Named Entity Recognition and Relation Extraction with Graph Neural Networks in Semi Structured Documents

Manuel Carbonell*†, Pau Riba*, Mauricio Villegas†, Alicia Fornés* and Josep Lladós*
*Computer Vision Center, Computer Science Dept., Universitat Autònoma de Barcelona, Spain
Email: {mcarbonell, priba, afornes, josep}@cvc.uab.es
†omni:us, Berlin, Germany. Email: {manuel, mauricio}@omnius.com

***Abstract*—The use of administrative documents to communicate and leave record of business information requires of methods able to automatically extract and understand the content from such documents in a robust and efficient way. In addition, the semi-structured nature of these reports is specially suited for the use of graph-based representations which are flexible enough to adapt to the deformations from the different document templates. Moreover, Graph Neural Networks provide the proper methodology to learn relations among the data elements in these documents. In this work we study the use of Graph Neural Network architectures to tackle the problem of entity recognition and relation extraction in semi-structured documents. Our approach achieves state of the art results in the three tasks involved in the process. Additionally, the experimentation with two datasets of different nature demonstrates the good generalization ability of our approach.**

***Index Terms*—Relation Extraction Name Entity Recognition, Semi-structured Documents, Administrative Documents, Graph Neural Networks**

I. Introduction

In the digital transformation era, Robot Process Automation (RPO) technologies have emerged as central processes in digital mailroom workflows [1], [2]. Heterogeneous documents coexist in AI-driven decision making processes that companies and organizations adopt for the sake of efficiency. Sectors as fintech, legaltech or insurance process an inflow of million of forms, invoices, id documents, claims, etc. every day. The success in the automation of these transactions relies on the ability to incorporate semantic understanding, beyond the traditional Optical Character Recognition (OCR) that merely transcribes the input. Information extraction (IE) is the task of automatically retrieving structured data from semi-structured machine-readable documents, both digitally born or scanned. Broadly speaking, it requires named entity recognition (NER) and relation discovery between document terms. Both tasks are mutually dependent because the correct retrieval of semantic terms from documents is boosted by their context, either geometric (where the information appears in the document) or semantic (which other terms this information is linked to, in the same document or in other ones). In Natural Language Processing (NLP) based information extraction systems, such as BERT models [3], this context is modelled in one dimension, and words are interpreted in a given sequence (pre and post words) roughly speaking. However, business documents do not have a linear wise reading order, but the interactions between text, graphical objects and the layout is highly relevant.

To successfully perform such tasks it is necessary to identify visual as well as linguistic patterns to recognize textual content and its layout, which provides a complementary aspect to the plain textual content. In the last years both domains have seen a huge leap forward due to the arrival of Convolutional Neural Networks (CNNs) [4], [5]. In Bahdanau *et al.* [6], the idea of selectively *attending* to different parts of the input data when sequentially producing predictions brought major improvements in the field of machine translation. This idea was improved and successfully applied to other types of sequential data, achieving state of the art on several NLP tasks such as question answering, named entity recognition or natural language inference [7]. With BERT architecture [3], the use of pretraining techniques further improved the performance of such models achieving current state of the art on most of the existing challenges of NLP.

These approaches are all based on a sequential *relational inductive bias* [8] that consists in making some relational assumptions to learn a model able to make correct predictions. Nevertheless, the high variability of the data in the task of finding layouts and relationships within document elements suggests that other arbitrary inductive bias should be allowed. Other approaches [9] [10] face the mentioned tasks with architectures originally designed for vision, which have locality inductive bias. These methods achieve acceptable results but also are not allowing the mentioned arbitrary relational inductive bias assumptions among the document, which motivates the use of Graph Neural Networks (GNNs). Several work has been done exploiting the combination of neural architectures with graph structured data with great success, extending their breakthrough on vision and natural language to many other domains such as quantum chemistry, knowledge graphs, or citation networks [11] [12] [13]. In the work of Velickovic *et al.*. [14] the idea of attention is brought together with GNNs leveraging masked self attention layers, having in this way a specially adequate architecture to efficiently solve not only problems such as node classification with prior known graph structure but also structure inferring problems such as link prediction. In this work we tackle the problem of finding relationships between elements in a document, *i.e.* predict links between entities by means of a Graph Neural Network model.

