Multimodal Graph-Based Dependency Parsing of Natural Language

Amr Rekaby Salama(✉) and Wolfgang Menzel

Department of Informatics, University of Hamburg, Hamburg, Germany
{salama,menzel}@informatik.uni-hamburg.de

Abstract. Dependency parsing is a popular approach for syntactic analysis of natural language utterances. It concerns building a dependency tree of the linguistic input relying only on a model of syntactic regularities. The cognitive process of human language processing, however, has also access to other sources of knowledge, like visual clues that can be used to improve language understanding.

In this paper, we approach integrating visual context and linguistic information to improve the reliability of dependency parsing. To achieve this goal, we modify a state-of-the-art dependency parser to make it accept visual information as extra features in addition to the original linguistic input. All these inputs (features) are considered in the learning process of the trained model. Experiments have been carried out to investigate the contribution of this additional multimodal information on ambiguity resolution and parsing quality.

Keywords: Multimodal integration · Graph-based dependency parsing · RBG parser

1 Introduction and Motivation

Natural language parsing usually suffers from the problem of ambiguity. Many ambiguities cannot be easily resolved using only the linguistic information. Trained models learn to disambiguate the syntactic structure according to the prior probability distribution found in the training data: The most frequently found interpretation will also be taken as the most plausible one, irrespective of any other factors which might contribute counter-evidence. Such behavior is highly undesirable in dynamic contexts, where the actual choice should also consider the current state-of-affairs in the world. In such a situation, including visual information into the decision process might help to find a better fitting interpretation.

One of the most noticeable sources of ambiguity in natural language is prepositional phrase (PP) attachment. As shown in Fig. 1, “I saw a girl with a telescope”, the decision between high and low attachment cannot be taken based on pure syntactic information, even lexical preferences don’t provide reliable clues. If the high attachment is adopted, the PP is attached to the verb which indicates its relation to the verb (I use the telescope to see the girl). In the low attachment case, PP is attached to the closest lexical item, which marks the coexistence of the PP with that item (I saw a girl who has a telescope with her). If we have available visual input in addition to the

A.E. Hassanien et al. (eds.), *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016*, Advances in Intelligent Systems and Computing 533, DOI 10.1007/978-3-319-48308-5_3

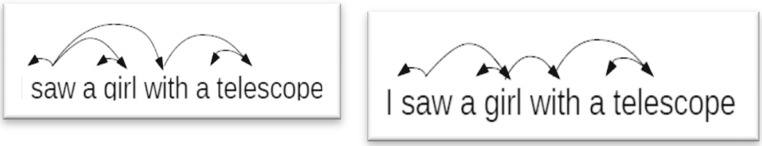

Fig. 1. (A) High attachment. (B) Low attachment

linguistic one, integrating them into the learning model might help in such a situation, if the kind of knowledge provided by the visual (context) information is beneficial to disambiguate the dependencies.

In this paper, we provide a multimodal dependency parser. The parser does not only depend on the linguistic input, but also on the non-linguistic modality. Although we consider the context (visual) information as the non-linguistic modality and inject such input into the parser by providing the relations between the elements in the context, we don't work on the level of relation extraction through image processing so far. Instead we focus on trying to overcome a range of other challenges in this research such as: introducing the context knowledge into the learning model of a graph-based parser, and manipulating the scoring function to take the decision based on the linguistic and non-linguistic modalities.

We use thematic roles in the form of triples as a description language stating the situation given in the non-linguistic context. The thematic roles contain information that helps in ambiguity resolution. Both linguistic and non-linguistic information are fed into the graph based dependency parser to improve its quality.

The paper starts with a review of dependency parsing methods (transition and graph based) and previous work on the integration of context information into a rule-based parser. In Sect. 3, we present our context-integrating model. The high-level architecture of the solution is presented in Sect. 4. Then a set of experiments is discussed in Sect. 5 before we state the conclusion and make proposals for future work.

