Exploring Different Methods for Solving Analogies with Portuguese Word Embeddings

Tiago Sousa
ISEC, Polytechnic Institute of Coimbra, Portugal
a21220135@isec.pt

Hugo Gonçalo Oliveira
CISUC, Department of Informatics Engineering, University of Coimbra, Portugal
hroliv@dei.uc.pt

Ana Alves
CISUC, University of Coimbra, Portugal
ISEC, Polytechnic Institute of Coimbra, Portugal
ana@dei.uc.pt

Abstract

A common way of assessing static word embeddings is to use them for solving analogies of the kind "*what is to king as man is to woman?*". For this purpose, the vector offset method ($king - man + woman = queen$), also known as 3CosAdd, has been effectively used for solving analogies and assessing different models of word embeddings in different languages. However, some researchers pointed out that this method is not the most effective for this purpose. Following this, we tested alternative analogy solving methods (3CosMul, 3CosAvg, LRCos) in Portuguese word embeddings and confirmed the previous statement. Specifically, those methods are used to answer the Portuguese version of the Google Analogy Test, dubbed LX-4WAnalogies, which covers syntactic and semantic analogies of different kinds. We discuss the accuracy of different methods applied to different models of embeddings and take some conclusions. Indeed, all methods outperform 3CosAdd, and the best performance is consistently achieved with LRCos, in GloVe.

2012 ACM Subject Classification Computing methodologies → Lexical semantics

Keywords and phrases analogies, word embeddings, semantic relations, syntactic relations, Portuguese

Digital Object Identifier 10.4230/OASIcs.SLATE.2020.9

1 Introduction

Computational representations of the words of a language and their meanings have followed two main approaches: symbolic methods like first-order logic and graphs, instantiated as lexical-semantic knowledge bases (LKBs), such as wordnets [7]; and distributional models, like word embeddings. The former organise words, sometimes also senses, often connected by relations, such as Hypernymy or Part-of, and may include additional lexicographic information (part-of-speech, gloss), while the latter follow the distributional hypothesis [10] and represent words as vectors of numeric features, according to the contexts they are found in large corpora. On distributional models, since 2013, the trend was to use efficient methods that learn word embeddings – dense numeric-vector representations of words, like word2vec [14] or GloVe [16]. Besides their utility for computing word similarity, such models have shown very interesting results for solving analogies of the kind "*what is to b as a* is to a?*" (e.g., what is to Portugal as Paris is to France?). So much that both previous tasks are extensively used for assessing word embeddings in different languages.

Popular analogy test sets cover syntactic and semantic relations of different types. Among them, the Google Analogy Test (GAT) [14], which contains syntactic and semantic analogies, was popularised by Mikolov et al. [14] and used in several experiments for assessing word embeddings. Even though there are other similar tests, to our knowledge, only GAT was translated to Portuguese, and rebaptised as LX-4WAnalogies [17].

The most popular method for solving analogy is the vector offset method, used by Mikolov et al. for assessing word2vec [14, 15], and for assessing Portuguese word embeddings [18, 11] in the LX-4WAnalogies. Also known as 3CosAdd, this method solves analogies with a simple operation on the word vectors of the given words ($king - man + woman = queen$).

In this paper, we also use the LX-4WAnalogies test but, more than comparing the performance of different word embeddings, we aim at testing alternative methods for solving analogies, namely 3CosMul [12], 3CosAvg and LRCos [5], in Portuguese word embeddings. An important difference of the last two methods is that they do not solve the analogy from a single pair of words. Instead, they consider a larger set with pairs of analogously-related words. Besides assessing the quality of word embeddings, the analogy solving task can be useful for many different tasks, from the automatic discovery of new word relations for populating knowledge bases [8], to text transformation [1].

The previous methods were tested in different models of word embeddings available for Portuguese [11], including GloVe [16], word2vec [14], fastText [2], and an alternative model, Numberbatch, part of the ConceptNet open data project [19]. Briefly, results achieved confirm what happens for English: all alternative methods tested outperform 3CosAdd and the best accuracy is consistently achieved with LRCos. Even for solving analogies from a single pair, 3CosMul achieves better accuracies than 3CosAdd. We may as well take some conclusions on the quality of the tested word embeddings for analogy solving, which may be broadly interpreted as how well they capture linguistic regularities. Again, as it happens for English, GloVe is often the model with best overall results. Numberbatch, not used in previous work, suffers from a lower word coverage. Yet, when uncovered pairs of words are ignored, it achieves the best accuracy in the semantic analogies.

The paper starts with a brief overview on the most common tasks for assessing word embeddings, with a focus analogy and available datasets for this purpose. After that, we describe the experimentation setup, covering the models used, the methods applied, the relations in the LX-4WAnalogies test, and the adopted data format, for compatibility with all methods and with the used framework. Before concluding, the results achieved are presented and discussed, with global figures as well as results for each type of analogy.

2 Background Knowledge

Models of static word embeddings are commonly assessed in two different tasks: word similarity and analogy solving. The goal of the former is to assign a suitable value for the semantic similarity between pairs of words (e.g., between 0 a 1). In static word embeddings, this value is often given by the cosine of the vectors that represent each word of the pair (e.g., $sim(a, b) = cos(\vec{a}, \vec{b})$). The higher the cosine, the higher the similarity (e.g., the similarity between *dog* and *cat* should be higher than the similarity between *dog* and *car*).

