

Building a Massive Corpus for Named Entity Recognition using Free Open Data Sources

Daniel Specht Menezes

Departamento de Informática

PUC-Rio

Rio de Janeiro, Brazil

danielssmenezes@gmail.com

Pedro Savarese

TTIC

Chicago, USA

savarese@ttic.edu

Ruy L. Milidiú

Departamento de Informática

PUC-Rio

Rio de Janeiro, Brazil

milidiu@inf.puc-rio.br

Abstract—With the recent progress in machine learning, boosted by techniques such as deep learning, many tasks can be successfully solved once a large enough dataset is available for training. Nonetheless, human-annotated datasets are often expensive to produce, especially when labels are fine-grained, as is the case of Named Entity Recognition (NER), a task that operates with labels on a word-level.

In this paper, we propose a method to automatically generate labeled datasets for NER from public data sources by exploiting links and structured data from DBpedia and Wikipedia. Due to the massive size of these data sources, the resulting dataset – SESAME¹ – is composed of millions of labeled sentences. We detail the method to generate the dataset, report relevant statistics, and design a baseline using a neural network, showing that our dataset helps building better NER predictors.

Index Terms—named entity recognition, distant learning, neural networks

I. INTRODUCTION

The vast amounts of data available from public sources such as Wikipedia can be readily used to pre-train machine learning models in an unsupervised fashion – for example, learning word embeddings [1]. However, large labeled datasets are still often required to successfully train complex models such as deep neural networks, collecting them remain an obstacle for many tasks.

In particular, a fundamental application in Natural Language Processing (NLP) is Named Entity Recognition (NER), which aims to delimit and categorize mentions to entities in text. Currently, deep neural networks present state-of-the-art results for NER, but require large amounts of annotated data for training.

Unfortunately, such datasets are a scarce resource whose construction is costly due to the required human-made, word-level annotations. In this work we propose a method to construct labeled datasets **without human supervision** for NER, using public data sources structured according to Semantic Web principles, namely, DBpedia and Wikipedia.

Our work can be described as constructing a massive, weakly-supervised dataset (i.e. a silver standard corpora). Using such datasets to train predictors is typically denoted *distant learning* and is a popular approach to training large deep neural networks for tasks where manually-annotated data

is scarce. Most similar to our approach are [2] and [3], which automatically create datasets from Wikipedia – a major difference between our method and [3] is that we use an auxiliary NER predictor to capture missing entities, yielding denser annotations.

Using our proposed method, we generate a new, massive dataset for Portuguese NER, called SESAME (Silver-Standard Named Entity Recognition dataset), and experimentally confirm that it aids the training of complex NER predictors.

The methodology to automatically generate our dataset is presented in Section III. Data preprocessing and linking, along with details on the generated dataset, are given in Section IV. Section V presents a baseline using deep neural networks.

II. DATA SOURCES

We start by defining what are the required features of the public data sources to generate a NER dataset. As NER involves the delimitation and classification of named entities, we must find textual data where we have knowledge about which entities are being mentioned and their corresponding classes. Throughout this paper, we consider an entity class to be either person, organization, or location.

The first step to build a NER dataset from public sources is to first identify whether a text is *about* an entity, so that it can be ignored or not. To extract information from relevant text, we link the information captured by the DBpedia [4] database to Wikipedia [5] – similar approaches were used in [6]. The main characteristics of the selected data sources, DBpedia and Wikipedia, and the methodology used for their linkage are described in what follows next.

A. Wikipedia

Wikipedia is an open, cooperative and multilingual encyclopedia that seeks to register in electronic format knowledge about subjects in diverse domains. The following features make Wikipedia a good data source for the purpose of building a NER dataset.

- **High Volume** of textual resources built by humans
- **Variety** of domains addressed
- **Information boxes:** resources that structure the information of articles homogeneously according to the subject

¹Available at <https://sesame-pt.github.io>

- **Internal links:** links a Wikipedia page to another, based on mentions

The last two points are key as they capture human-built knowledge about text is related to the named entities. Their relevance is described in more detail ahead.

