DATA PAPER

Applying a Context-based Method to Build a Knowledge Graph for the Blue Amazon

Pedro de Moraes Ligabue[1†], Anarosa Alves Franco Brandão[2], Sarajane Marques Peres[3], Fabio Gagliardi Cozman[4], Paulo Pirozelli[5]

[1]Escola Politécnica, Universidade de São Paulo, Rua Visconde da Luz, 60, São Paulo, São Paulo 05508-010, Brazi

[2]Escola Politécnica, Universidade de São Paulo, Av. Prof. Luciano Gualberto, tv 3, 158, São Paulo 05508-010, Brazil

[3]Escola de Artes Ciências e Humanidades, Universidade de São Paulo, Rua Arlindo Béttio, 1000, São Paulo 03828-000, Brazil

[4]Escola Politécnica, Universidade de São Paulo, Av. Prof. Luciano Gualberto, tv 3, 158, São Paulo 05508-010, Brazil

[5]Instituto de Estudos Avançados, Universidade de São Paulo, Rua da Praça do Relógio, 109, São Paulo 05508-050, Brazil

Keywords: knowledge graph; word embeddings; relationship triple extraction; Blue Amazon; Atlantic Ocean; Brazilian coast

Citation: Ligabue P.D.M., Brandão A.A.F., Peres S.M., et al.: Applying a Context-based Method to Build a Knowledge Graph for the Blue Amazon. Data Intelligence 6(1), 64–103 (2024). doi: https://doi.org/10.1162/dint_a_00223
Submitted: March 27, 2023; Revised: July 5, 2023; Accepted: August 10, 2023

ABSTRACT

Knowledge graphs are employed in several tasks, such as question answering and recommendation systems, due to their ability to represent relationships between concepts. Automatically constructing such a graphs, however, remains an unresolved challenge within knowledge representation. To tackle this challenge, we propose CtxKG, a method specifically aimed at extracting knowledge graphs in a context of limited resources in which the only input is a set of unstructured text documents. CtxKG is based on OpenIE (a relationship triple extraction method) and BERT (a language model) and contains four stages: the extraction of relationship triples directly from text; the identification of synonyms across triples; the merging of similar entities; and the building of bridges between knowledge graphs of different documents. Our method distinguishes itself from those in the current literature (i) through its use of the parse tree to avoid the overlapping entities produced by base implementations of OpenIE; and (ii) through its bridges, which create a connected network of graphs, overcoming a limitation similar methods have of one isolated graph per document. We compare our method to two others by generating graphs for movie articles from Wikipedia and contrasting them with benchmark graphs built from the OMDb movie database. Our results suggest that our method is able to improve multiple aspects of knowledge graph construction. They also

† Corresponding author: Pedro de Moraes Ligabue (E-mail: pedro.ligabue@alumni.usp.br; ORCID: 0000-0002-5004-7355).

highlight the critical role that triple identification and named-entity recognition have in improving the quality of automatically generated graphs, suggesting future paths for investigation. Finally, we apply CtxKG to build BlabKG, a knowledge graph for the Blue Amazon, and discuss possible improvements.

1. INTRODUCTION

Knowledge graphs are structures in which nodes represent entities (real, fictitious, or abstract), and edges represent relationships that exist between them [1]. Knowledge graphs have proved to be particularly useful at handling ambiguity, establishing connections between ideas and describing attributes and characteristics [1-3].

The usefulness of knowledge graphs has been consistently observed in prominent applications such as recommender systems and question answering (QA) [1, 3, 4]. Additionally, they have also been successfully employed in more specific situations, such as weather simulations [5] and medical evaluations [6].

The Google knowledge graph is an example of a knowledge graph created to assist with search results by utilizing relationships to find associated entities and attributes. It also serves as the source for the Google knowledge panels — cards containing essential information about people, locations etc., built using data linked to the searched terms in the knowledge graph [2, 7]. Another example is Wikidata, which acts as the central source of structured data for all the Wikimedia projects, including Wikipedia. Because of that, Wikidata covers an extremely vast number of topics and domains, and includes numerous relationship and association types [8].

While large-scale graphs are useful for those types of applications, it may be necessary to build new knowledge graphs when domain-specific information is required. These targeted knowledge graphs are unlikely to hold knowledge that does not pertain to the domain of interest, while including more specialized and in-depth information.

Constructing knowledge graphs, however, is still a persisting challenge within knowledge representation [3, 9]. Many of the largest and most popular knowledge graphs rely on volunteer crowd-sourcing for expansion and maintenance [3]. For instance, between 2014 and October 2022, the amount of edits to Wikidata by real users increased from 10% to around 59.87% [8, 10], indicating a considerable dependency on human participation—a scenario that represents a serious expenditure in terms of time and resources. To overcome this dependency, several techniques have been developed in order to extract knowledge graphs directly from text [3, 9, 11]. These techniques usually employ Natural Language Processing (NLP) pipelines to extract relationships, identify named entities and resolve coreferences.

Yet, there are important issues associated with current NLP approaches to knowledge-graph construction, namely (i) the loss of information, which comes from not fully interpreting the contents of the text; (ii) the redundancy of information, which happens when an excessive number of entities or relations are extracted from the text; and (iii) the overlapping of information, which is associated with the handling of change in attributes (e.g. how to express that an entity was previously in one state but now is in another) [3].

In this paper, we present a method for building knowledge graphs from text documents, which we refer to as **CtxKG** (Context-Based Knowledge Graph). The challenge was to develop a method which can work in an environment where resources are sparse. In our case, we are regularly dealing with small sets of documents, which are not accompanied by a backing knowledge base. For those reasons, we rely heavily on the syntactic and semantic information in the documents. In order to specify our method, we provide detailed descriptions of each of its parts and evaluate its results through direct comparisons to similar approaches [9, 12].

After presenting our method, we delineate how it was used to build **BlabKG**, a knowledge graph for the Blue Amazon. The Blue Amazon is a large area off the coast of Brazil that includes Brazil's territorial waters, the Exclusive Economic Zone and the continental shelf (Fig. 1). It is a region of great importance, as its 45,000,000 km^2 of sea cover 90% of Brazil's oil reserves and 77% of its gas reserves, as well as multiple different and unique ecosystems, including mangroves, estuaries and coral reefs [13-16].

Although **BlabKG** was first discussed in [17], we have added new results and examples, which should further illustrate its features and possible applications. Those include an example of how our entity unpacking extension works, along with a demonstration of how **BlabKG** can be used to find entities with shared characteristics.

The paper is organized as follows. Section 2 describes different knowledge-graph generation approaches and how they influenced **CtxKG**. Section 3 goes through our knowledge graph generation method in detail. Section 4 describes the evaluation process for our method, comparing it to other alternatives. Section 5 presents the corpus used to generate **BlabKG**, and describes and examines its features. A discussion about improvement opportunities for **CtxKG** is given in Section 6. Finally, Section 7 suggests some paths for future investigation.

2. KNOWLEDGE-GRAPH GENERATION

In this section, we discuss the state of knowledge-graph generation in the literature and describe how different techniques relate to **CtxKG**.

2.1 Related work

Many approaches to knowledge-graph generation from text are structured as a combination of three separate stages: entity recognition, relation extraction, and knowledge fusion [3]. Furthermore, knowledge graph completion is often added as a supplementary step [18].

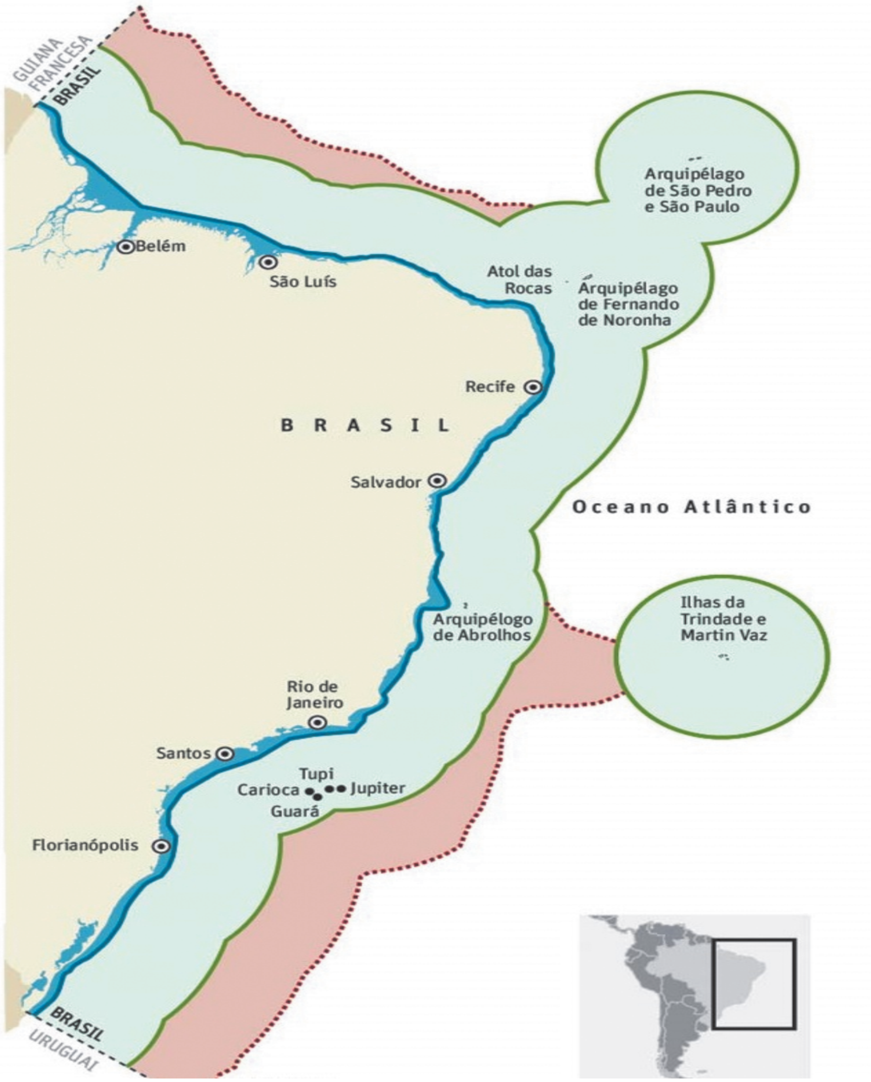


Figure 1. The Blue Amazon, which includes the territorial waters (in blue), the Exclusive Economic Zone (in green) and the continental shelf (in red). Names of locations are in Portuguese. Extracted from [53].

2.1.1 Entity recognition

The entity recognition stage is concerned with extracting the entities in a text and identifying those which are named entities. Named entities relate to specific objects, like people, organizations, or events, and are particularly useful for representing concrete and domain-specific information.

Early attempts at entity recognition used rule-based models. Essentially, they went through texts looking for predefined patterns and returned matching strings. Later, those methods were replaced by machine-learning ones, which proved more effective overall. Moreover, many modern approaches have employed, to great effect, deep-learning techniques, like those based on recurrent neural networks (RNN), especially LSTM networks, and transformers. These methods usually take advantage of large amounts of labeled and unlabeled data to train NLP models [3, 19, 20].

Peters, Ammar, Bhagavatula, *et al*. [21], for example, used bidirectional RNNs to generate embeddings that captured both semantic and syntactic information, all from unlabeled data. Those embeddings produced better results when compared to usual word embeddings. Xiao, Tan, Fan, *et al* [20], on the other hand, relied on BERT [22], one of the most popular transformer models, to encode sentences in a way that produced useful features for the other models in their named entity recognition (NER) pipeline.

2.1.2 Relation extraction

The relation extraction stage is intended to identify the relationships between entities as described in the text. A relation can be a verb that connects two entities, for example.

Similarly to the case of entity recognition, early attempts at relation extraction used rule-based models, which have been mostly replaced with machine-learning methods. These more current methods can be divided into different groups. There are the classic classification models, which can be either supervised, semi-supervised or distantly supervised, and modern deep-learning models, which make use of the aforementioned RNNs and transformers, as well as convolutional neural networks (CNN) [3, 20]. In fact, Guo, Zhang, Yang, *et al*. [23] present a system in which multiple techniques were combined to improve overall results. They utilize a bidirectional RNN to extract contextual features from the sentence tokens and combine it with a CNN and attention mechanisms in order to select the most important features for their classification system.

2.1.3 Knowledge fusion

The goal of knowledge fusion is to combine knowledge from heterogeneous sources and to weed out inconsistencies.

