8)
(5,1)	(5,1)	(10,2)	(3,6)	(3,3)	(3,6)	(3,6)

The results for the Synonyms dataset and for the Capitals dataset are shown in Figure 4. We can see that the normalized *F*-Score increases rapidly considering the first heads identified by the Self Metric and stops after 20 or 25 heads, confirming the capability of our metric to identify this desired behaviour and that the remaining heads perform some other types of operations not related to semantics or factual knowledge but probably most focused on grammatical properties [14]. The results are similar in the other domains we considered, except for the bert-multilingual-cased model in the Belongs to, Part of and Medical datasets and for the bert-base-cased model in the Medical dataset. These issues are discussed more thoroughly in Section 6. Therefore, thanks to this preliminary study, we selected $N = 20$, thus identifying 20 heads as the most promising for the next experiments across all the domains considered.

The results of our approach that extracts pairs of words for the most promising 20 heads are presented in Table 2 in terms of acc_{20}, and they are compared to the baseline, i.e., considering all heads (acc_{Tot}). In terms of acc_{20}, considering the Semantics domains, our approach obtains remarkable results. Here, the accuracy of the approach on 20 heads is at most 0.05 lower than the acc_{Tot}. For instance, considering bert-base-uncased model in the Synonyms domain, we obtained 0.84 in acc_{Tot} and 0.82 in acc_{20}. This means that those heads contain almost all the knowledge of the model on these domains and that our Self Metric correctly identified them.

The performance is very good also on the Geography domains. For the **States and Capitals** dataset, we can see a very small drop in performance between acc_{Tot} and acc_{20}, with performances always higher than 0.8. On the **Locations** dataset, the results even reach

0.97 in terms of acc_{20} with bert-base-uncased. However, we have a drop in accuracy, with 0.62 in terms of acc_{20} and 0.9 in terms of acc_{Tot} using the multilingual version of BERT, which probably stores this kind of knowledge in a different way.

Table 2. Accuracy of pairs correctly identified by all heads and the top 20 heads in the Self Metric for each model. We have highlighted in bold the best performance of our approach for each domain.

Model	Synonyms		Antonyms		Capitals		Locations	
	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}
bert-base-uncased	0.84	**0.82**	0.77	0.76	0.85	0.84	0.97	**0.97**
bert-base-cased	0.84	0.80	0.76	0.76	0.87	**0.86**	0.94	0.94
bert-multilingual-cased	0.84	0.79	0.77	0.74	0.84	0.82	0.90	0.62
bluebert	0.84	0.81	0.77	**0.77**	0.87	0.81	0.93	0.90
Model	**Belongs to**		**Part of**		**Medical**			
	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}		
bert-base-uncased	0.91	**0.75**	0.82	0.68	0.87	0.69		
bert-base-cased	0.91	0.52	0.81	**0.75**	0.88	0.21		
bert-multilingual-cased	0.91	0.67	0.81	0.53	0.87	0.46		
bluebert	0.91	0.70	0.82	0.56	0.88	**0.79**		

From this analysis and considering the fact the we considered (as can be seen in Table 1) basically the same heads, we claim that the models see the geography relations between a state and its capital and between a place (such as a region) and the state where it is located as semantically related, as if they were synonyms, providing an interesting perspective on how factual knowledge can be memorised by a pre-trained language model.

Considering the other domains, our approach still performs well. However, we can see that we have a larger difference between acc_{Tot} and acc_{20}. For instance, in the **Belongs to** dataset using bert-base-uncased, we obtain 0.75 in terms of acc_{20}, while acc_{Tot} reaches 0.91. The most significant case, however, is regarding the **Medical** dataset. In fact, considering the bert-base-cased model, we can see a 0.21 in terms of acc_{20}, while the accuracy calculated on all the heads is 0.88. Therefore, we claim that the knowledge between a drug and the disease it treats is not captured by the heads we selected and, with respect to the other domains, cannot be seen as semantically related.

An interesting fact can be observed by analysing the blue-bert model, which is fine-tuned on documents about medicine and biology, regarding the Medical domain. We can see that our approach produces better acc_{20} (0.79) than all the other models while maintaining basically the same acc_{Tot}. From these results, we can make a few observations. First, it seems that the relations between the most common drugs and the corresponding disease are also present in the general-purpose models, and a fine-tuning with medical documents does not increase that kind of knowledge. However, the fine-tuning process seems to modify the way it is stored inside the BERT architecture.

Please note that, in all configurations, the acc_{Tot} is never equal to 1, which means that not all the semantic relations or the factual knowledge are captured by the self-attention mechanism and they perhaps could be contained in other parts of the architecture, like the feed-forward layers [20].

