Few-exemplar Information Extraction for Business Documents

Daniel Esser, Daniel Schuster, Klemens Muthmann and Alexander Schill
Computer Networks Group, TU Dresden, 01062 Dresden, Germany

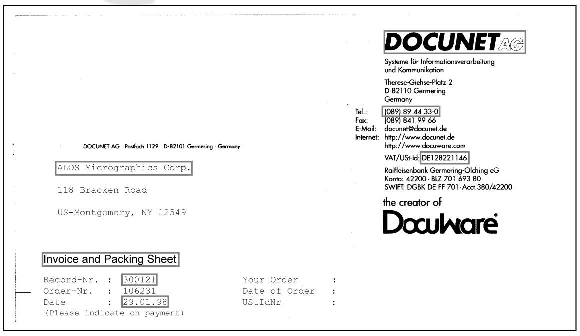
Keywords: Information Extraction, Few-exemplar Learning, One-shot Learning, Business Documents.

Abstract: The automatic extraction of relevant information from business documents (sender, recipient, date, etc.) is a valuable task in the application domain of document management and archiving. Although current scientific and commercial self-learning solutions for document classification and extraction work pretty well, they still require a high effort of on-site configuration done by domain experts and administrators. Small office/home office (SOHO) users and private individuals do often not benefit from such systems. A low extraction effectivity especially in the starting period due to a small number of initially available example documents and a high effort to annotate new documents, drastically lowers their acceptance to use a self-learning information extraction system. Therefore we present a solution for information extraction that fits the requirements of these users. It adopts the idea of one-shot learning from computer vision to the domain of business document processing and requires only a minimal number of training to reach competitive extraction effectivity. Our evaluation on a document set of 12,500 documents consisting of 399 different layouts/templates achieves extraction results of 88% F_1 score on 10 commonly used fields like document type, sender, recipient, and date. We already reach an F_1 score of 78% with only one document of each template in the training set.

1 INTRODUCTION

Today a huge amount of communication between business partners is still done using physical correspondence. The movement towards paperless offices and the need for archiving documents due to legal regulations require the digitization and storage of these letters. Commercial solutions like smartFix (Dengel and Klein, 2002) and Open Text Capture Center (Opentext, 2012) already provide solutions to automatically process digital and digitized documents like invoices, medical documents, or insurances. These systems identify relevant information (Figure 1) that later on can be stored as structural information to ERP systems or databases for improved search and retrieval or automated processing.

While current solutions work for large and medium-sized companies, they still require a high effort of on-site configuration or a high amount of example documents to reach acceptable extraction effectivity. Both requirements do not fit the needs of small office/home office (SOHO) users and private individuals. Rule-based systems reach very good extraction rates but need experts that initialize and update the rule base according to the needs of the institution. Especially very small companies and private persons do not have the funds and expertise to run a customized solution. Purely trainable systems can overcome the disadvantages of rule-based systems, but most of them need large sets of example documents and long periods of training to generate a knowledge base capable of high accuracy extractions.



DOCUNET AG
Systeme für Informationsverarbeitung
und Kommunikation

Therese-Giehse-Platz 2
D-82110 Germering
Germany

Tel.: (089) 89 44 33-0
Fax: (089) 841 99 66
E-Mail: docunet@docunet.de
Internet: http://www.docunet.de
http://www.docuware.com

VAT/USt-Id: DE128221146

Raiffeisenbank Germering-Olching eG
Konto: 42200 · BLZ 701 693 80
SWIFT: DGBK DE FF 701 · Acct.380/42200

the creator of
DocuWare

DOCUNET AG · Postfach 1129 · D-82101 Germering · Germany

ALOS Micrographics Corp.

118 Bracken Road

US-Montgomery, NY 12549

Invoice and Packing Sheet

```
Record-Nr.  :  300121          Your Order      :
Order-Nr.   :  106231          Date of Order   :
Date        :  29.01.98        UStIdNr         :
(Please indicate on payment)
```

Figure 1: Excerpt of a scanned real-world business document. Some relevant information (sender, recipient, date, subject, etc.) are highlighted with frames.

