DICR: AI Assisted, Adaptive Platform for Contract Review

Dan G. Tecuci, Ravi Palla, Hamid R. Motahari Nezhad,
Nishchal Ahuja, Alex Monteiro, Tigran Ishkhanov, Nigel Duffy
{dan.tecuci, ravi.k.palla, hamid.motahari, nishchal.ahuja, alex.monteiro, tigran.ishkhanov, nigel.p.duffy}@ey.com
EY AI Lab, Palo Alto, CA

Abstract

In the regular course of business, companies spend a lot of effort reading and interpreting documents, a highly manual process that involves tedious tasks, such as identifying dates and names or locating the presence or absence of certain clauses in a contract. Dealing with natural language is complex and further complicated by the fact that these documents come in various formats (scanned image, digital formats) and have different degrees of internal structure (spreadsheets, invoices, text documents). We present DICR, an end-to-end, modular, and trainable system that automates the mundane aspects of document review and allows humans to perform the validation. The system is able to speed up this work while increasing quality of information extracted, consistency, throughput, and decreasing time to decision. Extracted data can be fed into other downstream applications (from dashboards to Q&A and to report generation).

Introduction

Documents are central to the functioning of companies. The ability to read, understand and interpret business documents, collectively referred to as "Document Intelligence", is a critical and challenging application of artificial intelligence (AI) in business. While a variety of research has advanced the fundamentals of document understanding, the majority have focused on documents found on the web which fail to capture the complexity of analysis and types of understanding needed across business documents (Piskorski and Yangarber 2013). Recent interest in document understanding has significantly increased as evidenced by the availability of commercial and open-source product offerings including: IBM Watson[1], Microsoft Azure cognitive services[2], Amazon Textract [3], Google document understanding[4], Prodigy[5].

The closest effort in scope is (Staar et al. 2018), but the focus there is on the structure and layout of the document rather then interpreting its contents. The closest commercial product to what we have built is a contract review solution from Lawgeex[6], but compared to DICR, this is domain and contract type specific.

Our approach is novel in breaking down the problem of document review into several steps: separating the documents into snippets, finding relevant snippets for a particular information need through classification/ranking, and then using value extraction and entity recognition to find values of interest from those snippets. It offers a more reliable extraction mechanism in cases where the direct value extractions and entity recognition fails. It is trainable and applicable to document review across domains.

The System

We present DICR ("Document Intelligence for Contract Review"), an AI-based document review tool, developed for contract review that combines classification, ranking, and entity recognition to speed up and increase consistency of the contract review process. Under our approach the task of document review is reformulated as presenting to the user different document snippets based on their relevance to the users' information need. We have customized the application for review of contracts in the real estate domain and identified two types of generic information needs: value extraction for various data elements (e.g. the date when a lease contract starts, the amount of rent payable annually) and classification (e.g. whether the contract contains a termination clause). We adopt the user-in-the-loop paradigm and offer full control to the user over the output of the tool. The UI supports this paradigm in two ways: highlighting values identified by the tool and offering the user the context in order for her to make an informed judgement. DICR is a generic document intelligence platform that can be customized and trained for reviewing other types of documents (e.g. NDAs, purchase agreements, mortgages, etc.) in other types of domains (e.g. labor law, finances).

The typical workflow of DICR consists of two phases: an optional offline phase where a set of common data elements for a specific domain (called "extraction profile") is defined

[1]https://cloud.ibm.com/docs/services/discovery?topic=discovery-sdu

[2]https://azure.microsoft.com/en-us/services/cognitive-services/

[3]https://aws.amazon.com/textract/

[4]https://cloud.google.com/solutions/document-understanding/

[5]https://prodi.gy/

[6]https://www.lawgeex.com/

by a subject matter expert, data is manually annotated and extraction models are trained, and an online phase where the users interact with the tool in the normal course of business.

The online phase starts with the user uploading a contract, then text is extracted via OCR and grouped into snippets of text (referred to as "Units of Analysis" or UoAs). If an extraction profile has been defined, the user can proceed to performing document review using the predefined models. Otherwise, the user defines a data element (aka field) by providing information such as datatype, keywords, synonyms, description, and other related data elements. This definition is automatically incorporated into a domain ontology that may contain descriptions of the common concepts in the domain. Each data element is represented as a concept in the ontology along with relations between them.

