A Compositional-Distributional Semantic Model for Searching Complex Entity Categories

Juliano Efson Sales[1], **André Freitas**[1], **Brian Davis**[2], **Siegfried Handschuh**[1]
[1]Department of Computer Science and Mathematics - University of Passau
Innstrasse 43, ITZ-110, 94032 Passau, Germany
{juliano-sales,andre.freitas,siegfried.handschuh}@uni-passau.de
[2]Insight Centre for Data Analytics - National University of Ireland Galway
IDA Business Park, Lower Dangan, Galway, Ireland
brian.davis@insight-centre.org

Abstract

Users combine attributes and types to describe and classify entities into categories. These categories are fundamental for organising knowledge in a decentralised way acting as tags and predicates. When searching for entities, categories frequently describes the search query. Considering that users do not know in which terms the categories are expressed, they might query the same concept by a paraphrase. While some categories are composed of simple expressions (e.g. *Presidents of Ireland*), others have more complex compositional patterns (e.g. *French Senators Of The Second Empire*). This work proposes a hybrid semantic model based on syntactic analysis, distributional semantics and named entity recognition to recognise *paraphrases of entity categories*. Our results show that the proposed model outperformed the comparative baseline, in terms of recall and mean reciprocal rank, thus being suitable for addressing the vocabulary gap between user queries and entity categories.

1 Introduction

A significant part of search queries on the web target entities (e.g. people, places or events) (Pound et al., 2010). In this context, users frequently use the characteristics of the target entity to describe the search query. For example, to find *Barack Obama*, it is reasonable that a user types the query *Current President of United States*.

The combination of attributes and types of an entity in a grammatically correct fashion defines an *entity category*, which groups a set of entities that share common characteristics. Examples of entity categories are *French Female Artistic Gymnasts*, *Presidents of Ireland* and *French Senators Of The Second Empire*. Considering that users do not know in which terms the categories are expressed, they might query the same concept by a paraphrase, i.e. using synonyms and different syntactic structures.

The following text excerpt from Wikipedia shows an example where *Embraer S.A* is defined as *Brazilian aerospace conglomerate*:

> "*Embraer S.A. is a* ***Brazilian aerospace conglomerate*** *that produces commercial, military, executive and agricultural aircraft and provides aeronautical services. It is headquartered in São José dos Campos, São Paulo State.*"[1]

The flexibility and richness of natural language allow describing **Brazilian aerospace conglomerate** both as *Brazilian Planemaker*[2] or as *Aircraft manufacturers of Brazil*[3].

In addition to their occurrence in texts, entity categories are also available in the form of structured data. The Yago project (Suchanek et al., 2007) shares unary properties associating hundreds of thousands of descriptive categories manually created by the Wikipedia community to DBpedia entities (Auer et al., 2007). Thus, a mechanism to recognise paraphrases can make a shortcut between a natural language expression and a set of entities. Table 1 shows a list of entity categories and associated paraphrases.

This paper focuses on the recognition of *paraphrases of entity categories*, which is designed as an information retrieval task. To

[1] Extracted from https://en.wikipedia.org/wiki/Embraer

[2] In *Brazilian Planemaker Unveils Its Biggest Military Jet Yet* published by Business Insider.

[3] The Wikipedia category *Aircraft manufacturers of Brazil*.

*Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics (*SEM 2016)*, pages 199–208,
Berlin, Germany, August 11-12, 2016.

Original	***Paraphrased***
Prehistoric Canines	Ancestral Wolves
Soviet Pop Music Groups	Popular Musical Bands in the USSR
American Architectural Styles	Fashions of American Building Design
Defunct Companies of Finland	Bankrupt Finnish Businesses

Table 1: Examples of paraphrases.

deal with this problem, we propose an approach which combines syntactic analysis, distributional semantics and named entity recognition. To support reproducibility and comparability, we provide the test collection and the source code related to this work at `http://bit.ly/cat-test-collection` and `http://bit.ly/linse-code`.

2 Understanding the Structure of an Entity Category

An entity category names and classifies a set of entities. It is composed of a central concept, called *core*, and its *specialisations*. For example, the entity category *2008 Film Festivals* embraces *festivals*, which defines the category's core. More specifically, this category covers those *festivals* that are related to *film*s and occurred in *2008*. In its turn, *Populated Coastal Places in South Africa* embraces *places* (the core) that are *populated*, in the coast (*coastal*) and within *South Africa*. While *festivals* and *places* act as cores, all other terms work as *specialisations*, defining characteristics such as *temporality* (specialisations of time), *localization* (specialisations of place) and other general characteristics. These three types of terms are respectively classified as *temporal named entity*, *spatial named entity*, and *general specialisation*.