If we look at the models, we can see that bert-base-uncased is the model where our approach is more stable, but, in the Semantics and Geograpy domains, all the models perform great. During the experiments, we observed that, for bert-base-cased and bert-base-multilingual-cased, it was more difficult to distribute attention between words due to the presence of capital letters, which makes the same word correspond to two different tokens or makes the tokenizer split the word into several tokens. Moreover, the further reduction in performance in bert-multilingual-cased might be caused by the presence of other languages known by the model, Thus, it is less confident in sharing the attention between words. The results on the approach regarding bluebert show that its knowledge

is quite the same as bert-base-uncased, and it seems that fine-tuning does not affect the general knowledge of the model negatively.

5.2. Robustness to Different Prompts

We ran an additional experiment to check the behaviour of our approach with different prompts. We created new datasets of synonyms and antonyms, selecting five different prompts and forty entities, for a total of two-hundred samples per domain. We then analysed the results of our techniques, checking that they identified the same entities regardless of the prompt in which they were entered. The prompts we designed for the study on the Synonyms domain are:

- I could describe it as [X], or, to put it another way, it is [Y].
- If I had to put it into words, I would say it is [X], or, to use a synonym, it is [Y].
- You might call it [X], but I would describe it as [Y]—they are essentially synonyms.
- In my opinion, it is [X], or, to use a similar term, it is [Y].
- If you were to ask me for a word to describe it, I might choose [X], or, to put it synonymously, [Y].

Similarly, for the Antonyms domain, we designed the following prompts:

- You used the term [X], but I would argue that it is actually quite the opposite. I would describe it as [Y].
- While you described it as [X], I believe that it is actually the antithesis. I would say it is [Y].
- Your description of it as [X] does not quite fit; I think the antonym is more accurate. I would label it as [Y].
- Although you referred to it as [X], I think there is a more fitting antonym. I would say it is [Y].
- Your characterization of it as [X] does not match my perception; the opposite seems more appropriate. I would describe it as [Y].

The results of these experiments are available in Table 3. We can see that our method exhibits a remarkable level of robustness with respect to the prompt used, allowing for flexibility and adaptability in various contexts. Despite the diverse prompts employed, our methods obtain stable accuracy in both domains, demonstrating their robustness despite the context in which the entities are presented.

Table 3. Accuracy of pairs correctly identified by all heads and the top 20 heads in the Self Metric for each model on Synonyms and Antonyms domains, on each different prompt, to prove the robustness of our approach.

(a) Synonyms

Model	Prompt 1		Prompt 2		Prompt 3		Prompt 4		Prompt 5	
	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}
bert-base-uncased	0.9	0.8	0.9	0.83	0.9	0.88	0.9	0.85	0.9	0.88
bert-base-cased	0.88	0.83	0.90	0.80	0.90	0.78	0.90	0.85	0.90	0.88
bert-multilingual-cased	0.90	0.83	0.90	0.78	0.90	0.83	0.90	0.83	0.90	0.90
bluebert	0.88	0.75	0.90	0.90	0.90	0.78	0.90	0.85	0.88	0.88

(b) Antonyms

Model	Prompt 1		Prompt 2		Prompt 3		Prompt 4		Prompt 5	
	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}	acc_{Tot}	acc_{20}
bert-base-uncased	0.9	0.8	0.9	0.83	0.9	0.88	0.9	0.85	0.9	0.88
bert-base-cased	0.83	0.55	0.83	0.60	0.83	0.60	0.83	0.63	0.83	0.63
bert-multilingual-cased	0.83	0.78	0.83	0.75	0.83	0.78	0.83	0.83	0.83	0.80
bluebert	0.80	0.80	0.80	0.78	0.80	0.78	0.80	0.78	0.80	0.80

6. Discussion

In this section, we discuss more thoroughly our results and their consequences and we contextualize them with our initial goals.

Our first objective was to find a technique to identify heads that focus on basic semantic information, for example, if two words are synonyms or antonyms. We defined the Self Metric to highlight heads that strongly connect a word with itself inside a sentence, with the assumption that they also linked those semantically related words. However, from the results we reported in Table 1, we have found that the values of the Self Metric, and, therefore, the overall behaviour of the heads, did not regard only synonyms or antonyms but also factual knowledge in domains such as geography. We claim that this is one of the first results of our study because we show how semantics and factual knowledge can be treated in the same way by a Transformer-based model as if the name of a state and the name of its capital were treated as synonyms. Nonetheless, considering other domains could lead to further results, and we want to conduct a more in-depth study of this kind of behaviour as future work.

