# IDENTIFYING AND CHARACTERIZING CONCEPTS IN UNSTRUCTURED TEXTS USING AUTOMATIC ANNOTATION

Tiago Fraga[1], Orlando Belo[1] and Anabela Barros[2]
[1]*ALGORITMI Research Centre / LASI, School of Engineering, University of Minho, 4710-059 Braga, Portugal*
[2]*CEHUM, Centre for Humanistic Studies, University of Minho, 4710-059 Braga, Portugal*

**ABSTRACT**

Text annotation is an important and useful activity in semantic text analysis processes. The introduction of notes and marks in texts is one of the most common ways of valuing the content and revealing the semantics of a text, allowing its readers to have a more concrete idea of what is expressed in it. For a long time, text annotation processes were done manually often carried out in an *ad hoc* manner, without using a concrete method. It was a very time-consuming and labor-intensive process of different natures. Today, the great development of natural language processing, machine learning and mining text, promoted a strong emergence of applications in several domains. Text annotation was no exception. Through the combination of natural language processing and machine learning mechanisms, it is possible to develop systems that analyze texts, written in natural language, identifying words, creating contexts, and discovering and maintaining tags, their relationships and annotations, in an automatic way. In this paper, we present an automatic annotation system conceived and developed specifically for tagging the texts of the Book of Properties, a codex containing the inventory of the Archbishop's Table of Braga's properties (Portugal) in the 17th century. In addition to a general characterization of the system, we also describe the various stages of its annotation process with references taken from the Book of Properties.

**KEYWORDS**

Annotation Systems, Natural Language Processing, Machine Learning, Text Mining, Automatic Tagging, Data Analysis

## 1. INTRODUCTION

The area of semantic analysis of unstructured texts (Sinoara et al., 2017) has evolved a lot in recent years. The great emergence of tools and applications, particularly in the fields of text interpretation or content retrieval and extraction (Cornolti *et al.*, 2013) (Chu *et al.*, 2012) clearly proves this evolution. The increasing interest of researchers from various scientific domains, in knowledge contained in documents available in archives, libraries, databases, or in the Web, has promoted the development of much diversified solutions for the semantic analysis of these documents. Some of these solutions involve researching and application of text annotation techniques, structured or not, as a way of revealing and tagging, manually or automatically, textual elements with a particular semantic meaning. The need to make this annotation process more flexible, as well as to increase its speed and effectiveness, led to the development of systems to help the annotation of texts, both in initiatives that aimed at manual annotation and in other, more interesting and challenging ones, the use of automatic annotation systems.

A text annotation system (Moraes and Lima, 2008) is a mechanism capable of identifying concepts and relationships using a combination of text mining, natural language processing and machine learning techniques. In practice, this makes possible to develop text analysis systems — written in natural language — that are capable of (semi)automatically annotating sets of words, as well as their respective contexts of use, identifying concepts, characterizing them and establishing possible relationships. The extraction of concepts from texts is fundamental for any information retrieval process. However, it is difficult to perform, since obtaining annotated elements from a set of texts is a very complicated task. Automatic text annotation systems perform tagging tasks with a greater degree of speed and efficiency compared to conventional manual annotation processes. For cases involving a large volume of texts, manual annotation, even carried out by specialists with great experience, has become a task too expensive in time and resources (Cai and Hofmann, 2003). This is not

acceptable, because it is very slow and inefficient, taking into account the operational and analysis requirements of current specialists and scholars.

In research and analysis of texts content, two of the most demanding tasks in annotation processes, especially when dealing with unstructured texts, written in natural language, it is usual to apply natural language processing and machine learning techniques. When properly combined, these techniques allow for preparing texts, identifying sets of relevant words, establishing research and relationship patterns, or discovering concepts and their relationships. All these elements are very useful for establishing definition tags for text contexts. Furthermore, these techniques eliminate, or at least mitigate, most of the disadvantages associated with the preparation and annotation of texts manually. However, the use of these techniques does not remove all manual tasks in the general annotation process, such as, for example, checking and validating tags, for adjusting parameters and improving the quality of the annotation process.