1) *Infobox:* Wikipedia infoboxes [7] are fixed-format tables, whose structure (key-value pairs) are dictated by the article's type (e.g. person, movie, country) – an example is provided in Figure 1. They present structured information about the subject of the article, and promote structure reuse for articles with the same type. For example, in articles about people, infoboxes contain the date of birth, awards, children, and so on.

Through infoboxes, we have access to relevant human-annotated data: the article's categories, along with terms that identify its subject e.g. name, date of birth. In Figure 1, note that there are two fields that can be used to refer to the entity of the article: "Nickname" and "Birth Name".

Manfred von Richthofen	
Birth name	Manfred Albrecht Freiherr von Richthofen
Nickname(s)	"Red Baron"
Born	2 May 1892
Died	21 April 1918 (aged 25)
Allegiance	German Empire
Rank	Captain
Relations	Lothar von Richthofen (brother) Wolfram von Richthofen (cousin) see Richthofen for more

Fig. 1. Example of a Wikipedia infobox for a person entity. It consists of a key-value table whose keys depend on the type of the corresponding entity – for a person entity, common keys include name, birth date, and so on.

Infoboxes can be exploited to discover whether the article's subject is an entity of interest – that is, a person, organization or location – along with its relevant details. However, infoboxes often contain inconsistencies that must be manually addressed, such as redundancies e.g. different infoboxes for person and for human. A version of this extraction was done by the DBpedia project, which extracts this structure, and identifies/repairs inconsistencies [8].

2) *Interlinks:* Interlinks are links between different articles in Wikipedia. According to the Wikipedia guidelines, only the first mention to the article must be linked. Figure 2 shows a link (in blue) to the article page of Alan Turing: following mentions to Alan Turing in the same article must not be links.

[Alan Turing](#) went on to devise the basis for modern computing.

Fig. 2. Example of an interlink (blue) in a Wikipedia article. Interlinks point to other articles within Wikipedia, and follow the guideline that only the first mention to the article should contain a link.

While infoboxes provide a way to discover relevant information about a Wikipedia article, analyzing an article's interlinks provide us access to referenced entities which are not the page's main subject. Hence, we can parse every article on Wikipedia while searching for interlinks that point to an entity article, greatly expanding the amount of textual data to be added in the dataset.

B. DBpedia

DBpedia extracts and structures information from Wikipedia into a database that is modeled based on semantic Web principles [9], applying the Resource Description Framework (RDF). Wikipedia's structure was extracted and modelled as an ontology [10], which was only possible due to infoboxes.

The DBpedia ontology focused on the English language and the extracted relationships were projected for the other languages. In short, the ontology was extracted and preprocessed from Wikipedia in English and propagated to other languages using interlinguistic links. Articles whose ontology is only available in one language are ignored.

An advantage of DBpedia is that manual preprocessing was carried out by project members in order to find all the relevant connections, redundancies, and synonyms – quality improvements that, in general, require meticulous human intervention. In short, DBpedia allows us to extract a set of entities where along with its class, the terms used to refer to it, and its corresponding Wikipedia article.

III. BUILDING A DATABASE

The next step consists of building a structured database with the relevant data from both Wikipedia and DBpedia.

A. DBpedia data extraction

Data from DBpedia was collected using a public service access [11]. We searched over the following entity classes: people, organizations, and locations, and extracted the following information about each entity:

- The entity's class (person, organization, location)
- The ID of the page (Wiki ID)
- The title of the page
- The names of the entity. In this case the ontology varies according to the class, for example, place-type entities do not have the "surname" property

B. Wikipedia data extraction

We extracted data from the same version of Wikipedia that was used for DBpedia, October 2016, which is available as *dumps* in XML format. We extracted the following information about the articles:

- Article title
- Article ID (a unique identifier)
- Text of the article (in wikitext format)

C. Database modelling

Figure 3 shows the structure of the database as a entity-relation diagram. Entities and articles were linked when either one of two linked articles correspond to the entity, or the article itself is about a known entity.

