Combining Deep Learning and Reasoning for Address Detection in Unstructured Text Documents

*[1], *[1], *[1], *[2]
[1]REDACTED
[2]REDACTED
*, *, *, *

Abstract

Extracting information from unstructured text documents is a demanding task, since these documents can have a broad variety of different layouts and a non-trivial reading order, like it is the case for multi-column documents or nested tables. Additionally, many business documents are received in paper form, meaning that the textual contents need to be digitized before further analysis. Nonetheless, automatic detection and capturing of crucial document information like the sender address would boost many companies' processing efficiency. In this work we propose a hybrid approach that combines deep learning with reasoning for finding and extracting addresses from unstructured text documents. We use a visual deep learning model to detect the boundaries of possible address regions on the scanned document images and validate these results by analyzing the containing text using domain knowledge represented as a rule based system.

Introduction

Many businesses need to deal with scanned text documents, such as invoices or purchase orders, on a daily basis. This continuing trend is reflected by the importance of OCR solutions worldwide (Grand View Research Inc. 2019). Nowadays, many powerful tools like OCR engines or deep learning models, trained on large quantities of data, are freely available and can be used for the automated analysis of such documents. However, adoption of these generic tools for specific tasks is still error-prone, since human expert domain knowledge is required in many use cases. This *a priori* domain knowledge is usually not well represented or integrated in modern AI models (Stevens et al. 2020). For instance, before continuing to process the contents of a document, we found that a first crucial step for businesses is usually a proper identification of the correct meta data, like sender and receiver address, because this information might influence the subsequent processing flow. Humans can solve this task easily, since they know about typical locations of address information on document headers as well as the internal structure of addresses that all entities have to follow. Our experience showed, that pure machine learning solutions can heavily profit from such human reasoning assessments, provided these information can be represented and integrated in a learning system.

For tackling this problem of address detection, we built a processing pipeline, consisting of different data-driven and rule-based components, which was developed in cooperation with *REDACTED*. Many of the components originally come from the open source OCR-D project (Neudecker et al. 2019), which aims to provide means for analysis of scanned documents by enabling usage of other (heterogeneous) open source processing tools, like e.g. Tesseract OCR (Smith 2019), in a standardized way.

Our work is structured as follows: We first give a short overview of related work, followed by a comparison of data-driven and reasoning-based approaches in general. This forms the basis for further technology decisions during the pipeline implementation described in the subsequent section. Thereafter, we describe the evaluation of our approach using a document test set and conclude our work with an outlook and future work.

Related Work

Extracting address information from unstructured texts is not a new use case: Long before the rise of modern AI and deep learning technology, rule-based approaches and algorithms aimed to detect post addresses in digitized texts, for example on web pages (Lin et al. 2005). With the upcoming of machine learning techniques, more data driven approaches were applied, for example using Conditional Random Fields (Chang and Li 2010). However, when dealing with complex document layouts or noisy scanned documents, the correct identification and boundary detection in terms of entity extraction remains error-prone.

State-of-the-art document analysis models, like the multimodal LayoutLM architecture released by Microsoft, target this problem by combining both visual and text information in the model architecture, even for multiple languages (Xu et al. 2021). Nonetheless, results achieved by pure data driven approaches always retain some amount of uncertainty regarding plausibility and validity from the view of a human expert. Therefore, we propose an approach that merges machine intelligence and human reasoning for improved address detection.

