Probing Pretrained Language Models for Semantic Attributes and their Values

Meriem Beloucif and Chris Biemann
Language Technology Group, Dept. of Informatics, MIN Faculty, Universität Hamburg
{beloucif, biemann}@informatik.uni-hamburg.de

Abstract

Pretrained Language Models (PTLMs) yield state-of-the-art performance on many Natural Language Processing tasks, including syntax, semantics and commonsense reasoning. In this paper, we focus on identifying to what extent do PTLMs capture semantic attributes and their values, e.g. the relation between rich and high net worth. We use PTLMs to predict masked tokens using patterns and lists of items from Wikidata in order to verify how likely PTLMs encode semantic attributes along with their values. Such inferences based on semantics are intuitive for us humans as part of our language understanding. Since PTLMs are trained on large amount of Wikipedia data, we would assume that they can generate similar predictions. However, our findings reveal that PTLMs perform still much worse than humans on this task. We show an analysis which explains how to exploit our methodology to integrate better context and semantics into PTLMs using knowledge bases.

1 Introduction

Given the ability of pretrained language models (PTLMs), such as BERT (Devlin et al., 2019), to create useful text representations, they have become the standard choice when building NLP applications (Peters et al., 2018a; Devlin et al., 2019; Radford and Narasimhan, 2018). However, there has recently been a rising amount of research that uses probes to understand the level of linguistics PTLMs encode. Different probing experiments have been proposed to study the drawbacks of PTLMs in areas such as the biomedical domain (Jin et al., 2019), syntax (Hewitt and Manning, 2019), semantic and syntactic sentence structures (Tenney et al., 2019; Peters et al., 2018b), prenominal anaphora (Sorodoc et al., 2020), linguistics (Belinkov et al., 2017; Clark et al., 2020; Tenney et al., 2019) and commonsense knowledge (Petroni et al., 2019; Davison et al., 2019; Talmor et al., 2020).

In this paper, we expand on this line of research by probing PTLMs to investigate if they cover semantic attributes and their values. The closest work to ours has been proposed by Ribeiro et al. (2020), where they investigate if PTLMs capture checklists such as: red, green and yellow. In contrast to them, we focus on finding out whether pretrained language models capture the correlation between semantic attributes and their values.

An example of semantic attributes and their values is the relation that exists between old, age and date of birth, or the relation between rich, wealth and net worth. Looking up rich on FrameNet (Fillmore, 1982; Baker et al., 1998) would not result in a frame by itself, but would evoke the semantic frame wealthiness (Figure 1). In WordNet (Fellbaum, 1998) these associations are called an attribute-value relation, where the attribute is a noun for which adjectives express values. For instance, the
nound *weight* is an attribute, for which the adjectives *light* and *heavy* express values. Another example of these kind of associations is *rich*, which could be associated with wealthiness and net worth.

Knowledge bases (KBs), such as Wikidata (Vrandečić and Krötzsch, 2014), constitute a valuable resource for collecting attributes and their values. In general, KBs have been shown to help improve multiple NLP application as they contain structured information (Annervaz et al., 2018; Nakashole and Mitchell, 2015; Rahman and Ng, 2011; Ratinov and Roth, 2009). As matter of fact, it is fairly simple to answer factoid questions such as “How old is Joe Biden?” using Wikidata, by simply looking up his date of birth on Wikidata. An important step to make this happen is to match *old* and *date of birth* to each other. Similarly, to check “how rich is Jeff Bezos?”, we only need to extract his *net worth* from Wikidata, and identify that *rich* and *net worth* are related to each other. However, a simple task like this one requires us to have tools that can identify the relation between the attribute *net worth* and its qualitative value *rich*. Despite this task being straightforward to perform manually, it is not yet solved automatically due to linguistic challenges. In the previous example, rich and high net worth are not synonyms, and do not share the same part-of-speech tag since the former is an adjective and the latter is a noun. However, they tend to appear together in text: net worth occurs 34 times in the Wikipedia page of Jeff Bezos, while rich/er/est appears 35 times.