Analogy solving, on the other hand, aims to check how well linguistic regularities are kept in the embeddings. Its goal is to answer questions of the kind "*what is to b as* a^* *is to a*?" (e.g., what is to Portugal as Paris is to France?, for which the answer would be *Lisbon*). The most common method for this is the vector offset, also known as 3CosAdd ($b^* = b - a + a^*$), previously used for computing both syntactic and semantic analogies with word embeddings for different languages, including English [14, 15] and Portuguese [18, 11].

For English, popular datasets include the Microsoft Research Syntactic Analogies (MSR) [15] and the Google Analogy Test (GAT) [14], both used by Mikolov et al. when assessing word2vec. MSR contains 8,000 questions, covering eight different types of syntactic analogy. GAT covers nine types of syntactic analogy (e.g., adjective to adverb, opposite, comparative, verb tenses), roughly the same as MSR, plus five semantic (e.g., capital-country, currency, male-female) categories, with 20-70 unique example pairs per category, which may be combined in 8,869 semantic and 10,675 syntactic questions (see Section 3.3).

GAT was translated to Portuguese [17], rebaptised as LX-4WAnalogies, made publicly available[1], and originally used for assessing the LX-DSemVectors [18], based on word2vec. It was further used for assessing other word embeddings, such as the NILC embeddings [11], which cover different models (e.g., word2vec, GloVe, fastText) with vectors of different dimensions (50, 100, 300, 600, 1000).

Due to its simplicity, the previous experiments on analogy solving relied on the 3CosAdd method. However, other researchers proposed alternative methods that lead to significant improvements. Such methods include 3CosMul [12], 3CosAvg and LRCos [5]. The main difference of the last two is that they use more than a single pair $a : a^*$ for solving the analogy, and exploit a larger set of analogously-related pairs, which could be those in the same dataset (see Section 3.2). Both 3CosAvg and LRCos were presented along the creation of the Bigger Analogy Test Set (BATS) [8], which includes part of GAT, but is larger, covers more types of analogy, and is balanced among all covered types. Moreover, BATS adopted a different representation format, which suits the exploitation of the full dataset better and enables questions to have more than one possible answer (see Section 3.4).

3 Experimentation Setup

Our experimentation consisted of testing a set of methods for solving the analogies in the LX-4WAnalogies [17] dataset, using different pre-trained word embeddings, available for Portuguese. For this purpose, we used Vecto[2], a framework for testing word embeddings that includes the implementation of different methods and produces well-organised logs of the results. In order to test some of the methods in Vecto, we had to change the data format of the LX-4WAnalogies to a format closer to the BATS dataset. This section describes the tested embeddings, the tested methods, and the adopted data format for the dataset.

3.1 Word Embeddings

We tried to cover static word embeddings learned with different algorithms, for which pre-trained models are available in Portuguese. More precisely, we used the following models:

- GloVe, word2vec (CBOW and SKIP-gram) and fastText (CBOW and SKIP-gram), with 300-sized vector, available from the NILC repository of Portuguese word embeddings [11];
- Vectors of Portuguese words in the Numberbatch embeddings [19], version 17.02.

Even though Numberbatch is available in a similar vector format, also with size 300, it is significantly different from the others, because it was learned from several sources. Such sources include raw text (i.e., an ensemble of Google News word2vec, Common Crawl GloVe, Open Subtitles fastText) combined with the ConceptNet semantic network with retrofitting.

[1] https://github.com/nlx-group/LX-DSemVectors/tree/master/testsets
[2] https://github.com/vecto-ai

Selecting only the Portuguese words in Numberbatch is straightforward because all entries are identified by a URI that contains the language prefix (e.g., `/c/pt/banana` for the word *banana*). This made it possible to store Numberbatch in a 107MB text file, while all the other models are substantially larger, with sizes around 2.5GB. Moreover, we considered using the latest version of Numberbatch, 19.08. Yet, after doing the same process for extracting the Portuguese words, the file is about five times larger and some preliminary experiments showed that the new version contains many multiword expressions, which are not in the analogy tests. Therefore, we decided to test only the smaller old version.

3.2 Methods

Mikolov et al. [15] showed that word2vec vectors retain semantic and syntactic information and proposed the vector offset method for answering analogy questions such as "*what is to Portugal as Paris is to France?*". This method, also known as 3CosAdd, formulates the analogy as *a is to* a^* *as b is to* b^*, where b^* has to be inferred from a, a^* and b. More precisely, b^* will be the word with the most similar vector to the result of $a^* - a + b$ (see equation 1). Having in mind that, in a vector space, the similarity between two vectors is given by their cosine, the most similar vector will maximise its cosine with the resulting vector.

$$b^* = \underset{w \in V}{\operatorname{argmax}} \cos(w, a^* - a + b) \tag{1}$$

3CosMul (see equation 2) emerged as an alternative to the arithmetic operation of 3CosAdd, the sum. Using multiplication, Levy and Goldberg [12] refer that, this way, a better balance between the various aspects of similarity is achieved. This was confirmed when 3CosMul indeed achieved better performance in the MSR and GAT tests.

$$b^* = \underset{w \in V}{\operatorname{argmax}} \frac{\cos(b, w) \times \cos(w, a^*)}{\cos(w, a)} \tag{2}$$

3CosAvg and LRCos, both proposed by Drozd et al. [5], try to make the most out of the full test set, instead of a single pair of related words $(a : a^*)$. 3CosAvg computes the average offset between words in position a and words in position a^*, in a set of word pairs analogously related (see equation 3). The answer, b^*, must maximise the cosine with the vector resulting from summing the average offset to b.