One of the key aspects of knowledge fusion is coreference resolution, which seeks to replace context-dependent constructions, like pronouns, with the entity they reference, so that no information is lost once the context of the text is left behind. The main techniques for coreference resolution may be divided in three groups: rule-based, statistical, and machine learning. Rule-based methods take advantage of syntactic functions and relations that can be extracted from the parse tree. Statistical methods use models based on decision trees, genetic algorithms, Bayesian statistics etc., while modern machine-learning approaches apply deep learning, using techniques like word embeddings to identify semantic similarities [3, 27]. As an example, Lee, He, Lewis, *et al*. [28] present a well-known and effective coreference resolution model that uses a CNN to build word embeddings and a LSTM network to compare them and calculate similarity scores, so as to determine whether two words refer to the same concept.

The other aspect of knowledge fusion that should also be highlighted is entity alignment, which aims to link entities from separate graphs that have the same meaning or represent the same entity. Many approaches to this problem, much like in the case coreference resolution, make use of word embeddings to find similarities between entities. For instance, Bordes, Usunier, Garcia-Duran, *et al*. [29] developed a word-embedding-based method that has been adopted and expanded upon since TransE, as seen in proposals by Chen, Tian, Yang, *et al*. [30] and Zhu, Xie, Liu, *et al*. [31]. Other recent efforts, like Wang, Lv, Lan, *et al*. [32] and Wu, Liu, Feng, *et al*. [33], make use graph convolutional networks to build the entity embeddings, as these types of network are able to take the connections in the graph into account as well.

At the same time, there are also techniques like neighborhood matching, which are able to link entities in different graphs by comparing their surroundings, i.e. the other entities connected to them and the relations described by those connections [9, 34].

2.1.4 Knowledge-graph completion

The idea behind knowledge graph completion is to generate new entities or new relationships between entities for an existing knowledge graph. Knowledge-graph completion can be divided into two groups: closed environment, which seeks to build new knowledge using strictly what is already contained in the graph, and open environment, which uses outside sources to generate new knowledge within the graph.

Similarly to previous cases, classic completion methods have used rule-based approaches, as well as statistical ones. Most modern attempts, however, tend to rely on deep learning to build meaningful embeddings for the nodes in the graphs, employing tools such as CNNs, graph neural networks (GNN), and transformer-based approaches like BERT [18, 22, 35].

Schlichtkrull, Kipf, Bloem, *et al*. [36], for instance, introduced the Relational Graph Convolutional Network, a specific type of GNN that has become a standard when it comes to entity embedding in the context of knowledge graphs. Barr, Shaw, Abu-Khzam, *et al*. [37], on the other hand, adopted a node2vec-type algorithm, which relies on random walks to define neighborhoods and generate these graph embeddings [38].

2.2 Structuring our method: alignment and extension of existing work

Our method, which we describe in detail in Section 3, is based on AutoKG [9], also a knowledge-graph generation method. Its pipeline includes, to varying degrees, all of the four tasks described in previous subsections.

For entity recognition, relationship extraction and coreference resolution, we rely on the Stanford CoreNLP library and its OpenIE implementation [39]. OpenIE is a paradigm for extracting relationship triples directly from text. Its goal is to be domain independent and to rely mostly on linguistic features to determine the components of triples (i.e. subjects, relations and objects). The original OpenIE implementation depended on different levels of syntactic analysis [40]. More recent implementations, however, have adopted deep learning techniques, using CNNs and transformers to build word embeddings, which are processed by different models to determine the role of each token in the relationship triple [41, 42]. The reason we chose to use OpenIE in our method is that we do not have a knowledge base that could be used to define our domain, so we depend solely on the contents of the texts.

Stanford's implementation was specifically chosen due to its flexibility. It allowed us to extend it in order to take advantage of sentences' parse trees and to avoid the overlap of triples typically produced by base implementations of OpenIE (see Subsection 3.1). Instead, we were able to produce a single relationship triple for each subject-object relation and to build auxiliary triples for elements such as adjectives, adverbs etc. This reduces unnecessary redundancy by ensuring that the same entity does not appear multiple times with very slight variations, while also having separate — but connected — nodes for core entities and each of their attributes.

For the tasks which require word/entity embeddings, we decided to use BERT, a language model that has become ubiquitous in NLP since its release in bert. BERT's popularity is partially due to its ability to encode tokens while taking into account the context in which they appear in the sentence, which is a product of its transformer-based architecture. This means that BERT generates different embeddings for the same word, depending on how it is used in the sentence, which is particularly useful when handling homographs [22].

One of the key uses of BERT was for entity alignment. By generating contextualized word embeddings, we were able to connect graphs from different texts (i.e. different contexts) through what we call "bridges" (see Subsection 3.4). This allows a seamless transition from one graph to another, a property that AutoKG lacks, and it does so without merging all graphs into a single one, which could result in a loss of contextual information that might have been crucial.

It is this particular combination of open information extraction techniques with word embeddings that sets **CtxKG** apart from other graph generation approaches that either rely on backing knowledge structures (e.g. ontologies) or operate within a limited set of relations [3].

3. CTXKG: A CONTEXT-BASED KNOWLEDGE-GRAPH GENERATOR

In this section we describe our new method for building knowledge graphs from text documents, named **CtxKG** (Context-Based Knowledge Graph). **CtxKG** combines a standard NLP pipeline for knowledge graph generation with a technique for connecting and merging entities that uses word embeddings [22].

CtxKG's pipeline consists of four stages, which are detailed in the following subsections:

3.1. The extraction of relationship triples from the text documents using an extension of OpenIE [40], setting up the base graphs;
3.2. The identification of synonyms among entities using BERT, connecting triples that were initially separate;
3.3. The merging of synonyms into one single entity;
3.4. The building of bridges between graphs, connecting knowledge from separate documents.

The codebase is available at github.com/Pligabue/CtxKG.

3.1 Relationship triple extraction

The first stage is responsible for extracting the triples which will ultimately be the nodes and edges of the final knowledge graphs. Because this is the only stage in direct contact with the documents, our only source of knowledge, we strive to take full advantage of the them, not only applying information extraction techniques but also relying on their syntactic structure.

The first step is to extract the base triples, which are the product of putting the text through OpenIE. To achieve this, we set up an OpenIE pipeline using Stanford's CoreNLP library [39]. The pipeline includes stages for tokenization, sentence splitting, part-of-speech (POS) tagging, lemmatization, named-entity recognition (NER), dependency parsing, and coreference resolution.

Two of those stages are not directly associated with OpenIE and were an addition we made to the basic pipeline: the coreference resolution and the NER stages. Coreference resolution was included to solve pronominal ambiguity. Several sentences, specially those which included pronouns as subjects or objects, generated triples with excessively context-dependent knowledge, which are not particularly useful for knowledge graphs. The coreference resolution replaces these pronouns in the triples with the terms they reference, producing more meaningful representations.

The NER stage is useful for a different reason. In our triple extraction process, each document is given an ID, which is then used to generate a unique ID for each entity. The purpose of these IDs is to distinguish entities with the same text but different connotations, in so far as they appear in different parts of the text or entirely different documents.

For example, two entities for the word "city", which are considered **regular entities**, will get different IDs if they originate from different documents. **Named entities**, identified by the NER stage, exceptionally receive an ID that is shared across all documents. A named entity related to the year of 2007, for instance, receives the "NE-YEAR-2007" ID, regardless of the document in which it appears, as long as it is identified as such.

The second step in the extraction of relationship triples is a step we have created called **entity unpacking**. To recognize the importance of unpacking, one should understand how CoreNLP library generates triples. For a single sentence, CoreNLP is able to generate multiple triples by including or excluding auxiliary parts of the sentence, like adjectives. For instance, given the sentence "the quick brown fox jumps over the lazy dog", CoreNLP generates six distinct triples:

⟨ quick brown fox; jumps over; lazy dog ⟩
⟨ quick brown fox; jumps over; dog ⟩
⟨ brown fox; jumps over; lazy dog ⟩

⟨ brown fox; jumps over; dog ⟩
⟨ fox; jumps over; lazy dog ⟩
⟨ fox; jumps over; dog ⟩.

As can be seen, the three different subjects are simply variations of the noun phrase “quick brown fox”, while the two objects are variations of “lazy dog”. There are two main problems with this configuration. First, the number of triples grows exponentially with the size of the noun phrases. More importantly, if interpreted directly as nodes in a graph, there would be no connection between entities like “fox” and “quick brown fox”, resulting in some loss of information.

The point of entity unpacking is identifying subsets within the pool of entities and using them as the basis for new connections. In the example above, “brown fox” is a subset of “quick brown fox” and “fox” is a subset of both “brown fox” and “quick brown fox”. By applying this logic, it is possible to arrive at the following four triples:

⟨ fox; jumps over; dog ⟩
⟨ fox; is; quick ⟩
⟨ fox; is; brown ⟩
⟨ dog; is; lazy ⟩.

The identification of the main terms (“fox” and “dog”) and of the subset relations (“is”) is done through the analysis of the dependency tree (Fig. 2), which is generated in the dependency parsing stage of the CoreNLP pipeline. Both “fox” and “dog” are at the top of the tree associated with the phrasal verb “jumps over”, while the adjectives function as adjectival modifiers to the main terms, which can be represented by the verb “is” in the new triples [43].

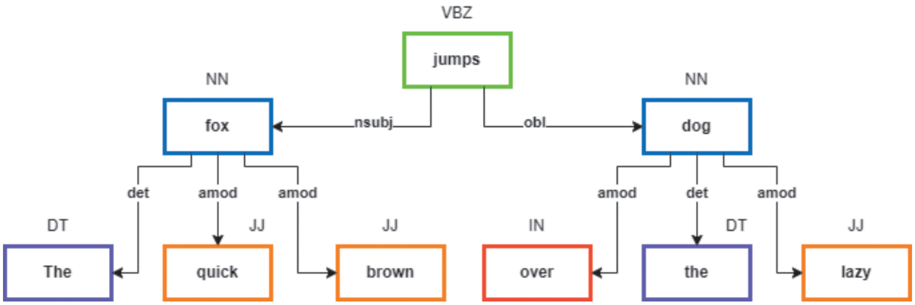


Figure 2. Visual representation of the dependency tree for the sentence “the quick brown fox jumps over the lazy dog.” The relations (e.g. “amod” for adjectival phrases, “det” for determiners etc.) are defined in [11], while the POS tags follow the standard described in [54].

After this extended OpenIE stage, each document has a set of triples that operates as the basis for the knowledge graph. The subjects and objects are the nodes, while the relationships are the edges.

3.2 Synonym identification

The second stage of the knowledge graph generation pipeline is the identification of synonyms within documents. Because there is no background knowledge base to provide such semantic information, we rely on contextualized word embeddings to introduce semantics and compare the meanings of different words.

For each document, each triple, formatted as a sentence, is put through BERT [10]. Then, for each triple embedding, the embeddings for the subject entity and the object entity are calculated by averaging the embeddings of the tokens that constitute them.

After entity embeddings are calculated, we compare them in terms of their cosine similarity. If the cosine similarity is above a certain predefined threshold—usually set between 0.8 and 0.9—the two entities are considered synonyms.

To illustrate, consider the text "The fox jumped over the dog. The dog chased the fox. The fox is quick." and the following set of extracted triples:

⟨ fox; jumped over; dog ⟩
⟨ dog; chased; fox ⟩
⟨ fox; is; quick ⟩.

Ideally, the word embeddings for all three instances of "fox" would be similar. Then, by comparing these embeddings, the three instances would be linked, even though they appear in different sentences, both as the subject and as the object.

After this stage, each document has a base version of the graph, which includes all the relations from the previous stage along the new synonym-link relations.

3.3 Graph reduction

The graph reduction stage complements the synonym identification by merging synonyms into one single final entity. From this point forward, we are extending AutoKG's pipeline, as synonym identification is AutoKG's final knowledge-graph generation step.

This is a more straightforward stage, in which the linked entities are compared in terms of the number of occurrences and the most recurring one is chosen as the one that will represent all the synonyms in the final graph. The only exception is named entities, which are always chosen over regular entities, even if they appear fewer times.

After this stage, each document has a clean version of the graph, in which each synonym group has been merged into one single entity. As a consequence, the synonym-link relations are no longer necessary.

3.4 Building bridges

The goal at this stage of the pipeline is to establish connections between the individual graphs generated in the previous stages. Up to now, documents have been treated as independent units. This, however, prevents information sharing among them. To overcome this shortcoming, we establish points of contact between the different graphs, which we call "bridges".