By analysing a large set of entity categories generated in a decentralised setting, Freitas et al. (2014) described them according to a group of recurring features: *contains verbs*, *contains temporal references*, *contains named entities*, *contains conjunctions*, *contains disjunctions* and *contains operators*. These features suggest a syntactic pattern that can be described as a combination of simple relations based on the lexical categories of their constituent terms (Freitas et al., 2014). In this manner, we apply a list of parsing rules to determine the graph structure/hierarchy according to Table 2, which defines *the core-oriented segmentation model*.

POS Pattern	**Core-side**
[VB, IN]	left
[NN, VBG]	left
[IN]	left
[”,”]	left
[POS]	right
[CC]	left

Table 2: Rules to construct the graph of an entity category.

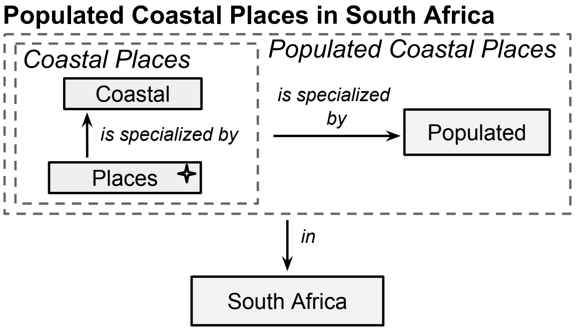

Figure 1: Graph of *Populated Coastal Places in South Africa*.

During the parsing process, categories are analysed from left to right. Once a pattern is identified, the *core-side* attribute specifies the side where the core is located. Both parts are then recursively analysed, where the opposite part is treated as specialisation(s). The order of the rules determines their precedence. To simplify the rule list, some tags are normalised, e.g. POS-tag *TO* is converted to *IN* and *NNPS* is converted to *NNP*. When no pattern is identified, the last term in the resulting chunk is admitted as the *core* and all others as *specialisations*, if any.

Figure 1 shows the graph generated by the core-oriented segmentation method for the entity category *Populated Coastal Places in South Africa*. The graph root (*places*) represents the core.

3 Semantic Approximation & Compositionality

From a finite set of words, it is possible to express unlimited utterances and ideas. This property is credited to the principle of *semantic compositionality* (Baroni et al., 2014a).

Distributional semantics is based on the hypothesis that words co-occurring in similar contexts tend to have similar meaning (Harris, 1954; Turney and Pantel, 2010). Distributional semantics supports the automatic construction of semantic models from large-scale unstructured corpora, using vector space models to represent the meaning of a word. The process to construct distributional models ranges from statistical methods to models based on machine learning (Dumais et al., 1988; Mikolov et al., 2013; Jeffrey Pennington, 2014).

Distributional semantics allows measuring the semantic compositionality by combining an appropriate *word representation* and a suitable method to *semantically compose* them. Its meaning representation supports the construction of more comprehensive semantic models which have semantic approximation at its centre. We compute the semantic similarity and relatedness between two terms using vector operations in the vector space.

4 Compositional-Distributional Model

This work proposes *a hybrid model that combines the core-oriented segmentation model with semantic approximation based on distributional semantics to provide a semantic search approach for entity categories*. This approach segments the entity categories and stores their constituent parts according to their type in a graph-based data model.

The graph data model has a signature $\Sigma = (C, Z, R, S, E)$, where C, Z, R and S represent the sets of *cores*, *general specialisations*, *temporal specialisations* and *spatial specialisations* respectively. E contains sets of edges, where each set represents a graph. The elements in C and Z are natural language terms indexed in distributional semantics spaces. The elements in R are closed integer intervals representing the temporal expressions in years. The elements in S are sets of equivalent terms referring to a geographic place and its demonyms. The proposed graph data model is inspired by the τ-*Space* (Freitas et al., 2011), which represents graph knowledge in a distributional space.

Distributional semantics spaces represent terms by distributional vectors. The distributional vectors are generated from a large external corpus to capture the semantic relation in a broader scenario. It allows that even when dealing with a small dataset, the semantic representation is not limited to that context. The distributional space allows searching by measuring the geometric distances or vector angles between the query term and the indexed terms.