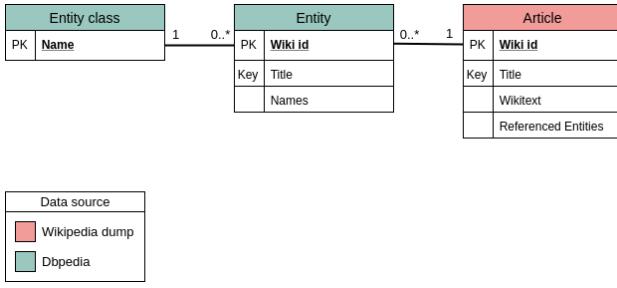


Fig. 3. Diagram of the database representing the links between entities and articles.

IV. PREPROCESSING

A. Wikitext preprocessing

We are only interested in the plain text of each Wikipedia article, but its Wikitext (language used to define the article page) might contain elements such as lists, tables, and images. We remove the following elements from each article's Wikitext:

- Lists, (e.g. unbulleted list, flatlist, bulleted list)
- Tables (e.g. infobox, table, categorytree)
- Files (e.g. media, archive, audio, video)
- Domain specific (e.g. chemistry, math)
- Excerpts with irregular indentation (e.g. outdent)

B. Section Filtering

Wikipedia guidelines include sets of suggested sections, such as early life (for person entities), references, further reading, and so on. Some of the sections have the purpose of listing related resources, not corresponding to a well structured text and, therefore, can be removed with the intent to reduce noise. In particular, we remove the following sections from each article: "references", "see also", "bibliography", and "external links".

After removing noisy elements, the Wikitext of each article is converted to raw text. This is achieved through the tool MWparser [12].

C. Identifying entity mentions in text

The next step consists of detecting mentions to entities in the raw text. To do this, we tag character segments that exactly match one of the known names of an entity. For instance, we can tag two different entities in the following text:

John Smith travelled to Rio de Janeiro. Visited Copacabana.
 PER LOC

Note that the word "Copacabana" can also correspond to a "Location" entity. However, some entity mentions in raw text might not be identified in case they are not present in DBpedia.

D. Searching for other entities

To circumvent mentioned entities which are not present in DBpedia, we use an auxiliary NER system to detect such mentions. More specifically, we use the Polyglot [13] system, a model trained on top of a dataset generated from Wikipedia.

Each mention's tag also specifies whether the mention was detected using DBpedia or by Polyglot. The following convention was adopted for the tags:

- Annotated (Anot) - Matched exactly with one of the the entity's names in DBpedia
- Predicted (Pred) - Extracted by Polyglot

Therefore, in our previous example, we have:

John Smith travelled to Rio de Janeiro. Visited Copacabana.
 Anot-PER Anot-LOC Pred-LOC

A predicted entity will be discarded entirely if it conflicts with an annotated one, since we aim to maximize the entities tagged using human-constructed resources as knowledge base.

E. Tokenization of words and sentences

The supervised learning models explored in this paper require inputs split into words and sentences. This process, called tokenization, was carried with the NLTK toolkit [14], in particular the "Punkt" tokenization tool, which implements a multilingual, unsupervised algorithm [15].

First, we tokenize only the words corresponding to mentions of an entity. In order to explicitly mark the boundaries of each entity, we use the BIO format, where we add the suffix "B" (begin) to the first token of a mention and "I" (inside) to the tokens following it. This gives us:

John Smith travelled to Rio de Janeiro. Visited Copacabana.
 B-PER I-PER B-LOC I-LOC I-LOC B-LOC

Second, we tokenize the remaining text, as illustrated by the following example: w_i denotes a word token, while s_i corresponds to a sentence token.