LRCos (see equation 4) considers the probability that a word w is of the same class as other words in position a^* as well as the similarity between w and b, measured with the cosine. A classifier, in this case, logistic regression, is used for computing the likelihood of a word belonging to the class of words a^*. Since all methods were applied with the default parameters of the Vecto implementation, the classifier is trained with all entries of the dataset, except the target one $(b : b^*)$, as positive examples, and the same number of negative pairs, each generated from two arguments in different entries, i.e., a is from an entry and a^* is from another, meaning that they should not be related, at least not in as the positive examples.

$$b^* = \underset{w \in V}{\operatorname{argmax}} \cos(w, b + avg\_offset) \tag{3}$$

$$b^* = \underset{w \in V}{\operatorname{argmax}} P(w \in target\_class) \times cos(w, b) \tag{4}$$

Besides the previous methods, we follow Linzen's [13] suggestion and also test to what extent simply using the most similar words is enough for solving analogies and how much different it makes to computing the previous methods. Though not exactly an analogy-solving method, due to its simplicity, the SimilarToB (see equation 5) can be seen as a baseline for this purpose. This method simply retrieves words similar to b, based on the vector cosine, thus, achieving the best accuracy with it means that more complex analogy solving methods are not doing any good.

$$b^* = \underset{w \in V}{\operatorname{argmax}} \cos(b, w) \tag{5}$$

We should add that, as it happens in other implementations of 3CosAdd for assessing word embeddings, in Vecto, when one of the words a or a^* in a pair are not in the model of embeddings, this pair is discarded and it is not considered for computing the average accuracy. This means that the model coverage will not be considered in this evaluation. Though, impact should be minimal in all but Numberbatch, because all other models were learned from the same corpus.

3.3 Relations

The methods previously described were applied to the selected word embeddings for answering the analogy questions in LX-4WAnalogies. We should note that there are two versions of LX-4WAnalogies, one in Brazilian (`LX-4WAnalogiesBr`) and another in European Portuguese (`LX-4WAnalogies`), with minor differences. We used the latter. As in GAT, the questions in LX-4WAnalogies cover 14 types of relation, including five semantic and nine syntactic. Table 1 shows all relation types with two examples for each, in English (from GAT) and in Portuguese, also including the number of questions in the Portuguese version.

A quick look at the data shows some translation issues, not always easy to deal, due to the more complex morphology of Portuguese. Although we did take care of these issues, they can have a negative impact on the accuracy of analogy solving methods. Thus, we point some of them out, and will consider fixing them in future work. For instance, in Portuguese, some of the comparative and superlative analogies are translated equally (e.g., both worse and worst to *pior*). But, in Portuguese, there are two superlative degrees: the relative uses the same word as the comparative, through differently (e.g., *o pior*); the absolute uses a single word (e.g. *péssimo*). In LX-4WAnalogies, both types seem to be used interchangeably. Another issue occurs with the interpretation of the analogy class. For instance, in Portuguese, the verb plurals become a relation between the infinitive to the third person of the singular in the present tense. Finally, in Portuguese, names of nationalities (e.g., as *albanês*) do not start with a capitalised letter, which could be a problem in a case-sensitive scenario.

3.4 Data Format

The questions of GAT are represented in a single text file where each line contains four words: the three necessary for formulating the question, followed by the correct answer, i.e., a a^* b b^*. The type of analogy is identified by lines starting with `:`, indicating that all the following lines have questions of that type. This format, also adopted by LX-4WAnalogies, is illustrated in Figure 1 with sample lines of both datasets.

However, this format was not adopted in the experiments carried out in the scope of this work, because it did not suit some of the methods, namely 3CosAvg and LRCos. Instead, we adopted a BATS-like format, supported by Vecto. This means that there is a file for each

Table 1 Relations covered by GAT, original examples and Portuguese translations in LX-4WAnalogies.

Semantic	**GAT**	**LX-4WAnalogies**
capital-common-countries	Athens, Greece	Atenas, Grécia
(506 questions)	Baghdad, Iraq	Bagdade, Iraque
capital-world	Abuja, Nigeria	Abuja, Nigéria
(4,524)	Accra, Ghana	Acra, Gana
city-in-state	Chicago, Illinois	Chicago, Ilinóis
(2,467)	Houston, Texas	Houston, Texas
currency	Algeria, dinar	Argélia, dinar
(866)	Angola, kwanza	Angola, kwanza
family	boy, girl	rapaz, rapariga
(462)	brother, sister	irmão, irmã
Syntactic	**GAT**	**LX-4WAnalogies**
gram1-adjective-to-adverb	amazing, amazingly	fantástico, fantasticamente
(930)	apparent, apparently	aparente, aparentemente
gram2-opposite	acceptable, unacceptable	aceitável, inaceitável
(756)	aware, unaware	consciente, inconsciente
gram3-comparative	bad, worse	mau, pior
(30)	big, bigger	grande, maior
gram4-superlative	bad, worst	mau, pior
(600)	big, biggest	grande, maior
gram5-present-participle	code, coding	programar, programando
(1,056)	dance, dancing	dançar, dançando
gram6-nationality-adjective	Albania, Albanian	Albânia, Albanês
(1,599)	Argentina, Argentinean	Argentina, Argentino
gram7-past-tense	dancing, danced	dançando, dançou
(1,560)	decreasing, decreased	diminuindo, diminuiu
gram8-plural	banana, bananas	banana, bananas
(1,332)	bird, birds	pássaro, pássaros
gram9-plural-verbs	decrease, decreases	diminuir, diminuem
(870)	describe, describes	descrever, descrevem

kind of analogy, where each row has a single pair of two related words: one to be used in the formulation of a question (b), and another to be used as the target answer (b^*). Although, in BATS, the latter could include more than a single word (i.e., more than one possible answer), this does not happen in LX-4WAnalogies. With this format, GAT-like questions can be formulated by combining two rows of the same file. This is also how Vecto applies the 3CosAdd and 3CosMul methods. Figure 2 illustrates how the LX-4WAnalogies lines in Figure 1 become in the adopted format, where a box representing a single file with file name on top.