2 Previous Work

2.1 Dependency Parsing

Dependency parsing extracts a syntactic dependency tree that describes binary relationships between the words of a sentence. The nodes of the tree correspond to the word forms in the sentence while the edges represent the dependency links between them in a child-parent relationship. These links are interpreted in terms of the functions that a lexical item fulfills with respect to its governor. These functions are described using labels attached to the edges. A valid dependency tree has to be an acyclic, and connected graph with a single head of each node (Nivre 2004).

Among the machine learning approaches for dependency parsing, there are two main methods: transition-based, and graph-based parsing. The transition-based (Shift-reduce) method constructs the tree incrementally, by attaching an incoming word immediately, or delaying its attachment until a better attachment point becomes available. The decision is based on an oracle which consults the history of prior attachments decisions. MALTparser is an example of this approach (Nivre et al. 2004).

Graph-based parsers start by creating a graph where each node represents a word from the sentence (Zhang et al. 2014a). All the nodes are connected to each other. A feature vector is assigned to each edge. The cost of each edge is dynamically learned based on a function of the training dataset. The parser finds a minimum spanning tree of the graph with the optimal score (Bohnet 2010). Different algorithms are used as alternatives for the creation of the minimum spanning tree: Chu-Liu/Edmond (Chu and Liu, 1965), and Hill-climbing (Zhanget al. 2014b). RBG parser (Lei et al. 2015) claims the state-of-the-art of graph-based parsing. It uses high-order features, different spanning tree decoding algorithms (Lei et al. 2014), a passive-aggressive online learning algorithm (MIRA), and parameter averaging (Crammer et al. 2006). It outperforms other dependency parsers quality.

2.2 Context Representation

The idea of utilizing context information from the visual environment in the dependency parsing was introduced by (McCrae 2009). He injected the visual information into a constraint-based parser (Weighted Constraint Dependency Grammar WCDG), and run his research on a German language dataset. He used the Web Ontology Language (OWL) to encode high-level descriptions of the visual input. Although OWL has two main components: t-box, a-box, he considered only the a-box to describe the relations between entities in the visual context. Under a-box representation, four thematic roles (Agent, Theme, Instrument, and Owner) are used to demonstrate the conceptual relationships in the context.

In this paper we implement our ideas by means of RBG parser to develop a proof of concept model, to demonstrate that the desired fusion of multimodal information can be achieved in a learning (graph-based) parsing model. The visual information is presented in form of thematic roles. Our experiments compare the results between the new implemented context-integrating parser that combines visual and linguistic information for English sentences against the original RBG parser as a benchmark.

3 Context-Integrating Dependency Parser

RBG parser considers only the linguistic input during model learning. It hypothesizes high order scoring functions to use them in minimum spanning tree extraction. To make it sensitive to visual information, we modify RBG parser to accept additional context information as features for the learning model. Our new version of RBG keeps linguistic features on the edges between combinatorial pairs of words in addition to newly introduced visual features between the entities (words) that have a relationship in the context (visual) input. As presented in Fig. 2, a visual relation has three parts: the relation type (agent, theme, etc.), the head of the relation which is the verb (except in the “owner” relation), and the modifier of the relation.

Figures 2 and 3 represent two pictures and their (visual) context information. The linguistic input corresponding to these figures are:

- 2: “The doctor with a coat feeds at this moment the journalist with a microphone”
- 3: “The journalist with a microphone feeds at this moment the doctor with a spoon”

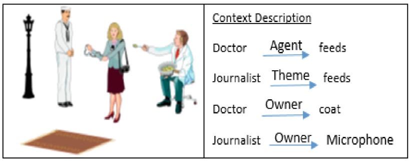

Fig. 2. The context information of an image (image taken from (Knoeferle 2005))

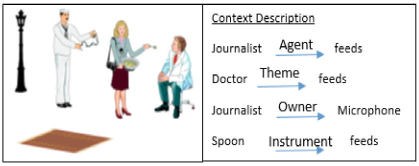

Fig. 3. The context features of an image (image taken from (Knoeferle 2005))

These sentences are confusing for a person if he/she hears them without the visual information. Using the visual information, however, one can differentiate that “microphone” in sentence 2 is an object with the journalist while the “Spoon” in sentence 3 is the tool for feeding and not an object with the doctor.