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Criteria	Deep Learning	Reasoning	Consequence
Availability of training data or example documents	Quantity of training data limited, especially in small businesses, leading to lacking model quality	Number of sample documents typically sufficient for the definition of simple rules	Risk that deep learning models generalize poorly (and produce too many false negatives) or generalize too much (and produce too many false positives)
Availability of reference databases (for example Customer Relationship Management system)	Typically ignored by standard deep learning approaches	Useful for implementing plausibility checks / validation rules	Integration of validation rules using reference database in a hybrid approach to reduce false positives
Use of standard layout for internal documents	Result in usually high-quality models, even with limited training data	Well-suited for writing visual rules	Possibility to use visual rules for processing of internal documents
High variability of layouts for external documents (i.e., from customers)	-	Increases the difficulty of defining rules	Need for deep learning models tackling the variety of designs
False positives (wrong address being extracted) lead to faulty document processing (e.g., routing to the wrong department or person)	Tend to be common	Tend to be rare when using validation rules	Need to limit false positives by combining reasoning and validation after deep learning extraction
False negatives (no address being extracted) lead to the need for manual data extraction	Tend to be rarer than with reasoning approaches	Tend to be common as rules generalize poorly	False negatives can be accepted, but they limit the possible degree of full automation
Debugging	Black-box character of deep learning models and limitations of XAI usually only allow the use of additional training data with debugging and improvement of model quality	Software developers typically can fine-tune the rules when provided with sample documents that were incorrectly processed	No significant difference between approaches as business customers tend to prefer stable and slowly evolving solutions

Table 1: Comparison of deep learning and reasoning approaches for address data extraction from German-language business correspondence

Comparison of Deep Learning and Reasoning for Address Extraction

Our work focuses on the extraction of sender or receiver address data in business correspondence, as they appear in invoices, purchase orders or customer letters. This use case was chosen based on the feedback of many project partners and customers: Extracting address data and matching the sender or receiver of documents with a database is very often the first step in automated document processing. The restriction to business correspondence implies some specificity that shows the strengths and limits of sole deep learning and reasoning approaches listed in Table 1, which led to the hybrid approach showcased in this work.

Recent work has shown that approaches such as transfer learning may reduce the downsides of deep learning when little training data is available (Martínek, Lenc, and Král 2020). The application of transfer learning, however, requires the availability of sufficiently good foundational models (Bommasani et al. 2021). These may be available for historical – mostly anglophone – business correspondence where training data is available, like the RVL-CDIP data set (Harley, Ufkes, and Derpanis 2015). To the best of our knowledge, however, no sufficient publicly available datasets exists for business correspondence in German. Overall, the comparison of both pure methods indicates, that a hybrid approach combining the generalization capability of deep learning to a variety of layouts and the reduction of false positives in typical reasoning, appears promising.

Hybrid Method for Address Extraction

Problem Definition

In our use case, the goal of address detection is achieved in two steps:

1. Detection of possible address candidates on the (scanned) pages of business documents
2. Assignment of the correct address class for each candidate (*sender*, *receiver*, *other*)

Note that the found *sender* and *receiver* address are limited to one instance per document page, while the amount of *other* addresses remains unlimited. An address candidate is considered as detected correctly, if the text within its region boundaries contains at least the ZIP Code and the city name of the correct address.

Pipeline Realization

We built a prototype in form of the processing pipeline which utilizes a hybrid workflow as is shown in Figure 1, including both: deep learning and rule-based pattern matching. The choice of technology for the individual pipeline steps was based on the considerations listed in Table 2.