In this work, we present an automatic annotation system for unstructured texts, which we developed for tagging texts of a very singular document: the Book of Properties (Barros, 2019) (Barros, 2021). This manuscript of the 17th century contains the inventory of the properties of the Archbishop's Table of Braga (Portugal) in the beginning of that period. It describes in detail all the rustic and urban properties of several districts located mostly in the north of Portugal, as well as presents the rents and payments due to their lease. The annotation of this book is very important for identifying and highlighting particular elements (people names, place names, types of land, properties, degrees of kinship, etc.) and application contexts, for establish automatically maps of tag relationships to discover similar contents or the renewal of their definitions over time. Additionally, we will describe each of the processing stages of the system and illustrate some of the results achieved. The remaining part of this paper is structured as follows. Section 2 describes some related work in the area of text annotation, while section 3 exposes and describes the system we developed, and how it works in each execution stage. Finally, section 4 presents some conclusions and future work.

## 2. TEXT ANNOTATION

A text can be seen as a set of sentences that convey some kind of information to its readers. However, if there are no additional information elements, markings or notes throughout the text, its interpretation will differ more than expected between different readers. This means that a particular word or sentence can convey different ideas to different readers, especially in manuscript texts from past centuries. Text annotation (Gosal, 2015) is a task that adds value and specification to texts. It is possible to provide additional elements about the text, using a well-defined set of tags for helping readers in its interpretation. For example, when analyzing a conventional text, numerous punctuation marks are easily to found. They correspond to a set of marks, whose function is to help readers of that text in their reading and interpretation; however, these marks are quite different in old unpublished manuscripts. Tags also have this function, being defined according to the content of the texts. Usually, we call them as “content-oriented annotations” (Ferreira, 2011). The use of tags intends to indicate the presence of different types of elements in the text, such as people names, nicknames, place names, professions, or products, among many other information elements. The greater the presence of labels in a text, the greater the percentage of annotated text. Consequently, the easier it is the interpretation and understanding of the text. Text annotation is often performed in an *ad hoc* manner, which obviously does not reveal to the reader the true usefulness of this technique. However, when we done the annotation process methodically, and guided by the context and content of the text, highlighting its main ideas, the understanding of the texts increases significantly. Thus, text annotation can help readers to analyze their own ideas and thoughts, allowing them to have a direct and faster access to the most pertinent ideas and elements of a text (Lynch, 2021).

In recent years, a lot of work was made in the field of semantic analysis of unstructured texts, which has given rise to many applications, in areas so diverse as text interpretation or content retrieval and extraction. Perhaps the main reason for this development was the high number of documents available in companies, libraries, databases and websites. For this volume of data, manual annotation of texts is not feasible, as it becomes too exigent in terms of time and money, requires the involvement of specialized human resources and is not always carried out in a methodical and efficient way, which may cause errors and failures in annotation processes (Cai and Hofmann, 2003). For these reasons, many researchers have promoted the development processes of automatic annotation systems, as a viable way for a large part of the annotation problems in

unstructured texts (Chu *et al.*, 2012). These initiatives quickly revealed a significant increase in the speed of execution of the annotation process and in the efficiency rate of recognition of textual elements of interest, in the identification and extraction of entities in texts, as well as dispensing with a large part of non-specialized human work. The development that has taken place in recent years in the areas of natural language processing, machine learning and text mining also contributed to this. The combination and use of tools from these areas makes possible to develop systems for analyzing structured or unstructured texts, written in natural language, taking note of words, contexts of use, concepts and their relationships. In some cases, they evolve as they do more and more annotation processes, that is, they have the ability to learn. We are, therefore, in another dimension of text annotation. However, they remain complex systems and difficult to implement (Finlayson and Erjavec, 2017). Despite these difficulties, the interest in automatic annotation systems by the research community increased significantly, according to the results observed in the applications that were being developed (Cornolti *et al*., 2013). Today, we can say that the growth in the use of these tools and applications is remarkable. We can take as an example the area of big data systems, in which analysts of e-commerce sites or social networks use automatic annotation of website documents to establish, for example, user behavior patterns or discover trends in the purchase of goods and services. Regardless of the development context and application area, an automatic annotation system must work according to a robust annotation scheme, which clearly defines the annotation strategy and rules, as well as the different tags to use. This work must be carried out with the support of specialists in annotation and in the field of application of the texts. In addition, we need to select suitable computational tools for the annotation process, which allow for the recognition and division of words, identifying and classifying later concepts and their relationships. Several academic and business initiatives have made significant efforts in the development of automatic annotation systems and their application in areas of great interest. For example: the EXACT system (Chen *et al*., 2019), developed at the University of Zhejiang, China, allows extracting entities from textual documents, through exploratory annotation operations that create attributes and associate them with tags; the Elketron system (Refinitiv, 2019), developed in an industrial context by Refinitiv, which has a concrete application in the areas of economics and financial markets; the Tagtop system (TagTop, 2022), which provides means for training a machine learning model for personalized annotation; or the UBIAI system (UBIAI, 2022), which is capable of performing Named Entity Recognition (NER) (Marrero *et al*., 2013), relation extraction and document classification tasks. In addition to these examples, there are companies such as Uber, Apple or Microsoft using several types of annotation procedures to analyze their reputation in the field of social networks.