Another aspect to be considered is that there is no guarantee that these heads are located in the same positions in different models. In fact, in Table 1, we can see that the first five heads identified by the Self Metric are completely different in the bert-base-uncased and in the bert-base-cased models. Given that these models have exactly the same architecture, we claim that this difference is simply due to the randomness of the training procedure. In fact, these two models are trained with the same data, apart from the fact that the uncased model does not deal with capital letters, and with the same algorithm and hyperparameters. Therefore, less-measurable aspects, such as weight initialization, can produce some differences regarding which heads do what. As a consequence, models trained with substantially different corpora of documents (such as the multilingual model or bluebert, which is trained also on medical documents) exhibit the same behaviour. From these experiments, we show that different types of BERT models have several heads into which tokens are mostly connected to themselves. This is coherent with the results shown in [15]. Therefore, we claim that these heads present something that should be learned by a model, and it is very useful for NLP tasks that BERT is able to solve.

Another important aspect that should be analysed is the number of heads we have to consider. As we reported in Section 5 (in particular, in Figure 4), we see that all models reach their performance in terms of the Cumulative Normalized *F*-Score considering about 20 heads for both the Synonyms and the Capitals datasets. Although similar results were obtained for most of the other cases we considered, there are a few notable exceptions: the Medical domain (except for the bluebert model); the bert-base-cased and bert-multilingual-cased models on the Belongs to dataset and the bert-multilingual-cased model on the Part of dataset. In these cases, more heads could be considered in order to obtain better performance. This behaviour can be seen also in Table 2, where these experiments are the ones with lower acc_{20}. In a further analysis, we noticed that the important heads are included in the first 30 and not the first 20, and, in general, all the models reach an acc_{20} equal to acc_{Tot} in the first 40 heads. Given these experiments, we claim that, for these models, this type of knowledge is captured by heads that are not easily identifiable by the Self Metric, and they exhibit different behaviour. For the Medical dataset, this could be due to the specificity of the task, which involves strictly medical terms regarding drugs and diseases, which are not exactly known to the models trained on corpora of general documents. In fact, bluebert (which is trained also on medical documents and research articles) obtains very good results in its first 20 heads, exactly as the other models for the more general tasks. Moreover, the difficulties of the multilingual model could be due to the fact that this model has to learn several languages, and, therefore, it has much more knowledge to encode and distribute across its heads. Moreover, this model has a much larger vocabulary, so it could be less confident in connecting important words. A more in-depth study of the behaviour of multilingual models could be performed as a future work.

We obtained great results over our target datasets, Synonyms and Antonyms. In these experiments, the acc_{20} is at most 5 points below the acc_{tot}, which means that the most important heads are correctly identified by our metric. This confirms our initial hypothesis, that the models see semantic-related words in the same way as repetition of the word itself. Seeing these results, we extended our approach to different types of knowledge belonging to real-word knowledge. We found that our approach works very well also on geographic knowledge. In this experiment, we reached an acc_{20} almost equal to acc_{Tot}, which means that, in this case as well, our approach identified the most important heads. The last domains, Belongs To, Part Of and Medical, have lower accuracy. This could be due to the fact that the models see this knowledge in a different way than before. Therefore, these entities are not seen as semantically related, but there are heads with different behaviours that connect them. This leads to the question of how the model classifies different types of knowledge and what head behaviour identifies them, and that could be answered with future works on the topic.

Finally, we conducted another experiment on the robustness of our method, evaluating the results of our approach with five different prompts for the Synonyms and Antonyms datasets. Our results show another important aspect of our work: in fact, we obtained very similar results considering different sentence prompts in the identification of synonyms and antonyms. Given that the behaviour of the heads identified by the Self Metric does not depend on context (most of the tokens simply are linked to themselves, with the notable exceptions of synonyms, antonyms and factual knowledge), we claim that this form of knowledge in a Transformer-based model is not strongly influenced by the overall context of the sentence but can be inherent to the model itself. However, more experiments should be conducted on this subject, considering longer and more complex prompts.

7. Conclusions and Future Work

In this work, we proposed a new metric to identify heads containing semantic knowledge in large language models based on attention, in this case based on the BERT architecture. In order to study the relationship between words, we designed an algorithm based on Mean Shift [25] that returns the word pairs with the highest attention weights.

Comparing the mostly related tokens according to the heads we identified together with the ground truth labels, we verified that these heads can capture synonyms and antonyms. Next, we applied our methods to different factual knowledge domains, such as geography, general knowledge and medicine. We found out that the approach works well in geographical domains as if the model sees these entities as semantically related. On the other hand, we also noticed some difficulties in particular areas, such as the medical domain. Therefore, we claim that the model has different ways of storing various kinds of knowledge. However, it is possible that slight variations in our technique could identify other types of heads and their knowledge.