John Smith travelled to Rio de Janeiro .
 B-PER I-PER B-LOC I-LOC I-LOC
 w_0 w_1 w_2 w_3 w_4 w_5 w_6 w_7
 s_0

However, conflicts might occur between known entity tokens and the delimitation of words and sentences. More specifically, tokens corresponding to an entity must consist only of entire words (instead of only a subset of the characters of a word), and must be contained in a single sentence. In particular, we are concerned with the following cases:

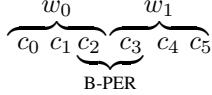
- (1) Entities which are not contained in a single sentence:

w_0 w_1 w_2 w_3
 O B-PER I-PER O
 s_0 s_1

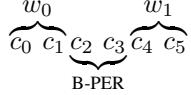
In this case, w_1 and w_2 compose a mention of the entity which lies both in sentence s_0 and s_1 . Under these circumstances, we concatenate all sentences that contain the entity, yielding, for the previous example:

w_0 w_1 w_2 w_3
 O B-PER I-PER O
 s_0

(2) Entities which consist of only subsets (some characters) of a word, for example:



In this case, we remove the conflicting characters from their corresponding word tokens, resulting in:



F. Dataset structure

The dataset is characterized by lines corresponding to words extracted from the preprocessing steps described previously, following the BIO annotations methodology.

Each word is accompanied with a corresponding tag, with the suffix PER, ORG or LOC for person, organization, and location entities, respectively. Moreover, word tags have the prefix "B" (begin) if the word is the first of an entity mention, "I" (inside) for all other words that compose the entity, and "O" (outside) if the word is not part of any entity. Blank lines are used to mark the end of an sentence. An example is given in Table I.

TABLE I

EXAMPLE OF A SENTENCE ("JOHN SMITH WENT TO RIO DE JANEIRO") FOLLOWING THE STRUCTURE OF THE PROPOSED DATASET, WHERE EACH LINE CONSISTS OF A TOKEN AND A CORRESPONDING TAG IN BIO FORM.

John	B-PER
Smith	I-PER
went	O
to	O
Rio	B-LOC
de	I-LOC
Janeiro	I-LOC
.	O

G. Semantic Model

Since our approach consists of matching raw text to a list of entity names, it does not account for context in which the entity was mentioned. For example, while under a specific context a country entity can exert the role of an organization, our method will tag it as a location regardless of the context. Therefore, our approach delimits an entity mention as a semantic object that does not vary in according to the context of the sentence.

H. SESAME

By following the above methodology on the Portuguese Wikipedia and DBpedia, we create a massive silver standard dataset for NER. We call this dataset SESAME (Silver-Standard Named Entity Recognition dataset). We then proceed to study relevant statistics of SESAME, with the goal of:

- 1) Acknowledging inconsistencies in the corpus, e.g. sentence sizes
- 2) Raising information relevant to the calibration and evaluation of model performance e.g. proportion of each

entity type and of each annotation source (DBpedia or auxiliary NER system)

We only consider sentences that have annotated entities. After all, sentences with only parser extraction entities do not take advantage of the human discernment invested in the structuring of the data of DBpedia.

1) *Sentences*: SESAME consists of **3,650,909** sentences, with lengths (in terms of number of tokens) following the distribution shown in Figure 4. A breakdown of relevant statistics, such as the mean and standard deviation of sentences' lengths, is given in Table II.

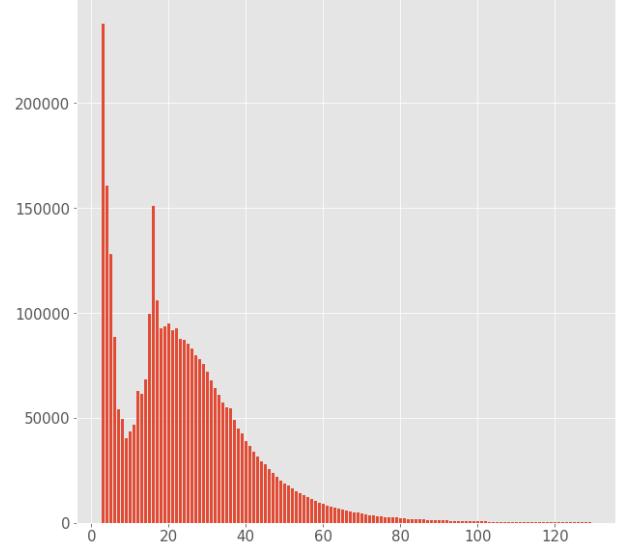


Fig. 4. Number of occurrences for different sentence lengths (number of tokens) in the generated corpus, with a mean of approximately 24 words.

