AURORA: An Information Extraction System of Domain-specific Business Documents with Limited Data

Conference Paper · October 2020

DOI: 10.1145/3340531.3417434

8 authors, including:

Minh-Tien Nguyen
Hung Yen University of Technology and Education
54 PUBLICATIONS **201** CITATIONS
SEE PROFILE

Dung Tien Le
Cinnamon AI
8 PUBLICATIONS **5** CITATIONS
SEE PROFILE

Do Hoang Thai Duong
Cinnamon
3 PUBLICATIONS **0** CITATIONS
SEE PROFILE

AURORA: An Information Extraction System of Domain-specific Business Documents with Limited Data

Minh-Tien Nguyen
CINNAMON LAB, 10th floor, Geleximco building, 36 Hoang Cau, Dong Da, Hanoi, Vietnam.
Hung Yen University of Technology and Education
Hung Yen, Vietnam.
tiennm@utehy.edu.vn

Dung Tien Le, Le Thai Linh, Do Hoang Thai Duong, Bui Cong Minh, Nguyen Hong Son, Nguyen Hai Phong and Nguyen Huu Hiep
CINNAMON LAB, 10th floor, Geleximco building, 36 Hoang Cau, Dong Da, Hanoi, Vietnam.
(nathan,linhlt,howard,matthew,levi,greg,hubert)@cinnamon.is

ABSTRACT

Information extraction is a well-known topic that plays a critical role in many NLP applications as its outputs can be considered as an entrance step for digital transformation. However, there still exist gaps when applying research results to actual business cases. This paper introduces AURORA, an information extraction for domain-specific business documents. The intuition of AURORA is to use transfer learning for extraction. To do that, it utilizes the power of transformers for dealing with the limitation of training data in business cases and stacks additional layers for domain adaptation. We demonstrate AURORA in the context of actual scenarios where users are invited to experience two functions: fine-grained and whole paragraph extraction of Japanese business documents. A video of the system is available at http://y2u.be/xHQpYE41Tqw.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

information extraction, business document analysis, transformers

ACM Reference Format:
Minh-Tien Nguyen and Dung Tien Le, Le Thai Linh, Do Hoang Thai Duong, Bui Cong Minh, Nguyen Hong Son, Nguyen Hai Phong and Nguyen Huu Hiep. 2020. AURORA: An Information Extraction System of Domain-specific Business Documents with Limited Data. In *Woodstock '20: ACM Symposium on Neural Gaze Detection, June 19–23, 2020, Woodstock, NY*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Information extraction (IE) is an important research topic of natural language processing (NLP), in which IE provides a way for converting unstructured to structured data. The conversion can be considered as a crucial step for digital transformation [6, 7]. With the rapid growth of unstructured data, IE has received attention from the research community [1, 2, 11, 15]. From the business side, IE provides an effective method for document analysis and many systems have developed for real business cases [8, 10, 11, 15]. The outputs of IE systems can be used in many NLP applications such as question answering, information retrieval [12], or the automatic generation of ontology [5].

Named entity recognition (NER) is an important sub-task of IE, in which it focuses on extracting specific predefined entities, such as organization names, personal names, addresses, etc with many studies [4, 8, 9]. There are two common approaches for NER: dictionary-based [14] and machine learning-based methods [8, 9]. While NER has successfully achieved promising results on relatively-easy tasks, e.g. extracting persons or organizations, it is not always straightforward for applying common NER techniques to actual business cases, which require to distill a large number of specific information.

In the context of business document assessment, we introduce an actual scenario of IE, which challenges common NER algorithms. From the business side, users usually face the problem of understanding business documents. For example, given a bidding document, they want to know who is the payee, who is the payer, or the deadline for applying qualification. As a result, building an IE system is a potential solution. However, creating this system, in fact, is a non-trivial task due to two challenges. The first challenge is the limited number of training samples. In business cases, because IE only focuses on narrow and specific domains, so providing a large number of training data is expensive and requires a lot of effort. For example, we usually receive a small number of documents, e.g. 100 for training IE models. It is different from CoNLL, which provides around 15,000 training examples for recognizing four entity types. The second challenge comes from the nature of IE for business documents, in which the identification of entity types cannot be merely the categories [12]. As mentioned, two types of organizations, such as payee and payer in bidding documents should be identical.