Liu *et al* [15] proposed a GNN based approach for NER in visually rich documents that successfully classifies named entities suggesting its potential capability of performing other tasks of information extraction. Recently, in the work of Riba *et al.* [16] a Graph Neural Network is trained to detect tables in different types of business documents, predicting relationships between table elements. Other notable contributions in the field are the LayoutLM model [17], and [18]. The first one is based in the idea that BERT [3] derived architectures provide a powerful resource to extract patterns in sequential data. Hence in their work they convert the input data in a sequential format comprising embedded layout as well as textual information to successfully classify entities. The latter one combines this idea with the use of GNNs to jointly predict the contents of documents with a predefined structure as in the case of the ICDAR 2019 Competition on Scanned Receipt OCR and Information Extraction [19]. Conversely, in our case we further extend this by giving to our model the possibility to predict links between the entities whose type and amount might be unknown a priory.

In this work, we propose a novel method to extract structured information from semi structured documents by means of GNNs. Inspired by [16] we extend this idea to a more generic context were also key-value pairs which are not strictly table elements are predicted, and also entities are classified in different categories. The whole system demonstrates the ability to solve the three tasks with state of the art performance. Summarizing, the main contributions of our work are:

- We cast the named entity recognition and relation extraction as a supervised message passing task.
- We surpass state-of-the-art performance of the three tasks involved.
- Our model generalizes to weakly structured documents, as we show in the experimental part validating it images of historical marriage licenses.

The rest of the paper is organized as follows. Section II introduces the proposed pipeline for named entity recognition and relation extraction, as well as the specific GNN chosen architecture for our work. Next in section III we describe the datasets and metrics to test the approach, and we show the obtained results. Finally, section IV draws the conclusions extracted from the experiments.

II. Methodology

In this section, we introduce our approach for name entity recognition and relation extraction. We focus on the steps of document understanding coming once the OCR has been already performed. Therefore, we consider that the raw textual content of the document is already available and to better isolate the problem we make use of the ground-truth transcriptions as well as bounding boxes.

A. Problem formulation

Given an input document the model has to be able to (i) detect the document entities *i.e.* groups of words with a semantic meaning; (ii) classify the detected entities into predefined categories and; (iii) discover the meaningful pairwise relationships between entities. These tasks are named as word grouping, entity labeling and entity linking respectively.

The proposed architecture is divided in several components. Each of them is trained for a single task independently from the others. Thus, in total three different GNN models, $f_1(\cdot)$, $f_2(\cdot)$ and $f_3(\cdot)$, are considered. The document is initially represented as a graph G_1 whose nodes are the words detected in the OCR process. Edges between words are created using k nearest neighbors (k-NN) based on the distances of the top-left corner of the word bounding boxes. The GNN first identifies groups of words corresponding to entities by doing edge classification. Subsequently, the graph is contracted according to the detected groups (graph G_2) in order to perform the tasks of entity labeling as a node classification approach and entity linking as link prediction pipeline. An overview of this approach is introduced in figure 1 for the first task and in figure 2 for the other ones.

B. Word Grouping

The first task towards a framework able to understand the complex structure of a document is to group the words which belong to the same semantic entity. This task requires to combine both sources of information, on the one hand, the textual content and, on the other hand, the pairwise relationships with other words. Thus, we consider the task of finding groups of words as a link prediction problem in the graph of the document.

