Named Entity Recognition in Semi Structured Documents using Neural Tensor Networks

Khurram Shehzad[1], Adnan Ul-Hasan[2], Muhammad Imran Malik[1,2], and Faisal Shafait[1,2]

[1] School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST), Islamabad, Pakistan.
{kshehzad.mscs15seecs, malik.imran, faisal.shafait}@seecs.edu.pk
[2] Deep Learning Laboratory, National Center of Artificial Intelligence, Islamabad, Pakistan.
adnan.ulhassan@seecs.edu.pk

Abstract. Information Extraction and Named Entity Recognition algorithms derive major applications related to many practical document analysis system. Semi structured documents pose several challenges when it comes to extract relevant information from these documents. The state-of-the-art methods heavily rely on feature engineering to perform layout-specific extraction of information and therefore do not generalize well. Extracting information without taking the document layout into consideration is required as a first step to develop a general solution to this problem. To address this challenge, we propose a deep learning based pipeline to extract information from documents. For this purpose, we define 'information' to be a set of entities that have a label and a corresponding value, e.g., application_number: ADNF8932NF and submission_date: 15FEB19. We form relational triplets by connecting one entity to another via a relationship, such as (max_temperature, is, 100 degrees) and train a neural tensor network that is well-suited for this kind of data to predict high confidence scores for true triplets. Up to 96% test accuracy on real world documents from publicly available GHEGA dataset demonstrate the effectiveness of our approach.

Keywords: Named Entity Recognition · Neural Tensor Networks · Semi Structured Documents.

1 Introduction

Information extraction from documents is crucial for many applications like invoice automation, knowledge management and preservation, and information retrieval. A key components in such systems is the entity extraction for relevant information. These entities could be personal information (like name, date of birth, address, emails, etc.), invoice data (like date, price of certain items, total amount, SKUs, etc.), technical data of components (like operating temperatures, power dissipation values, potential hazards, etc.), or legal data (like date of contract, expiration date, specific laws and regulations, etc.).

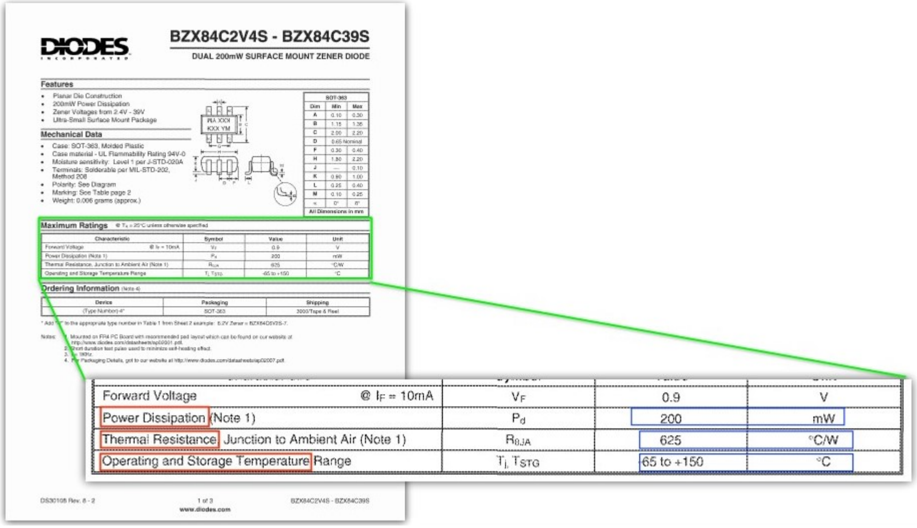
Entity Recognition can be applied to both speech and textual data. In the current scope of work, we are referring only to textual data and that too contained in paper based documents such as technical specifications, forms, legal contracts. One can divide documents into three broad categories: Unstructured (where information is contained in paragraphs and no explicit labels are present), Structured (where information is presented with specific labels), and Semi structured (where information is partially (un) structured). With respect to the complexity of information extraction from these three types of documents, one can imagine that the structured documents are the easiest form of documents, because each information pair is explicitly available in the document. On the other hand, it is very challenging to recognize entities in an unstructured documents due to the absence of specific tags or labels.