Temporal and spatial specialisations do not use the same representation strategy. In the case of spatial named entities, our tests have shown poor performance when using general-purpose distributional semantics models to compare them. The problem resides in the fact that places and demonyms have a high relatedness with common nouns. For example, in one distributional model[4], *American* has a higher relatedness with *war* than with *Texas*. To avoid this kind of misinterpretation, spatial expressions are compared using their names, acronyms, and demonyms.

Because of the numerical and ordered nature of temporal references, temporal specialisations are represented as year intervals. By this representation, two expressions of time are compared by computing the interval intersection. We consider them as semantically related if the intersection is not empty.

4.1 Constructing the Knowledge Representation Model

The first step is to build the data model based on the target set of entity categories. For each entity category in the set, the segmentation model presented in Section 2 generates a graph representation $G = (V, E)$. The set of vertices (V) is the union of the core term $\vec{c}$, the set of general specialisations (Z'), the set of temporal specialisations (R') and the set of spatial specialisations (S'), i.e. $V = \{c'\} \cup Z' \cup R' \cup S'$. Any of these three sets of specialisations can eventually be empty. The process of building the data space from a target set of entity categories `T` is described in Algorithm 1. In line 6, the category `t` is decomposed by the core-oriented segmentation model. Each term is indexed in their respective index according to their type: the core ($\vec{c}$) in the core space (C) and the specialisations in the general specialisation space (Z),

[4] Distributional models used in the context of this work are presented in Section 5.

temporal space (R) and spatial space (S).

Spatial specialisations are identified by the longest string matching method comparing against a dictionary which contains the name, acronym and demonym of places. Temporal expressions are converted to an interval of years. Terms that are considered neither spatial nor temporal specialisations fall into the general specialisation case.

Algorithm 1 Construction

```
input : $\texttt{T}$ : target set of entity categories.
output : $\Sigma$ : a filled graph data model.

$C \leftarrow \emptyset, Z \leftarrow \emptyset, R \leftarrow \emptyset, S \leftarrow \emptyset, E \leftarrow \emptyset$
for $\texttt{t} \in \texttt{T}$ do
    $\vec{c}, Z^{'}, R^{'}, S^{'}, E^{'} \leftarrow graphOf(\texttt{t})$
    $C \leftarrow \bigcup\{\vec{c}\}$
    $Z \leftarrow \bigcup Z^{'}$
    $R \leftarrow \bigcup R^{'}$
    $S \leftarrow \bigcup S^{'}$
    $E \leftarrow \bigcup\{flat(E^{'})\}$
return $\Sigma$
```

To illustrate visually, Figure 2 depicts a diagram where the entity categories *2000s Film Festivals* and *Populated Coastal Places in South Africa* are represented within the model. The cores *festivals* and *places* are stored in the core distributional space (C: geometric representation). The first category has two specialisations: the time interval 2000-2009, indexed in the temporal space (R: interval representation); and *film*, indexed in the general specialisation space (Z: also geometric representation). Next, the second category has three specialisations: the spatial named entity *South Africa*, indexed in the spatial index (S: expanded index); and the general specialisations *coastal* and *populated*, indexed in (Z). Dashed lines connecting the cores to their specialisations represent the flattened edges of the graphs, i.e. all specialisations are connected directly to their respective core.

4.2 Searching as Semantic Interpretation

Algorithm 2 describes the interpretation process that receives the query and the graph data model Σ as inputs. Queries are paraphrases that follow the same syntactic pattern of entity categories. The process starts by generating the graph of the input query (line 4). Considering the graph structure, each vertice becomes a sub-query to be submitted to their respective specific index (representation space).

The core defines the first sub-query. It needs to be semantically aligned to relevant cores in Σ. In line 5, $distSearch(\vec{c}, C)$ searches for cores semantically related to the query core $\vec{c}$. In addition to the simple searching of terms and synonyms, the vector cosine defines how related $\vec{c}$ is to the cores present in C. Given a threshold η, distributional search returns $K = \{(\vec{k}, h) | \vec{k} \in C, h = cosine(\vec{k}, \vec{c}), h > \eta\}$. The semantic relatedness threshold η determines the minimum distance or angle between the query core and the target cores that makes them semantically relevant. In the context of this work, η is defined dynamically according to the result set. Let X be the descending-order set of returned cosine scores, $(\eta = x_n | x_n \in X, x_{n+1} \in X, x_n/2 > x_{n+1})$. The distributional search returns a set of pairs $(\vec{k}, h)$ where $\vec{k}$ is a core term and h is the normalised $cosine(\vec{k}, \vec{c})$. Entity categories containing relevant cores are select for the next search step (lines 6, 7).