The process of building bridges is analogous to the construction of synonym links (see Subsection 3.2). Entities have their embeddings calculated by feeding the triples to BERT and the comparison is made using cosine similarity. Entities from separate texts that have a high similarity are considered related and a bridge is established. There are a few key differences in this process, though, compared to the synonym stage:

1. The entities of a text are compared to all entities of all other texts. This results in many more comparisons. To illustrate this point, with N documents, each with a set of M different entities, the synonym identification stage would need to make a total of

   $$N \cdot \frac{M(M-1)}{2!}$$

   comparisons, while this stage would need to make a total of

   $$\frac{N(N-1)}{2!} \cdot \frac{M(M-1)}{2!}$$

   comparisons. This difference causes this stage to be the most costly processing-wise.

2. The similarity threshold can be made lower at this stage, as it is not necessary for the entities on either side of the bridge to be treated as strict synonyms. In our experiments, the best results were achieved with a threshold of around 0.7 .
3. The number of connections per entity is limited to one. The idea behind this is to compensate the leniency of the lowered threshold by limiting the connections.
4. If two named entities have the same ID, indicating they refer to the same concept, they are connected.

After this stage, each document has a final individual graph, along with a set of bridges that connect it to other graphs. In this configuration, the graphs represent the knowledge in the texts, while the bridges provide alternatives should the sought-after piece of information not be present in the current graph.

3.5 Knowledge graph example

As a short example, Fig. 3 displays the graph generated from the following sentence:

> *BYD debuted its E-SEED GT concept car and Song Pro SUV alongside its all-new e-series models at the Shanghai International Automobile Industry Exhibition. The company also showcased its latest Dynasty series of vehicles, which were recently unveiled at the company's spring product launch in Beijing.*

This is the same sentence used in the showcase section of the 2019 ICDM/ICBK Contest [3]. The main highlights for our graph, compared to others in the contest, are: the ability to identify and isolate the main

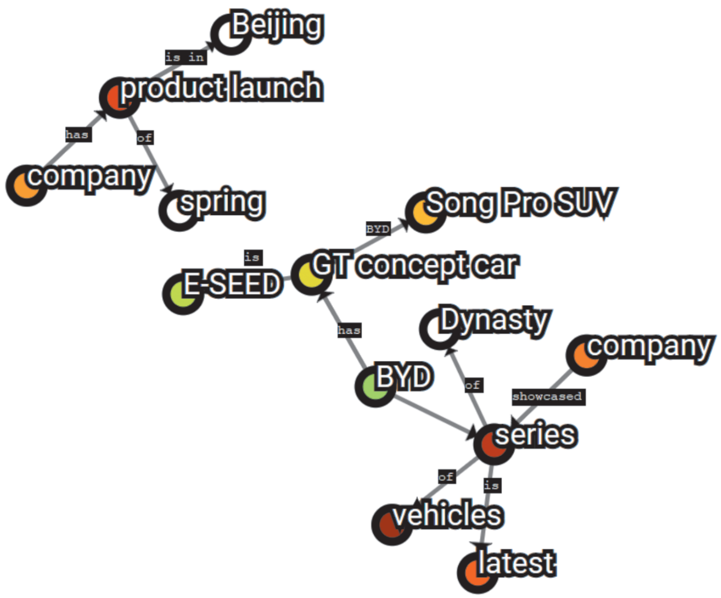


Figure 3. Sample knowledge graph created using our method. The input sentence was the same sequence used in the 2019 ICDM/ICBK knowledge-graph generation contest [3].

named entities (e.g. "Beijing", "Song Pro SUV" and "BYD") and the ability to unpack long sentences and produce more nuclear entities (e.g. *"the company's spring product launch in Beijing"* becomes four separate entities, "company", "spring", "product launch" and "Beijing").

4. EVALUATION OF CTXKG

Since **CtxKG** extends AutoKG [9], we used the same type of documents to evaluate our method, i.e. entries of movies on Wikipedia. More specifically, the complete articles for the 200 highest-grossing films were used to build the knowledge graphs.

The pages were fetched using MediaWiki's API [44] and the HTML was parsed to select the relevant sections: the summary at the beginning of the page, the description box that appears to the right of the summary, and the whole body of the article up to the reference section. The only parts excluded were items like tables and lists, which are not in ordinary text form.

Besides **CtxKG**, we also report the results for AutoKG and Miller *et al*. [12]. Because the codebases/repositories for these methods are not publicly available, we had to stick to the numbers and metrics provided in the original papers. Despite not being able to run these methods with the exact same dataset, we believe the reported numbers may provide an approximate measurement of how well **CtxKG** performs compared to those and other similar methods.

Table 1. List of abbreviations.

Abbreviations	
API	Application programming interface
BERT	Bidirectional Encoder Representations from Transformers, a language model
BLAB	Blue Amazon Brain, our project where we aim to build a conversational agent for the Blue Amazon
BlabKG	BLAB Knowledge Graph, our knowledge graph about the Blue Amazon
CNN	Convolutional neural network
CtxKG	Context-Based Knowledge Graph, our knowledge graph generation method
FPSO	Floating production storage and offloading, a type of vessel for oil extraction in the ocean
GNN	Graph neural network
LSTM	Long short-term memory, a type of neural network
NE-NE	Named entity to named entity, one of the three types of entity links
NE-RE	Named entity to regular entity, one of the three types of entity links
NER	Named-entity recognition
NLP	Natural language processing
POS	Part-of-speech, as in part-of-speech tagging
QA	Question answering
RE-RE	Regular entity to regular entity, one of the three types of entity links
RNN	Recurrent neural network

Table 2. List of notations.

Notations	
⟨ SUBJ; $\underline{\text{REL}}$; OBJ ⟩	A relationship triple, containing a subject, a relation and an object, in that order. The subject and the object are entities. The triple is enclosed by angle brackets (plus a space) and its three components are separated by semicolons. The relation is underlined.

4.1 Methodology

Graphs

The idea is to compare the generated graphs to benchmark graphs and determine the percentage of triples from the benchmark graphs that are present in the generated ones. There are four aspects of this evaluation method that ought to be highlighted:

Benchmark triples The benchmark triples are built on metadata about each movie from IMDb, made available via the OMDb API, combined with tags from MovieLens [45, 46]. The attributes that were included in Miller *et al*. [12], which is the basis for AutoKG, are: (1) the directors; (2) the writers; (3) the actors; (4) the release year; (5) the languages; (6) the genres; (7) the tags; (8) the IMDb rating; and (9) the IMDb votes.

Coverage Just as in AutoKG, a benchmark triple is considered covered if its object is included in any of the generated triples [9]. The subject of the benchmark triples is disregarded because it is always the title of the movie.

Coverage percentage While Miller *et al*. [12] built one benchmark triple for each of the nine attributes by grouping entities that have the same relationship to the movie (e.g. the triple "Blade Runner *starred actors* Harrison Ford, Sean Young, ..." includes all the actors in the movie in the object, separated by commas), **CtxKG** produces one triple per entity. To handle this difference in approach, the contribution of a given **CtxKG** triple to the overall hit percentage is defined as its contribution to the specific attribute divided by the total number of attributes. For example, if all nine attributes are being considered and the movie stars 12 actors, having a triple for one actor counts as

$$\frac{1}{12} \cdot \frac{1}{9} = \frac{1}{108} \approx 0.926\%.$$

Comparison AutoKG separates the benchmark triples into two groups: **reachable triples** (those that contain knowledge found in the *corpus*) and **unreachable triples** (those that contain knowledge that is not in the *corpus*). However, there is no clear definition of how the coverage of unreachable triples is calculated, which is specially important considering attributes like tags have variable length, that is, there is no limit to the number of tags a movie can have on MovieLens. For those reasons, our comparison will focus only on reachable triples, i.e. attributes (1) through (4), as these are the attributes that can be found reliably on Wikipedia articles [9].

Bridges

The evaluation of bridges was done manually, as bridges are not present in AutoKG and, therefore, cannot be directly compared like the graphs can. They were categorized according to the presence of named entities, since this is the most reliable type of entity, and evaluated in terms of semantic coherence. This means analyzing the relationship between the two entities that form a bridge and verifying the two graphs are connected in a way that would assist graph traversal.

For performance reasons, the bridges were built from just the summary sections of the Wikipedia articles, and not the entire pages, so as not to require the comparison of an extremely large number of entities (as mentioned in Subsection 3.4).

4.2 Results

As previously mentioned, the complete Wikipedia articles for the 200 highest-grossing films were used for the evaluation of **CtxKG**. The results are presented below.

Graphs

Focusing just on the reachable triples (Tables 3 and 4), it seems that they were reliably generated by **CtxKG**. The most elusive attribute was that of "writers". Not only were they found in fewer cases—only 84.17% of them—they also presented the highest standard deviation at 27.82. The most present attribute was the "year", which was found 98.99% of the time, with a standard deviation of 10.00. This matches what can be expected from a Wikipedia article about a movie, as directors, actors and the year tend to be mentioned more frequently than writers.

Table 3. Comparison between different methods. The four displayed attributes are the ones categorized as "reachable", meaning they can be reliably found on the articles used to generate the knowledge graphs. The individual values for miller-etal-2016-key and autokg are not included, as only average values were discussed in either paper.

Method	Reachable triples				Avg. (%)
	Directors	Writers	Actors	Year	
Miller *et al.*	–	–	–	–	83.69
AutoKG	–	–	–	–	90.00
CtxKG	93.89	84.17	93.47	98.99	92.63

Table 4. Average coverage and standard deviation for the reachable attributes in Table 3.

Attribute	Coverage (%)	SD
Directors	93.89	22.48
Writers	84.17	27.82
Actors	93.47	19.15
Year	98.99	10.00
Avg.	**92.63**	**15.13**

In total, **CtxKG** achieved a coverage of 92.63%. Moreover, Fig. 4 seems to indicate a slight correlation between the coverage percentage of reachable triples and the size of the Wikipedia article, suggesting that, after around 5,000 words, the coverage fluctuates between 90% and 100% for the most part. By comparison, when considering only reachable triples, AutoKG and Miller *et al*. [12] have a coverage of 90.00% and 83.69%, respectively. It seems, then, that **CtxKG** has a slight edge when it comes to reachable triples in the context of the selected documents.

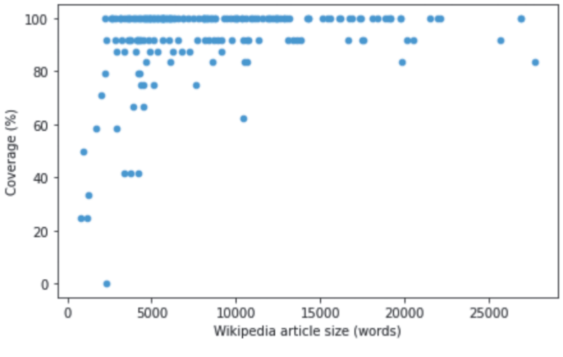


Figure 4. The coverage of reachable attributes (1 through 4) versus document size. After around 5,000 words, the coverage fluctuates between 90% and 100% for the most part.

The main drawback of the **CtxKG** lies in the extraction of triples from texts using the CoreNLP library [39]. While the additions of NER and the entity unpacking described in Section 3.1 seem to be working reasonably well, the initial triples, which exist before these two additional steps, are generally lacking, as exemplified in Table 5.

Table 5. Common issues with generated triples, taken from a graph generated for one of the movies used during evaluation: Spider-Man (2002).

	Subject	Relation	Object	Issue
(1)	Man	is	superhero film	This is a problem that occurs when CoreNLP defines a second entity with just part of the tokens of the real, expected entity. In this case, the original version of (1) had "Spider Man" as the subject. However, because (2) was also created, the entity unpacking detected "Man" as a subset of "Spider Man", changed the subject of (1) and created (3).
(2)	Man	is American	character	
(3)	Man	of	Spider	
(1)	Willem Dafoe	is	Kirsten Dunst	This is an issue specific to the CoreNLP library, which fails to identify comma-separated lists as such. It instead creates triples with the verb "is" connecting the items of the list two by two.
(2)	Tim Burton	is	Michael Bay	
(1)	superhero film	amount to	2002	This is an open issue with the unpacking, which is about the selection of the relationship text for a given derived triple. In the case of (1), the relation between "superhero film" and "2002" is of the type **nummod** (numeric modifier), which is mainly used to associate numbers with nouns, usually denoting quantity. However, since "2002" is a year, the default text for that relation, "amount to", does not actually make sense in context. This is also partially a product of the lack alternative to the **nummod** relation in this library, as a year is not quite a quantification [11].