With this aim, the graph $G_1 = (V_1, E_1)$ is constructed by considering each detected word as a node. To initialize the node features, we first calculate a fasttext word embedding model [20] by linearizing the text of the training documents ordered as given by the OCR process. An important benefit of using fasttext is that at prediction time it is possible to get meaningful embeddings for words not observed in the training set, which is a rather common occurrence in administrative documents.

Given a node $v_i \in V_1$, its initial hidden state vector $h_i^0 = [x_i, y_i, w_i, h_i, w_{\text{embed}}]$ is the concatenation of the word embedding with the corresponding bounding box width, height, and top left corner position normalized with respect to the page size. Having calculated $h^0 = \{h_1^0, \ldots, h_n^0\}$ we generate k-NN graph G_1 with $k = 10$ since the complete graph–all nodes connected with each other–makes the problem computationally unfeasible. The number of neighbors for constructing the graph has been chosen experimentally making sure the minimum number of candidate edge between words is missing while keeping the number of edges low. This hyper-parameter could be further tuned but it is beyond the main scope of this work. The generated graph is going to be further processed by our L layer GNN architecture $s = f_1(G_1)$ where s are the final link predictions.

To get the word groups from the link predictions we keep the edges whose predicted scores are greater than a threshold τ, and, by connected components, we define the entities.

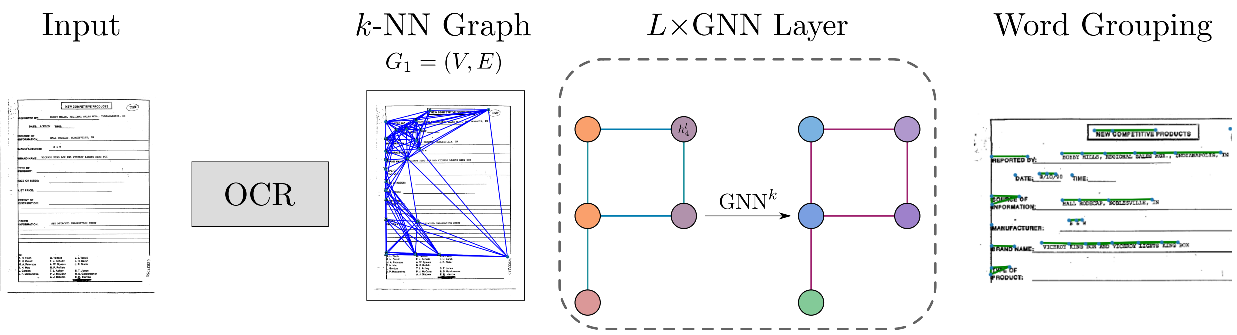


Fig. 1. Overview of the proposed word grouping approach. The text content and location of the words in the input document is encoded in a word level k-NN graph. This is fed into a GNN with L layers. The word grouping is formulated in terms of a binary edge classification problem, that is, 1's indicates that these words belong to the same entity.