# 3. THE ANNOTATION SYSTEM

Usually, we develop an annotation system for very specific application domains, in order to facilitate the processes of interpretation and research the contents of texts. Although, we can do manually a text annotation process, its automation is highly desired, since, when possible, it simplifies the annotation process and drastically reduces its completion time, as already referred. Furthermore, when properly configured, it automatically establishes relationships that exist between discovered (and annotated) tags and the various textual elements that may be associated with them. The construction of an automatic annotation system requires the use of natural language processing mechanisms, commonly combined with machine learning mechanisms. This combination of technologies enables the development of systems capable of analyzing texts in natural language, with the ability to identify sets of words and create contexts of use, according to an established set of prerequisites, as well as discover and manage tags through techniques of word processing, more expeditiously and with a very high level of correction.

## 3.1 The Application Domain

Over the last few years, we have been developing a document management system (Barros et al., 2020) to accommodate the content of the Book of Properties, an impressive manuscript from the 17th century, which contains a detailed inventory of the rural and urban properties of the archbishops of Braga (Portugal). According to Barros (2021), the manuscript, having 1288 large pages, presents in detail and precision the inventoried properties of the Portuguese counties of Valença, Vila Real, Chaves and Braga, and also some properties in Porto and Santarém, in Portugal, and Galicia, in Spain. Due to the detail of this information, it is

possible that researchers or readers of the Book of Properties, if they have roots in these locations, can find references or properties of their ancestors. The codex has a very impressive size, weight and binding, being handwrite in a very cultured and regular calligraphy, and presenting a detailed and extensive description of all the properties, rents and pensions of the Archbishop's Table of Braga. Only philologists, or researchers with experience in reading this type of documents, can read and interpret the information contained in the book. Other scholars, without this experience, may have difficulties in reading and handling the manuscript. In addition, the fact that the codex is not reproduced in digital format, and can only be consulted in person at the Braga District Archive, means that only one person can study it at a time. On the other hand, people far from the city have to travel to consult the book, which can sometimes lead to the abandonment of their study. With the increase in the number of researchers and due to the importance of the codex, the opportunity arose to store the contents of the manuscript, edited since 2015, in a digital format, overcoming all the difficulties for the study of the book mentioned. The document management system we implemented is a Web based application having the capacity to store, index and search the edited texts of the Book of Properties. It is a cross-platform system designed to run on the most relevant operating systems on the market. It is a client-server system, in which the backend received the native document management services, and the client sustain the user interaction and document loading and search services, running in conventional Web browsers. In practice, it is a web based application specially designed for the reception, processing and analysis of the texts contained in the Book of Properties, which has a set of specially created mechanisms for importing, cataloguing, modifying, removing, and analyzing texts stored in a document store.

## 3.2 The Annotation Mechanisms

In order to improve the research and analysis of texts of the document management system developed, as well as to reveal the various relationships between its various textual elements, a set of mechanisms, specifically oriented for annotating the document database, was incorporated into its structure. With the introduction of such mechanisms, the system received new means for incorporating tags into texts, which we consider relevant for the study of the Book of Properties, discovering and cataloguing anthroponyms, toponyms and microtoponyms, degrees of kinship, products, plants, properties and their location, etc. In this way, it was possible to maintain a base of tags indexing the most relevant information contained in the book. Furthermore, based on the tag specifications, the annotation mechanisms allow for analyzing the texts that are contained in the system and, similarly, suggesting a global annotation strategy for other tags, as well as generating a map of tag relationships that can be used to discover similar content. The quantity and diversity of the elements mentioned in the Book of Properties are surprising: all the names and surnames, nicknames, villages, professions, types of land and buildings, natural resources, cultures and ways of cultivation, among many others, are very important for the study and learning of geography, culture, agriculture, economy, architecture, religion and Portuguese language in the 17th century. The annotation of these elements expressively reveals their location in time and space, as well as their potential relationships, facilitating the study of the book and providing researchers, linguists, teachers and students with a valuable tool to reach and reinforce the knowledge about the codex.

