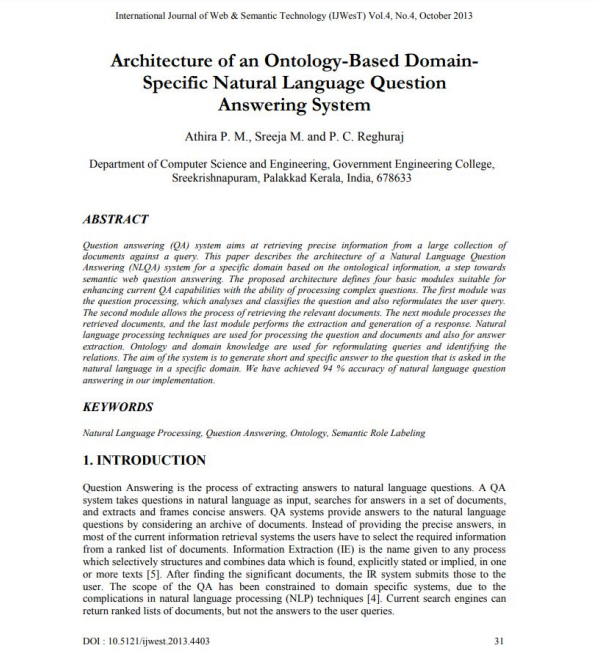
Arquitetura para um sistema de perguntas e respostas usando ontologias

Brian Alves Andreossi	11060215
Gustavo Zanfelice Dib	11023915
Marcelo Schirbel Gomes	11022014
Murilo Bolzan Dionisio	11107414

Artigo utilizado:

Architecture of an Ontology-Based Domain-Specific Natural Language Question Answering System



International Journal of Web & Semantic Technology (IJWesT) Vol.4, No.4, October 2013

Architecture of an Ontology-Based Domain-Specific Natural Language Question Answering System

Athira P. M., Sreeja M. and P. C. Reghuraj

Department of Computer Science and Engineering, Government Engineering College, Sreekrishnapuram, Palakkad Kerala, India, 678633

ABSTRACT

Question answering (QA) system aims at retrieving precise information from a large collection of documents against a query. This paper describes the architecture of a Natural Language Question Answering (NLQA) system for a specific domain based on the ontological information, a step towards semantic web question answering. The proposed architecture defines four basic modules suitable for enhancing current QA capabilities with the ability of processing complex questions. The first module was the question processing, which analyses and classifies the question and also reformulates the user query. The second module allows the process of retrieving the relevant documents. The next module processes the retrieved documents, and the last module performs the extraction and generation of a response. Natural language processing techniques are used for processing the question and documents and also for answer extraction. Ontology and domain knowledge are used for reformulating queries and identifying the relations. The aim of the system is to generate short and specific answer to the question that is asked in the natural language in a specific domain. We have achieved 94 % accuracy of natural language question answering in our implementation.

KEYWORDS

Natural Language Processing, Question Answering, Ontology, Semantic Role Labeling

1. INTRODUCTION

Question Answering is the process of extracting answers to natural language questions. A QA system takes questions in natural language as input, searches for answers in a set of documents, and extracts and frames concise answers. QA systems provide answers to the natural language questions by considering an archive of documents. Instead of providing the precise answers, in most of the current information retrieval systems the users have to select the required information from a ranked list of documents. Information Extraction (IE) is the name given to any process which selectively structures and combines data which is found, explicitly stated or implied, in one or more texts [5]. After finding the significant documents, the IR system submits those to the user. The scope of the QA has been constrained to domain specific systems, due to the complications in natural language processing (NLP) techniques [4]. Current search engines can return ranked lists of documents, but not the answers to the user queries.