This conversion resulted in some differences in the new LX-4WAnalogies. First, this dataset contains a small amount of duplicate rows, some of them originating from the English to Portuguese translation. For instance, GAT has entries such as:

- `father mother grandfather grandmother`
- `father mother grandpa grandma`
- `father mother dad mom`

With corresponding lines in LX-4WAnalogies:

- `pai mãe avô avó`
- `pai mãe avô avó`
- `pai mãe pai mãe`

```
:  capital-common-countries
Athens Greece Baghdad Iraq
Athens Greece Bangkok Thailand
Athens Greece Beijing China
...
Baghdad Iraq Bangkok Thailand
Baghdad Iraq Beijing China
Baghdad Iraq Berlin Germany
...
:  gram4-superlative
bad worst big biggest
bad worst bright brightest
bad worst cold coldest
...
:  gram8-plural
banana bananas bird birds
banana bananas bottle bottles
banana bananas building buildings
...
```

```
:  capital-common-countries
Atenas Grécia Bagdade Iraque
Atenas Grécia Banguecoque Tailândia
Atenas Grécia Pequim China
...
Bagdade Iraque Banguecoque Tailândia
Bagdade Iraque Pequim China
Bagdade Iraque Berlim Alemanha
...
:  gram4-superlative
mau pior grande maior
mau pior brilhante brilhantíssimo
mau pior escuro escuríssimo
...
:  gram8-plural
banana bananas pássaro pássaros
banana bananas garrafa garrafas
banana bananas edifício edifícios
...
```

Figure 1 Sample lines of format of GAT and corresponding lines in LX-4WAnalogies.

```
capital-common-countries.txt
Atenas Grécia
Bagdade Iraque
Banguecoque Tailândia
Pequim China
Berlim Alemanha
...
```

```
gram3-comparative.txt
mau pior
grande maior
brilhante brilhantíssimo
escuro escuríssimo
...
```

```
gram8-plural.txt
banana bananas
pássaro pássaros
garrafa garrafas
edifício edifícios
...
```

Figure 2 Sample lines of LX-4WAnalogies in a Vecto-compatible data format.

In the adopted format, this would also result in duplicate pairs, and thus duplicate lines, which were removed. After this, the number of questions that can be formulated for 3CosAdd and 3CosMul decreases for three types (see Table 2).

Table 2 Analogy types with less formulated questions in the conversion of LX-4WAnalogies.

Type	family	gram3-comparative	gram7-past-tense
#Questions (original)	462	30	1,560
#Questions (new)	380	20	1,482

Moreover, we noticed that, in some analogy types, LX-4WAnalogies does not contain all possible combinations of two related pairs. Since, with the adopted format, all combinations are tested, the number of analogies of five types increased in the conversion of the test (see Figure 3). The increase is especially high for the capital-world relations.

Table 3 Analogy types with more formulated questions in the conversion of LX-4WAnalogies.

Type	cap-world	city-in-state	currency	gr2-opposite	gr6-nat-adj
#Questions (original)	4,524	2,467	866	756	1,599
#Questions (new)	13,340	4,032	870	812	1,640

Our conversion of LX-4WAnalogies to the adopted data format was baptised as TAP, acronym for "Teste de Analogias em Português" (Test of Portuguese Analogies), and is available online[3], for anyone willing to use it.

[3] https://github.com/NLP-CISUC/PT-LexicalSemantics/tree/master/Analogies

4 Results

With Vecto, the LX-4WAnalogies test was solved with all combinations of selected methods (Section 3.2) and word embeddings (Section 3.1). Table 4 shows the macro and micro average accuracy of each combination, also splitted by the semantic and syntactic analogies. Macro averages consider that the accuracy for each relation is worth the same, no matter the number of questions of their type, and thus gives a better perspective on how balanced each combination is for different relations. On the other hand, for micro-averages, every single question is worth the same, meaning that each relation is weighted according to the number of questions its type.

Table 4 Average accuracies achieved with each method and each model of word embeddings.