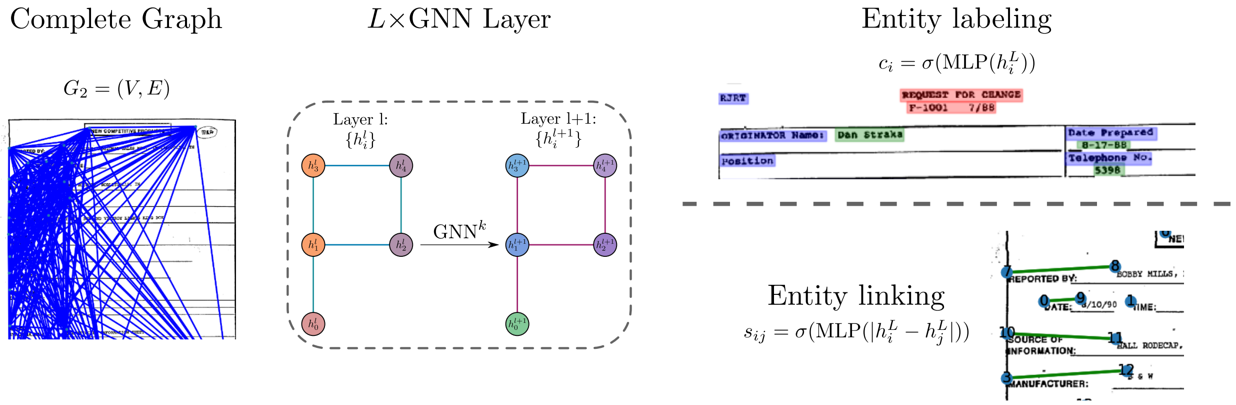


Fig. 2. Given the discovered entities (see figure 1), a complete entity level graph is generated and fed into L GNN layers. Thus, the tasks of entity labeling and entity linking are formulated in terms of node and edge classification respectively. The GNN is trained separately for each task.

C. Entity Labeling

Assuming that the previous word grouping task has been successfully solved, in this step we want to classify each group of words or equivalently *semantic entity* with its corresponding label. For this case, let us consider a graph $G_2 = (V_2, E_2)$ as the entity graph, where each node represents an entity. For this module we considered the complete graph since the number of nodes is drastically reduced. Then the label for a given entity is calculated in terms of node classification. Thus, following the notation mentioned above, $c = f_2(G_2)$ where c are the predicted entity labels.

D. Entity Linking

Similarly to the previous task, entity linking makes use of the complete graph G_2 as its input. However, this task is cast as an edge classification framework following the same pipeline introduced for the word grouping task. Therefore, our model binary classifies edges to predict the existence or absence of links between nodes. Thus, $s = f_3(G_2)$ where s are the predicted scores per each edge.

E. Architecture

Here we describe how our three graph models are built to solve the above described problem. With our approach the model extracts structured information combining two types of processes: (i) given a set of node vectors, find the structure of graph, i.e. predict the existing edges between them. This is used for the word grouping part as well as for entity linking; (ii) given a set of nodes, classify each of them in a predefined category. This is used for the entity labeling part.

The proposed tasks, do not only predict classes in the set of nodes, but also relationships among words and entities in a document. This second objective requires to infer the meaningful structure given a set of node data and partially known edge information rather than making use of static ground truth edge connectivity to predict values for nodes. For this type of task GAT layers have shown to be very adequate, therefore, we selected them as the base of our GNN architecture.

In the following lines, we describe the backbone of our architecture independently to the final task. Let $G = (V, E)$ be a graph where $e_{ij} \in E$ denotes the edge between nodes $v_i, v_j \in V$. Let $n = |V|$ be the number of nodes in the input graph then GAT layers receive a set of nodes features $h^l = \{h_i^l\}_{i=0^n} \in \mathbb{R}^{F_l}$ and return an updated set of those nodes $h^{l+1} = \{h_i^{l+1}\}_{i=0}^n \in \mathbb{R}^{F_{l+1}}$ according to the pairwise relationships defined in E. GAT layers follow the idea of attention in CNN's to decide which are the important connections. Therefore, for each pair of nodes (v_i, v_j) the *attention* *coefficients* α_{ij} are calculated:

$$\alpha_{ij} = \frac{\exp(\text{LeakyRelu}(V[Wh_i||Wh_j]))}{\sum_{k\in\mathcal{N}(v_i)}\exp(\text{LeakyRelu}(V[Wh_i||Wh_k))} \quad (1)$$

where $\mathcal{N}(v_i)$ is the set of neighboring nodes of v_i, W and V are weight matrices with learnable parameters and $||$ is the concatenation operator. Following the Transformer architecture practices [7] we use K attention heads. Hence, K attention coefficients are computed and aggregated in order to obtain the updated node hidden state h^{l+1}. Thus, a GAT layer is defined as:

$$h_i^{l+1} = g(h_i) = \Big|\Big|_{k=1}^{K} \sigma\left(\sum_{j\in\mathcal{N}_i}\alpha_{ij}^k W^k h_j^l\right). \quad (2)$$

In our experiments, we consider the backbone model of our functions $f_1(\cdot)$, $f_2(\cdot)$ and $f_3(\cdot)$ as L GAT layers.