The most commonly found methodology in literature to deal with documents is to first identify the layout and then to apply pattern recognition and machine learning algorithms for entity recognition. Layout based machine learning and rule based approaches mostly utilize the layout and format specific geometric features to extract information from documents [10], [6], [4], [12], [13]. These approaches work well when the test image and the training documents have comparable layouts. A challenge in information extraction is that information can be located anywhere on the document page and in any form of data structure, such as tables, figures, logos or the regular text paragraphs. This is a problem with layout based approaches. They are not general solutions to extract information from unseen layouts and document formats. They require human intervention to correct the extraction mistakes in such cases [10], [6]. Other challenges in information extraction include variation in fonts of the text, variation in field sizes associated with a label and presence of rare labels of interest that are not found in all documents [10]. Two examples of such documents are shown in Figure 1 and 2. Figure 1 shows a datasheet document where the labels and their corresponding values are arranged from left to right, horizontally. Figure 2 shows a patent document where the labels and their corresponding values are placed vertically and the field sizes are larger than one sentence.

Both text and geometric features can help in information extraction. For example, one of the ways dates are usually represented is MM/DD/YYYY and the knowledge that where the piece of text appears on the page further helps in associating it with its correct label. Geometric features are mostly layout dependent and if an information extraction system uses them alone, or even in conjunction with textual features, it makes the solution layout dependent as well. Varying layouts makes this problem difficult to solve.

Our work is an effort to develop a general system for information extraction that can work on a broad spectrum of document classes. In this regard, we have chosen to work with the textual features only because they are more general compared to geometric features, since information in text is mostly represented in the same way even if the document layouts change.

Natural language processing (NLP) is one of the domains in which machine learning and deep learning in particular have made great progress. One of the



DIODES INCORPORATED

BZX84C2V4S - BZX84C39S

DUAL 200mW SURFACE MOUNT ZENER DIODE

Features

- Planar Die Construction
- 200mW Power Dissipation
- Zener Voltages from 2.4V - 39V
- Ultra-Small Surface Mount Package

Mechanical Data

- Case: SOT-363, Molded Plastic
- Case material - UL Flammability Rating 94V-0
- Moisture sensitivity: Level 1 per J-STD-020A
- Terminals: Solderable per MIL-STD-202, Method 208
- Polarity: See Diagram
- Marking: See Table page 2
- Weight: 0.006 grams (approx.)

SOT-363		
Dim	Min	Max
A	0.10	0.30
B	1.15	1.35
C	2.00	2.20
D	0.65 Nominal	
F	0.30	0.40
H	1.80	2.20
J	—	0.10
K	0.90	1.00
L	0.25	0.40
M	0.10	0.25
α	0°	8°
All Dimensions in mm		

Maximum Ratings @ T_A = 25°C unless otherwise specified

Characteristic		Symbol	Value	Unit
Forward Voltage	@ I_F = 10mA	V_F	0.9	V
Power Dissipation (Note 1)		P_d	200	mW
Thermal Resistance, Junction to Ambient Air (Note 1)		$R_{\theta JA}$	625	°C/W
Operating and Storage Temperature Range		T_j, T_{STG}	-65 to +150	°C

Ordering Information (Note 4)

Device	Packaging	Shipping
(Type Number)-7*	SOT-363	3000/Tape & Reel

* Add "-7" to the appropriate type number in Table 1 from Sheet 2 example: 6.2V Zener = BZX84C6V2S-7.

Notes:
1. Mounted on FR4 PC Board with recommended pad layout which can be found on our website at http://www.diodes.com/datasheets/ap02001.pdf.
2. Short duration test pulse used to minimize self-heating effect.
3. f = 1KHz.
4. For Packaging Details, go to our website at http://www.diodes.com/datasheets/ap02007.pdf.

Forward Voltage	@ I_F = 10mA	V_F	0.9	V
Power Dissipation (Note 1)		P_d	200	mW
Thermal Resistance, Junction to Ambient Air (Note 1)		$R_{\theta JA}$	625	°C/W
Operating and Storage Temperature Range		T_j, T_{STG}	-65 to +150	°C