Model	Method	Macro-Accuracy			Micro-Accuracy		
		Sem	Synt	Avg	Sem	Synt	Avg
GloVe	SimilarToB	6.61%	10.05%	8.82%	4.35%	7.06%	5.75%
	3CosAdd	26.32%	29.79%	28.55%	17.97%	30.67%	21.95%
	3CosMul	29.04%	33.27%	31.76%	21.75%	33.75%	25.51%
	3CosAvg	34.51%	43.01%	39.98%	27.27%	42.01%	34.87%
	LRCos	**51.87%**	**48.34%**	**49.60%**	**56.13%**	**48.33%**	**52.11%**
word2vec CBOW	SimilarToB	2.00%	1.19%	1.48%	0.79%	1.12%	0.96%
	3CosAdd	8.26%	18.07%	14.57%	2.37%	14.60%	6.20%
	3CosMul	9.72%	21.25%	17.14%	2.73%	17.40%	7.33%
	3CosAvg	17.39%	27.01%	23.58%	10.67%	24.54%	17.82%
	LRCos	13.47%	30.46%	24.39%	8.30%	28.62%	18.77%
word2vec SKIP	SimilarToB	2.00%	2.18%	2.12%	0.79%	2.23%	1.53%
	3CosAdd	15.06%	21.33%	19.09%	7.26%	20.52%	11.42%
	3CosMul	16.37%	25.09%	21.98%	9.69%	24.32%	14.28%
	3CosAvg	23.16%	31.16%	28.30%	16.21%	29.74%	23.18%
	LRCos	30.54%	37.88%	35.26%	27.67%	37.92%	32.95%
fastText CBOW	SimilarToB	2.00%	0.40%	0.97%	0.79%	0.37%	0.57%
	3CosAdd	12.79%	28.05%	22.60%	4.85%	31.68%	13.23%
	3CosMul	15.11%	28.28%	23.57%	6.29%	32.42%	14.45%
	3CosAvg	15.51%	39.00%	30.61%	9.88%	39.41%	25.10%
	LRCos	34.30%	36.95%	36.00%	31.23%	37.17%	34.29%
fastText SKIP	SimilarToB	4.21%	4.31%	4.27%	2.37%	4.83%	3.64%
	3CosAdd	25.31%	35.75%	32.02%	15.35%	37.90%	22.39%
	3CosMul	28.62%	38.01%	34.65%	20.73%	39.62%	26.63%
	3CosAvg	29.52%	42.43%	37.82%	20.95%	43.49%	32.57%
	LRCos	50.00%	44.99%	46.78%	51.38%	46.47%	48.85%
Numberbatch	SimilarToB	14.17%	2.26%	6.51%	15.02%	2.97%	8.81%
	3CosAdd	21.45%	8.97%	13.43%	18.21%	10.22%	15.72%
	3CosMul	23.94%	16.15%	18.93%	21.29%	12.12%	18.43%
	3CosAvg	29.99%	13.09%	19.13%	27.27%	11.90%	19.35%
	LRCos	43.81%	23.14%	30.52%	42.69%	22.30%	32.18%

By comparing the accuracy of different methods, for any model of embeddings, it becomes clear that the best accuracy is always achieved by the methods that exploit more than a single pair of related words. This is especially true for LRCos and suggests that this is the best option for solving this kind of problem, at least when a dataset of analogously-related pairs is available. However, we recall that the figures for 3CosAdd and 3CosMul imply many

more questions, i.e., when using each entry pair as $b : b^*$ when each of the remaining entries is used as $a : a^*$. On the other hand, for 3CosAvg and LRCos, a single question is made for each entry $b : b^*$, with all the remaining entries used at the same type.

Still, when a single pair is available, 3CosMul showed to be a better choice than the popular 3CosAdd, which it consistently outperforms. One the other hand, the worst accuracy is always for the SimilarToB, which was expected. Improving the accuracy of SimilarToB shows that all analogy solving methods are indeed doing more than simply looking at the most similar words, also confirming that linguistic regularities in the embedding space go further than just similarity.

Overall, both the best macro and micro accuracies are achieved by LRCos in the GloVe embeddings (50% and 52%, respectively). Despite the different language of GloVe, this is also the combination that achieved the best results in BATS [5]. Although LRCos seems to be the best option overall, some exceptions arise against this absolutism, namely the syntactic relations with 3CosAvg in fastText CBOW.

LX-4WAnalogies had previously been used for assessing Portuguese word embeddings [11], using all models used here, except Numberbatch, and always with 3CosAdd. However, our results do not match the previous. This happens, first, due to the adoption of the BATS data format, which made that, for 3CosAdd and 3CosMul, the number of formulated questions was not the same for all types of analogy (see Section 3.4). Second, for every pair $a : a^*$ for which the model did not include either a or a^* (i.e., they were unknown to the model), the answer was automatically considered incorrect. Both differences made the test more difficult, but we would also say that it increased fairness in the comparison of the models. Nevertheless, although our results with 3CosAdd are lower than in previous work, the main conclusions are the same for this method. The best results for semantic analogies are achieved with GloVe and for the syntactic analogies with fastText-SKIP. Since fastText also considers character n-grams, it makes sense that it handles morphology well. However, a curious outcome of our results is that this is no longer true when LRCos is used in fastText-SKIP. It is not only outperformed by GloVe, but the macro-accuracy is also higher for the semantic relations than for the syntactic. In fact, this is a consequence that, when comparing 3CosAdd and LRCos, the increase of performance is always higher for the semantic than for the syntactic analogies, suggesting that LRCos is more suitable for semantic relations.

Even though we considered that questions with unknown words were automatically incorrect, we also looked at the results when those questions were simply ignored. As expected, all performances increase slightly but, for Numberbatch, the increase is substantial. Table 5 shows the results computed this way for all methods in three models. Figures suggest that Numberbatch is indeed a very accurate model, especially concerning semantic relations. However, it is much smaller and its performance in the previous experiment was heavily affected by its lower coverage.

For a finer-grained analysis, Tables 6 and 7 show the specific results, respectively for each semantic and syntactic relation. Again, we consider that, when a word in the question is not covered, the question is automatically incorrect. The first impression is that accuracy varies significantly, depending on the relation, meaning that different relations pose different challenges, with different levels of difficulty. For instance, with few exceptions, all combinations struggle in the city-in-state and currency analogies. The language of the embeddings may contribute to both of them, especially for the former, as names of states and cities in USA may not appear too often in Portuguese text. Still, in city-in-state, LRCos achieves the best accuracy in all but one model. On the currency, the only accuracies above 0 are those with LRCos in fastText-SKIP (3%) and Numberbatch, especially with LRCos (27%),

Table 5 Average accuracies when questions with unknown words are ignored.