The tasks that we are facing for document understanding can be summed up in node classification and link prediction. The first one simply consists to assign a label $c_i \in C$ to each node v_i in the input graph G. The second one consists of predicting the existance or absence of an edge between each pair of nodes. For the first case we simply feed the hidden state node representation to a Multi Layer Perceptron (MLP) with a sigmoid activation function, predicting this way each class probability for node v_i:

$$c_i = \sigma(Wh_i^L), \quad (3)$$

where $W \in \mathbb{R}^{F_L\times C}$ is a learnable weight matrix, C is the number of classes and h_i^L is the node hidden state at the last layer.

In the case of link prediction we also use a MLP but in this case receiving a list of all the candidate node pairs and returning their link likelihood score:

$$s_{ij} = \sigma(W(|h_i^L - h_j^L|)), \quad (4)$$

where $W \in \mathbb{R}^{F_L\times 1}$ is a learnable weight matrix.

Note that with this approach we are not predicting directed links as we take the absolute value of the difference between hidden state vectors.

In all cases the GNN is trained with Stochastic Gradient Descent (SGD) on the Cross Entropy (CE) loss for both problems, node or edge classification. CE loss is defined as:

$$CE(y') = -(y \cdot log(y') + (1-y) \cdot log(1-y')) \quad (5)$$

where y are the ground-truth labels and y' are the predicted scores.

III. Experiments

In this section we present the experiments for our method on the benchmark datasets FUNSD [21] and IEHHR [22] for administrative and historical documents respectively. The code to reproduce the experiments are available here [1].

A. Datasets

1) FUNSD: As we introduced earlier, despite the abundance of research on extracting structured information from semi structured documents and the interest in the industry for obtaining a robust solution for the problem there is no universally accepted main benchmark for the task. An obstacle for the advance and refinement of a solution in the field is the confidential nature of the data in which companies need to run such algorithms. Jaume *et al.* [21] intend to unify efforts with a benchmark on this popular problem, reducing it to the tasks of grouping, labeling and linking. The dataset comprises 199 real, fully annotated, scanned forms extracted from the Truth Tobacco Industry Document6 (TTID), and archive comprising scientific research, marketing, and advertising documents of some of the largest US tobacco firms.

2) IEHHR: Besides testing our approach on modern bureaucratic document dataset we also want to investigate its versatility in even weaker structured documents, such the ones containing in the IEHHR competition dataset [22]. This database consists of historical handwritten records from the Archives of the Cathedral of Barcelona. Each record contains information about the husbands occupation, place of birth, husbands and wifes former marital status, parents occupation, place of residence, geographical origin, etc. In this case the word groups are also forming named entities, but restricted to information of members of the family in which marriages are taking place -wife, husband, wife's father, mother etc.- as well as their related locations, occupations or civil states. All entities corresponding to a family member are linked to the name of the corresponding members. Also wife and husband names are linked for each record. An example page with labeled entities can be seen in figure 3.

B. Metrics

The performance of the tasks faced in this work are measured with two different metrics. For the grouping part, since it consists of clustering elements we calculate the Adjusted Rand Index (ARI) [23].

For the tasks of entity labeling and link prediction we calculate the $F1$ score in the traditional way, being the harmonic mean between precision P and recall R.

$$F1 = 2 \cdot \frac{P \cdot R}{P + R}$$

C. Results

Table I presents the quantitative evaluation on the three tasks. Note that our model is not using any external data to train our architecture.