DS30108 Rev. 8 - 2 1 of 3 BZX84C2V4S - BZX84C39S
www.diodes.com

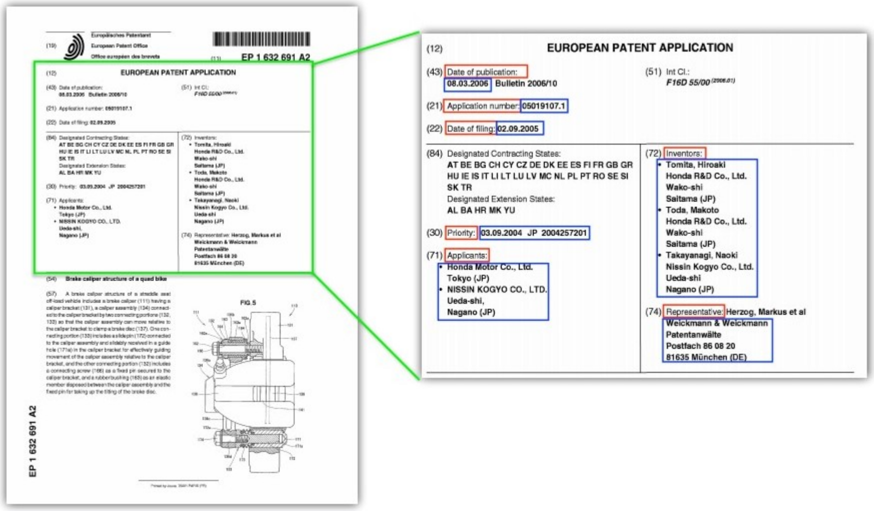
Fig. 1: A sample of datasheet document from GHEGA dataset. The red boxes contain the labels and the blue boxes have their corresponding values.

most prominent applications of NLP is search via speech where we can talk to our smart phone and our speech gets recognized as a query and the results are returned as speech again. The most popular commercial applications of this technology are found in Google's home, Amazon's Alexa, Window's Cortana and Apple's Siri assistant systems [9]. Other common applications of NLP include language modelling, language translation, speech recognition and image captioning.

Our hypothesis is that if the problem of information extraction can be modeled in a way that it can incorporate the kind of understanding that language models in modern machine learning have, we could get state-of-the-art results without using the knowledge of the document types. Since text is general and common to all documents, such a method could generalize to a variety of document classes. For example, a publication date can be present in a scientific paper, as well as a tender document and even a patent, and it is likely that all three are presented in the same way in text, but different geometrically.

2 Related Work

Documents are broadly classified into three classes based on their layouts [5]; highly structured documents that have a static well-defined layout, semi-structured documents that can be described in terms of templates partially but are not as well-defined as the former and loosely structured documents that cannot be associated with geometric layouts. Our solution mainly targets the first two document types; the highly structured and semi-structured documents.



(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 632 691 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
08.03.2006 Bulletin 2006/10

(51) Int Cl.:
F16D 55/00 $^{(2006.01)}$

(21) Application number: 05019107.1

(22) Date of filing: 02.09.2005

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR
HU IE IS IT LI LT LU LV MC NL PL PT RO SE SI
SK TR
Designated Extension States:
AL BA HR MK YU

(30) Priority: 03.09.2004 JP 2004257201

(71) Applicants:
- Honda Motor Co., Ltd.
Tokyo (JP)
- NISSIN KOGYO CO., LTD.
Ueda-shi,
Nagano (JP)

(72) Inventors:
- Tomita, Hiroaki
Honda R&D Co., Ltd.
Wako-shi
Saitama (JP)
- Toda, Makoto
Honda R&D Co., Ltd.
Wako-shi
Saitama (JP)
- Takayanagi, Naoki
Nissin Kogyo Co., Ltd.
Ueda-shi
Nagano (JP)

(74) Representative: Herzog, Markus et al
Weickmann & Weickmann
Patentanwälte
Postfach 86 08 20
81635 München (DE)

(54) Brake caliper structure of a quad bike

(57) A brake caliper structure of a straddle seat off-load vehicle includes a brake caliper (111) having a caliper bracket (131), a caliper assembly (134) connected to the caliper bracket by two connecting portions (132, 133) so that the caliper assembly can move relative to the caliper bracket to clamp a brake disc (137). One connecting portion (133) includes a slide pin (172) connected to the caliper assembly and slidably received in a guide hole (171a) in the caliper bracket for effectively guiding movement of the caliper assembly relative to the caliper bracket, and the other connecting portion (132) includes a connecting screw (168) as a fixed pin secured to the caliper bracket, and a rubber bushing (163) as an elastic member disposed between the caliper assembly and the fixed pin for taking up the tilting of the brake disc.

EP 1 632 691 A2

Printed by Jouve, 75001 PARIS (FR)

Fig. 2: A sample of patent document from GHEGA dataset. The labels are in the red boxes while the blue ones contain their values.