Model	Method	Macro-Accuracy			Micro-Accuracy		
		Sem	Synt	Avg	Sem	Synt	Avg
GloVe	SimilarToB	6.75%	10.11%	8.91%	4.44%	7.20%	5.86%
	3CosAdd	26.32%	29.79%	28.55%	17.97%	30.67%	21.95%
	3CosMul	29.04%	33.27%	31.76%	21.75%	33.75%	25.51%
	3CosAvg	35.25%	43.41%	40.50%	27.82%	42.80%	35.55%
	LRCos	52.93%	**48.87%**	**50.32%**	57.26%	**49.24%**	53.13%
fastText SKIP	SimilarToB	4.26%	4.36%	4.33%	2.42%	4.92%	3.71%
	3CosAdd	25.31%	35.75%	32.02%	15.35%	37.90%	22.39%
	3CosMul	28.62%	38.01%	34.65%	20.73%	39.62%	26.63%
	3CosAvg	30.14%	42.91%	38.35%	21.37%	44.32%	33.20%
	LRCos	51.06%	45.55%	47.51%	52.42%	47.35%	49.80%
Numberbatch	SimilarToB	21.12%	2.65%	9.25%	25.17%	6.11%	16.31%
	3CosAdd	21.45%	8.97%	13.43%	18.21%	10.22%	15.72%
	3CosMul	23.94%	16.15%	18.93%	21.29%	12.12%	18.43%
	3CosAvg	41.93%	16.13%	25.34%	45.70%	24.62%	35.94%
	LRCos	**63.82%**	33.13%	44.09%	**71.52%**	45.80%	**59.57%**

which might have benefited of the amount of world knowledge included in ConceptNet. The best accuracies are for the capitals and family analogies. This is especially true for the capital-common-countries, where the highest accuracies are achieved with LRCos (e.g., 78% with GloVe or fastText).

On average, accuracies are lower for the syntactic analogies. As expected, the SimilarToB baseline is still the less accurate method. For almost every model and method, the highest accuracies are for the present-participle (e.g., 66% in fastText with 3CosAvg) and for the comparative (e.g., 80% in GloVe with LRCos or 3CosAvg). However, for the latter relation, results are not very representative, as they are only based on 20 questions, for 3CosAdd and 3CosMul, and on 5 questions, for the remaining methods. On the other hand, the good accuracy for the present-participle makes sense, because the questions are quite regular, and going from one form to the other is just a matter of adding the suffix *-ndo*. Therefore, a single example, as in 3CosAdd or 3CosMul, might be enough for solving the analogy. This is probably why the difference between methods is not so pronounced in this relation. Also, higher regularity works well for 3CosAvg, which is the best method in some models.

In opposition to the comparative, the superlative relation stands out has having almost all combinations with accuracies equal 0 or close. This should be mostly due to issue discussed in Section 3.3, related to the different superlative degrees in Portuguese. In the examples in Figure 1, it is clear that both types of superlative are used interchangeably (e.g., *brilhantíssimo* is the absolute superlative of *brilhante*, but *(o) pior* is the relative superlative of *mau*). This is even more problematic because the relative superlative uses the same forms as the comparative, even though they are used differently (e.g., *pior do que* for comparative and *o pior* for superlative). The relation with the second lowest accuracy was the opposite, which was quite surprising, because most opposites are obtained by adding prefixes like *i(n/m)-*, *des-* or *anti-*. It also becomes clear that relying exclusively on the Portuguese part of Numberbatch is not a suitable approach for several relations, mainly because ConceptNet will cover mostly lemmatised words without information on inflection.

Table 6 Accuracy for semantic relations.

Model	Method	Accuracy				
		cap-common	cap-world	city-in-state	currency	family
GloVe	SimilarToB	13.04%	3.45%	1.56%	0.00%	15.00%
	3CosAdd	52.77%	20.25%	6.72%	0.00%	51.84%
	3CosMul	55.73%	25.28%	7.12%	0.23%	56.84%
	3CosAvg	65.22%	26.72%	15.63%	0.00%	**65.00%**
	LRCos	**78.26%**	**66.38%**	**54.69%**	0.00%	60.00%
word2vec CBOW	SimilarToB	0.00%	0.00%	0.00%	0.00%	10.00%
	3CosAdd	10.47%	1.99%	0.67%	0.00%	28.16%
	3CosMul	11.46%	2.32%	0.62%	0.00%	34.21%
	3CosAvg	26.09%	7.76%	3.13%	0.00%	50.00%
	LRCos	30.43%	6.90%	0.00%	0.00%	30.00%
word2vec SKIP	SimilarToB	0.00%	0.00%	0.00%	0.00%	10.00%
	3CosAdd	30.43%	7.37%	3.00%	0.00%	34.47%
	3CosMul	32.61%	10.65%	3.32%	0.00%	35.26%
	3CosAvg	43.48%	12.93%	9.38%	0.00%	50.00%
	LRCos	56.52%	29.31%	21.88%	0.00%	45.00%
fastText CBOW	SimilarToB	0.00%	0.00%	0.00%	0.00%	10.00%
	3CosAdd	19.37%	4.89%	0.74%	0.00%	38.95%
	3CosMul	25.69%	6.57%	0.89%	0.00%	42.37%
	3CosAvg	17.39%	8.62%	1.56%	0.00%	50.00%
	LRCos	56.52%	36.21%	18.75%	0.00%	60.00%
fastText SKIP	SimilarToB	4.35%	1.72%	0.00%	0.00%	15.00%
	3CosAdd	48.22%	17.64%	2.95%	0.11%	57.63%
	3CosMul	55.14%	24.84%	3.60%	0.57%	58.95%
	3CosAvg	56.52%	19.83%	6.25%	0.00%	**65.00%**
	LRCos	**78.26%**	61.21%	42.19%	3.33%	**65.00%**
Numberbatch	SimilarToB	13.04%	21.55%	6.25%	0.00%	30.00%
	3CosAdd	34.19%	22.77%	1.86%	2.64%	45.79%
	3CosMul	41.11%	26.78%	2.03%	4.25%	45.53%
	3CosAvg	43.48%	35.34%	7.81%	3.33%	60.00%
	LRCos	69.57%	56.90%	10.94%	**26.67%**	55.00%