This paper introduces AURORA, a prototype of our product based on transformers for transfer learning. The system bridges the gaps of IE between academia and industry by addressing the two challenges above. The goal of AURORA is to extract structured information from domain-specific business documents. The intuition behind the system is to utilize the power of transformers trained on a huge amount of general data and fine-tune the transformers into downstream tasks by using transfer learning. This simulates real cases in which we only receive a small number of training data for information extraction. AURORA has three crucial characteristics:

- It currently provides two main functions of extraction on two levels: fine-grained and whole paragraph extraction. The first function is for the case that users want to extract very

Woodstock '20, June 19–23, 2020, The draft version of the paper accepted by CIKM

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

detailed information. The second function is for users when they want to retain all information in paragraphs. We believe these two functions provide flexible solutions for users.
- It uses a small number of samples for training IE models. Precisely, we used 78 public bidding documents for training the fine-grained IE model and 45 public benefit pension plan documents for training the whole paragraph IE model. By using transfer learning based on transformers, AURORA can adapt to new domains with limited training data.
- It can extract a large number of entity types. For example, the fine-grained IE model can clip 24 information types (tags), which facilitate the analysis of users.

2 SYSTEM OVERVIEW

Figure 1 shows the architecture of the system. AURORA receives an input document that includes tags and segments (note that tags and segments were already annotated in the labeling process). A tag-segment pair is separated by a special token ([SEP], please refer to the original paper of BERT for more detail [3]). Each tag-segment pair is fed into the transformer layer to transform raw texts into hidden representation, which is the input of transfer learning. The information extraction layer detects the start and end positions of each segment as extracted information.

2.1 Transformers

A Transformer is a transduction model which relies on self-attention for computing the representation of its inputs and outputs, without using sequence-aligned RNNs or convolution [13]. The Transformer includes the architecture of encoder-decoder which uses stacked self-attention and point-wise, and fully connected layers. The attention is to map a query and a set of key-value pairs to an output. Then, the output is computed as a weighted sum of the values, where the weight corresponding to each value is computed by a compatibility function of the query with the correlated key.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

where dimension d_k of keys, and dimension d_v of values. Moreover, Transformer performs the attention function in parallel, resulting d_v-dimensional output values using "multi-head attention" as following: $MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O$ where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$.

It is possible to take any transformer to implement the system, however we selected BERT [13] as an example. The reason is that BERT pioneered a new way of training language models and obtained state-of-the-art results on many NLP tasks. More importantly, in actual cases, we usually receive a small number of training data which challenges the training process of IE models. To address this obstacle, AURORA employs BERT to utilize its power for dealing with the limitation of training data. The transformer layer outputs the hidden representation of input for the transfer learning layer.

2.2 Transfer learning

BERT provides an appropriate solution for input representation due to the contextual embeddings learned from a large amount of data. However, it needs to be adapted to downstream tasks. To do that, we employed transfer learning to fine-tuning BERT for our IE problem. The transfer learning was done by stacking a CNN layer on BERT due to the efficiency of CNN for capturing local context between tags and extracted information. We also tested with LSTM and BiLSTM but the model BERT+CNN outputs the best results.

The CNN layer includes two operations: convolution and pooling. The convolution transforms input vectors from BERT by using transformation functions to create feature maps. The pooling operates on the feature maps for removing unnecessary information. In practice, we used multiple kernel sizes ($N = 3$) to enrich representation and max pooling for avoiding over-fitting.

2.3 Information extraction

Once input data has transformed, the system extracts information. To do that, we formulated the extraction as a Question Answering (QA) task based on the suggestion from BERT. An input question (tag) and a passage as a single packed sequence were fed into the system. The prediction outputs the result of the dot product between token T_i and start (S)/end (E) vectors as the probability of word i being the start/end of the answer span (see Eq. 2). The final score of a potential answer spanned from position i to position j defined as $\max_{i,j}(ST_i + ET_j)$ with $j \geq i$.

$$P_{start_i} = \frac{e^{S.T_i}}{\sum e^{S.T_j}}; \qquad P_{end_i} = \frac{e^{E.T_i}}{\sum e^{E.T_j}} \quad (2)$$

The extraction uses the positions *start* and *end* to extract values (information) corresponding to input tags.

2.4 Implementation

For implementation, we used a multilingual BERT-base model trained for 102 languages (including Japanese) on a huge amount of texts from Wikipedia [3]. The BERT model has 12 layers, a hidden layer of 430 768 neurons, 12 heads, and 110M parameters.

For training the system, data was internally annotated by our QAs (quality assurance), who have at least the N3 certificate (Japanese-Language Proficiency Test - JLPT, with N1 is the highest level). Given a tag, the QAs assigned the starting position and ending position of a corresponding segment in the document. The starting and ending positions are an indicator showing that which segments belong to which tags. The training process was done in two steps: pre-training and fine-tuning. For the first step, the pre-trained weights of BERT were reused, while the weights of the rest layers were generated with a truncated normal distribution. For the second step, these weights were fine-tuned on new domains. Precisely, AURORA used 78 documents for training a fine-grained IE model and 45 documents for training a paragraph IE model. The system was fine-tuned in 20 epochs by using the cross-entropy loss function. The number of kernel sizes is 3 and the convolutional output size is 768. The training process was done with a single GPU.