To raise acceptance for information extraction systems within this group of users, we require a mechanism to speed up information extraction in the starting period. While most related work centers on the overall performance, the ability of these solutions to extract information with only a minimal set of docu-

Esser D., Schuster D., Muthmann K. and Schill A..
Few-exemplar Information Extraction for Business Documents.
DOI: 10.5220/0004946702930298
In *Proceedings of the 16th International Conference on Enterprise Information Systems* (ICEIS-2014), pages 293-298
ISBN: 978-989-758-027-7

ments as a training set is nearly unknown. Therefore this paper makes the following contributions: (1) Turn the attention of the community to the problem of few-exemplar extraction. (2) Provide metrics to evaluate common systems according to their starting behavior. (3) Present a solution for fast-learning information extraction out of scanned business documents as a starting point for this research area that reaches a nearly constant extraction effectivity independent from the size of the training set.

The remainder of this paper is organized as follows. In Section 2 we present related works from document classification and information extraction that already discuss the issue of few-exemplar extraction. Section 3 defines the problem. Section 4 deals with our approach to reach nearly constant extraction effectivity not depending on the size of the training set. In Section 5 a new measure to evaluate this behavior will be defined. Our system will be assessed using this measure to show its ability to few-exemplar extraction. Section 6 closes this paper with a conclusion and gives an overview of future work.

2 RELATED WORK

The research problem of few-exemplar learning in the area of document processing was first mentioned by Eric Saund (Saund, 2011). He observes that users do not appreciate that classification methods need explicit examples of allowable variations in what to a person are clearly the same document type. In his point of view the research attention has to focus on machine-learning techniques that are able to distinguish between document types from a minimal set of available example documents.

In the area of computer vision the problem of insufficient available annotated data is already known and described. Fei-Fei et al. coin the phrase of one-shot learning that describes the behavior of a learning system to generate enough information to classify new objects to a category out of only one single example object (Fei-Fei et al., 2006). Salperwyck and Lemaire deal with this issue by evaluating leading classifiers according to their ability to perform well on a small set of training documents (Salperwyck and Lemaire, 2011). While the authors only focus on the classification of different kinds of objects, the same rules match for self-learning methods in the context of information extraction out of business documents, which leads to the field of few-exemplar extraction.

The community of information extraction has only produced a small number of systems explicitly optimized and evaluated according to the ability to learn from few examples. Bart and Sarkar try to identify relevant information organized as tables and lists (Bart and Sarkar, 2010). The user has to provide one entry of a table or list in a document as feedback. From this information the authors generate knowledge about table and list structure and recognize other entries within this document. Using this kind of one-shot learning on the entry level, the work reaches an extraction effectivity of 92% Recall. Nevertheless the user has to manually annotate one entry per document, which is a high requirement the majority of SOHO users would not agree with. Medvet et al. are much closer to the problem of few-exemplar extraction. The authors evaluate their solution according to the number of documents with the same class in the training set (Medvet et al., 2011). Following a probabilistic approach they reach extractions rates of 62% with only one, 65% with two and 80% with three documents of the same class in training. In relation to the solutions overall effectivity of above 90%, the starting behavior offers a large gap to this value.

To summarize, the documentation of the learning behavior of common solutions for information extraction out of business documents is disappointing. Among hundreds of solutions only a small number focusses on the ability to immediately learn from scratch. The extraction effectivity of these matching solutions is far away from a level that is acceptable for SOHO users and private individuals.