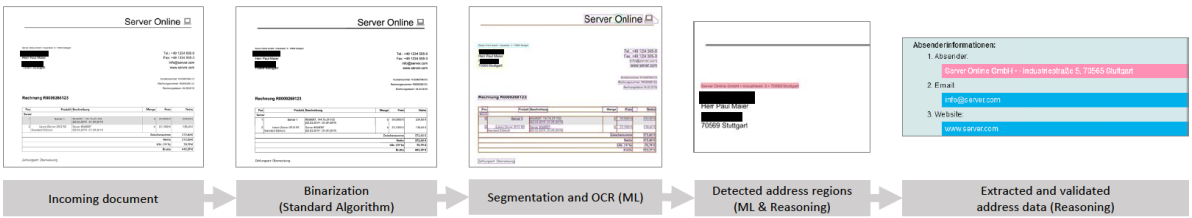

Figure 1: Processing pipeline for address detection in business documents

Pipeline Component	**Deep Learning**	**Reasoning**	**Consequence / Technology Choice**
Binarization	-	Standard image processing algorithms	Standard algorithms are sufficient here
Segmentation	For complex and diverse layouts suitable	Rules would quickly become too complex	Usage of pre trained data driven models
OCR	Nowadays, best results are achieved here with deep learning	Not used anymore	Usage of Tesseract 4.0 working based on deep learning approach
Address Region Detection	For complex and diverse layouts suitable	Rules would quickly become too complex	Visual Deep Learning approach was chosen.
Address Text Detection	Customized Named Entity Recognition Models thinkable, but training data is needed	Advantages of the use of reference databases and plausibility checks	Usage of rule based validation because of well defined structure of German addresses

Table 2: Description of the influence and application of data-driven and rule-based methods in the respective pipeline steps

In the following, we describe the realization of different processing steps in the pipeline. For many of the components, modules from the OCR-D project were adapted to our specific use case of address detection[1].

At the pipeline entry, all incoming documents are converted to 300dpi TIF files and the pages of each document are segregated. In the following step, the file is binarized with the tool Olena, included in the OCR-D project.

Afterwards, a general visual document segmentation is performed using Tesseract and OCRopy, an open-source document analysis framework which allows identification of table regions and structures (Breuel 2017). As a result of this step, we receive all text regions and lines that have been identified, together with their bounding box coordinates on the page. The OCR of Tesseract 4.0 is applied for extracting the textual contents of the detected lines.

For identification of address region candidates, we send the OCR results to our reasoning-based Address-API component. The module recognizes address-like sections using human crafted rules. Specifically, the module detects individual address components like ZIP codes, cities, streets or names of persons and organizations by using regular expressions and reference lists. The found entities are validated according to the rules of German address structures (e.g. street is followed by house number, followed by ZIP code with always 5 digits and city name, etc.). This way, a prediction can be made, whether a given section is a proper an address.

Finally, the address prediction for each text line is combined with the visual information, like region image, boundaries and location on the document page. This is done by feeding both visual and text classification information into a modified Deep Learning model for predicting the final address region boundaries together with their address label (*sender*, *receiver*, *other*). More precisely, we fine-tuned a Mask R-CNN model (Abdulla 2017) using both synthetically generated letters (with corresponding address labels) and about 300 annotated internal invoice documents.

At the end of the pipeline, the textual contents of the predicted address regions for *sender*, *receiver* and *other* are again validated by the same reasoning-based approach already used in the previous step (Address-API) to make sure we really received a valid address region from the model. Note that we have not implemented this final integration yet, thus the following evaluation is only based on the results predicted by the deep learning model mentioned in the last step. We expect the results to further improve by introducing this second level of (repeated) textual validation.

Evaluation

Test Data

For evaluating the quality of our address extraction approach, we built a test set consisting of 64 scanned documents with a resolution of 300 dpi. The data set provides a total of 104 pages, of which 67 pages contain a *sender* and 71 a *receiver* address. Additionally, 105 addresses of type *other* are distributed across 74 pages. There are 8 pages left without any addresses.

[1] for details to the individual components see *REDACTED*

	TP	FP	FN	F1
Sender	39	0	27	**0.7429**
Receiver	57	0	13	**0.8976**
Other	41	15	64	**0.5093**
All	137	15	104	**0.6972**

Table 3: Evaluation results on test set using the hybrid address extraction methodology

Results and Discussion

In our experiments, we consider an address as correctly detected and classified (TP) if the predicted region contains at least the core components of an address (postal code and city, and additionally street and house number if present in the ground truth) and is assigned the correct class label (*sender*, *receiver*, *other*). Regions that were predicted as addresses and contain no address information are defined as false positives (FP). Existing address regions in the test set that were not recognized or got the wrong address type assigned are false negatives (FN). The performance results on the whole test set are listed in Table 3. Note that we leave out the indication of true negatives (TN), since we would have to consider all other non-address text regions that were correctly not labeled as addresses as well.

