Semantic Annotation of Medical Documents in CDA Context

Diego Monti(✉) and Maurizio Morisio

Dipartimento di Automatica e Informatica, Politecnico di Torino, Turin, Italy
diegomichele.monti@studenti.polito.it

Abstract. The goal of this work is to recover semantic and structural information from medical documents in electronic format.

Despite the progressive diffusion of Electronic Health Record systems, a lot of medical information, also for legacy reasons, is available to patients and physicians in image-only or textual format. The difficulties of obtaining such information when needed result in high costs for health providers.

In this work we develop the concept of a system designed to convert legacy medical documents into a standard and interoperable format compliant with the Clinical Document Architecture model by the means of semantic annotation.

1 Motivation

In the healthcare domain different kinds of medical documents are produced by physicians in narrative form, relying on templates based on the scope of the document (e.g., progress note, discharge summary). Such templates are slightly different according to the healthcare provider.

Even if an Electronic Health Record (EHR) system is in place, clinical documents usually consist of free text blocks with no semantic encoding. When these documents are exported to a standard electronic format, like a PDF file, in order to send them to patients or other providers, all semantic and structural information is lost.

The difficulties caused by the exchange of medical information among different entities result in high costs for healthcare providers and time consuming activities for patients, that need to act as couriers and perform several times the same medical tests [11]. 25 billion dollars per year are spent in the USA because of unnecessary exams [4].

According to the European and Italian law all medical documents should be published in an interoperable format, but in practice this is not so common. Legacy documents, that contain the medical history of the previous 10–20 years, were initially produced in an electronic format, but nowadays are typically available only in a printed version.

These reasons justify the need of reconstructing the structure of medical documents and performing a semantic annotation on them.

M.E. Renda et al. (Eds.): ITBAM 2016, LNCS 9832, pp. 163–172, 2016.
DOI: 10.1007/978-3-319-43949-5_12

2 Related Work

2.1 Extraction of Semantic Information

Recognizing the structure of paragraphs in image only documents is a well-known problem and the proposed solutions are based on the analysis of the font size and the text placement [2,12]. Also the task of discovering the layout of a textual PDF file has been considered in literature [7]. These studies do not take into account the peculiar characteristics of medical documents, in which the division in sections may be more difficult to identify. On the other side medical documents contain semantic information that can be useful to solve this problem.

The extraction of information from textual medical documents is a very active field of research. Different studies take into account the task of processing non standardized medical data considering text mining and statistical methods in order to identify medical concepts [9], also exploiting a human feedback [20].

A popular approach deals with using Natural Language Processing (NLP) techniques to extract codes, mapping entities present in the text to medical ontologies [13]. cTAKES is an Apache project that aims to extract information from medical documents using the UMLS meta-thesaurus [16]. MetaMap is a program designed to discover UMLS concepts in biomedical text with indexing purposes [1]. MedEx is a medical information extraction system based on the Unstructured Information Management Architecture (UIMA) framework [19].

The limitations of these projects is that they work only with English documents, while we deal with documents in Italian, and that they are not designed with the objective of storing the result of the analysis in a standard medical format.

2.2 Clinical Document Architecture

The Clinical Document Architecture (CDA) is an XML based exchange model for clinical documents proposed by Health Level Seven International, a standards developing organization [8]. The purpose of this model is to enable the sharing of structured medical data among EHR systems of different providers. Three levels of semantic interoperability are defined by the standard.

A CDA document is composed of a header and a body. A level 1 compliant CDA document consists of a free text or an image with some metadata: the body is unstructured and the information about the author and the patience is located in the header. Level 2 compliance means that the body is structured in sections. A CDA document is level 3 compliant if the clinical markup of the body is semantically coded using healthcare code sets [17].

3 Approach

The purpose of this work is to develop a system able to perform an automatic conversion from the PDF version of a clinical document to a CDA level 3 compliant XML file. This operation is organized in three sequential steps (Fig. 1).