Although this work does not provide a full explanation of how BERT works, we claim that we have found something worthy of interest on the behaviour of attention mechanisms. However, further studies could be conducted. We defined only one of the possible metrics that can describe a behaviour of the model. Future studies can focus on identifying other behaviours and designing different metrics to analyse them.

These techniques can be applied to more domains, such as history, physics or chemistry, in order to test the performance of our methodology with other types of knowledge.

Moreover, for the Synonyms and Antonyms datasets, we also found that our method provides stable results with different prompts. As future work, we could expand this study considering more prompts and using more complex sentences taken directly from real documents.

In addition, our techniques could be redesigned or adapted to other BERT models or other architectures based on self-attention, like T-5, Reformer or GPT models. Finally, these methods can lead to a study on how the model manages different kinds of knowledge, designing different metrics.

Author Contributions: Conceptualization, L.P. and A.E.G.; Methodology, L.S., L.P. and I.S.; Software, L.S.; Validation, L.S. and A.E.G.; Investigation, L.P.; Data curation, L.S.; Writing—original draft, L.S. and L.P.; Writing—review and editing, L.S., L.P., A.E.G. and I.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the MIUR "Fondo Departments of Excellence 2018-2022" of the DII Department at the University of Brescia, Italy. The IBM Power Systems Academic Initiative substantially contributed to the experimental analysis.

Data Availability Statement: All the data and the source code of this work are available at the following GitHub repository: https://github.com/LorenzoSerina-UniBS/Synonyms-Antonyms-GeneralKnowledge-BERT-Heads.git (accessed on 26 June 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. 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, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Cedarville, OH, USA, 2019; Volume 1 (Long and Short Papers), pp. 4171–4186.
2. Peng, Y.; Yan, S.; Lu, Z. Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets. In Proceedings of the 2019 Workshop on Biomedical Natural Language Processing (BioNLP 2019), Florence, Italy, 1 August 2019; pp. 58–65.
3. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
4. Tenney, I.; Das, D.; Pavlick, E. BERT Rediscovers the Classical NLP Pipeline. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019.
5. Miaschi, A.; Brunato, D.; Dell'Orletta, F.; Venturi, G. Linguistic Profiling of a Neural Language Model. In Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), 8–13 December 2020; International Committee on Computational Linguistics: Cedarville, OH, USA, 2020; pp. 745–756.
6. Jawahar, G.; Sagot, B.; Seddah, D. What Does BERT Learn about the Structure of Language? In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Cedarville, OH, USA, 2019; Volume 1: Long Papers, pp. 3651–3657.
7. Lenci, A.; Sahlgren, M.; Jeuniaux, P.; Gyllensten, A.C.; Miliani, M. A comparative evaluation and analysis of three generations of Distributional Semantic Models. *Lang. Resour. Eval.* **2022**, *56*, 1269–1313. [CrossRef]
8. Jiang, Z.; Xu, F.F.; Araki, J.; Neubig, G. How Can We Know What Language Models Know. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 423–438. [CrossRef]
9. Petroni, F.; Lewis, P.S.H.; Piktus, A.; Rocktäschel, T.; Wu, Y.; Miller, A.H.; Riedel, S. How Context Affects Language Models' Factual Predictions. In Proceedings of the Conference on Automated Knowledge Base Construction, AKBC 2020, Virtual, 22–24 June 2020; Das, D., Hajishirzi, H., McCallum, A., Singh, S., Eds.; Association of Computational Linguistics: Cedarville, OH, USA 2020. [CrossRef]
10. Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.S.H.; Bakhtin, A.; Wu, Y.; Miller, A.H. Language Models as Knowledge Bases? In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; Inui, K., Jiang, J., Ng, V., Wan, X., Eds.; Association for Computational Linguistics: Cedarville, OH, USA, 2019; pp. 2463–2473. [CrossRef]
11. Elazar, Y.; Kassner, N.; Ravfogel, S.; Ravichander, A.; Hovy, E.H.; Schütze, H.; Goldberg, Y. Measuring and Improving Consistency in Pretrained Language Models. *Trans. Assoc. Comput. Linguist.* **2021**, *9*, 1012–1031. [CrossRef]
12. ElSahar, H.; Vougiouklis, P.; Remaci, A.; Gravier, C.; Hare, J.S.; Laforest, F.; Simperl, E. T-REx: A Large Scale Alignment of Natural Language with Knowledge Base Triples. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, 7–12 May 2018; Calzolari, N., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Hasida, K., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., et al., Eds.; European Language Resources Association (ELRA): Paris, France, 2018.
13. Rogers, A.; Kovaleva, O.; Rumshisky, A. A Primer in BERTology: What We Know About How BERT Works. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 842–866. [CrossRef]
14. Clark, K.; Khandelwal, U.; Levy, O.; Manning, C.D. What Does BERT Look at? An Analysis of BERT's Attention. In Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@ACL 2019, Florence, Italy, 1 August 2019; Association for Computational Linguistics: Cedarville, OH, USA, 2019; pp. 276–286.