TABLE II
STATISTICS ON THE LENGTH OF SENTENCES, MEASURED IN NUMBER OF TOKENS, ACROSS SESAME.

Metric	
Total	3,650,909
μ Mean	24.04
σ Standard deviation	20.41
Min	1
Max	8437
Q_1 Percentile 25%	11
Q_2 Percentile 50%	21
Q_3 Percentile 75%	33

2) *Tokens*: SESAME consists of **87,769,158** tokens in total. The count and proportion of each entity tag (not a named entity, organization, person, location) is given in III.

Not surprisingly, the vast majority of words are not related to an entity mention at all. The statistics among words that are part of an entity mention are given in Table IV, where over half of the entity mentions are of the type location.

Table V shows a size comparison between SESAME and popular datasets for Portuguese NER.

TABLE III
FREQUENCY OF EACH TAG ACROSS SESAME.

Class	Total	%
Not NE	81,357,679	92.69%
Organization	1,053,298	1.20%
Person	1,878,838	2.14%
Location	3,479,343	3.96%

TABLE IV
FREQUENCY OF EACH ENTITY TAG (ALL EXCEPT FOR "O", "OUTSIDE")
ACROSS SESAME.

Class	Total	%
Organization	1,053,298	16.42%
Person	1,878,838	29.30%
Location	3,479,343	54.26%

3) *Entity origin*: Table VI presents the proportion of matched and detected mentions for each entity type – recall that tagged mentions have either been matched to DBpedia (hence have been manually annotated) or have been detected by the auxiliary NER system Polyglot.

As we can see, the auxiliary predictor increased the number of tags by 33% relatively, significantly increasing the number of mentions of type organization and person – which happen to be the least frequent tags.

V. BASELINE

To construct a strong baseline for NER on the generated SESAME dataset and validate the quality of datasets generated following our method, we use a deep neural network that proved to be successful in many NLP tasks. Furthermore, we check whether adding the generated corpus to its training dataset provides performance boosts in NER benchmarks.

A. Datasets

In order to have a fair evaluation of our model, we use human-annotated datasets as validation and test sets.

TABLE V
COMPARISON BETWEEN NER DATASETS FOR PORTUGUESE.

Corpus	Sentences	Tokens
SESAME	3,650,909	87,769,158
Paramopama	12,500	310,000
First HAREM	5,000	80,000
Second HAREM	3,500	100,000
WikiNER	125,821	2,830,000

TABLE VI
FREQUENCY OF TOKENS ORIGINATED FROM ANNOTATION AND FROM
DETECTION BY THE AUXILIARY NER PREDICTOR.

Class	Annotated	Detected
All Classes	75.60%	24.39%
Organization	67.69%	32.30%
Person	65.76%	34.23%
Location	83.31%	16.68%

We use the first HAREM and miniHAREM corpus, produced by the Linguateca project [16], as gold standard for model evaluation. We split the dataset in the following manner:

- Validation: 20% of the first HAREM
- Test: 80% of the first HAREM, plus the mini HAREM corpus

Another alternative is to use the Paramopama corpus which is larger than the HAREM and miniHAREM datasets. However, it was built using an automatic refinement process over the WikiNER corpus, hence being a silver standard dataset. We opted for the smaller HAREM datasets as they have been manually annotated, rendering the evaluation fair.