2.5 Experimental results

We internally tested the accuracy of AURORA on two functions: fine-grained extraction and whole paragraph extraction. For the first function, AURORA was tested on two datasets: bidding and sale documents. The bidding documents are public data of competitive bids for development projects in Japan with 24 tags. The sale documents

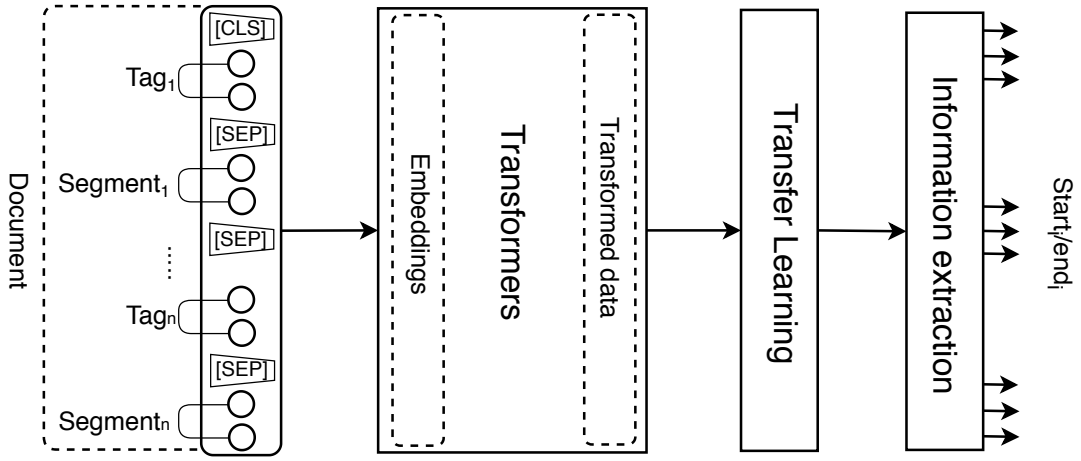

Figure 1: The system architecture of AURORA.

are public documents for selling hardware devices in Japanese with eight tags. For training, we used 78 bidding documents and 300 sale documents. For testing, AURORA was tested on 22 bidding documents and 165 sale documents. Table 1 shows the comparison.

Table 1: Comparison of methods according the average of F1-score. Bold is the best and *italic* is the second best. † shows that our model is significantly better with p-value ≤ 0.05.

Method	Bidding docs	Sale docs
BERT (QA)	*0.8607*	*0.8456*
BERT+CRF	0.1773†	0.1254†
BERT+LSTM+CRF	0.3817†	0.5572†
n-grams+MLP+Regex	0.8523	0.7177†
CNN+BiLSTM+CRF	0.6766†	0.7513†
AURORA	**0.9062**	**0.8614**

The evaluation used the average F-score over all tags (by fields) for comparison. As observed, AURORA achieves promising results compared to BERT, its extensions, and two methods based on n-grams features ($n = 4$) and CNN. BERT+CRF and BERT+LST+CRF output poor results because they were formulated as a NER task, not a QA task. The n-grams+MLP obtains the second-best results, showing that the n-grams features can be used for such IE task.

For whole paragraph extraction, AURORA was tested on 27 documents with five tags. As showed in Table 2, AURORA achieves the best results in terms of F-scores computed by char. The MLP method obtains very promising results by using n-grams features. Interestingly, the HAN (hierarchical attention network) model gives poor results due to the long-term dependency of tags and extracted information on this dataset.

3 DEMONSTRATION SCENARIO

The audience has the opportunity to experience Aurora on the web interface to extract information from business documents with two functions: fine-grained extraction and whole paragraph extraction. Figure 2 presents the interface.

Table 2: Comparison on the average of F1-score.

Method	Pension documents
BERT (QA)	*0.9251*
n-grams+MLP+Regex	0.9567
HAN	0.6436†
AURORA	**0.9738**

Users can upload an input document into the system. After uploading, the left side shows the original document and the right side contains extracted information. The interface shows some tags such as prefecture, the title of bidding, name of institution, address of demand, public announcement date, the deadline for delivery of the specification.