DOI : 10.5121/ijwest.2013.4403

Athira P. M., Sreeja M. and P. C. Reghuraj

Department of Computer Science and Engineering, Government Engineering College,
Sreekrishnapuram, Palakkad Kerala, India

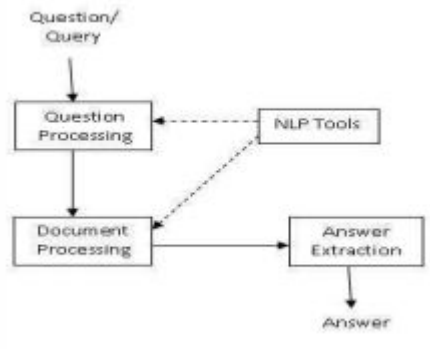
Novembro de 2013

- **19** Citações
- **11** Referências
- **10** Páginas

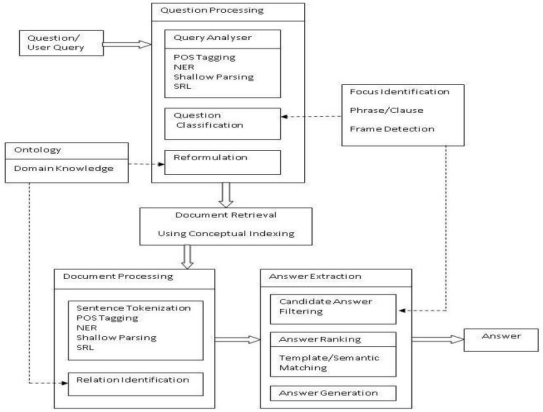
Ideia Geral

Uma arquitetura para um sistema de Perguntas e respostas que analisasse a ontologia das perguntas

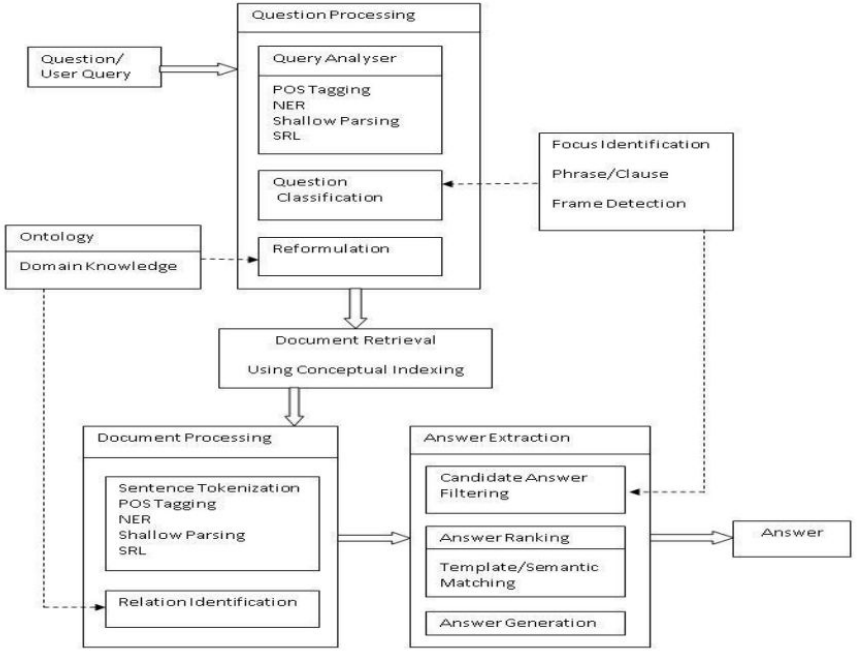
Arquitetura Comum:



Arquitetura Proposta:



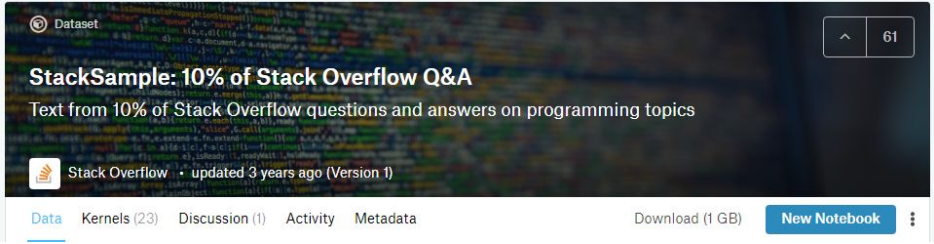
Arquitetura Proposta



Adaptação do Problema

Sistema de resposta automática baseado no Stack Overflow

Base de dados:



Bases:

Question.csv

Base de perguntas com Id, Score, Texto e corpo.