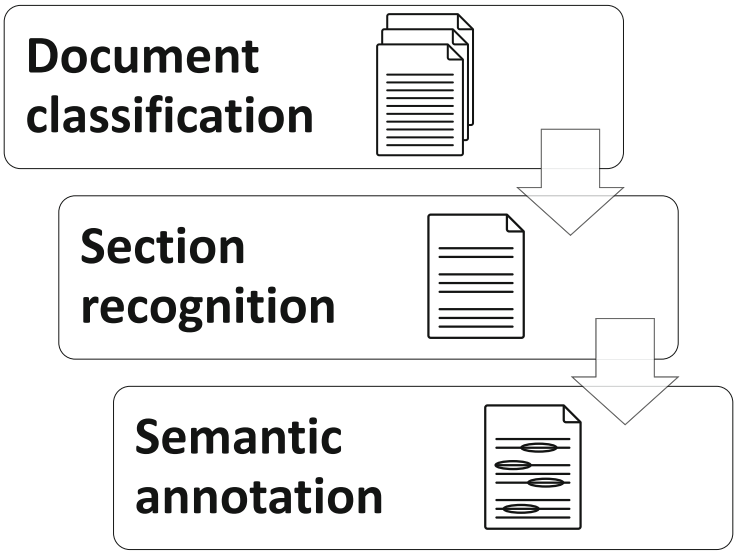

Fig. 1. Graphical representation of the process

3.1 Document Classification

The document is classified in one of the document level categories specified in the Consolidated CDA (C-CDA) standard, a library of CDA templates [10]. Examples of such templates are:

- *Consultation Note*: contains an opinion or advice from another clinician or is used to summarize an Emergency Room or Urgent Care encounter.
- *Discharge Summary*: contains information related to the admission of a patient to a hospital and to the care needed following the discharge.
- *Operative Note*: contains the report of a surgical or other high-risk procedure.
- *Progress Note*: contains a patient's clinical status during a hospitalization, outpatient visit or other healthcare encounter.
- *Transfer Summary*: contains critical information that needs to be exchanged between different providers when a patient moves between health care settings.

This operation is performed by the means of different text mining techniques, in order to select the most effective one. The classifiers considered are: N-gram-based, Naïve Bayes and Bernoulli.

Corpus Annotation. The corpus of input documents is a set of PDF files containing textual information. The following techniques can also be applied to image only documents, but first they must be converted to textual documents employing optical character recognition algorithms.

Different kinds of medical documents are included in the corpus, corresponding to the categories defined in the C-CDA standard. Each category should contain documents of the same type created by different providers. In order to obtain statistically significant results, the smallest category should contain at least a hundred documents.

The documents must be manually associated to their category. This work is known as *corpus annotation* and, in general, it is not a trivial task. State of the art practices demand that the same work is performed by at least two persons knowledgeable of the domain and that they must agree on the category of each document. If for some document there is no agreement, it must be removed from the corpus because it is ambiguous. The result of the manual annotation is called *gold standard* [15].

However, performing this task only for documents is normally easy and can be as simple as grouping the PDF files in different directories according to their category.

Text Extraction. The PDF files need to be converted into text only documents. Different tools can be exploited: for this work the Apache PDFBox Java library has been used.

Medical documents typically contain a header and a footer on each page that do not carry useful information, but they break the flow of the extracted text with, for example, the page number.

PDFBox gives the possibility to extract text only from a specific part of the page by defining a rectangle over the part that will be processed. Considering that a regular A4 page measures 595×842 points, it is generally advisable to avoid analyzing the first and the last 40 points in height. This, of course, needs to be checked for each document.

Classifier Training. LingPipe is a Java library *for processing text using computational linguistics*. Among other features, it is a robust framework to perform automatic text classification [5].

A classifier is trained specifying the possible categories and providing examples for each category. After the training it is possible to compile the classifier in order to produce a more efficient version. The compiled version cannot be trained anymore, but of course can be used to classify new documents. A compiled classifier can be serialized in a file and loaded in a future execution of the program.

Different classifiers are available in LingPipe. The simplest classifier is based on the sequences of characters available in the text: the user needs only to specify the number of characters to analyze together. This kind of classifier is successfully used to perform language recognition, but it is typically too simple to classify documents [6].

Classifiers more suitable for the proposed work are based on the concept of token, so the input text needs to be transformed into a set of tokens. The simplest approach is splitting the text using as separators whitespaces and punctuation

characters, transforming all tokens in lowercase characters and discarding short tokens (for example less than four characters) because they probably represent common words. More complex techniques involve statistical tokenizers that are specific for a particular language.