OCR error correction can be performed before it is used for information extraction [16]. The OCR text is corrected by string cleaning, substitution, syntactic and semantic checks based on the domain knowledge learned about the labels and their values during training. OCR error has not been taken into account for our work and we chose to work directly with raw OCR output.

Cesarini et al. [4] proposed a method for information extraction from invoices that utilized a three-step pipeline for information extraction. The total dataset they used contained 83 classes of documents with respect to the issuing firm and extracted only two labels, invoice total and invoice number. The labels are called tags and associated values are logical objects. Match is found between them on the basis of the text contained in the two entities or their positions. Document pages are divided into nine regions and distances between the entities on the document are expressed as vectors. A number of thresholds based on hit-and-trial are utilized for this recognition including thresholds for character recognition, average height and average width of bounding boxes. Overall the method is layout based and the class-dependent and even independent knowledge for inference comes from document layouts seen during training.

Esser et al. [6] proposed another layout dependent semi-automatic approach for indexing of required labels in documents. The system begins with an annotated set of training documents which can be augmented with user feedback. The system first performs template matching to find a subset of training documents that best matched the test image using K-nearest neighbour. This subset allows for extracting the positional index data information from the documents already annotated correctly and are transferred for extracting information from the test image. Template matching rate of 98% and information extraction accuracy over 90% for some classes is achieved attributing to the fact the class of business documents they used were assumed to be adhering to the same layouts with minor changes. They introduced feedback from the user into the system to enable addition of new layouts and correction of wrong extractions.

Rusinol et al. [12] proposed incremental structural templates for information extraction. A single image from each provider of the documents is used to train the model. Users manually annotate these images to build weighted star graphs connecting targets to labels where heavy weights are assigned to the physically closer bounding boxes. Relations are encoded by storing the polar coordinates of the vector between target and label bounding boxes (r, θ). At test time, the document class is determined, then the label of interest is located in the star graph to find all its occurrences and voting is done to assign a target value to this label. Star graphs are updated to store new documents and word labels for continuous learning. This approach performed well on a set of known document formats, but the method of connecting labels and targets relies on the document layouts and correct extractions depend on train and test layout similarity.

In Intellix system [13], the extraction pipeline begins by identifying the document class using a variant of kNN on a bag of words of the test document. Next, like other techniques mentioned before, template matching is done using another kNN getting the set of closest resembling layouts of documents in the training set that match the test document layout. Then three indexers are used in combination, namely a fixed-field indexer, a position-based indexer and a context-based indexer followed by a corrector that weighs and combines the scores and assigns a value with the highest score to the label of interest. The extraction steps are different compared to Rusinol et al. [12], but the pipeline seen as a black box as well as the dependence of the approach on seen document layouts is similar.

Medvet et al. [10] divided documents into a set of blocks, and defined rules for each label and a corresponding extraction function for assigning target values to labels. OCR output was used for finding the bounding boxes and a graphical user interface is provided to the users to label the training images of their choice. After that, they system would use the labeled examples to find out the extraction function parameters based on the text and positional relations between the bounding boxes of the labels and its values in the test image. This system, like the rest mentioned before, uses features that are typical to documents coming from the same provider or at least generated using the same software because it relies on layout for extraction.

RelNet, proposed by Bansal et al. [1] is an end-to-end model for entities and relations. RelNet models entities as abstract memory slots and relations between all memory pairs as an additional relational memory. This model is suitable for questioning answering tasks.

Strubell et al. [17] introduced iterated diluated convolutional neural networks as an alternative to Bidirectional LSTM networks for NER tasks. These networks exploits the power of GPUs massive speed advantages for several NLP tasks like sequence labeling, entity recognition, etc.