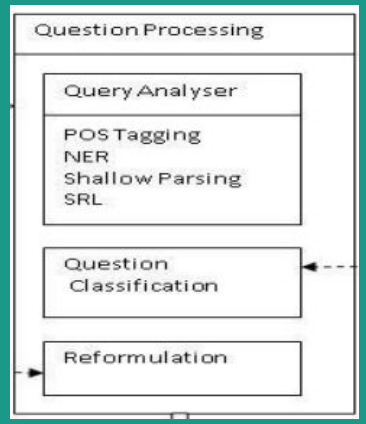
Tags.csv

Base de tags com Id da pergunta e Tags

Answers.csv

Base de Respostas com Id da resposta, Id da pergunta, Score e corpo da resposta

Question Processing

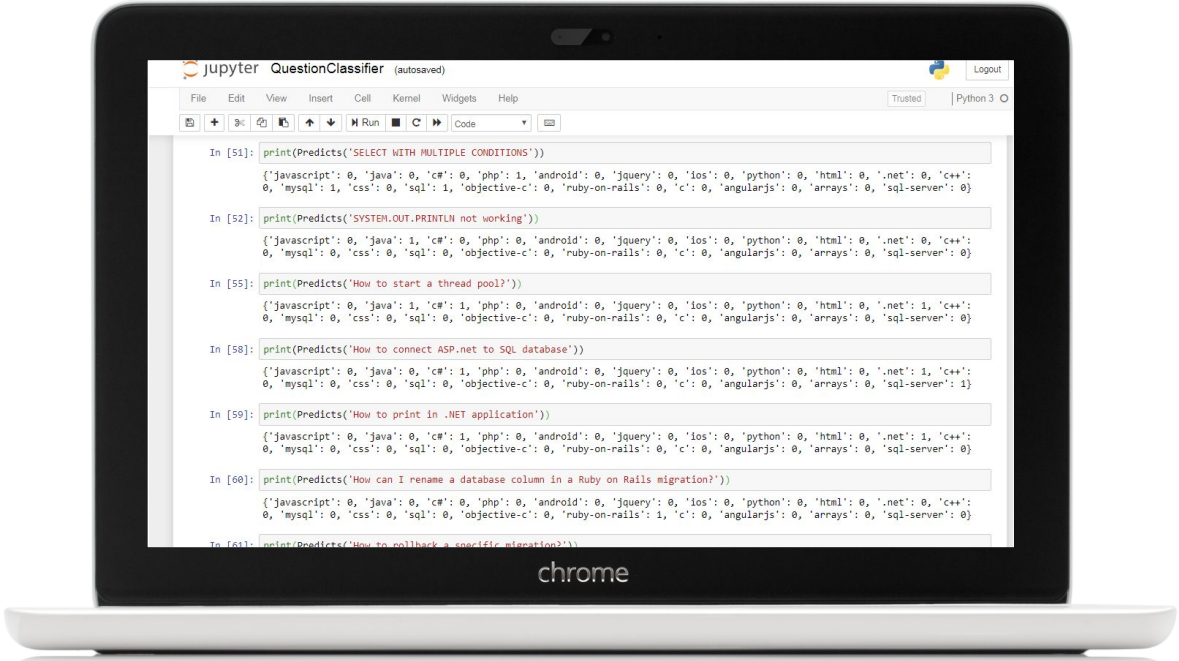


Modelo de Classificação

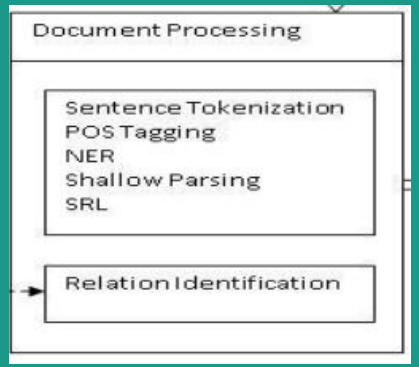
Como temos a base de Tags, utilizamos métodos de Aprendizado supervisionado para classificar uma nova pergunta com as tags do StackOverflow.

O método de melhor desempenho foi o Naive Bayes multinomial no paradigma One vs. All, para realizarmos a classificação com diversas Classes.