Bridges

From the 200 movie summaries, a total of 134,638 bridges were generated, including 7,690 different entities. These bridges can be separated in three different categories, in terms of how they relate to named entities:

1. **NE-NE**: connects a named entity to a named entity. They amount to **52,322** (38.86%) bridges;
2. **NE-RE**: connects a named entity to a regular entity. They amount to **3,016** (2.24%) bridges;
3. **RE-RE**: connects a regular entity to a regular entity. They amount to **79,300** (58.90%) bridges.

As for the **NE-NE** category, out of the 52,322 bridges, 52,070 of them (99.52%) connect two different named entities of the same type and 33,799 of them (64,60%) connect the exact same entity (identified by the unique ID) in distinct documents, the latter being the best scenario. At the same time, the fact that 35.40% of the bridges connect different named entities indicates there is likely room for improvement in this area: though the goal of the bridges is not to connect exact synonyms but to link entities that are related, this 65/35 distribution, combined with the inspection of some samples, indicates that part of those connections were not warranted. A recurring case was that of actors being linked because they had similar names, even though there was no real connection between them and their work.

The 0.48% of **NE-NE** bridges that connect entities of different types represent a subgroup which may be discarded in future versions of the pipeline, as it is unlikely that they establish meaningful or semantically useful connections, considering not only that the entities are different, but also that their categories do not match.

Regarding the types of entities in **NE-NE** bridges, the majority referenced nationalities (21,780), numbers (18,473) and dates (16,573). The rest was more evenly spread among other types (countries, organizations, people etc.).

In terms of the **NE-RE** bridges, although there are expected connections between related concepts (e.g. the named entity "Spider-Man" and the regular entity "Spider-Man trilogy"), a relevant chunk of them (23,21%) is just connecting a named entity to a regular version of that named entity, which was not correctly identified in the context of its document (e.g. one bridge connected the named entity "Disney", which is an organization, to a regular entity "Disney", which was not identified as a named entity in the triple extraction stage and thus received a regular ID). This is further evidence of the critical importance of NER when it comes to knowledge-graph generation.

In the case of the **RE-RE** bridges, many of them connected the same concept across different graphs. For example, many regular entities that represent the concept "film" in their respective documents are connected to each other, adding up to around 22,294 bridges just for this one term. Many adjectives also display this pattern (e.g. "positive", with around 1,485 bridges, and "success", with around 480). The main issue here is the same as for NE-RE bridges, i.e. the lack of identification of named entities.

5. BLABKG

With **CtxKG** as our knowledge-graph generation method, we now move to the second part of our work, in which we describe **BlabKG**, a knowledge graph for the Blue Amazon. In Subsection 5.1 we analyze the *corpus* which served as the input for **CtxKG** (available at github.com/C4AI/Pira), while in Subsection 5.2 we discuss the final graph (available at github.com/Pligabue/CtxKG).

5.1 Corpus

The *corpus* used to generate the knowledge graphs consists of 496 scientific paper abstracts in the domain of the Blue Amazon. Those abstracts were taken from Pirá, a reading comprehension dataset developed for this specific domain, also made in the context of BLAB [32].

Key information about the documents of the corpus can be seen in Tables 6, 7 and 8, including statistics about the *corpus*, the most prominent topics and the most recurring words, respectively .

Table 6. Document dimensions. Since the documents are research paper abstracts, it is expected that the average word count would fall near the 200 mark.

Metric	Avg.	SD
Word count	231.84	93.83
Sentence count	9.54	4.38
Sentence size (words)	24.39	11.70

Table 7. Subjects broached in the documents. The categorization was done manually by the BLAB team. As one could expect, natural resources, due to their economic importance, represent the biggest share of documents.

Category	#	%
Natural Resources & Extraction	196	39.52
Geology	84	16.94
Environment & Pollution	60	12.10
Oceanography	59	11.90
Business & Economy	32	6.45
Biology	25	5.04
Climate & Climate change	17	3.43
Logistics	16	3.23
Wind & Wave power	5	1.01
Territory & Security	2	0.40
Total	**496**	**100**

From those data we can extract the following:

1. The *corpus* itself is not very large, amounting to around 4,700 sentences. On top of that, part of those sentences are not necessarily relevant, since the documents are abstracts, which have specific format requirements. That is why the words "paper" and "study" appear so many times (table 8). This will inevitably take a toll on the quality of the knowledge graphs, at least in terms of containing irrelevant data. More on that in Subsection 5.3.
2. Oil and gas extraction is the main subject when it comes to this *corpus*. It represents approximately 39.52% of the documents (Table 7) and is directly associated with eight out of the 20 key recurring words (Table 8).

Samples

The following are some samples from Pirá [32]. We selected two abstracts which demonstrate the kinds of texts that served as input for **BlabKG**:

Abstract 1 This abstract is taken from "Extracting full-resolution models from seismic data to minimize systematic errors in inversion: Method and examples" [48].

> *Creating an accurate subsurface model is paramount to many* **geophysical and geological workflows**. *Examples are background models for seismic inversion, rock property models for reservoir characterization,*

Table 8. The 20 key recurring words (nouns and adjectives). These are words that have the potential to become entities in the knowledge graphs. For this reason, this table is one of the main points of comparison for similar tables that appear later on.

Noun	Description	#
Brazil	–	311
Petrobras	The state-owned Brazilian oil company.	201
Brazilian	–	192
Oil	Refers to petroleum.	192
Production	Refers to oil production.	176
Basin	Regions covered by seawater.	167
Water	–	163
Offshore	Refers to offshore drilling to extract oil.	153
Paper	Is mentioned in many scientific paper abstracts.	152
Study	Is mentioned in many scientific paper abstracts.	149
System	–	137
Development	–	126
Field	Usually referring to oil fields.	121
Gas	Refers to natural gas.	120
Analysis	Is mentioned in many scientific paper abstracts.	115
Campos	One of the basins in the Blue Amazon.	94
Sea	–	70
Santos	The city of Santos, in the state of São Paulo.	68
Reservoir	Refers to oil reservoirs.	52
Drilling	Refers to oil extraction.	51

> *and geological models of depositional elements for seismic morphological interpretation. The standard workflow for creating subsurface models using seismic data is* **stratal slicing**. *The stratal slicing approach, however, may break down in the case of complex stratigraphic or tectonic structuring, such as* **shelf-to-basin clinoforms**, **delta lobe switching**, **deep-water channel-fan complexes**, *and deformation due to salt tectonics. This paper illustrates how the results obtained with high-resolution inversion and the incorporation of a stratigraphically consistent low-frequency model generated through* **horizon mapping** *- called the HorizonCube - improves the quality of the estimation of the subsurface parameters in structural complex settings. Using two data examples with different seismic data and geological settings from the North Sea and offshore Brazil, the paper will demonstrate the increased accuracy of the final inversion result using a data-driven HorizonCube.*

Abstract 2 This abstract is taken from *Deepwater Installation of a Large Capacity FPSO with Large Number of Risers in the Marlim Field* [49]

> This paper describes the site installation of a turret moored Floating Production, Storage and Offloading System-FPSO-in 780 *meters of water in Campos Basin, offshore Brazil. The FPSO, a 270,000 dwt converted* **tanker**, *is the first of a series of two ordered by Petrobras for development of the Marlim Field. An* **internal bow mounted Turret system**, *anchored to the seafloor by 8 chain-wire rope-chain combined mooring legs, is used to permanently moor the FPSO in the location while allowing the vessel to freely weathervane. Thirty-five* **flexible risers**, *laid in a* **free-hanging catenary configuration**, *provide the flow path between the*

> *FPSO and the various subsea equipment on the seafloor. This paper describes the installation equipment and procedures employed.*

The underlined expressions (e.g. "this paper describes") exemplify some of the constructions that are present in most abstracts, since they are a staple of the research paper abstract format. They represent parts of the texts which may not generate particularly useful information for the knowledge graph, as it is generally just boilerplate text.

The bold expressions, however, represent very specific knowledge that exemplifies the type and the complexity of the content we encountered when building **BlabKG**. Terms like "FPSO" and "flexible risers" appear especially in the context of oil and natural gas extraction, which is the main economic activity in the Blue Amazon, while expressions like "geological workflows" and "clinoforms" appear in the context of geology, which is also the subject of a large percentage of research papers on the Blue Amazon.

5.2 Graphs

After running the graph generation pipeline for the *corpus*, a total of **496** graphs were generated, one for each document, and **348,033** bridges were established between those graphs. Together, they make up **BlabKG**.

Overall statistics for BlabKG can be seen in Table 9. They show that, with an average of around 54 entities and 52 relationship triples per document, each entity appears on roughly two different triples. Another data point worth highlighting is that named entities represent around 10% of all entities .

Table 9. Graph dimensions.

Metric	Total	Avg.	SD
Entity count	26,726	53.88	23.21
Named entity count	2,752	5.55	4.02
Relationship triple count	25,779	51.97	23.82
Synonym count	1,008	2.03	2.98

As to the retainment of information from the documents, a comparison between Tables 10 and 8 shows there is an overlap between the recurring nouns in the *corpus* and the recurring entities in the graphs, with named entities like "Brazil" appearing at the top of both lists. The number of occurrences, on the other hand, is not the same in both lists. The entity "Petrobras", for example, appeared in 201 documents and in 155 graphs. This difference suggests that there is still room for improvement when it comes to CtxKG's information extraction stage (see Subsection 3.1).

Table 11 complements Table 10, describing regular entities that appear in multiple graphs, part of which also appear on Table 8 (e.g. "paper" and "oil").

Table 10. The 20 key recurring named entities appearing in the generated knowledge graphs.

Named entity	Description	#
Brazil	–	185
Petrobras	The state-owned Brazilian oil company.	155
Brazilian	–	128
Basin	Regions covered by seawater.	98
Campos	One of the basins in the Blue Amazon.	78
Santos	The city of Santos, in the state of São Paulo.	44
Rio de Janeiro	The city of Rio de Janeiro.	26
FPSO	An offshore oil extraction vessel.	26
Atlantic Ocean	–	23
America	–	14
Gulf of Mexico	–	13
Guanabara Bay	Oceanic bay in the state of Rio de Janeiro.	13
North Sea	–	7
Bahia	Coastal state in the northeast of Brazil.	6
Cretaceous	Refers to the Cretaceous period.	6
Santa Catarina	Coastal state in the south of Brazil.	5
Pre-salt	Rock layer where a large oil reservoir sits.	5
Shell	Oil and gas company.	5
Marlim field	Oil field in the Campos basin.	5
Estuary	Region where rivers meet the sea.	4

Table 11. The 20 key recurring regular entities appearing in the generated knowledge graphs.

Named entity	Description	#
Offshore	Refers to offshore drilling to extract oil.	136
Paper	Is mentioned in many abstracts.	135
Results	Used in many different contexts.	102
Study	Usually mentioned in abstracts.	91
Data	–	79
Oil	Refers to petroleum.	68
Analysis	Is mentioned in many abstracts.	61
Production	Refers to oil production.	60
Water	–	50
Work	–	49
Field	Usually referring to oil fields.	47
Environmental	–	43
Methodology	Is mentioned in many abstracts.	40
Technology	–	39
Fields	Usually referring to oil fields.	39
Project	–	36
Basin	Regions covered by seawater.	35
Approach	Is mentioned mainly in abstracts.	34
Waters	–	31
Sediments	–	31

Table 12 shows the occurrences of each type of named entity identified by the named entity recognition step. Organizations are the most present type of named entity by a substantial margin. This is likely by virtue of the fact that a large portion of the documents, as shown in Table 7, are about oil extraction, which often includes mentions of companies like Petrobras or Shell. However, there are also some cases of misidentifications. For example, the FPSO oil extraction vessel was identified 26 times as an organization, even though "miscellaneous" is a better fit.

Table 12. Occurrences of each named entity type[a] in the generated knowledge graphs.

Type	#	Type	#
Organization	553	Percentage	56
Location	418	Money	23
Date	307	Cause of death	20
Title/position	273	Set	21
Country	250	State or province	17
City	242	Time	5
Nationality	151	Criminal charge	3
Misc.	140	Ideology	3
Person	113	Religion	3
Ordinal	83	URL	1
Duration	70		
Total: 2,572			

[a] These are types that are available in CoreNLP [39].

Another relevant group of named entities is the one associated with places, which includes the location, country, city, and state types. Because our domain is the Blue Amazon, there are several mentions of Brazil, of coastal states like Bahia and São Paulo, and of coastal cities like Rio de Janeiro, Santos and Salvador. Those entities serve as good bridges, connecting knowledge graphs through the locations they describe.