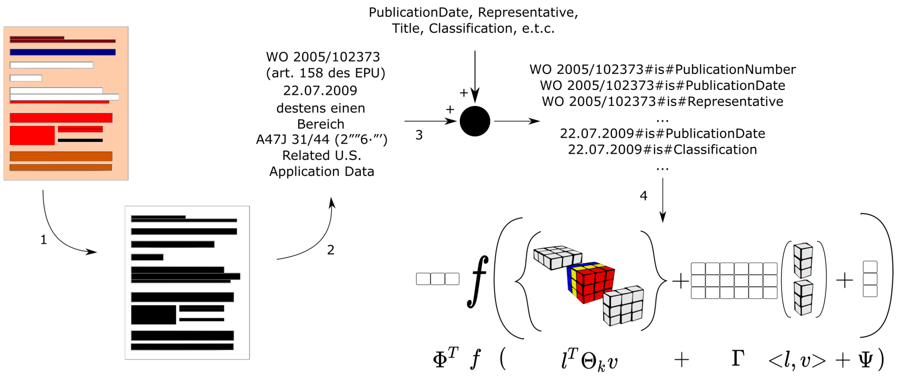

PublicationDate, Representative,
Title, Classification, e.t.c.
WO 2005/102373
(art. 158 des EPU)
22.07.2009
destens einen
Bereich
A47J 31/44 (2″″6·″′)
Related U.S.
Application Data
+
+
3
WO 2005/102373#is#PublicationNumber
WO 2005/102373#is#PublicationDate
WO 2005/102373#is#Representative
...
22.07.2009#is#PublicationDate
22.07.2009#is#Classification
...
4
1
2
f
Φ^T f ($l^T \Theta_k v$ + Γ $<l, v>$ + Ψ)

Fig. 3: Our proposed pipeline for information extraction. First, a test image is binarized and deskewed at Step-**1**. Then, OCRopus 2.0 is used for the OCR to get text boxes and text lines at Step-**2**. The obtained text boxes are combined with the predefined set of all the labels to extract, by forming all the possible permutations of labels and text boxes at Step-**3** and relational triplets are formed by combining the labels and the text boxes via the 'is' relation. Finally at Step-**4**, the triplets are given as input to the neural tensor network as the embedding vectors of the two entities, i.e., the labels and the possible corresponding value in the text box to get a prediction score. The last step shows a visualization of a neural tensor network [11], with $k = d = 3$. The 3d cube in the curly braces is the representation of the Θ_k tensor, with size $d \times d \times k$, a 3rd order tensor.

Socher et al. [15] proposed Neural Tensor Networks for relational classification. We surmise that this might be a very effective way of modeling the connection between a label and its corresponding value, that we call information. Our work mainly investigates this hypothesis. For knowledge base completion, they compared a variety of network architectures including distance model, single layer fully-connected model, hadamard model and bilinear model with their own proposed neural tensor network to classify whether two entities are connected by a relationship 'R' by forming relational triplets. Our approach for information extraction is inspired by their work and their network is explained in more detail in the next section where we present our solution pipeline.

3 Proposed Solution

We propose the pipeline shown in Figure 3 for extracting information from documents. As the first step, we preprocess the input document image, which includes adaptive binarization [14] and de-skewing [19] followed by OCR using OCRopus 2.0 [2]. The relational triplets are formed by combining the text boxes in the OCR output at sentence level with the labels of our interest via a relation, by forming all possible permutations. For our research, we utilize only the "is" relation because we want to verify whether temperature "is" 45 degrees, distance "is" 9km and so on. All of the relational triplets are passed for classification to the neural tensor network which is optimized for information extraction. The entities are character strings, so we convert them to their vector representation, which we obtain by training the Word2Vec algorithm on our whole dataset of all the available datasheets and patents. We train from scratch the continuous bag-of-words (CBOW) architecture for getting our entity vectors. Then, the neural tensor network is trained to output high scores for correct triplets, based on which we can extract all the values associated with every single label that we want to look for in the document.