In Fig. 4, we present the adapted graph representation of the sentence in the context integrating RBG parser. We can find (t) words, $\{fv_{a=1\ldots m}\}$ linguistic feature vectors (the original ones from the RBG parser). As presented in Eq. 1, each vector has n features encoding the linguistic properties of the pair of words (i,j). It has additionally p.

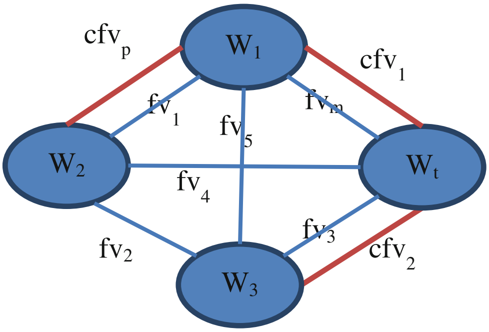

Fig. 4. Graph representation of the context-integrating dependency parsing

context feature vectors (newly introduced). These vectors consist of q visual features for the words pairs that have correlation in the context input.

As shown in Eq. 3, we build the learning model using multimodal inputs. In the testing phase, the model gets the linguistic information of the sentence in addition to the context information to find the optimal dependency tree. In equions 3, 4: x_i. is the input sentence, c_i is the context input for sentence (i), and $\tilde{y}$ the extracted dependency tree. For each sentence there is a set of possible trees $T(x_i)$, and a gold standard one $\hat{y}_i$. To represent the feature vector of the context input we use $\ddot{f}(c_i, y)$ with parameters (ω). $f(x_i, y)$ is used for the linguistic features and θ for the parameters.

$$fv_{a=1\ldots m} = \begin{pmatrix} f_{i,j,1} \\ \ldots \\ f_{i,j,n} \end{pmatrix} \tag{1}$$

$$cfv_{a=1\ldots p} = \begin{pmatrix} cf_{i,j,1} \\ \ldots \\ cf_{i,j,q} \end{pmatrix} \tag{2}$$

$$\tilde{y} = \max_{y \in T(x_i)} \{\theta.f(x_i, y) + \omega.\ddot{f}(c_i, y) + \|y - \hat{y}_i\|\} \quad \textit{Train} \tag{3}$$

$$\tilde{y} = \max_{y \in T(x)} \{\theta.f(\mathrm{x}, y) + \omega.\ddot{f}(\mathrm{c}, y)\} \quad \textit{Test} \tag{4}$$

For example, the context feature (HPp_HP_MAGP_MAGPn) describes four different aspects of a relation:

- HPp: The part of speech (POS) of the previous word of the Head.
  H: head of the visual relation. p: previous word. P: pos.
- HP: The head's POS.
  H: head of the visual relation. P: pos.
- AGP: The POS of the agent modifier.
  M: modifier of the visual relation. AG: agent relation. P: pos.
- MAGPn: The POS of the next word of the agent modifier.
  M: modifier of the visual relation. AG: agent relation. P: pos. n: next word.

Now we present how we encode this example of feature. At the beginning of the parsing process, the parser builds a dictionary of available POS tags and assign an ID to each one. We use this mapping to encode the visual relation. As shown in Fig. 2, "Doctor" is "Agent" of the "feeds" action. Here, "feeds" is the head of the visual relation, “agent” is the relation type, and “Doctor” is the modifier. “HPp” refers to the POS of the previous word of “feeds.” “HP” is the POS of “feeds.” “MAGP” is the POS of “Doctor.” “MAGPn” is the POS of the next word to “Doctor.”. Table 1 shows the encoding of this feature, and how it consists of different POS’s ID referring to the mentioned dictionary. This encoding is used as a feature ID in the parser learning process. This feature is added to the visual feature vectors between the two words in the adapted graph represented above.