In this paper, we conduct a case study to identify to what extent do language models that are pretrained on massive amounts of data learn these attribute-value correlations. In other words, we ask the question: “do PTLMs understand that a given value is associated with a specific attribute?”. We assume that PTLMs should learn these relations given that they are trained on everyday’s online data, which contains a wide amount of attribute-value pairs. Our goal is to see our work aspire future works by: a) enabling more efficient comparative QA and factoid QA such as “Q: Who’s older Obama or Trump”, b) improving PTLMs’ abilities to faithfully capture attribute-value relations, and c) a step towards finding resources to fine-tune PTLMs towards semantic objectives.

### 2 Methodology

In this paper, we aim at showing to what degree do PTLMs contain abstract semantically-based relations like attributes and their values. For instance, when asking "which is denser, New York or Hong Kong?", humans automatically link density in this specific case to urban cities and population. In order to show what PTLMs are able to understand when it comes to attribute-value pairs, we start by defining and collecting data from the predefined set of patterns shown in Table 1. Next, we randomly select a sample from the collected data to probe three different pretrained language models: BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and XLNet (Yang et al., 2019).

#### Patterns

The Wikipedia page of Jeff Bezos contains the statement: *Bezos was named the richest man in modern history after his net worth increased to X*. This type of sentences appears frequently on the web, which is why we construct a dataset based on similar patterns. To the best of our knowledge, there is no available dataset that contains statements of attribute-value pairs. In order to come up with patterns that are likely to happen, we decided on two types of patterns: 1) a single object statement, and 2) a comparative statement between two objects. Table 1 shows a sample of patterns used for data collection from Wikidata. Our initial patterns include: 1) basic linguistic phenomena such as hypernyms and hyponyms (A, B and C); 2) single object statements such as *A is [VALUE], hence it has more [MASK] and A is [MASK], hence it has more [ATTRIBUTE]*; and 3) comparative statements containing two objects, to test if having two objects would increase the likelihood of predicting the correct entity or value. The integrality of our patterns include 15 different ones: 2 hypernyms and hyponyms, the 5 comparatives from Table 1 with the masked attributes, the 5 same ones with

<table>
<thead>
<tr>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. [MASK] and B.</td>
</tr>
<tr>
<td>What’s [VALUE] or [MASK], A or B?</td>
</tr>
<tr>
<td>What’s [VALUE] and [MASK], A or B?</td>
</tr>
<tr>
<td>A is [MASK], thus they have a [VALUE]</td>
</tr>
<tr>
<td>A is [VALUE], thus it has more [MASK] per m².</td>
</tr>
</tbody>
</table>

To know which is [VALUE], A or B, you need [MASK].

What’s [VALUE], thus has a higher [MASK], A or B? We need the [MASK] to know what’s [VALUE], A or B. We need the [VALUE] to know who is [MASK] A or B. A is famous for [VALUE], thus it has more [MASK].

Table 1: The list of all patterns created for collecting probing data from Wikidata (Vrandečić and Krötzsch, 2014).
### Table 2: Statistics about the data extracted from Wikidata.

<table>
<thead>
<tr>
<th>Domain</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>People (20%)</td>
<td>Gates is richer, thus has a higher net worth.</td>
</tr>
<tr>
<td>Chemistry (23%)</td>
<td>We need the mass to know if gold is heavier.</td>
</tr>
<tr>
<td>Geography (21%)</td>
<td>Lima is denser, and thus has more people in a given area.</td>
</tr>
<tr>
<td>Politics (26%)</td>
<td>Obama is younger than Trump, hence he was born after.</td>
</tr>
<tr>
<td>Art (10%)</td>
<td>Paris is famous for art, thus it has more museums.</td>
</tr>
</tbody>
</table>

Table 2 contains our data distribution per subject. Our goal is to determine what is the prediction that PTLMs yield for each case, and whether the prediction changes if we blindly change the object. Our final dataset\(^1\) contains 18,327 English statements on 15 different target masks.