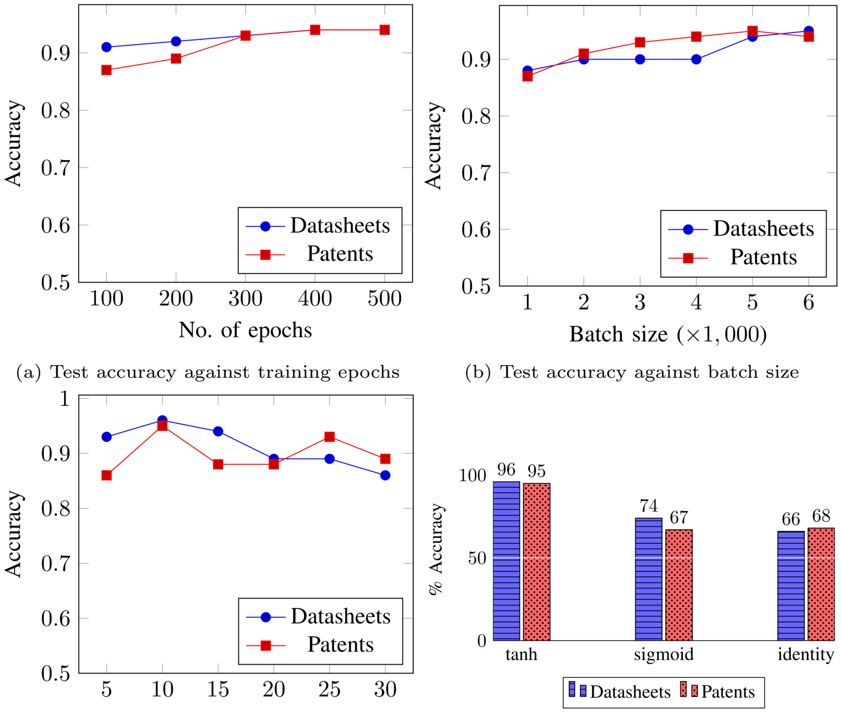
A neural tensor network differs from a fully-connected neural network in the sense that instead of having two dimensional weight matrices (which are 2nd order tensors), the weights are tensors of 3rd order which interact with the input entity vectors [15]. The network function is written as

$$p(l, R, v) = \Phi^T f(l^T \Theta_k v + \Gamma {<} l, v {>} + \Psi) \tag{1}$$

where f is a non-linear activation function and l and v are both 100-dimensional entity vectors, label and value vectors respectively. ${<}l, v{>}$ is the row concatenation of the two entity vectors. The parameter 'k' in the weight tensor, Θ, is called the *slice size*, which gives a third dimension to the weights, making the weight tensor $\Theta_k \in \mathbb{R}^{d \times d \times k}$, where 'd' is the dimension of entity vectors, and converts an ordinary fully-connected linear multiplication to a bilinear tensor product $l^T \Theta_k v$. Also $\Gamma \in \mathbb{R}^{k \times 2d}$ and $\Phi, \Psi \in \mathbb{R}^k$. More recently, new work has been done on the same problem [8], [7], [3], [18] but we want to demonstrate the effectiveness of modelling information extraction problem in this manner and not the problem of reasoning itself. The problem of knowledge base completion is beyond the scope of our research and we directly use the neural tensor model proposed by [15] and propose a different dimension to look at information extraction from a knowledge modelling point of view that makes the whole solution pipeline independent of the document layout.

For this work, we considered all of the OCR output from the available set of data-sheets and patents for training and testing. Many of the text blocks in the OCR are not associated with any label from our set of labels, so we assign them a sentinel label "unknown". The network was trained by passing the true triplets in the dataset as well as *corrupt triplets* that were constructed by replacing the associated value of a label with any random entity from the dataset. Against each true triplet, we tested different numbers of corrupt triplets for best performance. The ratio of number of corrupt triplets to the number of true triplets is called the *corrupt size*. No GPU was needed for this training because the model is small in size and fits in main memory, which means that inference on a document with a huge number relational triplet permutations can also run on the CPU. The slice size 'k' of the weight tensor is 3 and the words embedding were taken to be 100-dimensional. The following Hinge cost function is used for training the network

$$C(Y) = \sum_{i=1}^{N_E} \sum_{c=1}^{N_C} \max(0, 1 - p(T^{(i)}) + p(T_c^{(i)})) + \lambda ||Y||_2^2 \tag{2}$$



(a) Test accuracy against training epochs

(b) Test accuracy against batch size

(c) Test accuracy against corrupt size

(d) Test accuracy against activation functions

Fig. 4: The results of hyper parameter tuning experiments.

where N_E is the number of training examples, N_C is the number of corrupt examples taken against one true triplet (corrupt size), $Y = (\Phi, \Theta_k, \Gamma, \Psi)$ is the set of all the weights that are to be optimized, and $T^{(i)} = (l^{(i)}, R^{(i)}, v^{(i)})$ is a correct training relational triplet while $T_c^{(i)} = (l^{(i)}, R^{(i)}, v_c^{(i)})$ is a corresponding corrupt triplet. Similar to Socher et al. L-BFGS is used for optimizing the weights of the network [15].