5.3 Bridges

To evaluate the **348,033** bridges, the relation between the entities on each end of the bridges must be validated. To do this, the bridges may first be separated into the three groups described in Subsection 4.2:

1. **NE-NE**: amount to **61,835** bridges;
2. **NE-RE**: amount to **15,010** bridges;
3. **RE-RE**: amount to **271,188** bridges.

In the case of NE-NE bridges, the evaluation is more straightforward, as bridges are built when nodes in different graphs relate to the same named entity. The entity "Petrobras", for example, serves as a bridge between multiple graphs.

Table 13 lists the key recurring named entities that appear as bridges. As expected, the list of named entities overlaps considerably with Table 8 and Table 10, indicating that the key concepts from Table 6 are making it through the pipeline and becoming key entities in the graph bridges.

Table 13. The 20 key named entities used in NE-NE bridges.

Named entity	Description	#
Brazil	–	35,833
Petrobras	The state-owned Brazilian oil company.	24,639
Brazilian	–	18,032
Basin	Regions covered by seawater.	9,696
Campos	One of the basins in the Blue Amazon.	7,187
Santos	The city of Santos.	3,562
Pre salt	Rock layer with a large oil resevoir.	642
Rio de Janeiro	The city of Rio de Janeiro.	705
FPSO	An offshore oil extraction vessel.	690
Estuary	Region where rivers meet the sea.	213
Atlantic Ocean	–	417
Mexico	–	225
Gulf of Mexico	–	139
Cretaceous	Refers to the Cretaceous period.	177
Argentina	–	107
Venezuela	–	79
Guanabara Bay	Bay in the state of Rio de Janeiro.	77
Bahia	Coastal state in the northeast of Brazil.	82
North Sea	–	75
São Paulo	The city of São Paulo.	60

Table 14 describes the types of named entities that are included in these bridges. Compared to the values in table 12, it is evident that country and organization now lead by a much larger margin. This is due to the fact that "Brazil" (country) and "Petrobras" (organization) are predominant entities in the *corpus*.

Table 14. Types of named entities in NE-NE bridges.

Type	#	Type	#
Country	36,564	Misc.	275
Organization	27,128	Duration	229
Nationality	18,276	Set	134
City	12,050	Money	124
Location	11,752	State or province	61
Title/position	7,790	Cause of death	29
Ordinal	4,147	Religion	6
Date	3,250	Ideology	4
Percentage	1,126	Criminal charge	2
Person	723		
Total: 123,670			

Table 15 provides general statistics about the NE-NE bridges. The majority, around 80.80% of them, connect instances of the same named entity, which is the ideal scenario, while approximately 13.77% of them connect different entities of the same type, which is acceptable. The remaining 5.43% of them connect entities of different types, which is generally undesirable, in so far as the linked entities are unlikely to actually be related, as mentioned in Subsection 4.2.

Table 15. General properties about NE-NE bridges.

Property	#
Connects the same named entity	49,958
Connects different named entities of the same type	8,519
Connects different named entities of different types	3,358
Total	**61,835**

With respect to the NE-RE group, which consists of bridges connecting named entities to regular entities, to assess their quality, we verified which of these three scenarios is prevalent:

1. The regular entity is actually the named entity, but it was not identified as such during the first stage;
2. The regular entity is not the named entity, but it is related to the named entity, making the connection warranted;
3. The regular entity is unrelated to the named entity, which is the least desirable scenario.

Table 16 shows that the textual content of the named entity and of the regular entity do not usually match, so the first scenario is unlikely. Furthermore, in only about a third of bridges the regular entity contains the named entity, making the second scenario also unlikely. The third scenario seems to represent the majority of NE-RE bridges.

Table 16. Matching texts in NE-RE bridges.

Property	#
Named entity and regular entity have the same text	2,780
Named entity and regular entity have different texts	12,230
Total (exact match)	**15,010**
Regular entity contains the named entity	4,979
Regular entity does not contain the named entity	10,031
Total (contains)	**15,010**

These three scenarios can be seen in Table 17. The entity "basin" represents the first two, with most of its bridges being desirable, usually connecting a named entity version to a regular entity version. The entity "marine", however, represents the problematic third scenario and is the reason for the substantial prevalence of the title/position type in Table 18. In most cases, it should not have been identified as a named entity, as if it were referring to the navy, and many of its bridges include unrelated entities (e.g., "geologic", "tubular"). This indicates that it may be beneficial to drop bridges that represent this third scenario.

Table 17. The 20 key named entities that appear in NE-RE bridges.

Named entity	Description	#
Marine	–	2,723
Basin	Regions covered by seawater.	2,270
Petrobras	The state-owned Brazilian oil company.	891
Model	–	667
Pre salt	Rock layer with a large oil resevoir.	543
Brazilian	–	540
Campos	One of the basins in the Blue Amazon.	452
Present	–	371
Bay	–	332
Recently	–	326
Current	–	323
First	–	300
FPSO	An offshore oil extraction vessel.	256
Currently	–	194
Santos	The city of Santos.	187
Layer	Referring to rock layers.	161
Cretaceous	Refers to the Cretaceous period.	154
River	–	154
Annual	–	152
Climatic	–	146

Table 18. Types of named entities in NE-RE bridges.

Type	#	Type	#
Title/position	4,371	Person	92
Location	3,341	Duration	92
Date	1,740	Money	61
Organization	1,733	Cause of death	52
City	1,104	Percentage	19
Misc.	945	State or province	6
Nationality	604	Religion	5
Ordinal	414	Ideology	2
Set	329	Time	2
Country	98		
Total: 15,010			

The third and final group, which consists of bridges connecting regular entities to other regular entities (RE-RE), is the largest one, amounting to 77.92% of the total bridges. At this point, some of the words in Table 8 which do not refer named entities appear again, such as "offshore" and "paper". At the same time, as Table 20 shows, part of the key recurring entities is comprised of very context-dependent or semantically meaningless entities, such as pronouns (e.g. "we") and functional words (e.g. "to"). Ideally, such entities ought to be removed during the knowledge-graph generation process, as they do not add any useful information, they appear in a large number of documents and they end up cluttering the bridge building.

Table 19. General properties about RE-RE bridges.

Property	#
Connects regular entities with the same text	91,355
Connects regular entities with different texts	179,833
Total (match)	**271,188**
One regular entity contains the other	102,896
No regular entity contains the other	168,292
Total (contains)	**271,188**

Table 20. The 15 key entities that appear in RE-RE bridges.

Entity	Description	#
Offshore	Refers to offshore drilling to extract oil.	11,663
To	–	9,962
Paper	Is mentioned in many abstracts.	9,925
Seismic	Related to earthquakes.	9,431
High	–	8,257
New	–	7,330
We	–	7,136
Main	–	7,042
Coastal	Related to the coast.	4,788
Also	–	4,267
Environmental	–	4,131
Several	–	4,056
Study	Is mentioned in many abstracts.	4,048
Continental	–	3,906
Well	Both as an adverb and to refer to oil wells	3,642

Another property of RE-RE bridges can be seen in Table 19. It shows that, like in the case of NE-RE bridges, the pattern of one of the entities being equal to the other or containing the other is not seen in most of the cases. The segment of bridges that do not hold any of the two properties, which amounts to 62.06% of the RE-RE bridges, is the one that should be treated with the most caution, as it is the least likely to hold any semantic meaning.

The most interesting aspect of RE-RE bridges, however, relates to the number of bridges which connect the same regular entity across different graphs, as seen in Table 21. In the case of "paper", for example, out of the 5,231 bridges that include it, 4,694 of them connect two instances of "paper" from different graphs. These can be good connections, as the uses of the term "paper" are usually similar enough that these bridges make semantic sense. If one wanted to answer a question such as "What are the subjects broached by papers on the Blue Amazon?" using these graphs, the best answers would probably come from traversing the graphs using these bridges. The flip side is that unwanted entities like "we" also end up appearing in the list with a high rate of self-connection, once again pointing to the need to remove them early on in the process.

Table 21. Self-connections for the 15 key entities in RE-RE bridges (Table 20). Self-connections are those that connect two instances of the same regular entities in different graphs.

Entity	%[a]	Entity	%[a]
Offshore	48.23	Coastal	18.60
To	99.92	Also	49.46
Paper	89.73	Environmental	26.37
Seismic	7.96	Several	33.16
High	55.59	Study	58.43
New	81.08	Continental	21.99
We	99.11	Well	34.24
Main	37.16		

[a] Percentage out of all bridges with this entity.

After evaluating the three groups, it seems that the NE-NE bridges are definitely the more reliable one. In fact, when traversing the knowledge graphs, it might be safer to just ignore regular entity bridges until their consistency can be improved.

5.4 Selected excerpts

To better illustrate our results, we present snippets of graphs and triples extracted from some of the abstracts in the corpus.

5.4.1 Graph overview

These first examples use the abstract from "Petrography, geochemistry and origin of South Atlantic evaporites: The Brazilian side" [50].

Fig. 6 displays a section of a knowledge graph generated mainly from the first sentence in the abstract, which reads "*The discovery and production, by Petrobras, of over 50 billion barrels in place of pre-salt oil in Brazil's offshore South Atlantic Santos and Campos basins has drawn worldwide attention to its km-thick Cretaceous salt seal since 2007*". As can be verified, the key entities in the text were identified, including named entities (in white) such as "Petrobras", "Brazil", "South Atlantic" and "Campos".

Fig. 7 represents a passage that appears later in the abstract, which reads "*Aptian volcanic activity in the South Atlantic formed the Rio Grande Rise - Walvis Ridge that was the southern barrier of the salt basin*" [39]. Like the previous passage, it mentions the "South Atlantic". Because of that, the two passages became connected in the graph through the "South Atlantic" entity (right of Figure 6 and left of Figure 7).

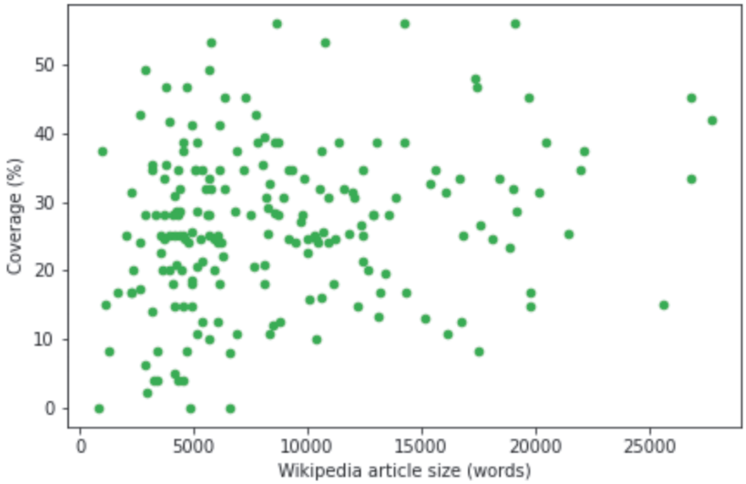


Figure 5. Coverage of unreachable attributes (5 through 9) versus document size. No correlation appears to exist between document size and coverage.

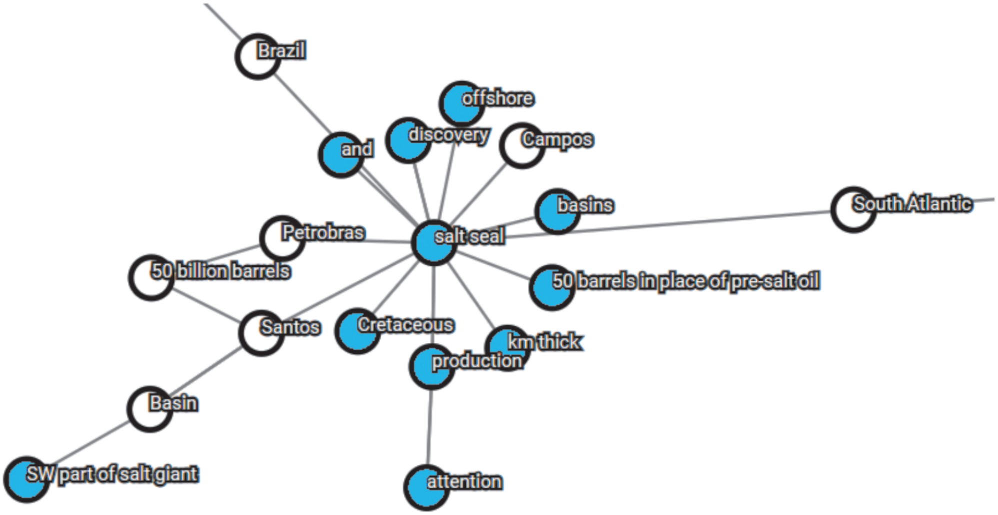


Figure 6. Section of the knowledge graph generated from the abstract of [50].