When we made a detailed study of several existing automatic annotation systems – *e.g*. Benikova *et al*., 2010), Ahmadi and Moradi (2015), Chen *et al*. (2019) or Dias *et al*. (2020) – we analyzed their most relevant models and features, as well as identified their functional architectures. In general, all these systems had very similar elements. Combining some of these models and their respective features, we sketched the working model of our automatic annotation system. Despite of its design having in mind its application to the Book of Properties, we consider that the system obtained is of widespread application, incorporating the most relevant functionalities that should be present in an automatic annotation system. The automatic annotation process we developed considers four working stages, namely: selection, tagging, validation and learning. Figure 1 presents an algorithm, in pseudo-code, which shows the most relevant operations performed by the system in each execution stage. In the next section, we will describe in detail each one of the mentioned stages.

```
def mainExecution():
   // 1.Selection
   text = selectAndReadText();
   // 2.Tagging
   tagsList = readTagsInfo();
   modernizationRules = readModernizationRules();
   textFormmated =
     formatText(modernizationRules, text);
   wordsList =
     applyTokenizationAndClassification(textFormmated);
   // 2.1 Automatic Tagging
   wordsCombination = calculateCombination(wordsList);
   for word in wordsCombination:
     for tag in tagList:
        dictionary = tag.getTermsDictionary()
        if word in dictionary:
           annotateWord(word, wordsList)
// 2.1 Manual Tagging
startConditionList = readStartConditions(tagList)
stopConditionList = readStopConditions(tagList)
for word in wordsList:
  if word in startConditionList:
     startPosition = word.getPosition()
     stopPosition = findStopWordInText(startPosition,
       wordsList, stopConditionList)
     annotateWordsBetweenStartAndStop(startPosition,
       stopPosition,wordsList)
// 3. Valitation
annotatedText = showText(wordsList)
// 4.Learning
saveAnnotatedText(annotatedText)
updateDictionaries(annotatedText)
updateIndex(annotatedText)
```

Figure 1. The annotation process in pseudo-code

### 3.2.1 Selecting Texts

This first stage (selection) is the simplest part of the system. Here, the system simply displays the texts that are available for annotation. When selected, texts are loaded into memory from a specific data collection in the system's document store. This step is not very demanding in terms of computational resources (memory and processor). Despite the large number of texts in the Book of Properties, each one of them, individually, does not have a large dimension, which makes the annotation process simple and fast.

### 3.2.2 Tagging Texts

During the tagging stage, the system performs five very distinct specific tasks on the text being processed, namely its classification, modernization, annotation using dictionaries (automatic tagging), annotation using NER (manual tagging) and, finally, aggregation. In the first task, classification, the system divides the words and aggregates them in a specific data structure, for using them later in other annotation tasks. This data structure receives the classification of words and stores the position of each of them in the text, as well as their homogenized form (in lowercase and without accents). The homogenized form is used throughout the dictionary search process, along with a logical value, which informs whether the word has already been annotated or not. The division and aggregation of the words in the text were implemented with the natural language processing tool LinguaKit (Gamallo and Garcia, 2017). This tool provided us mechanisms for diving text into words and making its morphological classification, so that it was possible to verify if a word would be a noun, an adjective, a verb, an adverb, etc. – the identification of these elements is essential in any annotation process. Then, the system performs some modernization of the orthography. As previously mentioned, the Book of Properties was written in classical Portuguese. The dictionaries of place names we used, for example, are in contemporary Portuguese, which imposes updating the spelling of the text. As most words do not follow current orthographic standards, and present multiple variants, or graphic forms, the detection of entities is difficult. To overcome this difficulty, we developed an automatic process for modernizing words, which works based on dictionaries and on a set of previously established lexical updating rules (Table 1). These rules match the classical patterns with the corresponding contemporary equivalents. The results of the word modernization process, the updated text and the data structure of the conversion dictionary, are stored in the system's document store, so that in all the time, whenever necessary, it will be possible to reverse the conversion (modernization) carried out and recover the original text. In Table 2 we can see an example of an original sentence extracted from a text (line 1) and the sentence that resulted from the modernization process (line 2).