4 Experiments

4.1 Data set

We used Ghega dataset for demonstrating our results, which is the same dataset used by [10], except their invoices have not been made publicly available. It consists of 136 patents and 110 data-sheets documents. One sample from each is shown in Figure 1 and Figure 2. The ground truth containing the OCR files for these two document types are also publicly available. For the patents, the labels we considered as information for our work include *Title, Priority, Inventor,*

Embeddings	**Datasheets**	**Patents**
Learned	**0.96**	**0.96**
Random	0.75	0.77

Table 1: Accuracy against embedding

Applicant, Classification, Filling Date, Publication Date, Representative, Abstract 1st line, Application Number, and *Publication Number*. For data-sheets, we considered *Model, Type, Case, Voltage, Weight, Power Dissipation, Thermal Resistance*, and *Storage Temperature*. The number of entities identified are 7847 in the data-sheet documents and 6218 in patents.

4.2 Evaluation Metrics

We evaluate the performance of our approach using the following metrics.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{3}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{4}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{5}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \tag{6}$$

where TP, TN, FP and FN are True Positives, True Negatives, False Positives and False Negatives.

4.3 Results and Discussion

We have tested our approach with different hyper parameters for the neural tensor network. Please refer to Figure 4 for the results of hyper parameter tuning. A 80/20 train and test split was used for our experiments. We performed cross-validation by constructing five different train and test splits for both document types. The lowest accuracy for any train and test split for our experiments was 93% for data-sheets and 92% for patents. We found that consistent results are obtained for both types of documents by having a tensor network with slice size of 3 and Γ and Ψ parameters set to 0, thus reducing the network to the bilinear product only. If all three components are used with different slice sizes, then a slice size of 1 yields up to 96% accuracy for data-sheets, but the patents perform slightly worse at 93% even with 3 slices. Our best results with slice size 3 and Γ and Ψ set to 0 are summarized in Tables 1-3.

Our accuracy peaks at a corrupt size of 10. For this result, tanh activation was used with learned embeddings. We also tested our method for learned and

Label	Precision	Recall	F1
Title	0.87	0.91	0.89
Applicant	0.90	0.95	0.92
Inventor	0.97	0.92	0.94
Representative	0.92	0.99	0.95
Filling Date	0.91	0.95	0.93
Publication Date	0.89	0.96	0.92
Application Number	0.93	0.80	0.86
Publication Number	0.89	0.99	0.94
Priority	0.96	0.89	0.92
Classification	0.97	0.91	0.94
Abstract 1st line	0.92	0.86	0.89
Unweighted Average	0.92	0.92	0.92

Table 2: Per-class results for patents. The sum of values in each column is divided by the number of values to get their average values.

random embedding vectors for the labels and target values and found that the learned embeddings work better. For this dataset, a tanh activation outperforms sigmoid and identity non-linearities by a big margin. For all the settings we tested, we found that power dissipation in datasheets and abstract 1st line in patents performed the worst. Per-class results are provided in Table 2 and Table 3.

We have also compared our results with those reported in Medvet et al. [10]. They report a "success rate" or precision of 91% for a fixed training set of 15 documents with no human supervision while we get 94% precision for data-sheets and 92% for patents. Since [10] uses layout information, fewer examples lead to better results because the layouts seen at test time are similar to the training inputs. For our approach, since we are learning from text, a larger training set is required to perform at the same level. Medvet et al. also experimented with feedback via human intervention by letting the user correct the results of incorrect extractions. They found out that supervision increases precision to 96%, and our system is capable of producing competitive results using just textual information without any human intervention.

Label	**Precision**	**Recall**	**F1**
Model	0.96	0.98	0.97
Type	0.98	0.97	0.97
Case	0.92	0.98	0.95
Power Dissipation	0.94	0.89	0.92
Storage Temperature	0.90	0.93	0.91
Voltage	0.91	0.95	0.93
Weight	0.91	0.99	0.95
Thermal Resistance	0.99	0.96	0.98
Unweighted Average	0.94	0.96	0.95

Table 3: Per-class results for datasheets. The sum of values in each column is divided by the number of values to get their average values.

5 Conclusion

We have presented a layout independent pipeline for information extraction from document images. We are able to obtain highly accurate results on patents and data-sheets for relational classification. Our problem formulation and feature selection are general and layout independent. Results were obtained by forming relational triplets of entities found in the OCR output provided with the dataset with a set of labels defined beforehand. In future, we would develop an end-to-end system, that would incorporate this method of information extraction and work holistically on a document by obtaining OCR, getting the bounding boxes at different levels and testing all the permutations to obtain a fully extracted information result. Newer methods of modelling information other than the neural tensor network of [15] should also be tested for finding more optimal solutions in terms of processing time for faster performance. Since the input features are textual, we would want to test our approach on a variety of other document types such as invoices and tender documents to test its effectiveness.