The next step deals with the specialisations. Spatial and temporal named entities found in the query are searched in their respective subsets, identifying equivalent spatial representations (lines 11-13) and comparing the time intervals (lines 14-20). Temporal expressions out-of-range are penalised by a negative score (line 20). The pairing of general specialisations (lines 22-24) follows the same principle of the core search. When there are two or more general specialisations, the method $maximiseMatching$ aims to avoid that two terms from one side match to the same term on the other side, selecting the pairs that maximise the final score.

The final score is determined by the composition of all scores proportionally to the number of terms in the categories according to the expressions in the lines 26-29.

In the example of Figure 2, *2008 Movie Celebrations* is the query which is segmented in *celebration* (core), *movie* (general specialisation) and *2008* (temporal interval). The core term *celebrations* feeds a sub-query in the distributional core space. The alignment is defined by computing a distributional semantic relatedness function between *celebrations* and all cores in the core space and by filtering out all the elements which are below the semantic relatedness threshold η.

Navigating over the graph structure, the query

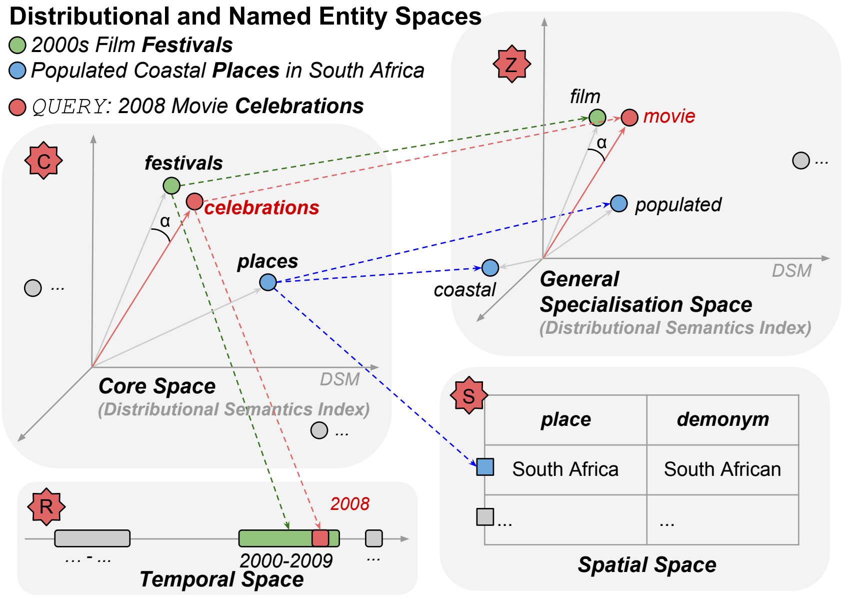

Figure 2: Depiction of the structured distributional vector space model.

terms representing specialisations are searched in the subspaces according to their type. In the given query example, *movie* is semantically aligned in the general specialisation space applying the same approach described in the core space. In its turn, the intersection is calculated for the temporal specialisation *2008* in the temporal space.

5 Evaluation

The evaluation focuses on comparing the compositional-distributional model to baseline approaches and assessing the performance of different distributional semantic models in combination with our representation model. The evaluation scenarios are designed to measure the individual contribution of each component.

5.1 Setup

The evaluation has three comparative baselines:
Bag-of-words search: Target entity categories are indexed in a state-of-the-art information retrieval system treating each category as a separate document. Additionally, the document is enriched by synonyms obtained from WorNet (Miller, 1995). Lucene[5] 4.10.1 is the information retrieval system used in the experiment.
Pure core-oriented segmentation: The core-oriented segmentation model incorporated by this work is applied in an isolated fashion, i.e. without the distributional component but making use of simple string matching, WordNet expansion and temporal and spatial named entity indices.
Sum-algebraic-based method: Entity categories are compared by an algebraic operation that sums up component vectors using the resulting vectors to calculate the cosine similarity. This method results in many scenarios, one for each distributional model.