3 PROBLEM DEFINITION

Few-exemplar extraction is the ability of a self-learning information extraction system to reach constant and high extraction effectivity independent from the available set of training documents. This section focusses on formalizing the problem and defining a measure to evaluate information extraction systems according to few-exemplar learning. For each document d from a set of test documents $\mathcal{D}$, the structure of the currently available training set $\mathcal{T}$ has to be analyzed. Equation 1 defines a function *sim* that returns a set $\mathcal{T}_{sim} \subset \mathcal{T}$ of similar training documents that directly influences extraction results from document d. In our case, function *sim* identifies documents in $\mathcal{T}$ that share the template/layout of document d. The evaluation results are categorized according to the size k of this set.

$$\forall d \in \mathcal{D}\ \exists k \in \mathrm{N} : |sim(d, \mathcal{T})| = |\mathcal{T}_{sim}| = k \qquad (1)$$

Using an evaluation metric p_k for measuring and averaging the results of all test documents with k similar documents in the training set, the performance of the system has to be (1) constant for all documents independent from the number *k* of similar documents in the training and (2) near to the extraction effectivity the fully trained system reaches.

As presented in Section 5, we define our similarity function *sim* based on training documents with the same template and our evaluation metric *p* on top of common metrics Precision, Recall and F_1 score. To compare the learning behavior of different systems, we implemented a measure, called Few-Exemplar Extraction Performance (FEEP), whose calculation is presented in Equation 2. By calculating the average over the relative performances according to the system's maximum performance p_{max} for a number of bins with $k \leq t$, we get an indicator how good a system works with few examples in its starting period according to its maximum performance. Due to an often very uneven number of instances for each *k*, we use the average performance p_{avg} of the system instead of the maximum performance p_{max}.

$$FEEP_t = \frac{1}{t} \sum_{k=1}^{t} \frac{p_k}{p_{max}} \qquad (2)$$

4 TEMPLATE-BASED APPROACH

To find an approach that fits the users' requirements and performs a few-exemplar extraction, we analyzed content and layout of business documents in detail. Most documents are based on document templates. While many related works define a template as a schema explicitly describing the document and its relevant information, from our point of view a template consists of a theoretical function, which transforms index data, fixed textual elements, and layout components to a graphical representation of the document, a so-called template instance.

The key idea of our information extraction system is to reverse this transformation to identify the index data used to create the representation. While we do not have any information about the function itself, we try to identify documents with the same template and benefit from commonalities between them. By grouping documents according to their layout and generating extraction rules out of at least a minimal number of similar training examples on-the-fly, we want to reach the proposed enhancement of the extraction effectivity and speed-up in the starting period. Ideally, template instances are similar enough to contain sufficient extraction knowledge out of only one single instance in the training set. Details on our approach are shown in Figure 2. It is a part of the Intellix process (Schuster et al., 2013), which focusses on the extraction of information out of business documents with a high overall effectivity. Due to the focus of this work to the ability of few-exemplar extraction, we reduce the description of our algorithms to a minimum. Further details can be found in the referred paper.

The input of our extraction system are XML files describing the content and layout of business documents. Starting with a document image taken by one of various source devices, i.e. scanner, printer, smartphone, or computer, the document is preprocessed and transformed by a commercial OCR to a hierarchical representation. This XML file describes the structure of the document starting from page level down to character level. For each element additional information like position, bounding box, font details, and formatting styles are detected. While this information is delivered by an external OCR, we do not focus on any optimizations.

4.1 Template Document Detection

Similar to common solutions, the first step of our approach is a classification. The template document detection searches the model of available training examples for similar documents called template documents. We try to identify training documents based on the same template as the extraction document. We do not have any formal definition what a template looks like. Hence we analyze textual and structural characteristics to find similarities which lead to the decision that two documents are based on the same template. Due to a high dependency of following algorithms on the results of the template document detection, we focus on reaching a very high Precision with values of 99% and higher. Technically, we use a two-step approach to find template documents.

In a first step, we use the search engine Lucene with a tf-idf-based ranking as a fast heuristic. Due to its independence from the size of training and its ability to immediately learn new documents, it is most suitable to the SOHO use case. As features we combine the document's words with its positions. For this purpose we overlay each document with a grid of the same size and add the coordinates of the cell the upper left corner of the bounding box of the word matches. Validation runs have shown a perfect grid size of 6 by 3. A word "Invoice" in grid cell with coordinates x=2 and y=4 will result in the feature "Invoice_0204". Querying Lucene returns a ranked list of k training documents that match the input document.