Fig. 8 and Fig. 9 represent how bridges can work. In the initial state (Fig. 8), all entities originate from the same graph. By exploring two bridges for the "Brazil" entity, a new set of entities is added to the graph, as indicated by their different colors[1] (Fig. 9).

[1] With the exception of named entities, which are always white.

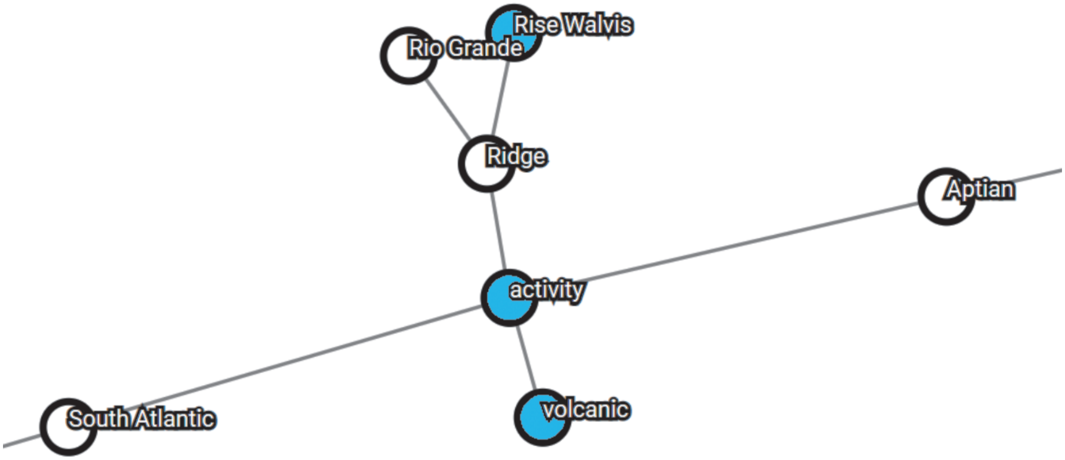


Figure 7. Other section of the knowledge graph generated from the abstract of [50].

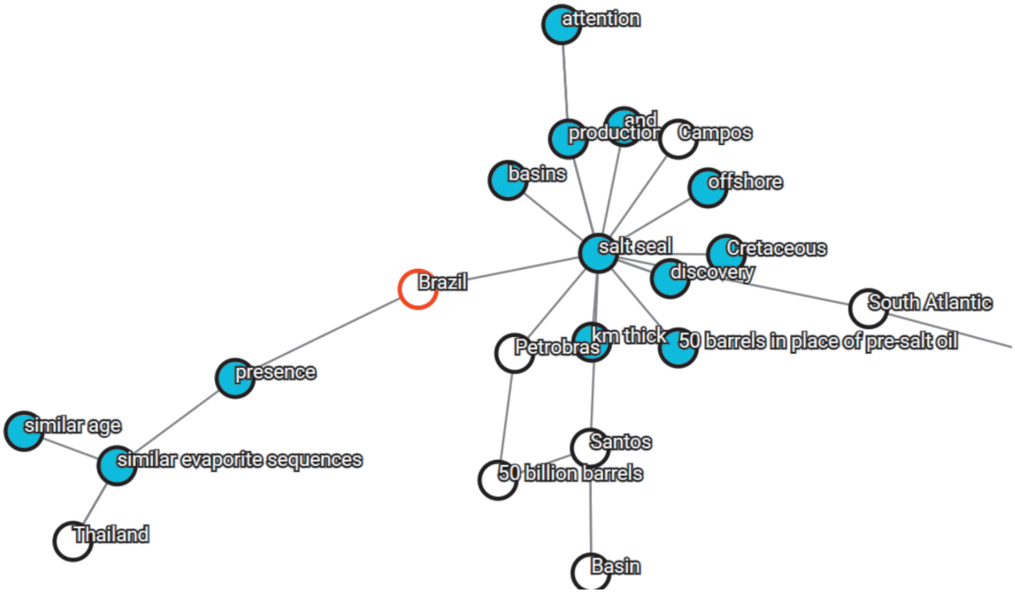


Figure 8. Base knowledge graph, before any bridges are explored.

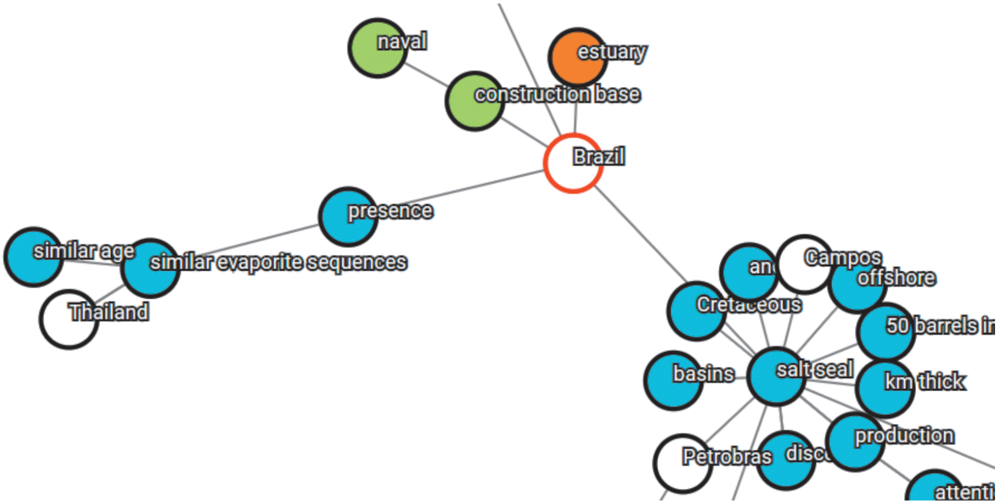


Figure 9. Knowledge graph from Fig. 8 once two bridges are explored. The added nodes are the ones in green and in orange.

5.4.2 Entity unpacking

The following excerpts exemplify how the entity unpacking (see Subsection 3.1) materializes in the knowledge graphs.

For the summary of [51], OpenIE originally extracted the following triples:

⟨ enriched calcareous algae; are transported; over time to the beach by wave action ⟩
⟨ calcareous algae; are transported; over time to the beach by wave action ⟩
⟨ algae; are transported; over time to the beach by wave action ⟩
⟨ enriched calcareous algae; are transported by; wave action ⟩
⟨ calcareous algae; are transported by; wave action ⟩
⟨ algae; are transported by; wave action ⟩.

After the entity unpacking extension—which identified, for example, that “algae“ is a subset of “calcareous algae”, which is is subset of “enriched calcareous algae”—those entities are broken up into the nuclear entities that can be seen in Fig. 10.

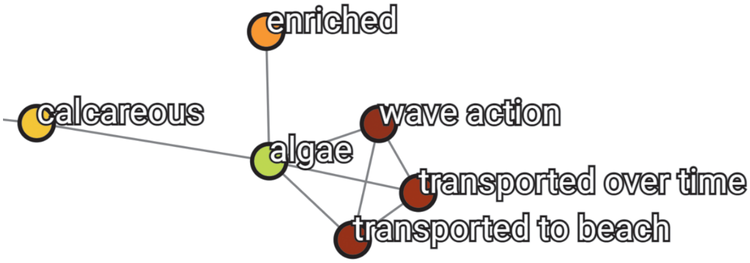


Figure 10. A display of entity unpacking in a graph (I). In this instance, "enriched calcareous algae" was split into three entities, one describing the central concept ("algae") and the other two describing its attributes ("calcareous" and "enriched").

In the second example, based on the summary of [34], the entities initially identified by OpenIE were: "data", "geochemical data", "carbonate mineralogy", "cycles", "repeated cycles", "cycles of lake level variation" and "repeated cycles of lake level variation".

This is an slightly different scenario from the last, in that there is not a direct subset-superset relation between all the associated entities, namely between "repeated cycles" and "cycles of lake level variation". Both are subsets of "repeated cycles of lake level variation", but they are not subsets/supersets of one another. Still, the unpacking works all the same and the nuclear entities are produced as shown in Fig. 11.

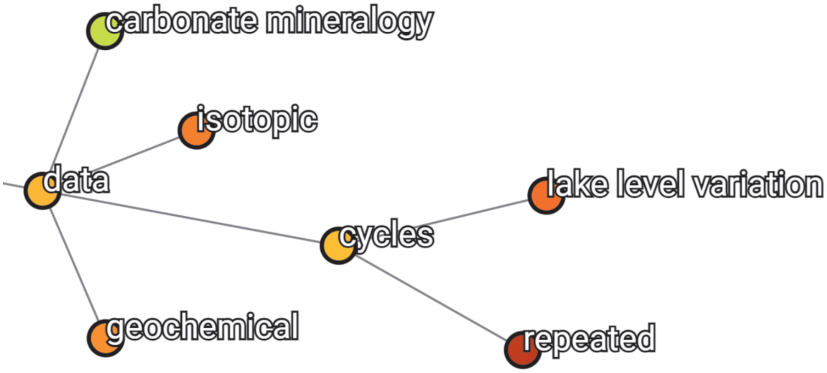


Figure 11. A display of entity unpacking in a graph (II). Similarly to what was seen in Fig. 10, the original entity "repeated cycles of lake level variation" is also split, resulting in one entity containing the central concept of "cycles", while the auxiliary entities "lake level variation" and "repeated" characterize these "cycles".

5.4.3 Shared attributes

In the case of Fig. 12, we see that the exploration of the "deep" RE-RE bridges reveals multiple entities that are associated with that characteristic, such as environments, technologies and maritime activities. Once the consistency issue mentioned in Subsection 5.3 is resolved, these kinds of bridges may become very useful in types of scenario in which there is an attempt to find an entity by its attributes.

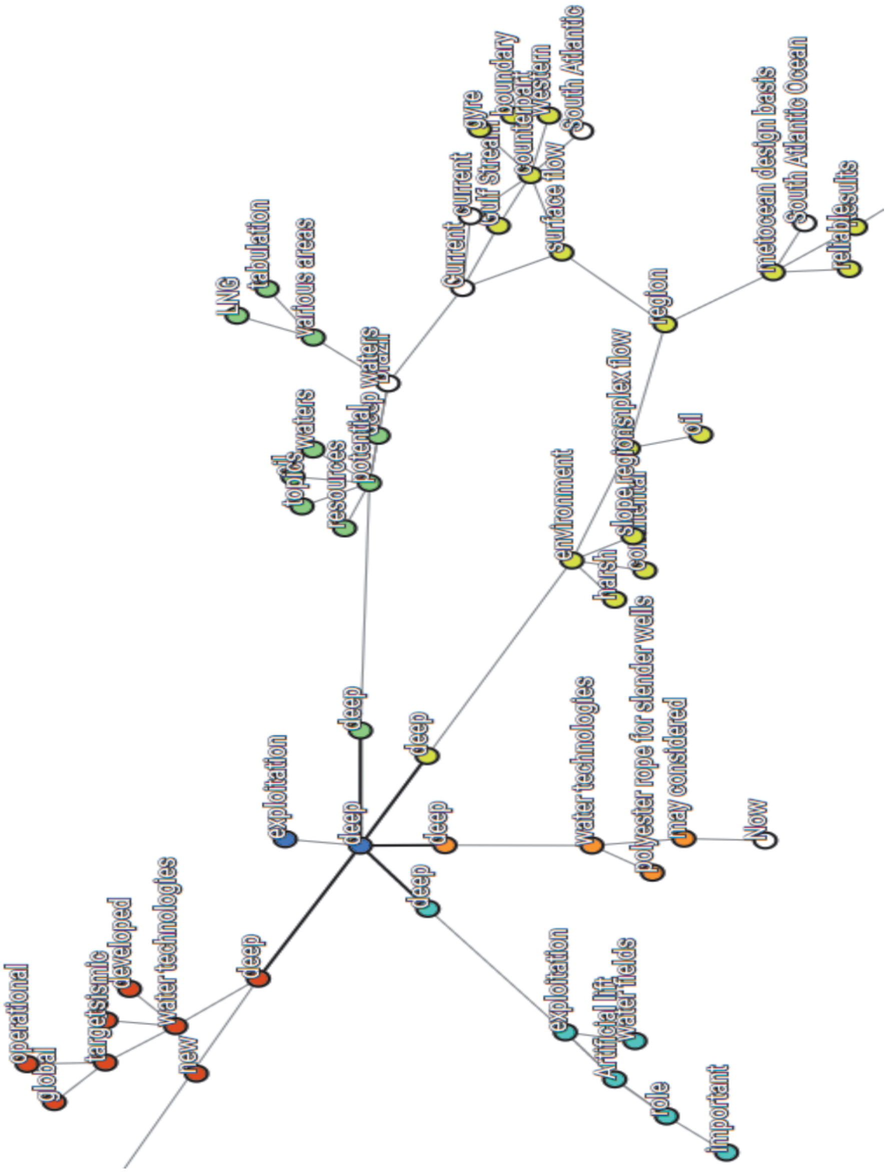


Figure 12. Using RE-RE bridges to find entities with shared characteristics. Each color group represents a completely different knowledge graph, all of which include an entity that can be described as "deep.".