Five different models are analysed in this work:
Latent Semantic Analysis (Dumais et al., 1988): LSA is a distributional semantic space that extracts statistical relations between words in narrow context windows. It is characterised for executing a costly operation to reduce the space dimensionality.
Random Indexing (RI) (Sahlgren, 2005): Random Indexing was proposed to avoid the dimensional reduction. It dynamically accumulates *context vectors based on the occurrence of words in contexts* to generate the semantic space.
Explicit Semantic Analysis (Gabrilovich and Markovitch, 2007): ESA uses entire documents as contexts. It was created under the assumption of *concept hypothesis*[6] which states that a portion of information such as an article or document is associated with a particular concept, and the space model could take advantage of this information.
Continuous Skip-gram Model (W2V) (Mikolov et al., 2013): Skip-gram is a vector space model created by deep learning techniques focused on lo-

[5] `http://lucene.apache.org/`

[6] Studies contest the existence of this hypothesis (Gottron et al., 2011).

Algorithm 2 Semantic Interpretation Process

```
input : query and $\Sigma = (C, Z, R, S, E)$
output : Z : related categories and their score.

$\vec{c}, Z^q, R^q, S^q, E^q \leftarrow graphOf(\texttt{query})$
$U \leftarrow distSearch(\vec{c}, C)$
for $(\vec{k}, h) \in K$ do
    $D \leftarrow selectGraphsByCore(\vec{k}, E)$
    for all $D^{'} \in D$ do
        $\vec{k}, Z^c, R^c, S^c, E^c \leftarrow D^{'}$
        $a \leftarrow 0$
        for $s^c \in S^c$ do
            if $\exists s \in S^q \mid s^c \equiv s$ then
                $a \leftarrow a + 1$
        $b \leftarrow 0$
        for $r^c \in R^c$ do
            if $\exists r \in R^q \mid r^c \equiv r$ then
                $b \leftarrow b + 1$
            else
                if $R^q \neq \emptyset$ then
                    $b \leftarrow b - 0.5$
        $X \leftarrow \emptyset$
        for $\vec{o^c} \in O^c$ do
            $J \leftarrow distSearch(\vec{o^c}, O^q)$
            $X.append(J)$
        $Y \leftarrow maximiseMatching(X)$
        $n^q \leftarrow \mid E^q \mid$
        $n^c \leftarrow \mid E^c \mid +1$
        $u \leftarrow h + a + b + (\sum_{x=1}^{n} y_x | y_x \in Y)$
        $u \leftarrow u * (n^q / n^c)$
        $U.append(D^{'}, u)$
return $sort(U)$
```

cal context windows.

Global Vectors (GloVe) (Jeffrey Pennington, 2014): GloVe aims to conciliate the statistical co-occurrence knowledge present in the whole corpus with the local pattern analysis (proposed by the skip-gram model) applying a hybrid approach of conditional probability and machine learning techniques.

DINFRA (Barzegar et al., 2015), a SaaS distributional infrastructure, provided the distributional vectors. We generated all five ditributional models using the English Wikipedia 2014 dump as a reference corpus, stemming by the Porter algorithm (Porter, 1997) and removing stopwords. For LSA, RI and ESA, we used the SSpace Package (Jurgens and Stevens, 2010), while W2V and GloVe were generated by the code shared by the respective authors. All models used the default parameters defined in each implementation.

5.2 Test Collection

The test collection is composed of a knowledge base of more than 350,000 entity categories obtained from the complete set of Wikipedia 2014 categories, but removing those containing non-ASCII characters. Each category has between one to three paraphrases.

The creation of the queries was guided by seed target categories. The use of seed entity categories was deliberately decided to ensure the presence of one paraphrase equivalence for each query.

Queries were generated by asking a group of English-speaking volunteers to paraphrase the subset of 105 categories. They were instructed to describe the same meaning using different words and, if possible, different syntactic structures. After that, we applied a curation process conducted by two researchers to validate the paraphrase's equivalence intuitively. In the end, we admitted a set of 233 paraphrased pairs.

To create various degrees of difficulty in the topics, we balanced the test collection with categories varying in size (two to ten terms), in the occurrence of places and demonyms references, in the presence of temporal expressions and, in the occurrence of noun phrase components (verbs, adjectives, adverbs).

Test collection files are available at `http://bit.ly/cat-test-collection`.

5.3 Results and Discussion

We evaluate our approach in three scenarios. The first considers the TOP-10 list of each execution. The second considers the TOP-20 list and the third the TOP-50.