Table 1. Coding of the example feature

Feature code	1100	0010	0100	0010	0011
	Visual Feature ID	*POS id of “coat”*	*POS id of “feeds”*	*POS id of “Doctor”*	*POS id of “with”*

4 Solution Architecture

In this section, we present the high-level architecture of the context-integrating RBG parser and how we introduce the visual modality in it. As shown in Fig. 5, there are three types of components:

- Components of the RBG parser that are kept without modification (Old).
- Components of the RGB parser that have been changed to be compatible with multi-modal parsing (Changed).

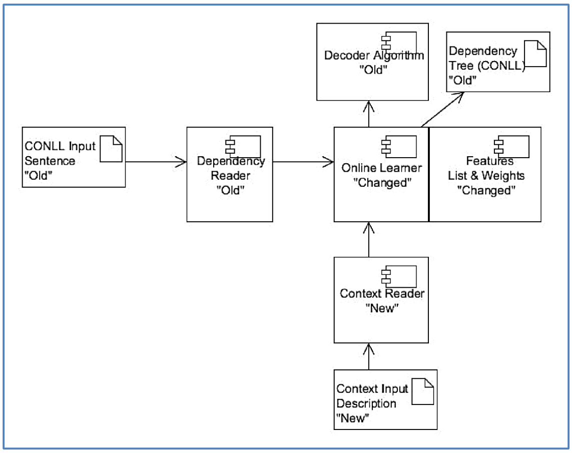
Decoder Algorithm "Old"
Dependency Tree (CONLL) "Old"
CONLL Input Sentence "Old"
Dependency Reader "Old"
Online Learner "Changed"
Features List & Weights "Changed"
Context Reader "New"
Context Input Description "New"

Fig. 5. The architecture of context-integrating dependency parser

- Newly introduced components (New).

The Online Learner is an existing component in RBG parser. It uses the Passive-Aggressive algorithm. We modify it to consider the additional features. Therefore, the features list and corresponding weights in RBG parser are also modified for the same purpose. On the other hand, the RBG component "Decoder Algorithm" is left unchanged. It is responsible for the minimum spanning tree decoder and implements different algorithms.

5 Experiments

5.1 Dataset Preparation

In our experiments, we developed two small corpora:

1. An extended version of Baumgärtner's dataset (Baumgärtner 2013). The original dataset condensed 24 images and 96 sentences describing these images. All sentences follow the same structure: Subject, verb, and object with adverbial modifiers. We translated this dataset from German to English, and extended it to 500 sentences that are equally distributed into the following groups:
   - The original dataset. Ex: "The Princess washes obviously the pirate."
   - A group has subject, object, and descriptions for both of them. Ex: "The Princess with long hair washes obviously the pirate with a woody leg."
   - A group has a descriptive subject, object, and a description of the action's instrument. Ex: "The Princess with long hair washes obviously the pirate with a brush."
   - A group has a subject with a description, an object with a description, and a description of the action. Ex: "The Princess with long hair washes obviously the pirate with a woody leg with a brush."
   - Sentences with subject and object in a passive form. Ex: "The pirate is washed by the Princess."
2. Part of the ILLIONS image corpus (Young et al. 2014) with 35 images and three corresponding sentences for each of them. This dataset is created through crowdsourcing to describe the content of the pictures. Therefore, the structure of the sentences varies in contrast to the first dataset.