6. DISCUSSION

After evaluating **CtxKG** using movie summaries and applying it to build **BlabKG**, three main points of improvement should be highlighted:

Triple identification Although the extraction of triples directly from text, without predefined relationships and entity types, is bound to sometimes produce incorrect, irrelevant or ill-formed triples, the bigger problem is inconsistency. Slight changes in sentence construction can cause important relations to be missed entirely and incoherent triples to be extracted instead.

The most noticeable example involves sentences with commas, usually separating items in a list. Ideally, the same relationship would be applied to each item in the list. However, what usually happens is that the items in the list end up connected to each other through meaningless relations (as seen in the second case in Table 5), and often the main relationship is not even identified. This is a considerable issue, since lists are a common way of expressing information in textual form.

Named entity recognition Though not as problematic as the previous issue, identifying named entities correctly and more consistently should increase the overall quality of the graphs, in so far as the reliability of entities goes. Named entities are helpful not only due to the fact that they represent real concepts, but also because they make for good bridges, since they share the same IDs across all graphs.

Word embeddings One of the most notable problems of the **RE-RE** bridges stems from the fact that, in 62.06% of them, the two regular entities being connected are not at all related. While there may be more reasons behind that, one of them certainly has to do with the word embeddings that are used to calculate the similarity between entities.

One way the word embeddings could be improved would be to retain more of the original text when calculating them, rather than creating them based solely on the contents of the triples. That should ensure that the embedding will better represent the semantic meaning of the entity.

7. CONCLUSION

In this work, we described a new knowledge graph generation method named **CtxKG**; assessed its quality using a dataset of Wikipedia articles; and presented a knowledge graph for the Blue Amazon, **BlabKG**, which combines knowledge graphs from multiple documents in a domain-specific *corpus*. We then compared the contents of **BlabKG** to the contents of the documents in order to highlight its strengths, such as correct identification of the key entities in the *corpus*, and weaknesses, such as the establishment of undesirable or too context-dependent bridges.

Regarding our method, **CtxKG** achieved results superior to competing approaches, with both the extension to OpenIE (entity unpacking) and the bridge building making positive contributions to the graphs. The unpacking extension, which used the parse tree to identify subsets among entities and split them into more nuclear entities, also helped with the stages of synonym identification and bridge building, as word embeddings were built for nuclear entities, rather than for the longer chunks of text that existed prior to the extension.

Moreover, our results indicate that the information extraction stage (i.e. entity recognition and relation extraction) really is the most critical part of the whole process. Not only is it crucial for extracting good, valuable information from the texts, it may also represent a major bottleneck for the whole pipeline, unlike later stages, which are more geared towards fine-tuning.

In addition, named entity recognition proved to be an important piece that can have a great impact on the quality of the graphs. A better and more extensive identification of named entities may improve the extraction stage by preventing named entities from being broken up and make the bridge building stage more straightforward, as named entities are matched directly by their IDs.

Consequently, our aim moving forward is to replace Stanford's CoreNLP OpenIE implementation with a better-performing relation and entity extraction technique . Additionally, we intend to have the named entity recognition and possibly the coreference resolution done separately, using dedicated software. As for the later stages, another possible improvement can come from using different approaches for building word embeddings, in order to improve entity matching.

As for **BlabKG**, our main focus is the *corpus*. Because the input for the knowledge-graph generation is a collection of scientific abstracts, there is a lot of unnecessary or confusing information that ends up cluttering the generated graphs. First, there is a great deal of complex jargon that is mentioned without much context, as it is supposed to be understood by specialists. This caused some graphs to have multiple groups of nodes that are not connected to each other because their underlying connections are implicit. Second of all, the paper abstract format includes recurring phrases and words, such as introductory expressions (e.g. "in this paper"), which are not directly relevant to the Blue Amazon domain.

ACKNOWLEDGMENTS

The authors of this work would like to thank the Center for Artificial Intelligence (C4AI-USP) and the support from the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and from the IBM Corporation.

Fabio G. Cozman acknowledges partial support by CNPq grant Pq 305753/2022-3.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Finance Code 001

AUTHOR CONTRIBUTION STATEMENT

Pedro de Moraes Ligabue performed the research, analyzed the data, wrote and revised the manuscript. **Anarosa Alves Franco Brandão** proposed the research problems, collected the data, wrote and revised the manuscript. **Sarajane Marques Peres** proposed the research problems, collected the data, wrote and revised the manuscript. **Fabio Gagliardi Cozman** proposed the research problems, wrote and revised the manuscript. **Paulo Pirozelli** collected the data, wrote and revised the manuscript.

REFERENCES

[1] IBM Cloud Education, What is a knowledge graph? https://www.ibm.com/cloud/learn/knowledge-graph, [Online; accessed 19-June-2022], Apr 2021

[2] A. Singhal, Introducing the knowledge graph: Things, not strings, https://blog.google/products/search/introducing-knowledge-graph-things-not, [Online; accessed 19-June-2022], May 2012

[3] X. Wu, J. Wu, X. Fu, J. Li, P. Zhou, and X. Jiang, "Automatic knowledge graph construction: A report on the 2019 ICDM/ICBK contest," vol. 2019-November, IEEE, Nov. 2019, pp. 1540–1545, isbn: 978-1-7281-4604-1. doi: 10.1109/ICDM.2019.00204. [Online]. Available: https://ieeexplore.ieee.org/document/8970862/

[4] P. Do, T. H. V. Phan, and B. B. Gupta, "Developing a Vietnamese tourism question answering system using knowledge graph and deep learning," ACM Trans. Asian Low-Resour. Lang. Inf. Process., vol. 20, no. 5, Jun. 2021, issn: 2375-4699. doi: 10.1145/3453651. [Online]. Available: https://doi.org/10.1145/3453651

[5] H. Noueihed, H. Harb, and J. Tekli, "Knowledge-based virtual outdoor weather event simulator using Unity 3D," J. Supercomput., vol. 78, no. 8, pp. 10 620–10 655, May 2022, issn: 0920-8542. doi: 10.1007/s11227- 021- 04212- 6. [Online]. Available: https://doi.org/10.1007/s11227-021-04212-6

[6] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition, and applications," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 2, pp. 494–514, Feb. 2022, Appendix E. doi: 10.1109/tnnls.2021.3070843. [Online]. Available: https://doi.org/10.1109%2Ftnnls.2021.3070843

[7] Knowledge Panel Help, About knowledge panels, https://support.google.com/knowledgepanel/answer/9163198?hl=en, [Online; accessed 11-July-2022]

[8] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," Communications of the ACM, vol. 57, pp. 78–85, 10 Sep. 2014, issn: 15577317. doi: 10.1145/2629489

[9] S. Yu, T. He, and J. R. Glass, "Constructing a knowledge graph from unstructured documents without external alignment," CoRR, vol. abs/2008.08995, 2020. arXiv: 2008.08995. [Online]. Available: https://arxiv.org/abs/2008.08995

[10] Wikimedia. "Wikidata statistics." [Online; accessed 28-November-2022]. (2022), [Online]. Available: https ://stats.wikimedia.org/#/wikidata.org/contributing/edits/normal%7Cline%7C2012-10-01~2022-11-28%7Ceditor_type~anonymous*group-bot*name-bot*user%7Cmonthly

[11] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 2124–2133. doi: 10.18653/v1/P16-1200. [Online]. Available: https://aclanthology.org/P16-1200

[12] A. Miller, A. Fisch, J. Dodge, A.-H. Karimi, A. Bordes, and J. Weston, "Key-value memory networks for directly reading documents," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 1400–1409. doi: 10.18653/v1/D16-1147. [Online]. Available: https://aclanthology.org/D16-1147

[13] United Nations General Assembly, Convention on the law of the sea, https://www.un.org/depts/los/convention_agreements/texts/unclos/unclos_e.pdf, [Online; accessed 12-July-2022], Dec. 1982

[14] N. Thompson and R. Muggah, The Blue Amazon: Brazil Asserts Its Influence Across the Atlantic, https://igarape.org.br/the-blue-amazon-brazil-asserts-its-influence-across-the-atlantic/, [Online; accessed 11-07-2022], Jun. 2015

[15] F. Ortiz, The Blue Amazon, Brazil's New Natural Resources Frontier, https://www.ipsnews.net/2015/05/the-blue-amazon-brazils-new-natural-resources-frontier/, [Online; accessed 11-07-2022], May 2015

[16] F. Frayssinet, BRAZIL: “Flying Blind” in Pre-Salt Oil Fields, https://www.ipsnews.net/2011/12/brazil-flying-blind-in-pre-salt-oil-fields/, [Online; accessed 11-07-2022], Dec. 2011

[17] P. de Moraes Ligabue, A. A. Franco Brandão, S. M. Peres, F. G. Cozman, and P. Pirozelli, “BlabKG: A knowledge graph for the Blue Amazon,” in 2022 IEEE International Conference on Knowledge Graph (ICKG), 2022, pp. 164–171. doi: 10.1109/ICKG55886.2022.00028

[18] H. Chen, C. Zhang, J. Li, P. S. Yu, and N. Jing, “KGGen: A generative approach for incipient knowledge graph population,” IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 5, pp. 2254–2267, 2022. doi: 10.1109/TKDE.2020.3014166

[19] V. Yadav and S. Bethard, “A survey on recent advances in named entity recognition from deep learning models,” in Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 2145–2158. [Online]. Available: https://aclanthology.org/C18-1182

[20] Y. Xiao, C. Tan, Z. Fan, Q. Xu, and W. Zhu, “Joint entity and relation extraction with a hybrid transformer and reinforcement learning based model,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, pp. 9314–9321, Apr. 2020. doi: 10.1609/aaai.v34i05.6471. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6471

[21] M. E. Peters, W. Ammar, C. Bhagavatula, and R. Power, Semi-supervised sequence tagging with bidirectional language models, 2017. doi: 10.48550/ARXIV.1705.00108. [Online]. Available: https://arxiv.org/abs/1705.00108

[22] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” CoRR, vol. abs/1810.04805, 2018. arXiv: 1810.04805. [Online]. Available: http://arxiv.org/abs/1810.04805

[23] X. Guo, H. Zhang, H. Yang, L. Xu, and Z. Ye, “A single attention-based combination of cnn and rnn for relation classification,” IEEE Access, vol. 7, pp. 12 467–12 475, 2019. doi: 10.1109/ACCESS.2019.2891770

[24] P. Do, T. Phan, H. Le, and B. B. Gupta, “Building a knowledge graph by using cross-lingual transfer method and distributed MinIE algorithm on Apache Spark,” Neural Comput. Appl., vol. 34, no. 11, pp. 8393–8409, Jun. 2022, issn: 0941-0643. doi: 10.1007/s00521-020-05495-1. [Online]. Available: https://doi.org/10.1007/s00521-020-05495-1

[25] L. Del Corro and R. Gemulla, “ClausIE: Clause-based open information extraction,” in Proceedings of the 22nd International Conference on World Wide Web, ser. WWW ’13, Rio de Janeiro, Brazil: Association for Computing Machinery, 2013, pp. 355–366, isbn: 9781450320351. doi: 10.1145/2488388.2488420. [Online]. Available: https://doi.org/10.1145/2488388.2488420

[26] K. Gashteovski, R. Gemulla, and L. del Corro, “MinIE: Minimizing facts in open information extraction,” in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 2630–2640. doi: 10.18653/v1/D17-1278. [Online]. Available: https://aclanthology.org/D17-1278

[27] R. Sukthanker, S. Poria, E. Cambria, and R. Thirunavukarasu, “Anaphora and coreference resolution: A review,” Information Fusion, vol. 59, pp. 139–162, 2020, issn: 1566-2535. doi: https://doi.org/10.1016/j.inffus.2020.01.010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253519303677

[28] K. Lee, L. He, M. Lewis, and L. Zettlemoyer, End-to-end neural coreference resolution, 2017. doi: 10.48550/ARXIV.1707.07045. [Online]. Available: https://arxiv.org/abs/1707.07045

[29] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, “Translating embeddings formodelingmulti-relational data,” in Advances in Neural Information Processing Systems, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds., vol. 26, Curran Associates, Inc., 2013.