Tagging using dictionaries is the third task of the annotation stage. Here, the system annotates entities from the word dictionaries referring to each of the tags that we want to use. For example, for tagging locations, it was necessary to create a specific dictionary – a gazetteer – with all locations in the Portuguese territory, with particular emphasis on the initial definition of annotation of locations, so that, in the future, we could apply the same or similar strategy to other types of tags. In this particular case, the dictionary was built from a specific dataset obtained from the Portuguese Public Administration Open Data portal (Dados.Gov, 2022), which disposes information about all districts, municipalities and parishes in Portugal. For the other types of tags established as essential, such as people names, types of land and houses, products, and others, we implemented similar processes.

Table 1. Some examples of spelling update rules

| Classic | Modernized | Classic | Modernized |
|---|---|---|---|
| y | i | d' | de |
| ll | l | co' | com |
| uu | uv | q' | que |
| th | t | hu' | um |
| j [ consonant ] | l | nn | n |
| [ vowel ] u [ vowel ] | v | ee | e |

Table 2. A modernized sentence from the Book of Properties

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Original > | e | de | largo | pella | banda | do | sul | sessenta | e | quatro |
| Modernized > | e | de | largo | pela | banda | do | sul | sessenta | e | quatro |

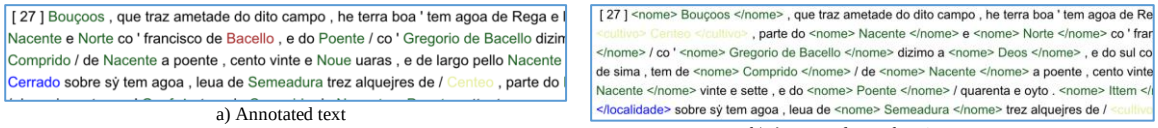
Finally, the system compares the words contained in the text with the terms registered in the tag dictionaries that are stored in the system and maintained over time. The comparison process used only words in lower case, without accents. With this strategy, it was possible to cover a greater number of cases, which was essential for increasing the level of accuracy and effectiveness of the tagging using dictionaries. Next, we have the tagging using NER task. In this task, the system identifies elements that are not present in the system dictionaries and to which no tag has been associated. The annotation performed in this task uses NER for identifying and recognizing entities in the text. As mentioned, the Book of Properties is a manuscript from the early of 17th century. As such, its texts contains references to names of people and places, lands or professions, which have changed over the years. The book also contains words that we do not use anymore, and thus automatically escape from the previous annotation process. For example, there are references to locations in the manuscript whose names and writing have simply changed. Cases like these are not present in the dictionaries. To handle them, we had to develop a process that was able to detect and record them properly, in a manual way. In this new process, we implemented a mechanism to verify the beginning and the end of class indicators. These words use to precede or belong to the term that we need to identify. For example, when processing words from the text, as soon as the system detects a start-of-class indicator it is quite likely that following words can be categorized by a specific tag. Once detected the starting point for a possible annotation, the system determines the different stopping points. When the system finds, for example, a determiner, a verb or an adjective, it defines a breakpoint. If a preposition was detected, the system checks the next word, which, if it belongs to the grammatical class of the stopping cases, determines the stop of the annotation of that case. Then, the system reuses the classification mechanism from the previous task, organizing the words in the text according to their morphological class. This allows for finding stopping cases contained in the texts and detect potential entities to annotate. At the end, we got an annotated text, having notes such as: "... e do nascente e sul com <name>Dona Maria </name> de semeadura" (in Portuguese). In this small example, the annotated name was identified based on the class start identifiers, which are the words that mark the beginning of the entity search, which in this case is "Dona". In turn, the stopping case is the preposition "de", which marks the end of the search. After identifying the referred limits, the system is able to identify the name "Dona Maria" and write it down correctly with the respective tag (<name>). After done the tagging tasks (automatic and manual), the system proceeds to the aggregation of annotated (and unannotated) words task. All the words were stored in a temporary JSON document, having a structure owning the attributes "word", "tag", "position", "annotated", "type" and "color". The "word" and "tag" attributes represent the form of the word, which may or may not contain a tag. For example, in an unannotated word, both attributes will have the same value. However, the same is no longer true for an annotated word. In this case, the "word" attribute will contain the word "surrounded" by the tag. As for the other attributes, they indicate, respectively, the position of the word in the text ("position"), whether or not the word is annotated ("annotated"), the type of tag used ("type") and the color that identify the tag in the text ("color").