For each query in the test collection, we calculate the recall and mean reciprocal ranking, together with their aggregate measures (Table 3). Figure 3 provides a visual representation of the recall scores. In the experiment, we assumed that only one category corresponded to the correct answer. This assumption makes *precision* a redundant indicator since it can be derived from *recall* ($precision = recall/range|range \in \{10, 20, 50\}$).

The evaluation shows that distributional semantic models address part of the semantic matching tasks since distributional approaches outperform simple stemming string search and WordNet-

Approaches	***Recall***			***MRR***		
	Top 10	***Top 20***	***Top 50***	***Top 10***	***Top 20***	***Top 50***
Lucene	0.0904	0.1040	0.1357	0.0410	0.0420	0.0429
Core-Oriented Segmentation	0.0985	0.1126	0.1361	0.0613	0.0623	0.0630
Sum-algebraic-based method	-	-	-	-	-	-
with LSA	0.1126	0.1621	0.2117	0.0595	0.0631	0.0645
with RI	0.0630	0.0945	0.1216	0.0348	0.0371	0.0379
with ESA	0.0540	0.0900	0.1486	0.0271	0.0296	0.0312
with W2V	0.2657	0.3333	0.3963	0.1356	0.1403	0.1422
with GloVe	0.2702	0.3558	0.4324	0.1417	0.1476	0.1501
Our proposed method	-	-	-	-	-	-
with LSA	0.3545	0.4000	0.4590	0.1981	0.2013	0.2033
with RI	0.3073	0.3743	0.4078	0.1768	0.1813	0.1823
with ESA	0.2818	0.3182	0.4000	0.1822	0.1846	0.1872
with W2V	**0.3727**	**0.4364**	**0.4909**	**0.2448**	**0.2491**	**0.2510**
with GloVe	**0.3727**	0.4090	0.4500	0.2274	0.2300	0.2314

Table 3: Results for recall and mean reciprocal rank (MRR).

based query expansion. By applying either *sum-algebraic-based method* and *our proposed method*, most of the distributional models present significant performance improvement in comparison to non-distributional methods. It is also important to stress that Word2Vec and GloVe consistently deliver better results for the test collection. Apart the controversies about predictive-based and count-based distributional models (Baroni et al., 2014b; Lebret and Collobert, 2015; Levy and Goldberg, 2014), in the context of this work, these results suggest that predictive-based distributional models outperform count-based methods (despite the proximity of LSA results).

Regarding the compositional method, the results of the *core-oriented strategy* combined with the named entity recognition exceeded all results delivered by the *sum-algebraic-based method* when comparing the same distributional model. The performance increases not only in the recall, which represents more entity categories retrieved but also in the mean reciprocal rank, reflecting that the target categories are better positioned in the list. Our proposed method succeed in almost 50% of the test collection when considered the Top-50 scenario.

Sales et al. (2015) shows a prototype demonstration of this work.

5.4 Analysing Unsuccessful Cases

The most significant limitation is the restriction of comparing words one-by-one, assuming that each word in a paraphrase is semantically equivalent to only one word in the target categories and vice-versa. For example, the pair (*Swedish Metallurgists*, *Metal Workers from Sweden*) is ranked at #1173 when using W2V. It occurs because *metallurgists* and *workers* have low relatedness (0.0031). Comparing the relatedness of *metallurgists* to *metal workers* would have a higher score.

Concerning named entities, we observed three relevant issues. Our approach uses a simple longest string matching method to identify places. Categories containing terms such as *Turkey* are always considered a spatial named entity. In the pair (*American Turkey Breeds* and *Chicken Breeds Originating in the US*) the terms *turkey* and *chicken* would not be semantically compared, since *Turkey* is always considered a spatial named entity. Secondly, when searching for *Water Parks in the USA*, all parks at *Texas*, *Tennessee* or *Pennsylvania* are also relevant for the user. Our model does not contain this hierarchical information to provide a geographic match. Finally, expressions such as *WWI* and *USSR* should be identified as the paraphrasing of *Wold War I* and *the Soviet Union* or even other variations, what is not available in our model.