The most famous classifier is the Naïve Bayes classifier. The *naïve* assumption is that each token is independent from other tokens. Despite the name, this classifier usually obtains good results. A variant of this classifier is the Bernoulli model, where for each document only the presence or the absence of a token is considered, not the number of times it appears.

Classifier Validation. The results of different classifiers need to be checked and compared using a technique known as cross-validation.

The training corpus is initially permuted in a random way in order to remove local dependencies. The corpus is then partitioned in a fixed number of folds, typically ten. Each classifier is trained on nine folds and tested on the tenth fold. This operation is repeated ten times, continuously changing the training folds and the test fold. Each time a confusion matrix is computed and for each classifier a mean confusion matrix is created.

The results of different classifiers are finally compared, considering also possible variations in the initialization parameters of each classifier and how the tokens generation is performed.

3.2 Section Recognition

The document is subsequently split into paragraphs analyzing the typographical features available in the PDF file. From the first paragraph the demographic information about the patient, the physician and the provider are extracted using deterministic rules and simple knowledge bases. The following paragraphs are mapped to the section level categories of the C-CDA standard, exploiting the results of the previous classification, statistical classifiers trained for this task and deterministic rules.

Paragraph Extraction. The task of identifying paragraphs in a PDF file is particularly challenging because typically no information about the organization of the document is provided [7].

A PDF file is similar to a vector image: the position of each element is expressed using absolute coordinates. PDFBox supposes that two elements with the same y coordinate are part of the same line. When a line ends it is, in general, impossible to say if the sentence ended or no more space is available on that line.

A PDF file may contain information about its logical structure: this is extremely important for accessibility tools used by blind people. Unfortunately, such tags are typically not present in medical documents if they are exported from a EHR system.

A possible solution to the problem consists in identifying as paragraph title the lines that are written with a different character, but in rare cases it is also possible that the titles are written with the same character of the text.

Another solution is to consider the frequencies of distinct lines for the same category of documents: lines that appear exactly the number of available documents are likely to be paragraph titles.

CDA Header. The first paragraph of a medical document corresponds to the header of the CDA model. The typical problem is that not all the information required by the standard is available in the analyzed file. If possible, a default value should be assigned in this case.

An example of information that is not present in the analyzed file is the unique identifier of the document. The CDA specification requires that an identifier is assigned to the provider using the Object IDentifier (OID) standard and that the provider assigns a unique code to each document that produces [14].

The CDA header also contains the kind of medical document, the time of redaction, the confidentiality level and the language code. The kind of medical document is the result of the previous classification and must be expressed as a code of the LOINC code system. The time of redaction is typically the last date present in the document. For the confidentiality level and the language code it is necessary to use default values.

Finally, the CDA header contains information about the patient, the physician and the provider. For each of them it is necessary to specify a unique code, a name, an address and a telephone number. For the patient also the birthdate, the birthplace and the gender are required. A lot of other information can be optionally added and other fields may be required according to the locale.

Typically, all the information regarding the patient is present, while the data about the provider is buried in an image part of the template and the physician is identified only by his name. The most effective way of discovering the fields of interest is to use regular expressions or a similar formal language designed considering the header of medical documents from different providers.

The usage of simple knowledge bases should be considered to retrieve information that is present in the document only in an implicit way. For example, a list of all possible cities can be exploited to map a birthplace to the corresponding postal code.

Paragraph Classification. The C-CDA standard specifies, for each type of medical document, a list of possible paragraphs. The classification of the remaining paragraphs is therefore linked to the result of the previous classification.

Two different techniques can be adopted: statistical classifiers similar to the ones used to perform the first step of the analysis or a deterministic technique based on the presence of specific keywords in the title or the text of the paragraph.

The usage of a statistical classifier is meaningful only if it was possible to automatically divide the document in sections. A different paragraph classifier

is needed for each category of medical documents: if the titles are similar it is advisable to prefer the deterministic method.

In rare situations it is possible that a paragraph has not a corresponding section in the CDA template: this typically means that two or more paragraphs of the analyzed document need to be merged together.

CDA Body. The body of a CDA document consists of a list of sections. Each section must be identified by the relative code in the LOINC code system, discovered thanks to the previous classification. At least the text of the corresponding paragraph in the processed document must be added to the section [10].