Exemplo de resposta

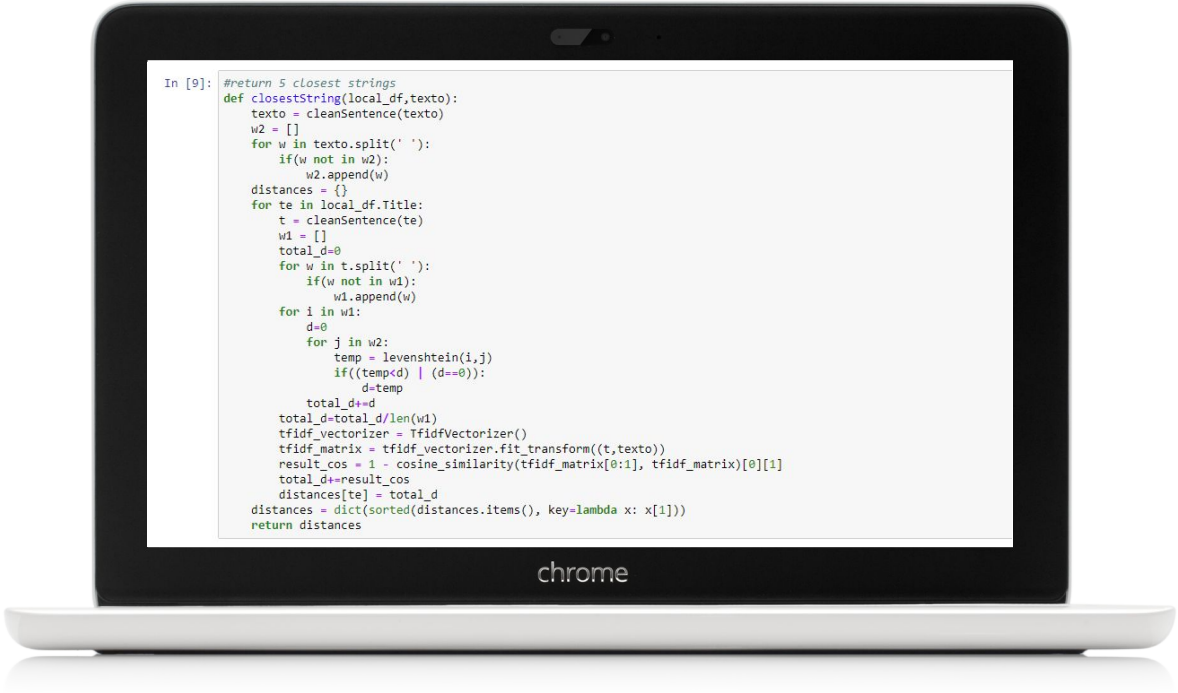


Document Processing



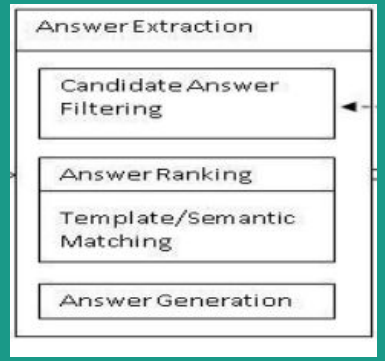
Distância entre as perguntas

Usamos a distância de Levenshtein para obter uma aproximação entre todas as palavras das frases e depois a distância do cosseno para comparar as frases inteiras, por fim, somamos os dois valores.



