Exploring Open Information Extraction for the Portuguese language: An integrated monolithic approach in Cloud environment

Augusto Sampaio Barreto Daniela Barreiro Claro
FORMAS Research Center - Institute of Computing
Federal University of Bahia – Salvador - Bahia - Brazil
augusto.barreto@ufba.br dclaro@ufba.br

Abstract

This work addresses Open Information Extraction (Open IE) as a crucial area for structuring unstructured data, aiming to identify and represent information through triples. The Open IE approach proposes a domain-independent paradigm that extracts potential relationships between entities using generalized patterns. Despite advancements, the lack of studies in the Portuguese language emphasizes the need to explore specific techniques. This is worse regarding easy access to take advantage of triples extracted by OpenIE models. In this context, we present an integrated hub of services in the Open IE domain, allowing users to extract triples and compare their results. This hub is developed under a monolithic architecture with the Django framework, coupled with deployment on Google Cloud to reinforce the efficiency and adaptability of inclusion and remotion of services. Within this framework, it would be possible for non-computer experts to use the advantages of OpenIE triples from the Portuguese languages.

1 Introduction

Open Information Extraction (Open IE) is a crucial area that aims to structure data from unstructured sources, with the main goal of identifying and representing information through triples that express relationships. The Open IE approach represents a domain-independent extraction paradigm, using generalized patterns to extract all potential relationships between entities (Etzioni et al., 2008).

Triples are essential for capturing the meaning of information present in unstructured data. Despite advances in the last decade, most of this progress has focused on the English language, with few studies dedicated to Portuguese in the last five years (Bender, 2019).

The limitation of studies in the Portuguese language emphasizes the need to explore and enhance the application of Open IE techniques in such language. However, works in this area publish their results as a package, making it difficult to use by non-computer experts. This article aims to contribute to this gap by presenting a hub of services that integrate methods from Open IE in Portuguese, enabling a single environment for users to extract triples and compare the results.

Our framework deserves a hub of Open IE services that implements each service as an Open IE method for the Portuguese language. We call this hub of Open IE services as FORMAS Open IE Framework.

This article is structured as follows: the next section describes the background. Section 3 presents each Open IE method. Section 4 describes our hub of Open IE services. Section 5 discusses our conclusions and some envisioning work.

2 Background

The advantages of Open IE, as outlined by (Gamallo, 2012), encompass domain independence, unsupervised extraction, and greater scalability. These features highlight Open IE's adaptability, efficiency, and scalability in handling diverse subjects and extensive volumes of unstructured text.

Concerning architecture, the system's architectural choice is pivotal in shaping Open IE systems' development and maintenance. Various paradigms, including monolithic, microservices, layered, and event-driven architectures, offer distinct characteristics influencing scalability, flexibility, and modularity (Fowler, 2003; Newman, 2015a; Richardson, 2018).

A brief exploration of these architectural approaches follows:

> *Monolithic Architecture*: Integrates all components into a single application, simplifying development and maintenance. While suitable for applications with precise requirements, it

may face scalability challenges as the system grows (Fowler, 2003).

Microservices Architecture: Divides a system into independent services, promoting scalability and independent evolution. Facilitates modular development, updates, and maintenance (Newman, 2015a).

Layered Architecture: Organizes the system into abstraction levels, promoting modularity. While simplifying maintenance, inter-layer dependencies may limit scalability (Fowler, 2003).

Event-Driven Architecture: Components communicate through asynchronous events, favoring scalability and dynamic responsiveness. Allows distributed processing but requires careful management of asynchronous events (Richardson, 2018).

The choice of architecture depends on project-specific needs, with monolithic architectures offering simplicity, microservices providing scalability, layered architectures ensuring modularity and event-driven architectures supporting reactivity in distributed systems. Each decision entails profound implications for system evolution and maintenance, emphasizing the importance of considering project characteristics and requirements.

3 Services

Services are implemented as Open IE methods. A set of OpenIE methods for the Portuguese language was selected to be part of the first version of this hub of services: DptOIE, PTOiE-Flair, and ChatGPT. We detailed each one as follows.

3.1 DptOIE

DptOIE (Oliveira et al., 2023) is a method developed for Open Information Extraction (OIE), specifically designed for the Portuguese language. The main objective of DptOIE is to extract valuable information or "facts" from sentences by analyzing their syntactic structure and dependencies. DptOIE has three main phases: Pre-processing, triple extraction and special cases.

The prepossessing carries out a *Tokenization*, *Part-of-Speech* (POS) tagging, and *Dependency Analysis* to inputted sentences. DptOIE identifies triples (subject, relation, object) by traversing the dependency tree using a Depth-First Search (DFS) approach. Each triple consists of an argument (Arg1), a relation, and the second argument (Arg2).

DptOIE addresses specific linguistic constructions, such as:

- Coordinative Conjunctions (CC): Handles conjunctions like "and" or "or" to generate multiple triples from a single sentence.
- Subordinate Clauses:Manages adjective, adverbial, and substantive clauses, linking them to the main clause to form coherent triples.
- Appositives: Derives additional triples from sentences with appositives, creating synthetic clauses.

Consider the sentence "O diretor do hospital, Júlio, vendeu sua fazenda." DptOIE extracts the main triple: (O diretor do hospital; vendeu; sua fazenda). Moreover, it recognizes the appositive "Júlio" and generates a new triple: (O diretor do hospital; é; Júlio). Additionally, it applies transitivity to create an additional triple: (Júlio; vendeu; sua fazenda).

3.2 PTOiE-Flair

The PTOiE-Flair is a new OpenIE model based on deep neural networks that enables the generation of triples given a sentence. PTOiE-Flair was trained with two datasets, LSOI and S2, achieving SOTA results for the Portuguese language.