Concerning the grouping task in *FUNSD*, we see that the model is able to correctly predict most links between words, despite the vast amount of edges in the k-NN graph. Although it would be ideal, with this approach it is not intended that every single edge is going to be correctly predicted, remind that we intend to cluster the nodes based on densely connected regions with a semantic meaning. In many cases the groups will be correctly predicted despite some of the links between

[1] https://github.com/manucarbonell/gcn-form-understanding

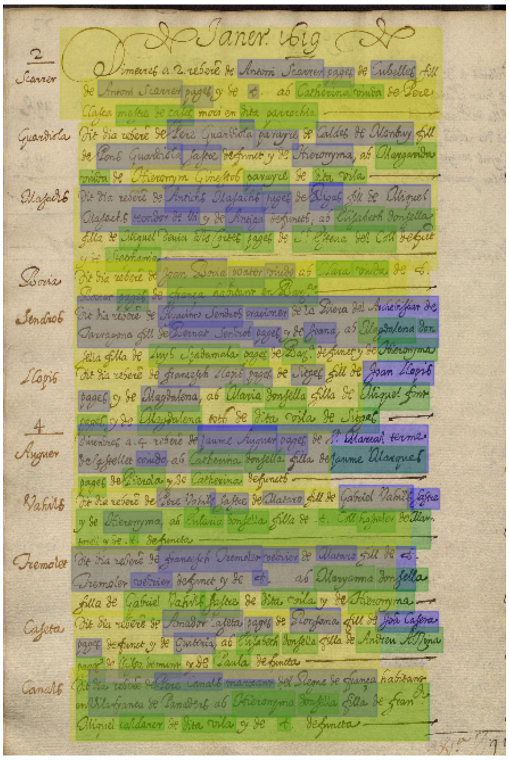


Fig. 3. Entity label ground truth on a IEHHR page. The amount of words in the groups vary greatly depending on the type of entity.

nodes in the are missing, *i.e.* a false negative link is likely to be harmless to the performance on this step as the aggregation is still correct. On the other hand, false positive links create a bigger problem. They may join two groups that should be separated for a proper detection. Using the validation scores during training, we set the threshold τ to the value above which an existing edge is considered a link on the grouping step. Hence, τ has been set to 0.65, and 0.9 for FUNSD and IEHHR respectively, avoiding as much false positives as possible. Predictions on a k-NN graph from a page can be observed in figure 4.

Regarding the entity labeling task, we outperform the BERT + MLP approach proposed in the FUNSD baseline [21]. The same task is performed at word level by the pretrained LayoutLM [17]. Their reported results are convincing, however, they are not directly comparable neither to the FUNSD approach [21] nor our current work. Our results follow the original paper, therefore the F1 is calculated at entity level.

Concerning entity linking, the model performs significantly better than the previously proposed method [21] but with a moderated performance when considering it in a generic context. We are convinced that this could be strongly improved using a dataset with a significant higher amount of training samples.

When observing qualitative results on an unseen page (see figure 5) we notice that the model does some wrong link predictions in which a rule restriction based on the content of the entities could give better results. However the scope of this work is to investigate how good a pure learned graph neural model could perform in such a task of finding relationships within the document, without having to classify a layout into a known one but learning to identify pairs of keys and values and other relevant related entities instead.