We developed context description for each sentence in both datasets. Baumgärtner's dataset had already initial context description to build up on, but with the ILLIONS dataset we had to start from scratch. Four thematic roles have been used: agent, theme, instrument, and owner. To prepare the training data, we used the online demo of "Noah's ARK" Turbo parser (Thomson et al. 2014). The output (CONLL format) was verified manually against "Stanford dependency manual" (de Marneffe and Manning 2015).

5.2 Experiments Results

We present a set of experiments carried out to verify the effectiveness of context integration and its impact on the dependency parsing quality. We use three metrics:

- Unlabeled attached score (UAS): the percentage of correct lexical-parent attachments in the testing data.
- Labeled Attached Score (LAS): the percentage of correct labeled lexical-parent attachments in the testing data.
- Complete Attached Score (CAS): the percentage of complete sentences that have been correctly analyzed.

Experiment 1. Here we use the first dataset mentioned above. The training uses 440 sentences, and the testing data has 60 sentences. We implement two different degrees for the influence of the context features:

- "Strong Context" condition: We treated the context features during learning with an extra confidence (3 times the weight of a normal linguistic feature).
- "Normal Context" condition: The features added from the context have the normal influence on the final decision like the linguistic features.

As presented in Fig. 6, integrating the context information into the learning process slightly improves the UAS and LAS scores. This (relatively small) improvement of the attachment across all the words in the testing data, has, however, a quite big impact on the CAS score. That illustrates the benefit of using multimodal information as input for the graph based dependency parser. It helps to properly disambiguate more attachments which improves the overall CAS score by 18 %.

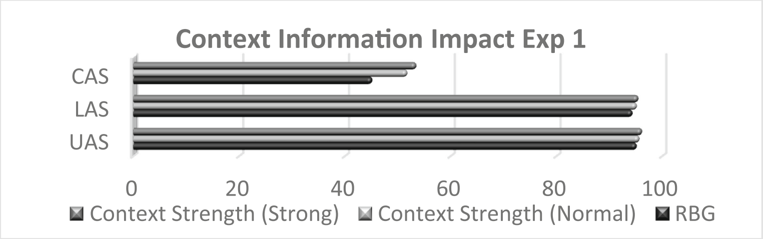

Fig. 6. Context information impact (Experiment 1)

Experiment 2. In this experiment, we used the second data set mentioned in the dataset section. Due to the small size of this dataset, and the online learning process of RBG using Passive-Aggressive (PA) algorithm, we use in this experiment the whole dataset as training data, as well as testing data. In the PA algorithm, the feature weights are updated according to the currently processed input sentence neglecting the influence of this update on the previously handled cases. Therefore, we train and test on the same dataset to check the effect of the context information. The context-integrating parser improves the overall CAS score by 8 % (Fig. 7).

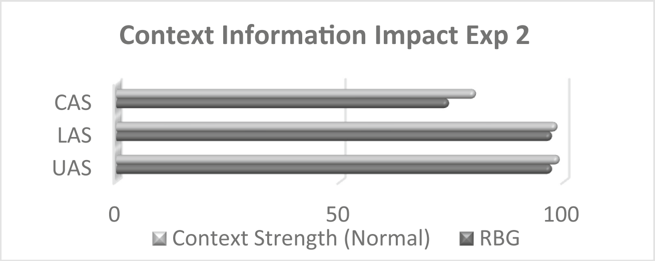

Fig. 7. Context Information Impact (Experiment 2)

Experiment 3. We trained the parser with the first dataset while testing with the second one. As shown in Fig. 8, there is no noticeable improvement in this case. We find that due to the completely different sentence structures and lexicons between the training and testing data, the context information didn't remarkably help towards better disambiguation in this case.