The low occurrence of FP indicates that our hybrid approach is able to predict address regions with a high degree of reliability, which is due to our reasoning-based textual validation mechanism, whose result is directly used as secondary input to the visual deep learning model. The high amount of false negatives shows that we are missing a lot of addresses. In many cases, we were able to trace back the reason for this to insufficient quality of the OCR results, which leads to incorrect address validation predictions by the reasoning-based component. This is due to the strict address formatting rules enforced by the module. Hence, if for example single characters in either the postal code, city or street name are incorrectly extracted by the OCR process, the match to our reference list will fail with high probability. In future works, we aim to tackle this problem by improving the OCR component and allowing more fuzzy matching in the reasoning and rule-based parts of the pipeline.

Conclusion and Future Work

Our experiments have shown that combining the strengths of data-driven and reasoning-based approaches for the use case of address extraction (in form of domain knowledge about structure and contents of country-specific addresses) can improve stability and reliability of results. For our future work, we plan to optimize each step of the pipeline. For the deep learning models, this means acquiring more training and fine tuning on bigger data sets. For the reasoning-based parts, this implies to soften the strictness of rule validation by allowing a small amount of fuzziness. Additionally, we aim to extend our approach to other tasks of information extraction, in particular intelligent analysis of complex or nested table structures.

References

Abdulla, W. 2017. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow. https://github.com/matterport/Mask_RCNN.

Bommasani, R.; Hudson, D. A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M. S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. 2021. On the opportunities and risks of foundation models. Stanford University: Center for Research on Foundation Models (CRFM), Stanford Institute for Human-Centered Artificial Intelligence (HAI).

Breuel, T. 2017. OCRopus: document analysis and optical character recognition (OCR) system. https://github.com/ocropus/ocropy. Accessed: 2021-11-10.

Chang, C.-H.; and Li, S.-Y. 2010. MapMarker: Extraction of Postal Addresses and Associated Information for General Web Pages. volume 1, 105–111.

Grand View Research Inc. 2019. Optical Character Recognition Market Size Report, 2021-2028. https://www.grandviewresearch.com/industry-analysis/optical-character-recognition-market. Accessed: 2021-11-10.

Harley, A. W.; Ufkes, A.; and Derpanis, K. G. 2015. Evaluation of Deep Convolutional Nets for Document Image Classification and Retrieval. In *International Conference on Document Analysis and Recognition (ICDAR)*.

Lin, C.; Zhang, Q.; Meng, X.; and Lin, W. 2005. Postal Address Detection from Web Documents. In *2005 International Workshop on Challenges in Web Information Retrieval and Integration (WIRI 2005), 8-9 April 2005, Tokyo, Japan*, 40–45. IEEE Computer Society.

Martínek, J.; Lenc, L.; and Král, P. 2020. Building an efficient OCR system for historical documents with little training data. *Neural Computing and Applications*, 32(23): 17209–17227.

Neudecker, C.; Baierer, K.; Federbusch, M.; Boenig, M.; Würzner, K.-M.; Hartmann, V.; and Herrmann, E. 2019. OCR-D: An End-to-End Open Source OCR Framework for Historical Printed Documents. In *Proceedings of the 3rd International Conference on Digital Access to Textual Cultural Heritage*, DATeCH2019, 53–58. New York, NY, USA: Association for Computing Machinery.

Smith, R. 2019. Tesseract OCR: an optical character recognition engine for various operating systems. https://github.com/tesseract-ocr/tesseract. Accessed: 2021-11-10.

Stevens, R.; Taylor, V.; Nichols, J.; Maccabe, A. B.; Yelick, K.; and Brown, D. 2020. AI for science, Report on the Department of Energy (DOE) Town Halls on Artificial Intelligence (AI) for Science.

Xu, Y.; Lv, T.; Cui, L.; Wang, G.; Lu, Y.; Florencio, D.; Zhang, C.; and Wei, F. 2021. LayoutXLM: Multimodal Pre-training for Multilingual Visually-rich Document Understanding.