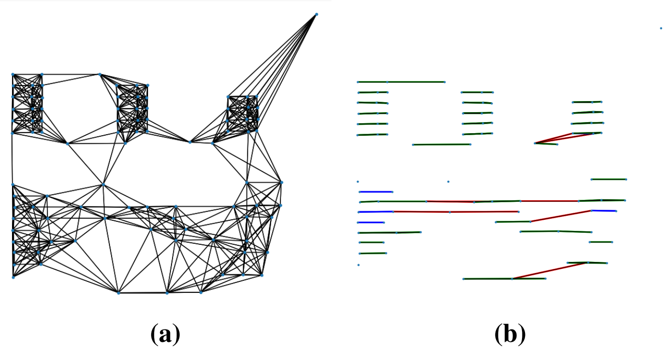
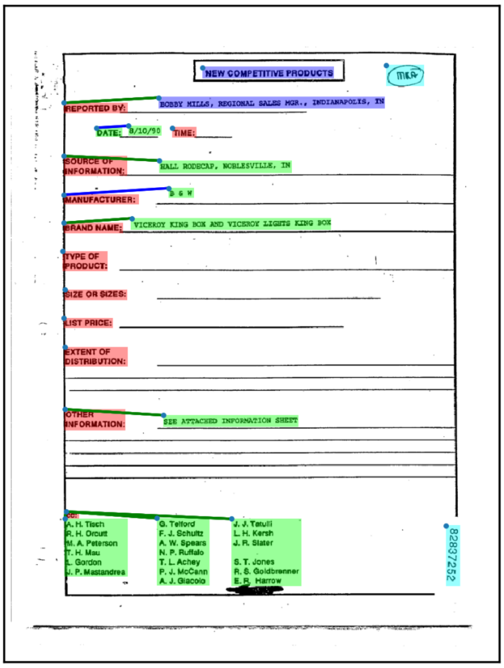


Fig. 4. (a) Input k-NN graph fed to the GNN for word grouping on a FUNSD page. (b) Word group predictions on the same document. Green edges are true positives, red are false positives and blue false negatives. We do not plot true negatives and the background to ease interpretation. Node positions are normalized with respect to the page image size.



NEW COMPETITIVE PRODUCTS

MEA

REPORTED BY: BOBBY MILLS, REGIONAL SALES MGR., INDIANAPOLIS, IN

DATE: 8/10/90 TIME:

SOURCE OF INFORMATION: HALL RODECAP, NOBLESVILLE, IN

MANUFACTURER: B & W

BRAND NAME: VICEROY KING BOX AND VICEROY LIGHTS KING BOX

TYPE OF PRODUCT:

SIZE OR SIZES:

LIST PRICE:

EXTENT OF DISTRIBUTION:

OTHER INFORMATION: SEE ATTACHED INFORMATION SHEET

cc:

A. H. Tisch	G. Telford	J. J. Tatulli
R. H. Orcutt	F. J. Schultz	L. H. Kersh
M. A. Peterson	A. W. Spears	J. R. Slater
T. H. Mau	N. P. Ruffalo	
L. Gordon	T. L. Achey	S. T. Jones
J. P. Mastandrea	P. J. McCann	R. S. Goldbrenner
	A. J. Giacoio	E. R. Harrow

82837252

Fig. 5. Entity linking and labeling predictions on FUNSD. Green and blue lines show true positive and false negative links between entities. Keys, values, headers and other are labeled with red, green, blue and turquoise boxes respectively.

Regarding *IEHHR*, the grouping model gives an acceptable performance, specially taking into account the strongly regular nature of the paragraphs in each page. Despite this regularity, the difficulty in the labeling part becomes clear, since we have to classify each entity in one of the predefined 20 categories with only 80 pages for training, to which we attribute the low performance in this step. Despite leaving room for improvement our model again gets to successfully

TABLE I
RESULTS FOR THE THREE DOCUMENT UNDERSTANDING TASKS ON FUNSD AND IEHHR DATASETS.

	Word Grouping (ARI)	**Entity Labeling (F1)**	**Entity Linking (F1)**	**External data**	**# Params**
			FUNSD [21]		
[21]	0.41	0.57	0.04	✓	340M
[17]	-	0.79[2]	-	✓	160M
Ours	0.65	0.64	0.39	-	201M
			IEHHR [22]		
Ours	0.65	0.53	0.67	-	201M

solve the linking of entities proving that the approach can also be suitable for this type of task.

IV. CONCLUSION

In this work we have presented a method to perform named entity recognition and relation prediction in semi structured documents with Graph Neural Networks, bringing promising results in the process of structured information extraction. Our method has been initially designed for administrative document understandig, but we have shown that it can be adapted to other domains, as for example historical manuscripts. The experimental results show that there is still room for improvements, probably due to the reduced size of the open available data sets. For this reason, further research tuning the method and testing on larger data sets could confirm the feasibility of the approach as a generic solution for extracting structured information from semi-structured documents.