3.1 Testing datasets

We released two small testing datasets for the user's experience of AURORA. The first dataset includes five bidding samples collected from Japan Oil, Gas and Metals National Corporation (JOGMEC).[1] This dataset is used for testing the fine-grained extraction model trained on 78 bidding documents stated in Section 2.4. The second dataset contains five benefit pension plan documents. These samples are used for testing the whole paragraph extraction model trained on 45 pension plan documents stated in Section 2.4. All the samples can be download in the **Help** menu in Figure 2.

To provide a better assessment of AURORA, we introduce a larger dataset named CinBidding[2] for evaluating the quality of the fine-grained IE model. This is because in actual cases the extraction of fine-grained information receives higher demand than the extraction of whole paragraphs. The dataset includes 124 documents with correct answers, in which 82 documents are for training, 22 for development, and 20 for testing. The agreement computed by Cohen Kappa[3] among annotators is 0.8275. Also, note that we did not re-train the fine-grained IE model on this dataset; therefore, we leave the comparison on this dataset as a future task.

[1] http://www.jogmec.go.jp/news/bid/search.php
[2] https://github.com/DungLe13/bidding-dataset
[3] http://graphpad.com/quickcalcs/kappa1.cfm

Figure 2: The web interface of Aurora.

3.2 User interaction

AURORA[4] offers two main scenarios for user experience on the interface of the system. After logging the system with the user as **`cinnamon`** and the password as **`cinnamon`**, users can use these functions. Firstly, we encourage users to read the guideline of the system in the **`Help`** menu. This guideline explains the system, main functions, data, and how to use the system for IE. Users should download samples for testing the system. Once the samples are ready, users can select one of two functions: fine-grained extraction and whole paragraph extraction. For testing the fine-grained IE, users are encouraged to use both samples and the `CinBidding` dataset. It provides a better way to observe extracted information from the system and correct answers created by humans. After uploading testing samples and clicking on the process button, AURORA processes the user's requests. Finally, the extracted information is showed on the right side of the interface denoted in Figure 2.

REFERENCES

[1] Gabor Angeli, Melvin Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging Linguistic Structure for Open Domain Information Extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 344-354.*

[2] Luciano Del Corro and Rainer Gemulla. 2013. Clausie: Clause-based Open Information Extraction. In *Proceedings of the 22nd international conference on World Wide Web, pp. 355-366.*

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171-4186.*

[4] Jenny Rose Finkel and Christopher D. Manning. 2009. Nested Named Entity Recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Volume 1-Volume 1, pp. 141-150. Association for Computational Linguistics.*

[5] Michael Fleischman and Eduard Hovy. 2002. Fine Grained Classification of Named Entities. In *Proceedings of the 19th International Conference on Computational Linguistic, Volume 1, pp. 1-7. Association for Computational Linguistics.*

[6] Lindsay Herbert. 2017. *Digital Transformation: Build Your Organization's Future for the Innovation Age.* Technical Report. Bloomsbury Publishing.

[7] Bill Inmon and Anthony Nesavich. 2007. *Tapping into Unstructured Data: Integrating Unstructured Data and Textual Analytics into Business Intelligence.* Pearson Education.

[8] Meizhi Ju, Makoto Miwa, and Sophia Ananiadou. 2018. A Neural Layered Model for Nested Named Entity Recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 1446-1459.*

[9] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 260-270.*

[10] James Manyika, Michael Chui, and Mehdi Miremadi. 2017. *A Future that Works: AI, Automation, Employment, and Productivity.* Technical Report. McKinsey Global Institute Research, Tech. Rep, 60.

[11] Minh-Tien Nguyen, Viet-Anh Phan, Le Thai Linh, Nguyen Hong Son, Le Tien Dung, Miku Hirano, and Hajime Hotta. 2019. Transfer Learning for Information Extraction with Limited Data. In *Proceedings of 16th International Conference of the Pacific Association for Computational Linguistics.*

[12] Sonse Shimaoka, Pontus Stenetorp, Kentaro Inui, and Sebastian Riedel. 2016. An Attentive Neural Architecture for Fine-grained Entity Type Classification. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction, pp. 69-74.*

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Advances in Neural Information Pro- cessing Systems, pp. 6000–6010.*

[14] Yotaro Watanabe, Masayuki Asahara, and Yuji Matsumoto. 2007. A Graph-based Approach to Named Entity Categorization in Wikipedia Using Conditional Random Fields. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 649-657.*

[15] Ruixue Zhang, Wei Yang, Luyun Lin, Zhengkai Tu, Yuqing Xie, Zihang Fu, Yuhao Xie, Luchen Tan, Kun Xiong, and Jimmy Lin. [n.d.]. Rapid Adaptation of BERT for Information Extraction on Domain-Specific Business Documents. ([n. d.]). arXiv preprint arXiv:2002.01861.

[4] https://aurora-demo.cinnamon.is/login