### 3.2.3 Validating Tags

During the tag validation stage, the system verifies the annotation made in the previous stage, having user supervision. Based on the results of the previous stage, the system displays the tags defined and applied to the text. The user confirm the tags that have been defined, or cancel the validation process, which will cause the system to discard all the annotation work performed. The reviewing tasks do not require large computational resources. However, it obviously requires some work from system user. Additionally, the user can make new manual annotations or change one or more annotations of entities made previously. This process happens mainly when user wants to correct a tagging error or add new tags, in case of annotation failure.

### 3.2.4 Learning and Incorporating Tags

Learning is the last stage of the system. The first task of this step stores the text previously annotated in the system document store. Thus, any user will be able to view the annotated text, without having to perform any annotation task, in any access they make to the system. In a second task, the system updates the various tag indexes it has, identifying the occurrences of each of the tags in the texts. This task indicates the texts in which we can find references to relevant data elements, such as localities, professions, terrains, etc. Finally, the system updates tag dictionaries that it usually uses in the automatic annotation task. Updating dictionaries over time, after each annotation process carried out, we can say that the annotation system has some learning capabilities. This is because, during the annotation process, new entities can be discovered, from the manual annotation task or from annotations made by users in the validation state. As result, the system acquires and reflects appropriately in its storage structures that tagging actions, enriching the dictionaries and improving its own annotation process in future iterations. In Figure 2 we can see two tagging views for a (fragment of a) text: a) the annotated text, having the words annotated colored, and b) the text annotated, with the annotated words and their respective tags in sight. In this state of the annotation process, if necessary, it is possible to recover the original text. The system's document store maintains the various versions of the text created during annotation process.




a) Annotated text

b) Annotated text showing tags

Figure 2. Two distinct views of a text fragment

## 4. CONCLUSIONS AND FUTURE WORK

In this paper, we presented an automatic annotation system specially designed and implemented for tagging the texts of the Book of Properties. The system provides a set of services for the automatic identification and annotation of various data elements referred in the book, such as names, locations, professions, terrains, as well as other elements with high historical interest of a large Portuguese territory. Such elements are very important to researchers, professors and students, for studying sociocultural, economic, architectural, agricultural, linguistic and religious aspects of the 17$^{th}$ century in the north of Portugal. At this point, we have available an operational version of the system, working as planned. However, as expected, the implementation system was not simple. It raised a very diverse range of challenges, ranging from modelling to tagging. It was difficult to integrate a supervised machine-learning module into the system, due to the lack of data for training the model we designed. Once the Book of the Properties was written in classical Portuguese, it was not feasible to use current texts in the training. In addition, the editing of this codex, from handwritten to digital form, is still in progress. It is a very time-consuming task, and at this moment the edition, already resulted in four books with more than two thousand pages. During the design and development phase of the system, the number of texts stored in the system was not large. Even so, we developed the bases for training and implementing the tagging system, obtaining a nice set of annotated texts. This allowed us to demonstrate the usability and utility of the system. As future work, we intend to increase significantly the number of texts in the system's document store (a task wich is recomended only when the last edition criteria will be definitelly fixed, after the completion of the codex edition, since it will affect all the text orthography), in order to improve the automatic annotation techniques, with new tagging services and learning mechanisms, increasing system robustness and performance. With this, we expect to open the use of the system to a selected community of researchers, professors, and students.

## ACKNOWLEDGEMENT

This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020.

## REFERENCES

Ahmadi, F., Moradi, M., 2015. A Hybrid Method for Persian Named Entity Recognition. 7th Conference on Information and Knowledge Technology, IKT 2015, May. DOI: 10.1109/IKT.2015.7288806.