It is advisable that the text is correctly divided into sentences. In order to achieve this, the most general solution is to remove any new line character and then to apply a sentence detector. LingPipe only implements deterministic sentence detectors, but also statistically based ones are available in other libraries. In any case better results are obtained if the used technique is dependent on the language.

This solution will work only if it is applied to blocks of text: it is also necessary to consider the possibility that the original paragraph consists of a list of short statements. In this case the new lines should not be removed and every statement should be considered a sentence.

3.3 Semantic Annotation

The text of each paragraph is finally analyzed using NLP techniques in order to identify key concepts (e.g., diseases, procedures, drugs) and to map them to the most appropriate medical ontology, such as LOINC or SNOMED CT. The entities that can be recognized in each block depend on the guessed category of the section. All the extracted information is serialized in a XML file following the CDA specifications.

Code Systems. The first code systems in the medical domain were created for financial reasons, to have procedures justified by a diagnosis in the claims for the health insurance. Nowadays they are also essential to describe laboratory analysis results and prescriptions of drugs without ambiguity [18].

An ontology is a code system where the relationships among different entities are represented in a graph structure. Each entity is at least characterized by a code, a normalized name, a list of possible synonyms and the code of the entity of which it is a generalization or a specialization.

Different medical ontologies are available: the most famous ones are the International Classification of Diseases (ICD), SNOMED CT, LOINC and RxNorm. ICD is a list of diseases maintained by the World Health Organization, SNOMED CT is a general collection of medical terms, LOINC contains codes for medical laboratory observations and RxNorm is used to encode information about drugs.

The same concept may be represented in more than a single ontology. For this reason, it is advisable to exploit a meta-ontology, an ontology that links together entities from different code systems.

UMLS is the most comprehensive medical meta-ontology: it contains 2 million concepts from about 200 ontologies [3].

Even if the 70 % of the descriptions in UMLS are in English, the goal of the project is to create a multilingual knowledge source. If a concept is translated in only a certain ontology, it is possible to discover the relative code in another ontology exploiting the links present in UMLS.

Mapping Strategies and Disambiguation. The problem of discovering entities in a text is known in computational linguistic as Named Entity Recognition (NER). In general, it is possible to exploit rule-based techniques, statistical approaches and dictionary-based lookup to perform this task. In the CDA context, the usage of an ontology is compulsory in order to discover the code related to the discovered entity; rules are only useful to identify dates or physical quantities.

Discovering concepts present in a section and mapping them to the most appropriate UMLS entity is not an easy problem because of the inherent ambiguity of the natural language, the presence of many abbreviations for commonly used medical terms and the fragmented syntax used in medical documents [13].

The most effective approach is to look in the text for sequences of words that match the description of an entity, completely or at least partially. For each section, only the most appropriate categories of entities should be used. For example, if a section of the C-CDA standard contains a list of prescriptions, only the entities that are a drug will be considered.

Disambiguate a concept means selecting the most relevant mapping amount a set of possible candidates. A distance among the different pairs of strings needs to be computed and then the most similar string is selected. If no candidate is clearly winning, it is advisable to avoid the encoding of such concept.

Coreference Resolution. It is not enough identifying key entities. Depending of the type of section other information may be required by the C-CDA standard. As usual it possible that such requirements are not completely fulfilled by the analyzed document.

The problem of grouping together semantically related concepts is called coreference resolution. In a clinical document the entities that are linked to a concept can be identified using a rule-based approach. Such rules need to be carefully designed and are dependent on the section.

For example, after having recognized a drug, it is necessary to look for a dosage and a time period; after a laboratory exam, a result with a physical unit. Given the richness of the C-CDA model, it is necessary to start analyzing the documents that need to be converted in order to look for common patterns that can be mapped to a specific CDA construct.

4 Conclusion

The conversion of medical documents from an unstructured PDF to a XML file following the C-CDA specifications is a challenging task consisting of many different sub problems.

EHR system vendors should consider increasing the amount of information included in the generated PDF files. At least the structure of the document should be present, also for accessibility reasons.

Because of the flexibility of the PDF format, it is possible to include in the same file the CDA version of the document: this is an interesting solution to create a medical document that is easily processable both by humans and machines.