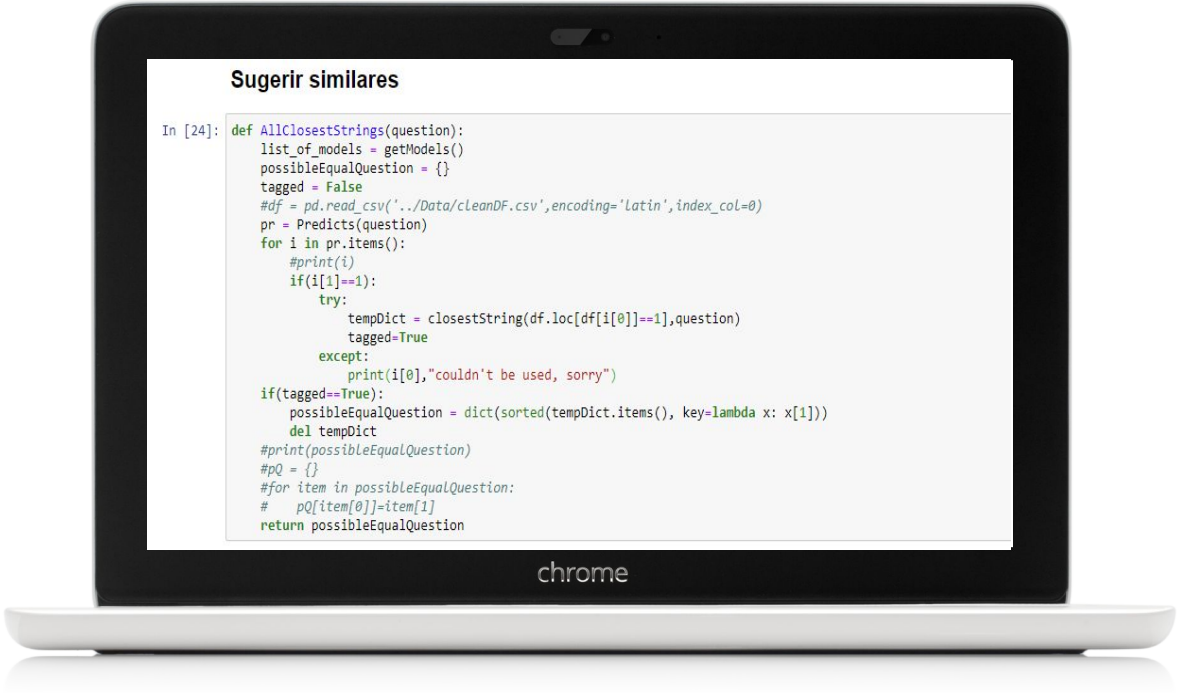

```
In [9]: #return 5 closest strings
        def closestString(local_df,texto):
            texto = cleanSentence(texto)
            w2 = []
            for w in texto.split(' '):
                if(w not in w2):
                    w2.append(w)
            distances = {}
            for te in local_df.Title:
                t = cleanSentence(te)
                w1 = []
                total_d=0
                for w in t.split(' '):
                    if(w not in w1):
                        w1.append(w)
                for i in w1:
                    d=0
                    for j in w2:
                        temp = levenshtein(i,j)
                        if((temp<d) | (d==0)):
                            d=temp
                    total_d+=d
                total_d=total_d/len(w1)
                tfidf_vectorizer = TfidfVectorizer()
                tfidf_matrix = tfidf_vectorizer.fit_transform((t,texto))
                result_cos = 1 - cosine_similarity(tfidf_matrix[0:1], tfidf_matrix)[0][1]
                total_d+=result_cos
                distances[te] = total_d
            distances = dict(sorted(distances.items(), key=lambda x: x[1]))
            return distances
```

Answer Extraction



Extraindo Perguntas

A base de busca é filtrada pelas tags preditas.
O retorno dessa função é um dicionário com as melhores perguntas, 5 melhores.
Caso o usuário não insira nenhuma pergunta das selecionadas, retornamos as 5 próximas seguintes.



Perguntas do Professor

Por que é necessário um sistema de perguntas e respostas?

O primeiro ponto é termos a capacidade de obter a resposta mesmo offline.

O segundo ponto é que obtemos a melhor resposta da base de dados, sem depender encontrar uma thread certeira.

Quais técnicas de PLN utilizamos?

Naive Bayes Multinomial(one versus all), Distância de Levenshtein, Distância de Cosseno,

Quais os sistemas de perguntas e respostas conhecidos atualmente? Qual tecnologia eles usam?

Quora, Google Answers e Yahoo Answers!

Algumas das tecnologias estão descritas no livro Introduction to Information Retrieval, disponível no seguinte link:
https://nlp.stanford.edu/IR-book/

Estão inclusas: Naive Bayes, Flat Clustering, Regex e Probabilistic Information Retrieval

Como é feita a validação da proposta?

Se a dúvida do usuário for sanada.

Para fazermos uma validação mais a fundo, podemos fazer uma pesquisa em outros sites de perguntas e respostas.