Expert Systems With Applications 102 (2018) 100–112

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An attentive neural architecture for joint segmentation and parsing and its application to real estate ads

Giannis Bekoulis*, Johannes Deleu, Thomas Demeester, Chris Develder

IDLab, Department of Information Technology, Ghent University – imec, Technologiepark Zwijnaarde 15, Ghent 9052, Belgium

ARTICLE INFO

Article history:
Received 2 October 2017
Revised 18 January 2018
Accepted 20 February 2018

Keywords:
Neural networks
Joint model
Relation extraction
Entity recognition
Dependency parsing

ABSTRACT

In processing human produced text using natural language processing (NLP) techniques, two fundamental subtasks that arise are (i) *segmentation* of the plain text into meaningful subunits (e.g., entities), and (ii) *dependency parsing*, to establish relations between subunits. Such structural interpretation of text provides essential building blocks for upstream expert system tasks: e.g., from interpreting textual real estate ads, one may want to provide an accurate price estimate and/or provide selection filters for end users looking for a particular property – which all could rely on knowing the types and number of rooms, etc. In this paper, we develop a relatively simple and effective neural joint model that performs both segmentation and dependency parsing together, instead of one after the other as in most state-of-the-art works. We will focus in particular on the real estate ad setting, aiming to convert an ad to a structured description, which we name *property tree*, comprising the tasks of (1) identifying important entities of a property (e.g., rooms) from classifieds and (2) structuring them into a tree format. In this work, we propose a new joint model that is able to tackle the two tasks simultaneously and construct the *property tree* by (i) avoiding the error propagation that would arise from the subtasks one after the other in a pipelined fashion, and (ii) exploiting the interactions between the subtasks. For this purpose, we perform an extensive comparative study of the pipeline methods and the new proposed joint model, reporting an improvement of over three percentage points in the overall edge F_1 score of the property tree. Also, we propose attention methods, to encourage our model to focus on salient tokens during the construction of the *property tree*. Thus we experimentally demonstrate the usefulness of attentive neural architectures for the proposed joint model, showcasing a further improvement of two percentage points in edge F_1 score for our application. While the results demonstrated are for the particular real estate setting, the model is generic in nature, and thus could be equally applied to other expert system scenarios requiring the general tasks of both (i) detecting entities (*segmentation*) and (ii) establishing relations among them (*dependency parsing*).

1. Introduction

Many consumer-oriented digital applications rely on input data provided by their target audience. For instance, real estate websites gather property descriptions for the offered classifieds, either from realtors or from individual sellers. In such cases, it is hard to strike the right balance between structured and unstructured information: enforcing restrictions or structure upon the data format (i.e., predefined form) may reduce the amount or diversity of the data, while unstructured data (i.e., raw text) may require non-trivial (i.e., hard to automate) transformation to a more structured form to be useful/practical for the intended application. In the real estate domain, textual advertisements are an extremely useful but highly unstructured way of representing real estate properties. However, structured descriptions of the advertisements are very helpful, e.g., for real estate agencies to suggest the most appropriate sales/rentals for their customers, while keeping human reading effort limited. For example, special search filters, which are usually used by clients, cannot be directly applied to textual advertisements. On the contrary, a structured representation of the property (e.g., a tree format of the property) enables the simplification of the unstructured textual information by applying specific filters (e.g., based on the number of bedrooms, number of floors, or the requirement of having a bathroom with a toilet on the first floor), and it also benefits other related applications such as automated price prediction (Nagaraja, Brown, & Zhao, 2011; Pace, Barry, Gilley, & Sirmans, 2000).

* Corresponding author.
E-mail addresses: giannis.bekoulis@ugent.be (G. Bekoulis), johannes.deleu@ugent.be (J. Deleu), thomas.demeester@ugent.be (T. Demeester), chris.develder@ugent.be (C. Develder).

https://doi.org/10.1016/j.eswa.2018.02.031
0957-4174/

The new real estate structured prediction problem as defined by Bekoulis, Deleu, Demeester, and Develder (2017) has as main goal to construct the tree-like representation of the property (i.e., the *property tree*) based on its natural language description. This can be approached as a relation extraction task by a pipeline of separate subtasks, comprising (i) named entity recognition (NER) (Nadeau & Sekine, 2007) and (ii) relation extraction (Bach & Badaskar, 2007). Unlike previous studies (Li & Ji, 2014; Miwa & Bansal, 2016) on relation extraction, in the work of Bekoulis et al. (2017), the relation extraction module is replaced by dependency parsing. Indeed, the relations that together define the structure of the house should form a tree, where entities are *part-of* one another (e.g., a floor is *part-of* a house, a room is *part-of* a floor). This *property tree* is structurally similar to a parse tree. Although the work of Bekoulis et al. (2017) is a step towards the construction of the *property tree*, it follows a pipeline setting, which suffers from two serious problems: (i) error propagation between the subtasks, i.e., NER and dependency parsing, and (ii) cross-task dependencies are not taken into account, e.g., terms indicating relations (includes, contains, etc.) between entities that can help the NER module are neglected. Due to the unidirectional nature of stacking the two modules (i.e., NER and dependency parsing) in the pipeline model, there is no information flowing from the dependency parsing to the NER subtask. This way, the parser is not able to influence the predictions of the NER. Other studies on similar tasks (Kate & Mooney, 2010; Li & Ji, 2014) have considered the two subtasks jointly. They simultaneously extract entity mentions and relations between them usually by implementing a beam-search on top of the first module (i.e., NER), but these methods require the manual extraction of hand-crafted features. Recently, deep learning with neural networks has received much attention and several approaches (Miwa & Bansal, 2016; Zheng et al., 2017) apply long short-term memory (LSTM) recurrent neural networks and convolutional neural networks (CNNs) to achieve state-of-the-art performance on similar problems. Those models rely on shared parameters between the NER and relation extraction components, whereby the NER module is typically pre-trained separately, to improve the training effectiveness of the joint model.