Barros, A., 2019. Apontamentos lexicais sobre o Livro das Propriedades ou Tombo da Mitra Arquiepiscopal de Braga: designações de terras e outros aspectos das propriedades. In Estudos de linguística histórica: mudança e estandardização, Coimbra: Imprensa da Universidade de Coimbra, pp. 393-428.

Barros, A., 2021. A edição do Livro das Propriedades ou Tombo da Mitra Arquiepiscopal de Braga. In Os sete castelos. Congresso de Homenagem a D.Rodrigo de Moura Teles, Braga.

Barros, A., Belo, O., Gomes, J., Fraga, T., Martins, R., Carvalho, J.P., 2020. A Computational Instrument for Students Accessing and Exploring The Book of Properties of The Braga Archbishop's Table (17th Century), In Proceedings of 13th Annual International Conference of Education, Research and Innovation" (ICERI'2020), 9th-10th November.

Benikova, D., Yimam, S., Santhanam, P., Biemann, C., 2010. Germa-NER : Free Open German Named Entity Recognition Tool. 1(1):31–38.

Cai, L., Hofmann, T., 2003. Text Categorization by Boosting Automatically Extracted Concepts. In SIGIR '03 Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, edited by ACM, 8. Toronto, Canada.

Chen, K., Feng, L., Chen, Q., Chen, G., Shou, L., 2019. EXACT: Attributed Entity Extraction by Annotating Texts. In ACM, editor, SIGIR'19 Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, page 4, Paris, France. DOI: 10.1145/3331184.3331391.

Chu, B., Zahari, F., Lukose, D, 2012. Benchmarking T-ANNE: Text Annotation System. In ACM, editor, i-KNOW '12 Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies, page 5, Graz, Austria. DOI: 10.1145/2362456.2362464.

Cornolti M., Ferragina, P., Ciaramita, M., 2013. A Framework for Benchmarking Entity-Annotation Systems. In WWW '13 Proceedings of the 22nd international conference on World Wide Web, page 11, Pisa, Italy. University of Pisa, Italy, ACM. DOI: 10.1145/2488388.2488411.

Dados.Gov, 2022, Portal de Dados Abertos da Administração Publica, Web Site [online] <https://dados.gov.pt/en/> [Accessed in 25 August 2022].

Dias, M., Boné, J., Ferreira, J., Ribeiro, R., Maia, R., 2020. Named entity recognition for sensitive data discovery in portuguese. Applied Sciences (Switzerland), 10(7). DOI: 10.3390/app10072303.

Finlayson, M., Erjavec, T., 2017. Overview of Annotation Creation: Processes and Tools, pages 167–191. 06 2017. DOI: 10.1007/978-94-024-0881-2 5.

Gamallo, P., Garcia, M., 2017. LinguaKit: uma ferramenta multilingue para a análise linguística e a extração de informação. Linguamática, 9(1), pages 19–28, jul. DOI:10.21814/lm.9.1.243.

Ferreira, L., 2011. Medical Information Extraction in European Portuguese. PhD Thesis, Universidade de Aveiro.

Gosal, G., 2015. A Survey on Semantic Annotation of Text. International Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, Issue 9, September.

Lynch, E., 2021. Annotating Text Strategies That Will Enhance Close Reading, 2021. [online] < https://www.sadlier.com/school/ela-blog/teaching-annotation-to-students-grades-2-8-annotating-text-strategies-that-will-enhance-close-reading> [Accessed in 25 August 2022]

Marrero, M., Urbano, J., Sánchez-Cuadrado, S., Morato, J, Gómez-Berbís, J., 2013. Named Entity Recognition: Fallacies, challenges and opportunities, Computer Standards & Interfaces, Volume 35, Issue 5, Pages 482-489. DOI: 10.1016/j.csi.2012.09.004.

Moraes, S., Lima, V., 2008. Abordagem nao Supervisionada para Extração de Conceitos a partir de Textos. In ACM, editor, WebMedia'08 Companion Proceedings of the XIV Brazilian Symposium on Multimedia and the Web, page 5, Vila Velha, Espírito Santo, Brazil, DOI: 10.1145/1809980.1810066.

TagTop, 2022. The Text Annotation Tool to Train AI. [online] <https://www.tagtog.com/> [Accessed in 25 August 2022]

UBIAI, 2022, Transform Your Unstructured Data Into Intelligence. [online] <https://ubiai.tools/> [Accessed in 25 August 2022]