The HAREM corpus follows a different format than the one of SESAME: it uses a markup structure, without a proper tokenization of sentences and words. To circumvent this, we convert it to BIO format by applying the same tokenization process used for generating our dataset.

B. Evaluation

The standard evaluation metric for NER is the F_1 score:

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R}$$

where P stands for precision and R for recall. *Precision* is the percentage of entity predictions which are correct, while *Recall* is the percentage of entities in the corpus that are correctly predicted by the model.

Instead of the standard F_1 score, we follow the evaluation proposed in [17], which consists of a modified *First HAREM* F_1 score used to compare different models. Our choice is based on its wide adoption in the Portuguese NER literature [18], [19], [20].

In particular, for the *First HAREM* F_1 score: (1) as the corpus defines multiple tags for the same segments of the text, the evaluation also accepts multiple correct answers; (2) partial matches are considered and positively impact the score.

In this work, the configuration of the First HAREM evaluation procedure only considers the classes “person”, “location” and “organization”. Also, the HAREM corpus has the concept of “subtype” e.g. an entity of the type “person” can have the subtype “member”. We only perform evaluation considering the main class of the entity.

C. Baseline results

We performed extensive search over neural network architectures along with grid search over hyperparameter values. The model that yielded the best results consists of: (1) a word-level input layer, which computes pre-trained word embeddings [21] along with morphological features extracted by a character-level convolutional layer [22], (2) a bidirectional LSTM [23], (3) two fully-connected layers, and (4) a conditional random field (CRF). Table VII contains the optimal found hyperparameters for the network.

Additionally, the baseline was developed on a balanced resample of SESAME with a total of 1,216,976 sentences. The model also receives additional categorical features for each

TABLE VII
HYPERPARAMETERS FOR THE DEEP NETWORK THAT YIELDED OPTIMAL RESULTS FOR NER ON SESAME, FOUND THROUGH GRID SEARCH.

Word embeddings	
Embeddings (pre-trained)	GloVe
Size	100
Character-level convolution	
Window size	3
Filters	30
Dropout (output)	0.5
biLSTM	
Hidden Units	150
Variational Dropout	0.25
Dropout (input)	0.5
Fully-connected Layers	
Depth	2
Activation	tanh
Hidden units	100
Optimization	
Optimizer	SGD
Momentum	0.9
Gradient clipping	5.0
Mini-batch size	10
Learning rate	0.005

word, signalizing whether it: (1) starts with a capital letter, (2) has capitalized letters only, (3) has lowercase letters only, (4) contains digits, (5) has mostly digits ($> 50\%$) and (6) has digits only.

With the goal of evaluating whether SESAME can be advantageous for training NER classifiers, we compare the performance of the neural network trained with and without it. More specifically, we train neural networks on the HAREM2 [24] dataset, on SESAME, and on the union of the two – Table VIII shows the test performance on the first HAREM corpus. As we can see, while SESAME alone is not sufficient to replace a human-annotated corpus (the F_1 score of the network trained on the SESAME is lower than the one trained on the HAREM2 corpus), it yields a boost of 1.5 in the F_1 score when used together with the HAREM2 dataset.

TABLE VIII
BASELINE RESULTS: USING SESAME ALONG WITH A HUMAN-ANNOTATED CORPUS BOOSTS F_1 PERFORMANCE.

Training Data	F_1	Precision	Recall
SESAME	67.49	77.53	59.76
HAREM2	72.72	75.28	70.32
SESAME + HAREM2	74.22	77.58	71.14

VI. CONCLUSION

Complex models such as deep neural networks have pushed progress in a wide range of machine learning applications, and enabled challenging tasks to be successfully solved. However, large amounts of human-annotated data are required to train such models in the supervised learning framework, and remain the bottleneck in important applications such as Named Entity Recognition (NER). We presented a method to generate a massively-sized labeled dataset for NER in an automatic fashion, without human labor involved in labeling – we do

this by exploiting structured data in Wikipedia and DBpedia to detect mentions to named entities in articles.