---

**Data Collection**  
After choosing the patterns, we construct the dataset based on the previously described patterns by extracting objects from Wikidata. First, we vary object-entity-value and object1-object2-entity-value to include statements such as gold-mass-heavy and gold-silver-mass-heavy or Obama-date_of_birth-age and Obama-Trump-date_of_birth-age. We constructed different sentences that include several patterns on how we could possibly invoke these object-entity-value triplet, using single object statements and comparative statements. Next, we extract different objects from Wikidata (Vrandečić and Krötzsch, 2014). Wikidata is a collaborative knowledge base, containing triplets (entity_id, property_id, value_id) that define a type of relation holding between an entity and a value. For instance, the Wikidata page of Barack Obama contains multiple triplets such as instance of human where instance which has P31 is a property_id and human is a value with a value_id equal to Q5. For our task, and in one specific case among many others, we iterate over all US presidents and create all possible combinations for the (date_of_birth-age) attribute-value pair. Our dataset contains some statements that are not valid from a commonsense perspective, but this does not affect this specific task, since we are not judging if a statement is true, but only if an attribute-value pair happens simultaneously. Table 2 contains our data distribution per subject. Our goal is to determine what is the prediction that PTLMs yield for each case, and whether the prediction changes if we blindly change the object. Our final dataset\(^1\) contains 18,327 English statements on 15 different target masks.

**Probing PTLMs**  
We probe three different PTLMs, including BERT, RoBERTa and XLNET on a randomly selected test set. We use the HuggingFace code\(^2\) (Wolf et al., 2020) to probe three PTLMs on attributes and their values by using the predictions of PTLMs’ Masked Language Modelling (MLM) head. We randomly extract 10 sentences from object-entity-value set of our data to make sure that our final test set (120 samples) is uniformly distributed.

### 3 Results

We evaluate the predictions from the PTLMs manually using heuristics based on the relatedness between the predicted attribute-value pairs\(^3\). If an entity-value pair are semantically related, we give a score of 1, else we give a score of 0. For instance, in Table 3, examples 1, 2, 3, 6 and 7 receive a 1, while 4 and 5 receive a 0. The results of our manual evaluation are summarized in Figure 3. We asked one person to predict the masked word for

---

\(^1\)[https://github.com/uhh-lt/semantic-probing]

\(^2\)[https://huggingface.co/bert-base-uncased]

\(^3\)The results reported here have been last checked on the 17th of July, 2021.
**Masked Sentence**

1. What's heavier or [MASK], gold or silver? lighter, softer, thinner, heavier, light
2. What's heavier and [MASK], gold or silver? lighter, softer, stronger, heavier, brighter
3. What's heavier, and thus has a higher [MASK], gold or silver? value, price, worth, weight, content
4. To know which is heavier, gold or silver, you need [MASK].
5. We need the [MASK], to determine which is heavier, gold or silver.
6. [MASK] such as gold, silver and copper.
7. Gold, [MASK] and iron.

**Prediction**

1. lighter, softer, thinner, heavier, light
2. lighter, softer, stronger, heavier, brighter
3. value, price, worth, weight, content
4. formula, alloy, ratio, weight, elements
5. metals, elements, metal, commodities, materials
6. silver, copper, tin, nickel, platinum