ACKNOWLEDGMENTS

This work has been partially supported by the Spanish project RTI2018-095645-B-C21, the European Commission H2020 SME Instrument program, under Grant Agreement no. 849628, project OMNIUS, the grant 2016-DI-095 from the Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya, the Ramon y Cajal Fellowship RYC-2014-16831, the FPU15/06264 and the CERCA Programme / Generalitat de Catalunya.

REFERENCES

[1] R. Jain and C. Wigington, "Multimodal document image classification," in *International Conference on Document Analysis and Recognition*, 2019, pp. 71–77.

[2] K. Li, C. Wigington, C. Tensmeyer, H. Zhao, N. Barmpalios, V. I. Morariu, V. Manjunatha, T. Sun, and Y. Fu, "Cross-domain document object detection: Benchmark suite and method," in *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, 2019, pp. 4171–4186.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1097–1105.

[5] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 260–270.

[6] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *International Conference on Learning Representations*, 2015.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30*, 2017, pp. 5998–6008.

[8] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. F. Zambaldi, M. Malinowski, A. Tacchetti *et al.*, "Relational inductive biases, deep learning, and graph networks," *CoRR*, vol. abs/1806.01261, 2018.

[9] M. Carbonell, A. Fornés, M. Villegas, and J. Lladós, "A neural model for text localization, transcription and named entity recognition in full pages," *Pattern Recognition Letters*, vol. 136, pp. 219–227, 2020.

[10] P. Zhang, Y. Xu, Z. Cheng, S. Pu, J. Lu, L. Qiao, Y. Niu, and F. Wu, "Trie: End-to-end text reading and information extraction for document understanding," 2020.

[11] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, pp. 61–80, 2009.

[12] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *International Conference on Learning Representations*, 2017.

[13] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning*, vol. 70, 2017, pp. 1263–1272.

[14] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *International Conference on Learning Representations*, 2018.

[15] X. Liu, F. Gao, Q. Zhang, and H. Zhao, "Graph convolution for multimodal information extraction from visually rich documents," in *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.

[16] P. Riba, A. Dutta, L. Goldmann, A. Fornés, O. Ramos, and J. Lladós, "Table detection in invoice documents by graph neural networks," in *International Conference on Document Analysis and Recognition*, 2019, pp. 122–127.

[17] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou, "Layoutlm: Pre-training of text and layout for document image understanding," in *The ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2020.

[18] W. Yu, N. Lu, X. Qi, P. Gong, and R. Xiao, "Pick: Processing key information extraction from documents using improved graph learning-convolutional networks," *ArXiv*, vol. abs/2004.07464, 2020.

[19] Z. Huang, K. Chen, J. He, X. Bai, D. Karatzas, S. Lu, and C. V. Jawahar, "Icdar2019 competition on scanned receipt ocr and information extraction," in *International Conference on Document Analysis and Recognition*, 2019, pp. 1516–1520.

[20] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *arXiv preprint arXiv:1607.04606*, 2016.

[21] G. Jaume, H. K. Ekenel, and J.-P. Thiran, "Funsd: A dataset for form understanding in noisy scanned documents," in *International Conference on Document Analysis and Recognition Workshops*, vol. 2, 2019, pp. 1–6.

[22] A. Fornes, V. Romero, A. Baro, J. Toledo, J. Sanchez, E. Vidal, and J. Llados, "Icdar2017 competition on information extraction in historical handwritten records," in *International Conference on Document Analysis and Recognition*, 2017, pp. 1389–1394.

[23] L. Hubert and P. Arabie, "Comparing partitions," in *Journal of Classification*, vol. 2, 1985, pp. 193–218.

[2] Not directly comparable, evaluation at word level