15. Kovaleva, O.; Romanov, A.; Rogers, A.; Rumshisky, A. Revealing the Dark Secrets of BERT. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Cedarville, OH, USA, 2019; pp. 4364–4373.
16. Vig, J. A Multiscale Visualization of Attention in the Transformer Model. In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Cedarville, OH, USA, 2019; Volume 3: System Demonstrations, pp. 37–42.
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R., Eds.; NeurIPS Foundation, Inc.: San Diego, CA, USA, 2017; pp. 5998–6008.
18. Arici, N.; Gerevini, A.E.; Putelli, L.; Serina, I.; Sigalini, L. A BERT-Based Scoring System for Workplace Safety Courses in Italian. In Proceedings of the AIxIA 2022—Advances in Artificial Intelligence—XXIst International Conference of the Italian Association for Artificial Intelligence, AIxIA 2022, Udine, Italy, 28 November–2 December 2022; Dovier, A., Montanari, A., Orlandini, A., Eds.; Volume 13796, Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2022; pp. 457–471.
19. Kassner, N.; Dufter, P.; Schütze, H. Multilingual LAMA: Investigating Knowledge in Multilingual Pretrained Language Models. *arXiv* **2021**, arXiv:2102.00894.
20. Dai, D.; Dong, L.; Hao, Y.; Sui, Z.; Chang, B.; Wei, F. Knowledge Neurons in Pretrained Transformers. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, 22–27 May 2022; Muresan, S., Nakov, P., Villavicencio, A., Eds.; Association for Computational Linguistics: Cedarville, OH, USA, 2022; pp. 8493–8502. [CrossRef]
21. Anelli, V.W.; Biancofiore, G.M.; De Bellis, A.; Di Noia, T.; Di Sciascio, E. Interpretability of BERT Latent Space through Knowledge Graphs. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta GA USA, 17–21 October 2022; CIKM '22; pp. 3806–3810. [CrossRef]
22. Putelli, L.; Gerevini, A.E.; Lavelli, A.; Mehmood, T.; Serina, I. On the Behaviour of BERT's Attention for the Classification of Medical Reports. In Proceedings of the 3rd Italian Workshop on Explainable Artificial Intelligence Co-Located with 21th International Conference of the Italian Association for Artificial Intelligence (AIxIA 2022), Udine, Italy, 28 November–3 December 2022; Musto, C., Guidotti, R., Monreale, A., Semeraro, G., Eds.; CEUR Workshop Proceedings: Aachen, Germany; Volume 3277, pp. 16–30.
23. Putelli, L.; Gerevini, A.E.; Lavelli, A.; Olivato, M.; Serina, I. Deep Learning for Classification of Radiology Reports with a Hierarchical Schema. In Proceedings of the Knowledge-Based and Intelligent Information & Engineering Systems: 24th International Conference KES-2020, Virtual Event, 16–18 September 2020; Cristani, M., Toro, C., Zanni-Merk, C., Howlett, R.J., Jain, L.C., Eds.; Procedia Computer Science; Elsevier: Amsterdam, The Netherlands, 2020; Volume 176, pp. 349–359.
24. Putelli, L.; Gerevini, A.E.; Lavelli, A.; Maroldi, R.; Serina, I. Attention-Based Explanation in a Deep Learning Model For Classifying Radiology Reports. In Proceedings of the Artificial Intelligence in Medicine—19th International Conference on Artificial Intelligence in Medicine, AIME 2021, Virtual Event, 15–18 June 2021; Tucker, A., Abreu, P.H., Cardoso, J.S., Rodrigues, P.P., Riaño, D., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12721, pp. 367–372.
25. Comaniciu, D.; Meer, P. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 603–619. [CrossRef]
26. Ghassabeh, Y.A. On the convergence of the mean shift algorithm in the one-dimensional space. *Pattern Recognit. Lett.* **2013**, *34*, 1423–1427. [CrossRef]
27. Princeton University. Princeton University, About Wordnet. 2010. Available online: https://wordnet.princeton.edu/ (accessed on 26 June 2023).