5 Conclusions

We have tested different methods that exploit word embeddings for solving analogies of the kind *What is to b as a^* is to a?*, in Portuguese. Although this problem had been tackled before [11], the previous goal was mainly to compare embeddings of different sizes and learned with different algorithms, always using the same analogy solving method, 3CosAdd. Here, we tested alternative methods for this purpose, always outperforming 3CosAdd, especially those methods that exploit a set and not just a single pair of related words ($a : a^*$), namely 3CosAvg and LRCos [5].

Despite working on a different language, initial conclusions are not much different from those for English. Different types of analogy pose different challenges, with varying accuracies. Still, overall, GloVe embeddings showed to be good at keeping linguistic regularities, with best results achieved by LRCos. In fact, when more than one pair is available, LRCos proved to be the best method for any model. As far as we know, this was the first time when the alternative analogy solving methods were tested for Portuguese, in the LX-4WAnalogies.

Table 7 Accuracy for syntactic relations.

Model	Method	Accuracy								
		adj-adv	opposite	comp	superl	pres-part	nation-adj	past	plural	plural-v
GloVe	SimilarToB	3.23%	3.57%	40.00%	4.00%	12.12%	0.00%	2.56%	21.62%	3.33%
	3CosAdd	4.95%	3.20%	60.00%	0.83%	49.81%	53.48%	26.52%	41.82%	27.47%
	3CosMul	6.02%	3.57%	70.00%	1.00%	52.84%	57.20%	32.39%	44.14%	32.30%
	3CosAvg	16.13%	17.86%	**80.00%**	0.00%	60.61%	68.29%	33.33%	67.57%	43.33%
	LRCos	25.81%	14.29%	**80.00%**	0.00%	60.61%	68.29%	56.41%	**72.97%**	**56.67%**
word2vec CBOW	SimilarToB	0.00%	10.71%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	3CosAdd	2.47%	6.77%	55.00%	0.67%	37.59%	12.68%	19.64%	9.68%	18.16%
	3CosMul	3.66%	6.53%	65.00%	1.00%	41.29%	15.37%	26.45%	12.24%	19.77%
	3CosAvg	9.68%	14.29%	60.00%	0.00%	54.55%	26.83%	28.21%	16.22%	33.33%
	LRCos	16.13%	17.86%	60.00%	0.00%	48.48%	29.27%	33.33%	32.43%	36.67%
word2vec SKIP	SimilarToB	0.00%	10.71%	0.00%	0.00%	3.03%	0.00%	2.56%	0.00%	3.33%
	3CosAdd	3.66%	7.51%	45.00%	0.67%	43.75%	29.88%	21.26%	14.86%	25.40%
	3CosMul	6.02%	7.02%	55.00%	0.83%	46.40%	38.35%	27.94%	17.04%	27.24%
	3CosAvg	12.90%	14.29%	60.00%	0.00%	57.58%	46.34%	25.64%	27.03%	36.67%
	LRCos	29.03%	17.86%	60.00%	0.00%	54.55%	56.10%	46.15%	40.54%	36.67%
fastText CBOW	SimilarToB	0.00%	3.57%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	3CosAdd	14.52%	9.26%	30.00%	0.33%	56.72%	36.52%	47.23%	29.73%	28.16%
	3CosMul	11.72%	9.39%	30.00%	0.33%	56.91%	42.93%	47.23%	29.43%	26.55%
	3CosAvg	22.58%	10.71%	60.00%	0.00%	**66.67%**	48.78%	58.97%	43.24%	40.00%
	LRCos	19.35%	17.86%	60.00%	0.00%	51.52%	51.22%	51.28%	51.35%	30.00%
fastText SKIP	SimilarToB	3.23%	3.57%	0.00%	0.00%	9.09%	0.00%	0.00%	16.22%	6.67%
	3CosAdd	9.46%	10.19%	60.00%	0.67%	64.49%	52.99%	43.25%	47.22%	33.45%
	3CosMul	11.72%	10.45%	70.00%	0.83%	63.92%	57.32%	46.96%	47.52%	33.33%
	3CosAvg	19.35%	14.29%	60.00%	0.00%	**66.67%**	60.98%	53.85%	56.76%	50.00%
	LRCos	29.03%	17.86%	60.00%	0.00%	63.64%	63.41%	**61.54%**	59.46%	50.00%
Numberbatch	SimilarToB	3.23%	0.00%	0.00%	0.00%	0.00%	17.07%	0.00%	0.00%	0.00%
	3CosAdd	8.92%	2.38%	20.00%	0.50%	0.19%	45.24%	0.00%	1.20%	2.30%
	3CosMul	14.09%	5.56%	70.00%	2.50%	0.38%	49.63%	0.00%	1.13%	2.07%
	3CosAvg	9.68%	0.00%	40.00%	0.00%	0.00%	56.10%	0.00%	5.41%	6.67%
	LRCos	**41.94%**	**32.14%**	40.00%	**8.00%**	6.06%	**70.73%**	0.00%	2.70%	6.67%

Another difference regarding previous work is that we included a different kind of word embeddings, Numberbatch. When ignoring questions with unknown pairs, performances achieved with this model are high, especially on semantic relations, where it achieved the best accuracy with LRCos. However, it is also a smaller model, with performance highly affected otherwise. In the future, we should test the larger newest version of Numberbatch (19.08). On the other hand, when solving analogies from a single pair, 3CosMul is generally better than 3CosAvg. In this specific case, fastText-SKIP is the best model for syntactic relations.