6 Related Work

Balog and Neumayer (2012) propose the *hierarchical target type identification problem* which aims to identify the most specific type grounded in a given ontology that covers all entities sought

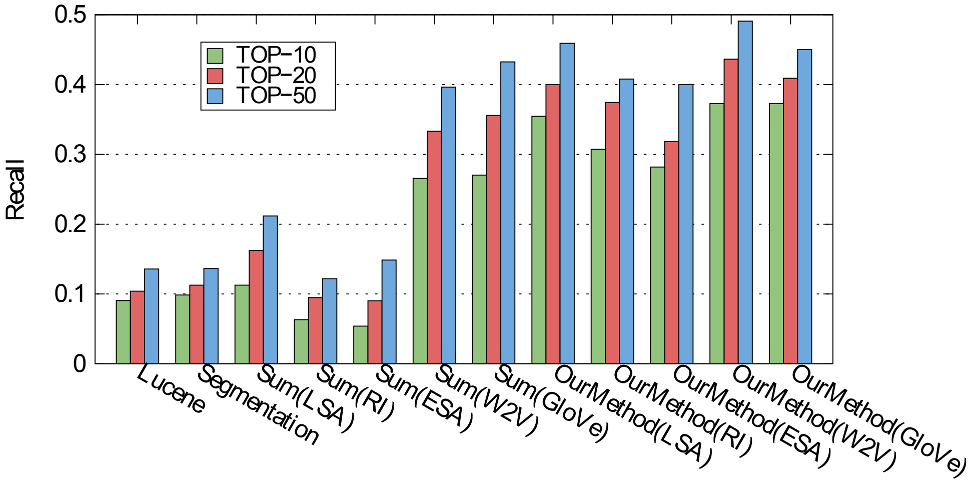

Figure 3: Chart of recall values grouped by different approaches.

by a query. Yao et al. (2013) propose an *entity type prediction* considering the *universal schema*. In this work, a predictor is expected to label a given entity with types. This schema is composed of all types from diverse available ontologies. To identify types from texts, they compose named entity recognition with dependency parsing. These works focus on identifying the ontological types that are sought by the query.

Regarding *entity similarity*, Moreau et al. (2008) propose a method to calculate entity similarity based on Soft-TFIDF. Liu and Birnbaum (2007) propose a method based on the Open Directory Project (ODP) to capture category names in all pages where the named entity appears to generate a vector space. Liu et al. (2008) describe a method that uses the set of URLs in which entities are present to measure similarity. The difference to these works is that they focus on comparing named entities, not based on their description, but based on non-linguistic attributes.

Other related topics are *paraphrasing* and *text entailment*. Androutsopoulos and Malakasiotis (2010) present an extension overview of datasets and approaches applied in these fields. Papers in this context deal with the paraphrasing of complete sentences (formed of subject and predicate) which cannot benefit from the core-oriented segmentation model. The different format of their target datasets inhibits a direct comparison, while their lack of association with entities does not create the required bridge between unstructured and structured data.

This work distinguishes mainly from existing approaches by proposing a novel compositional method grounded in syntactic analysis to combine distributional vectors and by using distributional semantics models generated from external resources. The target knowledge base (the dataset of categories) is not part of the data used to produce the distributional models. This isolation supports a more comprehensive semantic matching.

7 Conclusion

This work proposes a compositional-distributional model to recognise paraphrases of entity categories. Distributional semantics in combination with the proposed compositional model supports a search strategy with robust semantic approximation capabilities, largely outperforming string and WordNet-based approaches in recall and mean reciprocal rank. The proposed compositional strategy also outperforms the tradional *vector-sum method*.

This work also provides additional evidence to reinforce *(i)* the suitability of distributional models to cross the semantic gap (Freitas et al., 2012; Aletras and Stevenson, 2015; Agirre et al., 2009; Freitas et al., 2015) and *(ii)* suggest that prediction methods generate better semantic vectors when compared to count-based approaches. Considering the controversies about the comparisons between predictive-based and count-based distributional models (Baroni et al., 2014b; Lebret and Collobert, 2015; Levy and Goldberg, 2014), this evidence is restricted to the distributional models involved in the experiment and cannot be generalised. In the context of our work, we conjecture that the better performance is credited to the fact that our problem comprises much more *paradig-*

matic than *syntagmatic* relations.

Additionally, the use of distributional semantic models provides a better base for transporting the solution to multi-lingual scenarios, since it does not depend on manually constructed resources.

Future work will focus on the investigation of specialised named entity distributional methods in the context of the semantic search problem.