<table>
<thead>
<tr>
<th>City</th>
<th>BERT</th>
<th>RoBERTa</th>
<th>XLNET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singapore</td>
<td>vegetation, density, moisture</td>
<td>density, <strong>people</strong>, moisture</td>
<td>it, thus, (</td>
</tr>
<tr>
<td>Sofia</td>
<td>density, layer, leaves</td>
<td>volume, energy, density</td>
<td>it, to, d</td>
</tr>
<tr>
<td>New Delhi</td>
<td>vegetation, forests, soils</td>
<td>density, space, <strong>people</strong></td>
<td>New, it, to</td>
</tr>
<tr>
<td>Buenos Aires</td>
<td>vegetation, forests, density</td>
<td><strong>people</strong>, density, <strong>inhabitant</strong></td>
<td>in, off, &lt;eop&gt;</td>
</tr>
<tr>
<td>La Paz</td>
<td>density, vegetation, layers</td>
<td>space, density, tree</td>
<td>La, more, than</td>
</tr>
<tr>
<td>Panama City</td>
<td>vegetation, forests, trees</td>
<td><strong>people, inhabitant, residents</strong></td>
<td>Panama, Y, P</td>
</tr>
<tr>
<td>Addis Ababa</td>
<td>vegetation, leaves, density</td>
<td>density, space, <strong>people</strong></td>
<td>and, more, ness</td>
</tr>
<tr>
<td>Mbabane</td>
<td>vegetation, soils, forests</td>
<td>energy, density, carbon</td>
<td>.., M, S</td>
</tr>
<tr>
<td>Asunción</td>
<td>vegetation, density, <strong>inhabitants</strong></td>
<td><strong>inhabitants</strong>, density, trees</td>
<td>it, as, in</td>
</tr>
<tr>
<td>Freetown</td>
<td><strong>inhabitants</strong>, forests, density</td>
<td><strong>people</strong>, density, trees</td>
<td>thus, it, density</td>
</tr>
</tbody>
</table>

Table 3: BERT is good at hypernyms and hyponyms, like in *gold, silver and iron*, but it fails to accurately predict the right attribute-value pair.

Table 4: We probe different PTLMs on the sentence: [CITY] is denser, thus it has more [MASK].

Figure 3: Comparing BERT and RoBERTa to human predictions. The human prediction was collected from only one person, the accuracy was high since this kind of task is simple. We excluded XLNet from this evaluation, since its predictions were completely random.

We consistently observed throughout multiple examples that PTLMs are vulnerable to slight changes such as the name of the city or the name of the element that we are targeting. Figure 2 shows ten randomly selected cities from the input sentence: [City] is denser, thus it has more [MASK] in a given area, and the five most likely predictions from BERT. The examples show some random predictions (in red), such as moisture, layer and vowels. What we found interesting in our case study, is that changing only the city, changes the distribution of the predicted words, but in what seems to be a random fashion. We note from the figure that only two cities, namely Freetown and Asunción, trigger predictions related to inhabitants and population. While we initially conjectured that Freetown has also 'town' in its name, we were surprised that this did not apply for Panama City.

We argue that even though BERT is trained on Wikipedia content, the huge amount of data and the biased distribution of other linguistic phenomena makes it difficult to capture attribute-value rela-
tions. For that reason, it could be possible to train a robust semantically-aware PTLM by fine-tuning the current PTLMs to FrameNet content, and a first step would be to integrate Wikidata entities and their values according to the semantic frames of each entity and every word that is evoked by a value. One advantage of PTLMs is their capability to perform well on specific tasks and domains that were not part of their training regime via fine-tuning, i.e. the retraining of a pretrained model with domain-specific examples. We argue that for entities and their values, resources such as FrameNet and WordNet, while paired with massive resources such as Wikidata could be used to fine-tune PTLMs towards more semantically-based objectives, as a complementary work to ERNIE (Zhang et al., 2019), which showed that fine-tuning PTLMs towards knowledge graphs helps enhancing language representation with external knowledge.

5 Conclusion

We demonstrated that PTLMs are unable to capture semantic similarity between different words that refer to the same concepts. While PTLMs have been shown to improve the quality of many tasks and are not easy to train, our probing experiments show that an improvement is necessary. All the examples we extracted from Wikidata show that by enabling PTLMs to capture more semantically-based information by fine-tuning towards more semantically-based objectives like the ones found in FrameNet. All our examples are extracted from Wikidata to show that, resources such as Wikidata are rich and could be used as a resource for fine-tuning BERT towards more high-level semantics.

6 Acknowledgements

This work was supported by the DFG through the project “ACQuA: Answering Comparative Questions with Arguments” (grants BI 1544/7-1 and HA 5851/2-1) as part of the priority program “RATIO: Robust Argumentation Machines” (SPP 1999).

References


Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In ICLR.


Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, pages 365–375, Beijing, China.