A limitation of the GAT and LX-4WAnalogies tests is that they are not balanced among the covered relations. This makes it harder to compare the performance for each relation and to rely on micro-average for analysing the performance in the full dataset. This is why we mainly looked at the macro-average of the accuracy. This is an issue that the BATS dataset tries to answer. It does not only cover a broader set of relation types but has exactly 50 instances for each relation type.

One of the additional categories of relation covered by BATS is precisely lexicographic relations, which are extremely useful for testing how suitable a model of embeddings is for augmenting lexical-semantic knowledge bases. Besides assessing how well word embeddings capture linguistic regularities, and thus how suitable they are for exploitation in many different tasks, analogy solving can be useful for supporting or discovering new lexical-semantic relations automatically, for instance, for populating knowledge bases. The latter may consider general language knowledge bases, including wordnets [7], and also domain ontologies, especially if embeddings are learned from a corpus of the same domain.

Since LX-4WAnalogies does not cover lexicographic relations (i.e., those one would find in a dictionary or wordnet), we have recently explored available lexical knowledge bases on the creation of a new dataset for assessing Portuguese word embeddings in the discovery of such relations [9]. This was an alternative to avoid time-consuming manual translation of BATS and language issues that may arise with the process, such as those we have identified in LX-4WAnalogies. Our first impression is that lexicographic relations are significantly more challenging than most of the relations covered by GAT. Nevertheless, manually accepting good answers out of the top candidates should still be less time-consuming than populating or augmenting a knowledge base completely from scratch. In this case, evaluation measures that consider the ranked candidates (e.g., Mean Average Precision) are relevant. Furthermore, we should test if better results are achieved when we combine several of the methods tested here (e.g., in an ensemble), and possibly explore alternative methods proposed more recently [3].

Finally, better results on this task might be achieved with more recent language models, also known as contextual embeddings, like BERT [4], for which pre-trained models are available for Portuguese. Even though words in analogy tests are not lack context, recent work has showed that the first principal component of such contextualized representations in a given layer (apparently, the lower, the better) can outperform static word embeddings in analogy solving [6].

References

1. Benjamin Bay, Paul Bodily, and Dan Ventura. Text transformation via constraints and word embedding. In *Proc. 8th International Conference on Computational Creativity*, ICCC 2017, pages 49–56, 2017.
2. Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.
3. Zied Bouraoui, Shoaib Jameel, and Steven Schockaert. Relation induction in word embeddings revisited. In *Proceedings of the 27th International Conference on Computational Linguistics*, COLING 2018, pages 1627–1637, Santa Fe, New Mexico, USA, August 2018. ACL.
4. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of Human Language Technologies, Vol 1*, NAACL-HLT 2019, pages 4171–4186. ACL, 2019.
5. Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuoka. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings the 26th International Conference on Computational Linguistics: Technical papers (COLING 2016)*, COLING 2016, pages 3519–3530, 2016.
6. Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. 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)*, pages 55–65, 2019.
7. Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, 1998.
8. Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL 2016 Student Research Workshop*, pages 8–15. ACL, 2016.
9. Hugo Gonçalo Oliveira, Tiago Sousa, and Ana Alves. Tales: Test set of portuguese lexical-semantic relations for assessing word embeddings. In *Proceedings of the ECAI 2020 Workshop on Hybrid Intelligence for Natural Language Processing Tasks (HI4NLP)*, page In press, 2020.
10. Zelig Harris. Distributional structure. *Word*, 10(2-3):1456–1162, 1954.

11 Nathan S. Hartmann, Erick R. Fonseca, Christopher D. Shulby, Marcos V. Treviso, Jéssica S. Rodrigues, and Sandra M. Aluísio. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Proceedings 11th Brazilian Symposium in Information and Human Language Technology (STIL 2017)*, 2017.

12 Omer Levy and Yoav Goldberg. Linguistic regularities in sparse and explicit word representations. In *Proceedings of 18th Conference on Computational Natural Language Learning*, CoNLL 2014, pages 171–180. ACL, 2014.

13 Tal Linzen. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 13–18, Berlin, Germany, August 2016. ACL.

14 Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the Workshop track of ICLR*, 2013.

15 Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia, June 2013. ACL.

16 Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2014, pages 1532–1543. ACL, 2014.

17 Andreia Querido, Rita Carvalho, João Rodrigues, Marcos Garcia, João Silva, Catarina Correia, Nuno Rendeiro, Rita Pereira, Marisa Campos, and António Branco. LX-LR4DistSemEval: a collection of language resources for the evaluation of distributional semantic models of Portuguese. *Revista da Associação Portuguesa de Linguística*, 3:265–283, 2017.

18 João Rodrigues, António Branco, Steven Neale, and João Ricardo Silva. LX-DSemVectors: Distributional semantics models for Portuguese. In *Proceedings of 12th International Conference on the Computational Processing of the Portuguese Language PROPOR*, volume 9727 of *LNCS*, pages 259–270, Tomar, Portugal, 2016. Springer.

19 Robert Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of Thirty-First Conference on Artificial Intelligence (AAAI)*, pages 4444–4451, 2017.