References

1. Aronson, A.R.: Metamap: mapping text to the UMLS metathesaurus. NLM, NIH, DHHS, Bethesda, pp. 1–26 (2006)
2. Bloomberg, D.S., Chen, F.R.: Document image summarization without OCR. In: International Conference on Image Processing, vol. 2, pp. 229–232 (1996)
3. Bodenreider, O.: The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Res. **32**(1), D267–D270 (2004)
4. Burton, R., Coleman, E., Lipson, D.J., Agres, T., Schwartz, A., Dentzer, S.: Health policy brief: care transitions. Health Affairs (2012). http://www.healthaffairs.org/healthpolicybriefs/brief.php?brief_id=76
5. Carpenter, B., Baldwin, B.: Text Analysis with LingPipe 4. LingPipe Inc., New York (2011)
6. Cavnar, W.B., Trenkle, J.M., et al.: N-gram-based text categorization. Ann Arbor MI **48113**(2), 161–175 (1994)
7. Chao, H., Fan, J.: Layout and content extraction for PDF documents. In: Marinai, S., Dengel, A.R. (eds.) DAS 2004. LNCS, vol. 3163, pp. 213–224. Springer, Heidelberg (2004)
8. Dolin, R.H., Alschuler, L., Boyer, S., Beebe, C., Behlen, F.M., Biron, P.V., Shabo, A.: HL7 clinical document architecture, release 2. J. Am. Med. Inform. Assoc. **13**(1), 30–39 (2006)
9. Holzinger, A., Schantl, J., Schroettner, M., Seifert, C., Verspoor, K.: Biomedical text mining: state-of-the-art, open problems and future challenges. In: Holzinger, A., Jurisica, I. (eds.) Interactive Knowledge Discovery and Data Mining in Biomedical Informatics. LNCS, vol. 8401, pp. 271–300. Springer, Heidelberg (2014)
10. International, H.L.S: HL7 implementation guide for CDA release 2: consolidated CDA templates for clinical notes (2014)
11. Kripalani, S., LeFevre, F., Phillips, C.O., Williams, M.V., Basaviah, P., Baker, D.W.: Deficits in communication and information transfer between hospital-based and primary care physicians: Implications for patient safety and continuity of care. JAMA **297**(8), 831–841 (2007)
12. Mao, S., Rosenfeld, A., Kanungo, T.: Document structure analysis algorithms: a literature survey. In: Electronic Imaging 2003, pp. 197–207. International Society for Optics and Photonics (2003)

13. Meystre, S.M., Savova, G.K., Kipper-Schuler, K.C., Hurdle, J.F.: Extracting information from textual documents in the electronic health record: a review of recent research. IMIA Yearb. Med. Inform. **35**, 128–144 (2008)
14. Paskin, N.: Digital object identifier (DOI) system. Encycl. Libr. Inf. Sci. **3**, 1586–1592 (2008)
15. Pustejovsky, J., Stubbs, A.: Natural Language Annotation for Machine Learning. O'Reilly Media, Sebastopol (2012)
16. Savova, G.K., Masanz, J.J., Ogren, P.V., Zheng, J., Sohn, S., Kipper-Schuler, K.C., Chute, C.G.: Mayo clinical text analysis and knowledge extraction system (cTAKES): architecture, component evaluation and applications. J. Am. Med. Inf. Assoc. **17**(5), 507–513 (2010)
17. Trotter, F., Uhlman, D.: Hacking Healthcare: A Guide to Standards, Workflows and Meaningful Use, pp. 172–182. O'Reilly Media, Sebastopol (2011). Chap. 11
18. Trotter, F., Uhlman, D.: Hacking Healthcare: A Guide to Standards, Workflows and Meaningful Use, pp. 144–159. O'Reilly Media, Sebastopol (2011). Chap. 10
19. Xu, H., Stenner, S.P., Doan, S., Johnson, K.B., Waitman, L.R., Denny, J.C.: MedEx: a medication information extraction system for clinical narratives. J. Am. Med. Inform. Assoc. **17**(1), 19–24 (2010)
20. Yimam, S.M., Biemann, C., Majnaric, L., Šabanović, Š., Holzinger, A.: An adaptive annotation approach for biomedical entity and relation recognition. Brain Inform. **3**, 1–12 (2016)