[Online]. Available: https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf

[30] M. Chen, Y. Tian, M. Yang, and C. Zaniolo, Multilingual knowledge graph embeddings for cross-lingual knowledge alignment, 2016. doi: 10.48550/ARXIV.1611.03954. [Online]. Available: https://arxiv.org/abs/1611.03954

[31] H. Zhu, R. Xie, Z. Liu, and M. Sun, "Iterative entity alignment via joint knowledge embeddings," in Proceedings of the 26th International Joint Conference on Artificial Intelligence, ser. IJCAI'17, Melbourne, Australia: AAAI Press, 2017, pp. 4258–4264, isbn: 9780999241103

[32] Z. Wang, Q. Lv, X. Lan, and Y. Zhang, "Cross-lingual knowledge graph alignment via graph convolutional networks," in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 349–357. doi: 10.18653/v1/D18- 1032. [Online]. Available: https://aclanthology.org/D18-1032

[33] Y. Wu, X. Liu, Y. Feng, Z. Wang, R. Yan, and D. Zhao, "Relation-aware entity alignment for heterogeneous knowledge graphs," in Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, International Joint Conferences on Artificial Intelligence Organization, Aug. 2019. doi: 10.24963/ijcai.2019/733. [Online]. Available: https://doi.org/10.24963%5C%2Fijcai.2019%5C%2F733

[34] Y. Zhu, H. Liu, Z. Wu, and Y. Du, Relation-aware neighborhood matching model for entity alignment, 2020. doi: 10.48550/ARXIV.2012.08128. [Online]. Available: https://arxiv.org/abs/2012.08128

[35] Z. Chen, Y. Wang, B. Zhao, J. Cheng, X. Zhao, and Z. Duan, "Knowledge graph completion: A review," IEEE Access, vol. 8, pp. 192 435–192 456, 2020. doi: 10.1109/ACCESS.2020.3030076

[36] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. v. d. Berg, I. Titov, and M. Welling, Modeling relational data with graph convolutional networks, 2017. doi: 10.48550/ARXIV.1703.06103. [Online]. Available: https://arxiv.org/abs/1703.06103

[37] J. R. Barr, P. Shaw, F. N. Abu-Khzam, T. Thatcher, and T. D. Hocking, "Graph embedding: A methodological survey," in 2022 Fourth International Conference on Transdisciplinary AI (TransAI), 2022, pp. 142–148. doi: 10.1109/TransAI54797.2022.00031

[38] A. Grover and J. Leskovec, node2vec: Scalable feature learning for networks, 2016. doi: 10.48550/ARXIV.1607.00653. [Online]. Available: https://arxiv.org/abs/1607.00653

[39] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 55–60. doi: 10.3115/v1/P14-5010. [Online]. Available: https://aclanthology.org/P14-5010

[40] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction from the web," in Proceedings of the 20th International Joint Conference on Artificial Intelligence, ser. IJCAI'07, Hyderabad, India: Morgan Kaufmann Publishers Inc., 2007, pp. 2670–2676

[41] X. Han, T. Gao, Y. Yao, D. Ye, Z. Liu, andM. Sun, "OpenNRE: An open and extensible toolkit for neural relation extraction," in Proceedings of EMNLP-IJCNLP: System Demonstrations, 2019, pp. 169–174. doi: 10.18653/v1/D19-3029. [Online]. Available: https://www.aclweb.org/anthology/D19-3029

[42] Y. Ro, Y. Lee, and P. Kang, "Multi2OIE: Multilingual open information extraction based on multi-head attention with BERT," in Findings of the Association for Computational Linguistics: EMNLP 2020, Online: Association for Computational Linguistics, Nov. 2020, pp. 1107–1117. doi: 10.18653/v1/2020.findings-emnlp.99. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.99

[43] M.-C. de Marneffe, T. Dozat, N. Silveira, et al., "Universal Stanford Dependencies: A cross-linguistic typology," in Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland: European Language Resources Association (ELRA),

May 2014, pp. 4585–4592. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2014/pdf/1062_Paper.pdf

[44] MediaWiki, API:parsing wikitext—MediaWiki, https://www.mediawiki.org/w/index.php?title=API:Parsing_wikitext&oldid=5020192, [Online; accessed 22-June-2022], 2022

[45] D. Kotkov, A. Maslov, and M. Neovius, "Revisiting the tag relevance prediction problem," in Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, ser. SIGIR '21, Virtual Event, Canada: Association for Computing Machinery, 2021, pp. 1768–1772, isbn: 9781450380379. doi: 10.1145/3404835.3463019. [Online]. Available: https://doi.org/10.1145/3404835.3463019

[46] J. Vig, S. Sen, and J. Riedl, "The tag genome: Encoding community knowledge to support novel interaction," ACM Trans. Interact. Intell. Syst., vol. 2, no. 3, Sep. 2012, issn: 2160-6455. doi: 10.1145/2362394.2362395. [Online]. Available: https://doi.org/10.1145/2362394.2362395

[47] A. F. A. Paschoal, P. Pirozelli, V. Freire, et al., "Pirá: A bilingual Portuguese-English dataset for question-answering about the ocean," in Proceedings of the 30th ACM International Conference on Information & Knowledge Management, ser. CIKM '21, Virtual Event, Queensland, Australia: Association for Computing Machinery, 2021, pp. 4544–4553, isbn: 9781450384469. doi: 10.1145/3459637.3482012. [Online]. Available: https://doi.org/10.1145/3459637.3482012

[48] F. Brouwer, A. Huck, N. Hemstra, and I. Braga, "Extracting full-resolution models from seismic data to minimize systematic errors in inversion: Method and examples," The Leading Edge, vol. 31, no. 5, pp. 546–554, 2012. doi: 10.1190/tle31050546.1. eprint: https://doi.org/10.1190/tle31050546.1. [Online]. Available: https://doi.org/10.1190/tle31050546.1

[49] Deepwater Installation of a Large Capacity FPSO with Large Number of Risers in the Marlim Field, vol. All Days, OTC Offshore Technology Conference, OTC-10722-MS, May 1999. doi: 10.4043/10722- MS. eprint: https://onepetro.org/OTCONF/proceedings-pdf/99OTC/All-99OTC/OTC-10722-MS/1923497/otc-10722-ms.pdf. [Online]. Available: https://doi.org/10.4043/10722-MS

[50] P. Szatmari, C. Moré de Lima, G. Fontaneta, et al., "Petrography, geochemistry and origin of South Atlantic evaporites: The Brazilian side," Marine and Petroleum Geology, vol. 127, p. 104 805, 2021, issn: 0264-8172. doi: https://doi.org/10.1016/j.marpetgeo.2020.104805. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0264817220305882

[51] W. Baeyens, N. Mirlean, J. Bundschuh, et al., "Arsenic enrichment in sediments and beaches of Brazilian coastal waters: A review," Science of The Total Environment, vol. 681, pp. 143–154, 2019, issn: 0048-9697. doi: https://doi.org/10.1016/j.scitotenv.2019.05.126. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0048969719321515

[52] R. Pietzsch, L. R. Tedeschi, D. M. Oliveira, C. W. D. dos Anjos, J. C. Vazquez, and M. F. Figueiredo, "Environmental conditions of deposition of the Lower Cretaceous lacustrine carbonates of the Barra Velha formation, Santos Basin (Brazil), based on stable carbon and oxygen isotopes: A continental record of pCO2 during the onset of the Oceanic Anoxic Event 1a (OAE 1a) interval?" Chemical Geology, vol. 535, p. 119 457, 2020, issn: 0009-2541. doi: https://doi.org/10.1016/j.chemgeo.2019.119457. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0009254119305868

[53] Wikimedia Commons, File:Brésil-ZEE et plateau continental.jpeg — Wikimedia Commons, the free media repository, [Online; accessed 27-March-2023], 2023. [Online]. Available: https://commons.wikimedia.org/w/index.php?title=File:Br%C3%A9sil_-_ZEE_et_plateau_continental.jpeg&oldid=729478241

[54] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of English: The Penn Treebank," Computational Linguistics, vol. 19, no. 2, pp. 313–330, 1993. [Online]. Available: https://aclanthology.org/J93-2004

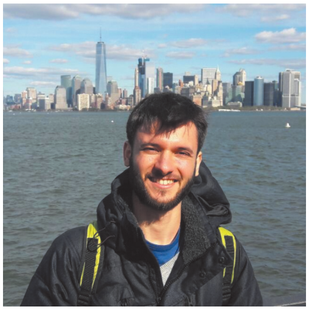
AUTHOR BIOGRAPHY

Pedro de Moraes Ligabue is an electrical engineer with a bachelor degree from the University of São Paulo (2020). Currently in the process of completing his MSc in computer engineering, he has been focused on the generation of knowledge graphs directly from text, both in English and in Portuguese.

Anarosa Alves Franco Brandão is an Associate Professor at Computing and Digital Systems Department of Escola Politécnica-Universidade de São Paulo. She earn degrees in Sciences-Mathematics in 1990 (bachelor) and 1994 (master) from the Universidade of São Paulo. Her PhD in Sciences-Informatics was finished in 2005, at the Pontifical Catholic University of Rio de Janeiro. Her research interests are related to Computer Science and Education, with an emphasis on Informatics in Education and Artificial Intelligence, working mainly on topics related to multiagent systems and accessibility in web-based learning systems. She has been a professor in the Computer Engineering and Digital Systems department at Escola Politécnica from Universidade of São Paulo since December 2008, the same year her son Leonardo was born. She is editor-in-chief of the Revista Brasileira de Informática na Educação (RBIE), a publication of CEIE-SBC, in the period 2022-2024 and Coordinator of the Computer Engineering Area of the Graduate Program in Electrical Engineering at Escola Politécnica - USP (2022-2024). She has been a representative of the Escola Politécnica at the Brazilian Computer Society (SBC) since March 2021. She is part of the advisory committee for the WESAAC-Workshop School of Agent Systems, Their Environments and Applications workshop series and the Special Committee of Informatics in Education of the Brazilian Computer Society.

Sarajane Marques Peres is an Associate Professor at the University of São Paulo, Brazil. Ph.D. in Electric Engineering (2006) at the University of Campinas; Master of Manufacturing Engineering (1999) at the Federal University of Santa Catarina; Bachelor's in computer science (1996) at the State University of Maringá, Brazil. She co-wrote a Data Mining textbook published in Portuguese, and worked as a guest researcher at the Vrije Universiteit Amsterdam (2018) and the Utrecht University (2019), Netherlands. She is a member of the Information System Master Program's coordination committee at the University of São Paulo, and researcher at the Center for Artificial Intelligence (C4AI). Her main research interests are computational intelligence, data mining, and machine learning applied to text mining, process mining, and human-robot interaction.

Fabio Gagliardi Cozman is a Full Professor at the Escola Politécnica from the Universidade de São Paulo (Dept. Mechatronic Engineering) since 2007, having joined the Escola Politécnica in 1990. He has a degree in Electrical Engineering from the Universdade de São Paulo (1989), a Master's degree in Engineering from the University of São Paulo (1991) and a PhD from Carnegie Mellon University (1997), Professorship from the Universidade de São Paulo (2003). He is director of the Artificial Intelligence Center USP/IBM/FAPESP (c4ai.inova.usp.br). He was coordinator of the Brazilian Computer Society's Artificial Intelligence committee, Associate Editor of the Int. Journal on Approximate Reasoning, Associate Editor of the Artificial Intelligence Journal, and Associate Editor of the Journal of Artificial Intelligence Research, as well as Area Chair of the Int. Joint Conf. on Artificial Intelligence and Program Chair at Conf. on Uncertainty in Artificial Intelligence. His research focuses on machine learning and decision processes under uncertainty, including knowledge representation and learning (topics: artificial intelligence, Bayesian networks, probability sets, graphical statistical models).



Paulo Pirozelli holds a Bachelor's degree in Philosophy (2010), followed by a Master's degree (2013), and a Ph.D. (2018), all earned from the University of São Paulo. He served as a visiting researcher at both Columbia University (2016) and the Institute for Quantitative Social Science, Harvard University (2018). Following his Ph.D., he completed a postdoctoral internship in Philosophy at the Federal University of Santa Catarina (2018-2020). Presently, he is a postdoctoral fellow at the Center for Artificial Intelligence (C4AI) at the University of São Paulo, working in natural language processing.