Consider the sentence "Os cachorros, que são mamiferos, são os melhores amigos do homem." PortNOIE extracts two triples: ["Os cachorros"/ARG0, "são"/V, "mamiferos"/ARG1, "são"/V, "os melhores amigos do homem"/ARG1], that is: (i) *Os cachorros; são; os melhores amigos do homem* and (ii) *Os cachorros; são; mamiferos*.

3.3 ChatGPT

The use of ChatGPT for triple extraction represents an innovative approach to integrating natural language technologies with the extraction of structured information. Considering the ChatGPT, carefully formulating prompts is essential to guide the model in generating structured responses.

Prompt example: "Provide a triple containing a subject, a relation, and an object based on the following statement: 'The event occurred when'."

We employed the ChatGPT as a service, providing custom prompts and receiving structured responses as a triple structure. Initially, we utilized the davinci-003 variant of the ChatGPT model. However, in future deployments, new models such

as GPT-4 or GPT-4-turbo could be seamlessly integrated without compromising performance.

Consider the prompt: "Describe a situation in which" followed by a specific context. ChatGPT generates a structured response, such as "(a scientist; makes; a significant discovery)".

4 FORMAS OpenIE Framework

The monolithic architecture based on Django framework simplifies the development and maintenance of robust systems. According to (Newman, 2015b; Fowler, 2014), monolithic architectures provide an integrated approach, making implementing and managing functionalities easier.

Our architecture comprises two modules as depicted in Figure 1: a Frontend and a Backend. Our front end was integrated with Django, and our back end was initially developed as a service hub with three methods: Chatgpt, Ptoie-flair, and DptOIE.

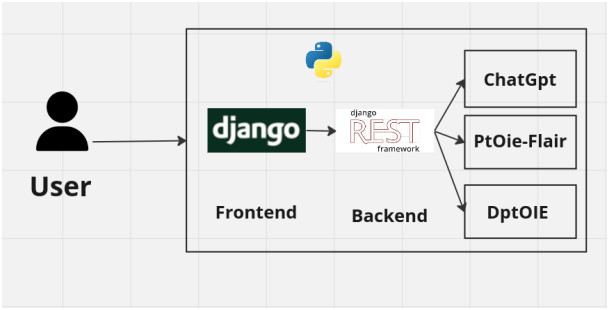

Figure 1: Architecture

Integration with Django enables the creation of a cohesive application, facilitating efficient communication between components. As mentioned by (Mokhtar, 2018), the use of monoliths is particularly advantageous when a pragmatic and efficiency-centered approach is sought, making it ideal for applications with clear and well-defined requirements.

In addition to the monolithic architecture, deploying the service on Cloud enhances resource adaptability for users. Utilizing cloud services provides dynamic scalability, allowing adjustments based on demand. According to (Kavis, 2014), Cloud services offers a reliable and flexible infrastructure, ensuring that users have access to resources tailored to their needs.

4.1 Graphical Interface

Our framework has a graphical interface depicted in Figure 2 that enables the use of visual elements to extract triples.

Figure 2: Interface

The user introduces a sentence in Portuguese, and our Framework answers with the triple extracted. The blue rectangle is omitted for anonymous review, and neither is the line under the title. Firstly, the user describes the sentence and selects which OpenIE method would prefer to extract the triples. Results are shown immediately after the button.

5 Conclusions and Future Work

The integration between the monolithic architecture using Django and deployment on the Cloud reinforces the robustness and adaptability of the Open Information Extraction services. This approach, supported by solid theoretical foundations, allows non-computer users to evaluate the extraction on three OpenIE methods.

In this work, while the primary goal was to provide a unified platform for users to access and utilize different OpenIE methods, the comparative analysis of these methods was not within the scope. Such comparisons entail diverse performance metrics and evaluation criteria, which could vary depending on the specific natural language processing tasks at hand. However, as future work, conducting comparative evaluations across various NLP tasks could offer valuable insights into the strengths and weaknesses of different extraction methods. By expanding the scope of evaluation beyond OpenIE,

we can further refine and optimize these methods to better serve the needs of the Portuguese language community.We envision enhancing the implementation and including more methods for the Open IE Portuguese language community.

Acknowledgments

This material is partially based upon work supported by the FAPESB under grant INCITE PIE0002/2022 and FAPESB TIC 0002/2015 and CAPES Financial code 001.

References

Emily M. Bender. 2019. On the lack of study of non-english languages in nlp. In *Proceedings of the First ACL Workshop on Ethics in NLP*, pages 7–13.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Commun. ACM*, 51(12):68–74.

Martin Fowler. 2003. *Patterns of Enterprise Application Architecture*. Addison-Wesley.

Martin Fowler. 2014. Microservices.

Pablo Gamallo. 2012. Overview of open information extraction. In *Open Information Extraction: Volume 377 of CEUR Workshop Proceedings*, pages 1–13.

Mike Kavis. 2014. *Architecting the Cloud: Design Decisions for Cloud Computing Service Models*. John Wiley & Sons.

Khaled Mokhtar. 2018. The advantages and disadvantages of monolithic and microservices architectures.

Sam Newman. 2015a. *Building Microservices: Designing Fine-Grained Systems*. O'Reilly Media.

Sam Newman. 2015b. *Building Microservices: Designing Fine-Grained Systems*, 1st edition. O'Reilly Media.

Leandro Oliveira, Daniela Barreiro Claro, and Marlo Souza. 2023. Dptoie: a portuguese open information extraction based on dependency analysis. *Artif. Intell. Rev.*, 56(7):7015–7046.

Chris Richardson. 2018. *Microservices Patterns: With Examples in Java*. Manning Publications.