To identify relevant documents within the ranked list, we rely on a common distance metric. In this second step we calculate a normalized and comparative

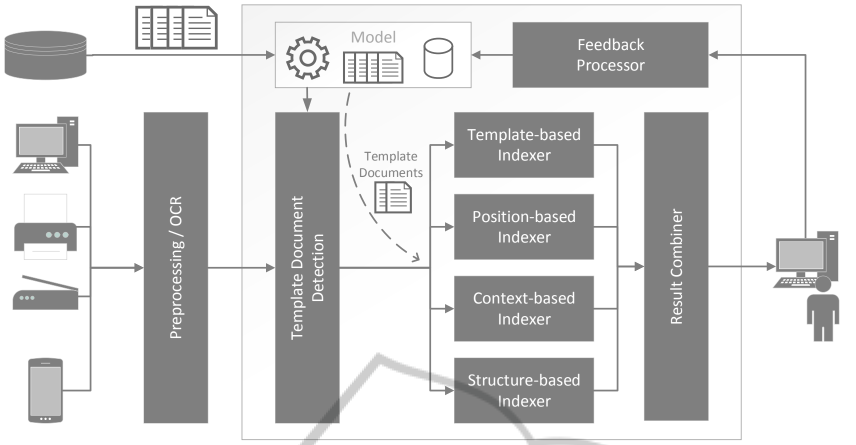


Figure 2: Intellix indexing process specialized to the area of few-exemplar extraction.

score for each similar document. Therefore we define features the same way we do it for our heuristic. By giving each document a weighting according to the occurrence of this feature, we get a vector space model. The distance between documents is calculated using the Cosine similarity. At last, we select a subset of documents by separating the results using a experiment-driven threshold, indicating these are the ones we assume are generated by the same template.

4.2 Generation of Extraction Rules

Our extraction algorithms analyze the template documents identified by our template document detection and generate different kinds of extraction rules on-the-fly. These rules are applied to the document that should be extracted to get relevant information. Each extraction algorithm produces a list of possible candidates including a score between 0 and 1 indicating how sure it is according to the correctness of each result. To focus on as much characteristics of documents as possible, we implemented different algorithms using different strategies to learn extraction patterns out of the template documents. Due to the independent processing, more algorithms can be easily added by integrating them in parallel to our proposed methods.

In Intellix, there already exist extraction algorithms that perform good for most index fields. The *Template-based Indexer* analyzes fixed fields (i.e. document type or sender) in the template documents and adopts the values. The *Position-based Indexer* combines the positions of values in template documents and searches for relevant information on comparable positions in the document to be extracted. Both algorithms improve when more template documents become available. They are not specialized to small sets of similar documents and do not significantly influence the starting period.

Therefore we developed two novel algorithms that require only one template document to find relevant information. Our *Context-based Indexer* focusses on fields, whose values do not change their position according to anchor words. While the total amount of an invoice will change its absolute position according to the number of records, its relative position according to some context word as “Total amount:” will stay the same. The indexer identifies such context words by overlapping the document to be extracted with all template documents and determining the best word in the intersection set regarding the shortest distance to the value. Afterwards it searches these words in the target document and extracts relevant information due to their relative positioning. Our *Structure-based Indexer* uses the hierarchical structure of the documents XML files and tries to map the position of index values in the DOM tree of a template document to the DOM tree of the document to be extracted. The consistent generation of XML files by OCRs guarantees similarities between both trees that allow a fuzzy mapping to identify relevant information.

From the perspective of few-exemplar extraction, the presented algorithms can already produce results with at least one similar document identified by the template document detection. Nevertheless precision of extraction patterns increases with more training documents becoming available.

4.3 Result Combination and Feedback

The results of each single extraction algorithm are merged to a set of final results. Based on the candidates and scores each algorithm produces, the result combiner aggregates them and calculates new scores. Hereby, the combiner considers the ability of each algorithm to extract fields by using weights to influence the share each algorithm has in the final results. Weights are dynamically calculated by analyzing the results of each algorithm in relation to feedback.