In this work, we propose a new joint model to solve the real estate structured prediction problem. Our model is able to learn the structured prediction task without complicated feature engineering. Whereas previous studies (Li, Zhang, Fu, & Ji, 2017; Li, Zhang, Zhang, & Ji, 2016; Miwa & Bansal, 2016; Zheng et al., 2017) on joint methods focus on the relation extraction problem, we construct the *property tree* which comes down to solving a dependency parsing problem, which is more constrained and hence more difficult. Therefore, previous methods are not directly comparable to our model and cannot be applied to our real estate task out-of-the-box. In this work, we treat the two subtasks as one by reformulating them into a head selection problem (Zhang, Cheng, & Lapata, 2017).

This paper is a follow-up work of Bekoulis et al. (2017). Compared to the conference paper that introduced the real estate extraction task and applied some basic state-of-the-art techniques as a first baseline solution, we now introduce: (i) advanced neural models that consider the two subtasks jointly and (ii) modifications to the dataset annotation representations as detailed below. More specifically, the main contributions of this work are the following:

- We propose a new joint model that encodes the two tasks of identifying entities as well as dependencies between them, as a single head selection problem, without the need of parameter sharing or pre-training of the first entity recognition module separately. Moreover, instead of (i) predicting unlabeled dependencies and (ii) training an additional classifier to predict labels for the identified heads (Zhang et al., 2017), our model already incorporates the dependency label predictions in its scoring formula.
- We compare the proposed joint model against established pipeline approaches and report an F_1 improvement of 1.4% in the NER and 6.2% in the dependency parsing subtask, corresponding to an overall edge F_1 improvement of 3.4% in the property tree.
- Compared to our original dataset (Bekoulis et al., 2017), we introduce two extensions to the data: (i) we consistently assign the first mention of a particular entity in order of appearance in the advertisement as the main mention of the entity. This results in an F_1 score increase of about 3% and 4% for the joint and pipeline models, respectively. (ii) We add the *equivalent* relation to our annotated dataset to explicitly express that several mentions across the ad may refer to the same entity.
- We perform extensive analysis of several attention mechanisms that enable our LSTM-based model to focus on informative words and phrases, reporting an improved F_1 performance of about 2.1%.

The rest of the paper is structured as follows. In Section 2, we review the related work. Section 3 defines the problem and in Section 4, we describe the methodology followed throughout the paper and the proposed joint model. The experimental results are reported in Section 5. Finally, Section 6 concludes our work.

2. Related work

The real estate structured prediction problem from textual advertisements can be broken down into the sub-problems of (i) sequence labeling (identifying the core parts of the property) and (ii) non-projective dependency parsing (connecting the identified parts into a tree-like structure) (Bekoulis et al., 2017). One can address these two steps either one by one in a pipelined approach, or simultaneously in a joint model. The pipeline approach is the most commonly used approach (Bekoulis et al., 2017; Fundel, Kffner, & Zimmer, 2007; Gurulingappa, MateenRajpu, & Toldo, 2012), treating the two steps independently and propagating the output of the sequence labeling subtask (e.g., named entity recognition) (Chiu & Nichols, 2016; Lample, Ballesteros, Subramanian, Kawakami, & Dyer, 2016) to the relation classification module (dos Santos, Xiang, & Zhou, 2015; Xu et al., 2015). Joint models are able to simultaneously extract entity mentions and relations between them (Li & Ji, 2014; Miwa & Bansal, 2016). In this work, we propose a new joint model that is able to recover the tree-like structure of the property and frame it as a dependency parsing problem, given the non-projective tree structure we aim to output. We now present related works for the sequence labeling and dependency parsing subtasks, as well as for the joint models.

2.1. Sequence labeling

Structured prediction problems become challenging due to the large output space. Specifically in NLP, sequence labeling (e.g., NER) is the task of identifying the entity mention boundaries and assigning a categorical label (e.g., POS tags) for each identified entity in the sentence. A number of different methods have been proposed, namely Hidden Markov Models (HMMs) (Rabiner & Juang, 1986), Conditional Random Fields (CRFs) (Lafferty, McCallum, & Pereira, 2001), Maximum Margin Markov Network (M^3N) (Taskar, Guestrin, & Koller, 2003), generalized support vector machines for structured output (SVM^{struct}) (Tsochantaridis, Hofmann, Joachims, & Altun, 2004) and Search-based Structured Prediction (SEARN) (Daumé, Langford, & Marcu, 2009). Those methods heavily rely on hand-crafted features and an in-depth review can be found