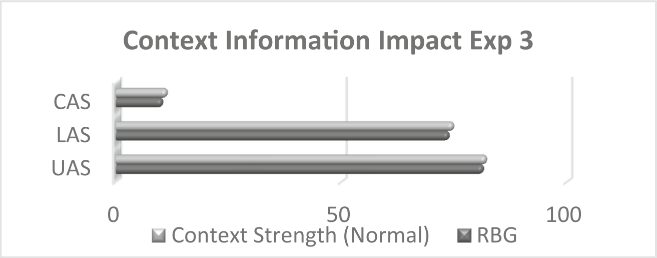

Fig. 8. Context Information Impact (Experiment 3)

6 Conclusion

In this paper, we present a context-integrating dependency parser by providing a new version of a graph-based dependency parser (namely RBG) accepting additional kind of features. These features present the visual context information of the sentence in a form of thematic roles. The experiments show an improvement of parsing quality between 8 % and 18 % in different experiments. We have shown the effectiveness of the idea on small datasets scale.

7 Future Work

While this paper is a first step in our context-integration parsing roadmap, we have identified some limitations in this work that should be tackled in future work. Currently, the system is not able to deal with a possible mismatch between the lexicons for the linguistic input and the context descriptions. So far we require them to be identical which is not a realistic assumption for richer linguistic stimuli.

In this research, we improved parsing quality using the cognitive influence of visual context information. In the future, we will work on enriching the context representations based on the linguistic input. We also need to apply the approach to a larger dataset. Additionally, we will study the behavior of the system in a situation where the context relationships contradict the linguistic content.

References

Baumgärtner, C.: On-line cross-modal context integration for natural language parsing. Ph.D. thesis Universität Hamburg, Hamburg (2013)

Bohnet, B.: Very high accuracy and fast dependency parsing is not a contradiction. In: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pp. 89–97. Beijing (2010)

Chu, Y.J., Liu, T.H.: On the shortest arborescence of a directed graph. Sci. Sinica **14**, 1396–1400 (1965)

Crammer, K., Dekel, O., Shalev-Shwartz, S., Singer, Y.: Online passive-aggressive algorithms. J. Mach. Learn. Res. **7**, 551–585 (2006)

de Marneffe, M.-C., Manning, C.D.: Stanford typed dependencies manual, 1 April 2015. Retrieved from http://nlp.stanford.edu/pubs/dependencies-coling08.pdf

Knoeferle, P.: The role of visual scenes in spoken language comprehension: evidence from eye-tracking. Saarlandes: Ph.D. thesis Universität des Saarlandes (2005)

Lei, T., Xin, Y., Zhang, Y., Barzilay, R., Jaakkola, T.: Low-rank tensors for scoring dependency structures. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL (2014)

Lei, T., Zhang, Y., Barzilay, R., Jaakkola, T.: RBGParser, 15 October 2015. Retrieved from github: https://github.com/taolei87/RBGParser

McCrae, P.: A model for the cross-modal influence of visual context upon language processing. In: The International Conference Recent Advances in Natural Language Processing, pp. 230–235 (2009)

Nivre, J.: Incrementality in deterministic dependency parsing. In: Proceeding Increment Parsing 2004 Proceedings of the Workshop on Incremental Parsing, pp. 50–57. Stroudsburg, PA, USA (2004)

Nivre, J., Hall, J., Nilsson, J.: Memory-based dependency parsing. In: Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL), Boston (2004)

Thomson, S., Kong, L., Martins, A.: ARK Syntactic & Semantic Parsing Demo, 1 December 2014. Retrieved from http://demo.ark.cs.cmu.edu/parse

Young, P., Lai, A., Hodosh, M., Hockenmaier, J.: From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions. Trans. Assoc. Comput. Linguist. **2**, 67–78 (2014)

Zhang, Y., Lei, T., Barzilay, R., Jaakkola, T., Globerson, A.: Steps to excellence: simple inference with refined scoring of dependency trees. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, pp. 197–207 (2014a)

Zhang, Y., Lei, T., Barzilay, R., Jaakkola, T.: Greedis goodif randomized: new inference for dependency parsing. In: Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1013–1024. Doha, Qatar: Association for Computational Linguistics (2014b)