Acknowledgments

This publication has emanated from research supported by the National Council for Scientific and Technological Development, Brazil (CNPq) and by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289. The authors also would like to thank Douglas N. Oliveira (Florida Institute of Technology) and the anonymous reviewers for the valuable critical comments.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09.

Nikolaos Aletras and Mark Stevenson. 2015. A hybrid distributional and knowledge-based model of lexical semantics. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, June.

Ion Androutsopoulos and Prodromos Malakasiotis. 2010. A survey of paraphrasing and textual entailment methods. *J. Artif. Int. Res.*, 38(1):135–187, May.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, pages 722–735, Berlin, Heidelberg. Springer-Verlag.

Krisztian Balog and Robert Neumayer. 2012. Hierarchical target type identification for entity-oriented queries. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM'12.

Marco Baroni, Raffaella Bernardi, and Roberto Zamparelli. 2014a. Frege in space: A program for compositional distributional semantics. *Linguistic Issues in Language Technology*, 9.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014b. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, June.

Siamak Barzegar, Juliano Efson Sales, Andre Freitas, Siegfried Handschuh, and Brian Davis. 2015. DINFRA: A one stop shop for computing multilingual semantic relatedness. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15.

S. T. Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester, and R. Harshman. 1988. Using latent semantic analysis to improve access to textual information. In *Proceedings of the Conference on Human Factors in Computing Systems*.

André Freitas, Edward Curry, João Gabriel Oliveira, and Seán O'riain. 2011. A distributional structure semantic space for querying RDF graph data. *International Journal of Semantic Computing*, 05(04):433–462.

André Freitas, Edward Curry, and Seán O'Riain. 2012. A distributional approach for terminological semantic search on the linked data web. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, SAC '12.

André Freitas, Rafael Vieira, Edward Curry, Danilo Carvalho, and João Carlos Pereira da Silva. 2014. On the semantic representation and extraction of complex category descriptors. In *Natural Language Processing and Information Systems*, volume 8455 of *Lecture Notes in Computer Science*.

Andre Freitas, Juliano Efson Sales, Siegfried Handschuh, and Edward Curry. 2015. How hard is this query? measuring the semantic complexity of schema-agnostic queries. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 294–304, London, UK, April. Association for Computational Linguistics.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07.

Thomas Gottron, Maik Anderka, and Benno Stein. 2011. Insights into explicit semantic analysis. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM'11.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23).

Christopher Manning Jeffrey Pennington, Richard Socher. 2014. Glove: Global vectors for word representation. In *Proceedings of the*

2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), October.

David Jurgens and Keith Stevens. 2010. The S-Space Package: An open source package for word space models. In *Proceedings of the ACL 2010 System Demonstrations*, ACLDemos '10, pages 30–35, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rémi Lebret and Ronan Collobert, 2015. *Rehabilitation of Count-Based Models for Word Vector Representations*, pages 417–429. Springer International Publishing, Cham.

Omer Levy and Yoav Goldberg. 2014. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 171–180, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Jiahui Liu and L. Birnbaum. 2007. Measuring semantic similarity between named entities by searching the web directory. In *Web Intelligence, IEEE/WIC/ACM International Conference on*, Nov.

Hui Liu, Jinglei Zhao, and Ruzhan Lu. 2008. Mining the URLs: An approach to measure the similarities between named-entities. In *Innovative Computing Information and Control, 2008. ICICIC '08. 3rd International Conference on*, June.

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

George A. Miller. 1995. WordNet: A lexical database for english. *Commun. ACM*, 38(11), November.

Erwan Moreau, François Yvon, and Olivier Cappé. 2008. Robust similarity measures for named entities matching. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, August.

Martin F. Porter. 1997. Readings in information retrieval. chapter An Algorithm for Suffix Stripping, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Jeffrey Pound, Peter Mika, and Hugo Zaragoza. 2010. Ad-hoc object retrieval in the web of data. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 771–780, New York, NY, USA. ACM.

Magnus Sahlgren. 2005. An introduction to random indexing. In *In Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE 2005*.

Juliano Efson Sales, André Freitas, Siegfried Handschuh, and Brian Davis. 2015. Linse: A distributional semantics entity search engine. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 1045–1046, New York, NY, USA. ACM.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA. ACM.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188, January.

Limin Yao, Sebastian Riedel, and Andrew McCallum. 2013. Universal schema for entity type prediction. In *Proceedings of the Workshop on Automated Knowledge Base Construction*.