As already mentioned, a user can hand in feedback for documents. Feedback is processed by a feedback processor, that adds each annotated document as a new training example to the model. The next extraction may already use this feedback to improve template detection and thus extraction effectivity.

5 IMPLEMENTATION AND RESULTS

We implemented the process described above in a prototype implementation at TU Dresden. Large parts of this implementation are already used in the commercial document management system developed by our project partner DocuWare.

As document corpus we used a set of 12,500 multilingual real-world business documents from the archive of DocuWare. We captured each document with a stationary scanner and classified it manually by its layout. It seems that nearly all documents were generated using templates thus we categorized them according to this basis. Altogether we identified 399 different kinds of layouts within the document set. To evaluate the information extraction we annotated each document according to commonly used fields in document archiving. Beside a minimal set of fields to enable structured archiving (document type, recipient, sender, date), we added further popular fields like amount, contact, customer identifier, document number, subject, and date to be paid based on an inedited survey carried out by DocuWare. All in all we identified 105.184 extractable values.

The extraction effectivity is evaluated using the common measures Precision, Recall, and F_1 score as adopted by Chinchor and Sundheim for MUC-5 (Chinchor and Sundheim, 1993). The authors categorize each extraction result as correct (COR), incorrect (INC), partial correct (PAR), missing (MIS), or spurious (SPU) and calculate Precision, Recall, and F_1 score on top of these values. As the user only expects correct results, we ignored the class of partial correct results and tackled this kind of extraction as incorrect. Overall values are calculated using a micro-averaging approach by averaging single results over all recognized labels.

To measure the ability to few-exemplar extraction, we used an incremental learning approach combined with an adapted measure based on Precision, Recall, and F_1 score. Incremental learning perfectly simulates the way SOHO and private individuals fill their system with training documents. We start with an empty model without any training and continuously enhance the training set by user feedback. Each document not recognized correctly by our system was added as a new training document to the model thus improving future indexing. The extraction effectivity in the starting period was measured by calculating Precision, Recall, and F_1 score in relation to the size and structure of the training set. For each test document, the number k of documents with the same template in the current training set was determined by our similarity function *sim* and the extraction results were grouped according to this number into classes correct (COR_k), incorrect (INC_k), missing (MIS_k), and spurious (SPU_k). Equations 3 and 4 describe the way our evaluation metrics p_k (Precision@k and Recall@k) were calculated. F_1@k is defined as the harmonic mean between Precision@k and Recall@k. Due to the fact that our metrics' effectivity depends very much on the quality of the first training documents, we did multiple evaluation runs and averaged all single results to get a significant final evaluation.

$$Precision@k = \frac{COR_k}{COR_k + INC_k + SPU_k} \tag{3}$$

$$Recall@k = \frac{COR_k}{COR_k + INC_k + MIS_k} \tag{4}$$

Figure 3 visualizes the results of our system using the iterative approach and the already defined metrics Precision@k, Recall@k, and F_1@k. For documents that do not have a similar document in the training set at extraction time, we reach an extraction effectivity of 22%. We expected this value to be zero. Instead, our template document detection sometimes selects wrong examples for documents that have no pendants with the same template in the training set. Using wrong but similar documents surprisingly leads to a low extraction effectivity unequal to zero. Much more relevant are the results the system produces with only one, two or three similar documents in the current training set. Our system reaches constant rates over all k starting with an already very high one-shot learning extraction effectivity of 78% F_1 score. Therefore a user has to manually annotate the relevant information in only one document of each template to nearly get an extraction effectivity of 80% that is comparable to the performance that can be reached with manual indexing (Klein et al., 2004). The averaged extraction effectivity (88% F_1 score) using the iterative evaluation approach bares the range of possible improvements for our fast learning system. Using our metric Few-Exemplar Extraction Performance based on the F_1 score with a threshold of $t = 5$, we reach $FEEP_5 = 0,93$.