When a specific contract and data element are selected, the extraction models are applied and the system displays the list of most relevant UoAs (called "AI suggestions") for that particular element and highlights the most likely value(s) for it. For ranking the UoAs relevant to a field, the system first applies classification models for each UoA in the contract and then ranks them based on the classification score for the field. For highlighting the most likely value(s) for a field, the system applies an entity recognition model. The system uses different types of classification models: for pretrained fields, the system uses a deep learning multilabel model, whereas for new fields an adaptive strategy consisting of semantic, similarity and binary MLP (multi-layer perceptron (Rosenblatt 1961) classifiers is used[7].

The **semantic model** is based on ontology driven classification. It identifies definitions, occurrences of defined terms/fields (ex: "Lease Commencement Date"); identifies entities (ex: dates); scores the UoA based on identified terms and entities, and extends the results to related terms based on the ontology. This model is used for new fields only until the number of examples for the class exceeds a predetermined threshold. The **similarity model** computes the cosine similarity between added examples for a field and the UoAs for which prediction needs to be done. The aggregate score for a given UoA with respect to a class is the average of the similarity scores between the UoA and top-k closest examples for the class. This approach is used for new fields only until the number of examples for the class exceeds a predetermined threshold.

The system currently employs two **online learning** strategies for classification/ranking: initially apply only the semantic model. As examples are added, the first strategy applies similarity model with increasing weight while reducing weight for semantic model. Once the number of examples reaches a predetermined threshold (t_1), the semantic and similarity models are disabled and an MLP model is trained for the new field. The MLP model is retrained in the background when the number of examples exceed by a predefined, configurable threshold (t2). The second online learning strategy continuously trains an MLP model in the background and switches to it once it starts performing better than the similarity model, as assessed by a continuous evaluation step for every batch of k examples provided by the user (k is configurable).

For extracting field values the system uses two types of **value extraction models**: NER for standard entities and a sequence labeling model for each pretrained, domain-specific entities. NER is used for new fields until enough data is accumulated through user interaction; afterward, new fields are treated just like a predefined ones. The NER model used by the system is a combination of spaCy NER[8] and regular expressions. For example, if a user defines a new field "Lease Start Date" with datatype "DATE", then the system first ranks the UoAs in the document based on their classification scores and then highlights all DATE entities in the top k results.

Conclusions and Future Work

We presented DICR, an end-to-end, modular, and trainable system that automates the mundane aspects of contract review. The system is currently in pilot in multiple client accounts and in different domains (real estate leases, loan and mortgage applications).

The system combines search, ranking, sequence labeling and online learning in a novel way to provide value to users even with a small amount of data. As more data is collected with usage of tool, accuracy of the system is improved. The system is able to speed up this work while increasing quality of information extracted, consistency, throughput, and decreasing time to decision.

While there has been considerable work done with respect to each of independent modules/models (search, ranking, sequence labeling, online learning), this systematic way of combining them is what enables the information extraction system to quickly deliver significant amount of value to the users.

References

Kim, Y.; Jernite, Y.; Sontag, D.; and Rush, A. M. 2016. Character-aware neural language models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, 2741–2749.

Piskorski, J., and Yangarber, R. 2013. Information extraction: Past, present and future. In *Multi-source, Multilingual Information Extraction and Summarization*. 23–49.

Rosenblatt, F. 1961. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Technical report, Cornell Aeronautical Lab Inc Buffalo NY.

Staar, P. W. J.; Dolfi, M.; Auer, C.; and Bekas, C. 2018. Corpus conversion service: A machine learning platform to ingest documents at scale. *CoRR* abs/1806.02284.

Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A. J.; and Hovy, E. H. 2016. Hierarchical attention networks for document classification. In *NAACL HLT 2016, San Diego California, USA, June 12-17, 2016*, 1480–1489.

[7] We also experimented with Hierarchical Attention Networks (Yang et al. 2016) with trainable word embedding layer and Hierarchical Attention Networks with trainable character embeddings based on (Kim et al. 2016)

[8] https://spacy.io/docs/usage/entity-recognition