Following the proposed method, we generate SESAME, a dataset for Portuguese NER. Although not a gold standard dataset, it allows for training of data-hungry predictors in a weakly-supervised fashion, alleviating the need for manually-annotated data. We show experimentally that SESAME can be used to train competitive NER predictors, or improve the performance of NER models when used alongside gold-standard data. We hope to increase interest in the study of automatic generation of silver-standard datasets, aimed at distant learning of complex models. Although SESAME is a dataset for the Portuguese language, the underlying method can be applied to virtually any language that is covered by Wikipedia.

REFERENCES

- [1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.
- [2] J. Nothman, N. Ringland, W. Radford, T. Murphy, and J. R. Curran, “Learning multilingual named entity recognition from wikipedia,” *Artificial Intelligence*, 2013.
- [3] C. Weber and R. Vieira, “Building a corpus for named entity recognition using portuguese wikipedia and dbpedia,” 2014.
- [4] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. G. Ives, “Dbpedia: A nucleus for a web of open data,” in *ISWC/ASWC*, 2007.
- [5] “Wikipedia,” <https://pt.wikipedia.org/>, accessed: 11/8/2018.
- [6] Y. Hahn, J. Park, K. Lim, Y. Kim, D. Hwang, and K.-S. Choi, “Named entity corpus construction using wikipedia and dbpedia ontology,” in *LREC*, 2014.
- [7] “Infocaxa,” <https://pt.wikipedia.org/wiki/Ajuda:Infocaxa>, accessed: 11/8/2018.
- [8] “Dbpedia datasets,” <https://wiki.dbpedia.org/services-resources/datasets/dbpedia-datasets> h434-10, accessed: 21/7/2018.
- [9] T. Berners-Lee, J. Hendler, and O. Lassila, “The semantic web,” *Scientific american*, 2001.
- [10] “Dbpedia datasets,” <http://dbpedia.org/ontology/>, accessed: 21/7/2018.
- [11] “Dbpedia datasets,” <https://dbpedia.org/sparql>, accessed: 21/7/2018.
- [12] “mwparserfromhell,” <https://mwparserfromhell.readthedocs.io/en/latest/>, accessed: 11/8/2018.
- [13] R. Al-Rfou, V. Kulkarni, B. Perozzi, and S. Skiena, “Polyglot-ner: Massive multilingual named entity recognition,” in *SDM*, 2015.
- [14] S. Bird and E. Loper, “Nltk: the natural language toolkit,” in *ACL*, 2004.
- [15] T. Kiss and J. Strunk, “Unsupervised multilingual sentence boundary detection,” *Comput. Linguist.*, 2006.
- [16] D. Santos, “Caminhos percorridos no mapa da portuguesificação: A linguoteca em perspectiva,” *Linguamática*, 2009.
- [17] D. Santos, N. Seco, N. Cardoso, and R. Vilela, “HAREM: An advanced NER evaluation contest for Portuguese,” in *LREC*, 2006.
- [18] C. N. d. Santos and V. Guimaraes, “Boosting named entity recognition with neural character embeddings,” *arXiv preprint arXiv:1505.05008*, 2015.
- [19] C. N. dos Santos and R. L. Milidiú, “Entropy guided transformation learning,” in *Foundations of Computational Intelligence*, 2009.
- [20] C. N. Aranha, “O cortex e a sua participação no harem,” *Reconhecimento de entidades mencionadas em português: Documentação e actas do HAREM, a primeira avaliação conjunta na área. Linguoteca*, 2007.
- [21] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *EMNLP*, 2014.
- [22] C. D. Santos and B. Zadrozny, “Learning character-level representations for part-of-speech tagging,” in *ICML*, 2014.
- [23] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional lstm and other neural network architectures,” *Neural Networks*, 2005.
- [24] C. Freitas, C. Mota, D. Santos, H. G. Oliveira, and P. Carvalho, “Second harem: Advancing the state of the art of named entity recognition in portuguese,” in *LREC*, 2010.