References

1. Bansal, T., Neelakantan, A., McCallum, A.: Relnet: End-to-end modeling of entities & relations. CoRR **abs/1706.07179** (2017), http://arxiv.org/abs/1706.07179
2. Breuel, T.M.: The ocropus open source ocr system. In: Document Recognition and Retrieval XV. vol. 6815, p. 68150F. International Society for Optics and Photonics (2008)

3. Cai, C.H., Ke, D., Xu, Y., Su, K.: Symbolic manipulation based on deep neural networks and its application to axiom discovery. In: 2017 International Joint Conference on Neural Networks (IJCNN). pp. 2136–2143. IEEE (2017)
4. Cesarini, F., Francesconi, E., Gori, M., Soda, G.: Analysis and understanding of multi-class invoices. Document Analysis and Recognition **6**(2), 102–114 (2003)
5. Dengel, A.R.: Making documents work: Challenges for document understanding. In: 7th International Conference on Document Analysis and Recognition. p. 1026. IEEE (2003)
6. Esser, D., Schuster, D., Muthmann, K., Berger, M., Schill, A.: Automatic indexing of scanned documents: a layout-based approach. In: Document Recognition and Retrieval XIX. vol. 8297, p. 82970H. International Society for Optics and Photonics (2012)
7. Liu, Q., Jiang, H., Evdokimov, A., Ling, Z.H., Zhu, X., Wei, S., Hu, Y.: Probabilistic reasoning via deep learning: Neural association models. arXiv preprint arXiv:1603.07704 (2016)
8. Liu, Q., Jiang, H., Ling, Z.H., Zhu, X., Wei, S., Hu, Y.: Combing context and commonsense knowledge through neural networks for solving winograd schema problems. In: AAAI Spring Symposium Series (2017)
9. López, G., Quesada, L., Guerrero, L.A.: Alexa vs. siri vs. cortana vs. google assistant: a comparison of speech-based natural user interfaces. In: International Conference on Applied Human Factors and Ergonomics. pp. 241–250. Springer (2017)
10. Medvet, E., Bartoli, A., Davanzo, G.: A probabilistic approach to printed document understanding. International Journal on Document Analysis and Recognition (IJDAR) **14**(4), 335–347 (2011)
11. Nieze, A.: How to draw a rubik's cube in inkscape (Sep 2014), http://goinkscape.com/how-to-draw-a-rubiks-cube-in-inkscape/
12. Rusinol, M., Benkhelfallah, T., Poulain d'Andecy, V.: Field extraction from administrative documents by incremental structural templates. In: 12th International Conference on Document Analysis and Recognition. pp. 1100–1104. IEEE (2013)
13. Schuster, D., Muthmann, K., Esser, D., Schill, A., Berger, M., Weidling, C., Aliyev, K., Hofmeier, A.: Intellix–end-user trained information extraction for document archiving. In: 12th International Conference on Document Analysis and Recognition. pp. 101–105. IEEE (2013)
14. Shafait, F., Keysers, D., Breuel, T.M.: Efficient implementation of local adaptive thresholding techniques using integral images. In: Document recognition and retrieval XV. vol. 6815, p. 681510. International Society for Optics and Photonics (2008)
15. Socher, R., Chen, D., Manning, C.D., Ng, A.: Reasoning with neural tensor networks for knowledge base completion. In: Advances in neural information processing systems. pp. 926–934 (2013)
16. Sorio, E., Bartoli, A., Davanzo, G., Medvet, E.: A domain knowledge-based approach for automatic correction of printed invoices. In: International Conference on Information Society (i-Society 2012). pp. 151–155. IEEE (2012)
17. Strubell, E., Verga, P., Belanger, D., McCallum, A.: Fast and accurate sequence labeling with iterated dilated convolutions. CoRR **abs/1702.02098** (2017), http://arxiv.org/abs/1702.02098
18. Trivedi, R., Dai, H., Wang, Y., Song, L.: Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 3462–3471. JMLR. org (2017)

19. Van Beusekom, J., Shafait, F., Breuel, T.M.: Combined orientation and skew detection using geometric text-line modeling. International Journal on Document Analysis and Recognition (IJDAR) **13**(2), 79–92 (2010)