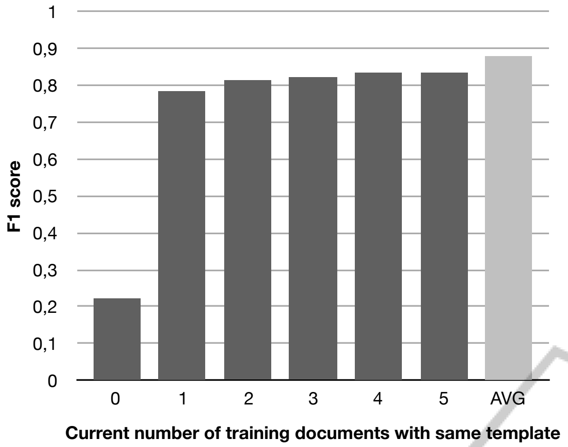


Figure 3: Extraction effectivity of the proposed approach in relation to the number of documents with the same template within the current training set using an iterative evaluation.

6 CONCLUSION

We discussed the problem of few-exemplar extraction in the area of document processing and presented a metric to measure information extraction systems according to their performance with a limited number of training documents. Based on this research, we developed an approach for few-exemplar information extraction for SOHO users and private individuals. It is based on a template document detection that identifies similar training documents using textual and layout-based features in combination with the search architecture Lucene. We use these similar documents as an input for our extraction algorithms to create single results from each component and combine them to a final result. We reach a one-shot extraction effectivity of 78% F_1 score on 10 commonly used fields in document archiving. SOHO users and private individuals, who do not have any training documents, just have to annotate one document per template to reach an acceptable extraction effectivity.

The results in Figure 3 reveal a very low performance for documents that do not have any similar document in the training set. While this case is not very surprising, our process is focussed on the existence of similar documents, we want to improve this performance by adding a cooperative information extraction. Trustworthy users can combine their systems in a secure way and provide extraction knowledge they already gained to increase the extraction effectivity of the whole group. Especially where there are no similar documents in the system, it can profit from the knowledge in another one.

ACKNOWLEDGEMENTS

This research was funded by the German Federal Ministry of Education and Research (BMBF) within the research program “KMU Innovativ” (fund number 01/S12017). We thank our project partners from DocuWare for insightful discussions and providing us with the document corpus used for evaluation.

REFERENCES

Bart, E. and Sarkar, P. (2010). Information extraction by finding repeated structure. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, DAS ’10, pages 175–182.

Chinchor, N. and Sundheim, B. (1993). Muc-5 evaluation metrics. In *Proceedings of the 5th conference on Message understanding*, MUC5 ’93, pages 69–78.

Dengel, A. and Klein, B. (2002). smartfix: A requirements-driven system for document analysis and understanding. *Document Analysis Systems V*, pages 77–88.

Fei-Fei, L., Fergus, R., and Perona, P. (2006). One-shot learning of object categories. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4):594–611.

Klein, B., Agne, S., and Dengel, A. (2004). Results of a study on invoice-reading systems in germany. In *Document Analysis Systems*.

Medvet, E., Bartoli, A., and Davanzo, G. (2011). A probabilistic approach to printed document understanding. *Int. J. Doc. Anal. Recognit.*, 14(4):335–347.

Opentext (2012). Opentext capture center. http://www.opentext.com/ What-We-Do/ Products/ Enterprise-Content-Management/ Capture/ OpenText-Capture-Center.

Salperwyck, C. and Lemaire, V. (2011). Learning with few examples: An empirical study on leading classifiers. In *The International Joint Conference on Neural Networks (IJCNN)*.

Saund, E. (2011). Scientific challenges underlying production document processing. In *Document Recognition and Retrieval XVIII (DRR)*.

Schuster, D., Muthmann, K., Esser, D., Schill, A., Berger, M., Weidling, C., Aliyev, K., and Hofmeier, A. (2013). Intellix - end-user trained information extraction for document archiving. In *Document Analysis and Recognition (ICDAR)*, Washington, DC, USA.