Multimodal Interactive Parsing

José-Miguel Benedí[1], Joan-Andreu Sánchez[1], Luis A. Leiva[1], Ricardo Sánchez-Sáez[1], and Mauricio Maca[2]

[1] Instituto Tecnológico de Informática, Universitat Politècnica de València, Spain.
{jbenedi,jandreu,luileito,rsanchez}@dsic.upv.es
[2] Departamento de Matemáticas, Universidad del Cauca, Colombia.
mmaca@unicauca.edu.co

Abstract. Probabilistic parsing is a fundamental problem in Computational Linguistics, whose goal is obtaining a syntactic structure associated to a sentence according to a probabilistic grammatical model. Recently, an interactive framework for probabilistic parsing has been introduced, in which the user and the system cooperate to generate error-free parse trees. In an early prototype developed according to this interactive parsing technology, user feedback was provided by means of mouse actions and keyboard strokes. Here we augment the interaction style with support for (non-deterministic) natural handwritten recognition, and provide confidence measures as a visual aid to ease the correction process. Handwriting input seems to be a modality specially suitable for parsing, since the vocabulary size involved in the recognition of syntactic labels is fairly limited and thus intuitively errors should be small. However, errors may increase as handwriting quality (i.e., calligraphy) degrades. To solve this problem, we introduce a late fusion approach that leverages both on-line and off-line information, corresponding to pen strokes and contextual information from the parse trees. We demonstrate that late fusion can effectively help to disambiguate user intention and improve system accuracy.

Keywords: syntactic parsing, interactive pattern recognition, multimodal interaction, late fusion

1 Introduction

Parsing, also known as grammatical or syntactic analysis, is considered a fundamental problem in Computational Linguistics [4]. Parsing consists of analyzing a sentence to determine its grammatical structure with respect to a given formal grammar. Such structure is given in the form of a *parse tree*, where noun phrases and predicate are detected together with the relations between their components, such as nouns, verbs, prepositions, etc. Parsing has been also applied to other research fields aside from Natural Language Processing (NLP), since the concept of "sentence" can be extended to other objects, e.g., mathematical expressions.

Having perfectly annotated parse trees is a critical task, since error-free trees allow to train and improve statistical models not only for probabilistic parsing but also for other NLP problems, such as machine translation, question

answering, or discourse analysis. However, until very recently the grammatical construction of such trees has been done manually, involving thus a really laborious task. When using automatic parsers as a baseline for building perfect parse trees, the traditional post-editing approach leads to the well-known two-step error correcting process, in which the system first generates an automatic output and then the user verifies or amends it [1,5].

The aforementioned post-editing paradigm is rather inefficient and uncomfortable for the human annotator. To this end, previous interactive annotation tools have been published elsewhere [2], including an Interactive Parsing (IP) framework to ease annotation tasks [8,9]. In such interactive (and iterative and predictive) framework the user and the system cooperate in order to decrease both human annotation effort and system recognition error. Other researchers have successfully deployed approaches in a similar vein in fields like Statistical Machine Translation [6] and Handwriting Transcription [7].

With the intention of making more comfortable the tree annotation process, instead of using deterministic feedback like mouse actions and keyboard strokes, we introduce on-line handwriting as a new input modality to allow the user to enter corrections. Handwriting input seems to be a modality specially suitable for parsing, since the vocabulary size involved in the recognition of syntactic labels is fairly limited and thus intuitively the number of recognition errors should be small. However, it is important to note that non-deterministic feedback decoding will never be error-free, as the system needs to accommodate different calligraphies, writing styles, and so on. This is true even for a highly skilled human translator. Therefore, recognition errors may increase as handwriting quality (i.e., calligraphy) degrades. Consequently, the design of a good (non-deterministic and multimodal) IP system ultimately should lead to achieving the best decoding accuracy by exploiting as much as possible the contextual information provided by the IP framework.

Following a similar approach to Romero et al. [7], we firstly propose a new formal framework, which is an extension of Sánchez-Sáez et al. [9], and then we introduce a late fusion approach to leverage on-line and off-line information, corresponding to (handwritten) pen strokes and contextual information. We name this approach *Multimodal Interactive Parsing* (MIP). As the new feedback modality to amend parsing errors consists of on-line handwriting, we rely on handwritten text recognition techniques. In addition, multimodal interaction is approached in such a way that both the main and the feedback data streams work together to optimize overall performance.

A series of synthetic experiments were performed to corroborate the feasibility of our approach. We demonstrate that late fusion can effectively help to disambiguate user intention and improve system accuracy. We therefore illustrate that handwriting input can be used as an input modality to annotate parse trees, and that recognition accuracy can be improved if the context is considered, specially when handwriting quality degrades. Together with our results, this work advances the state of the art on parsing, concretely in the interactive pattern recognition domain.

2 IP Overview

Given a sentence x and a Probabilistic Context Free Grammar (PCFG) $\mathcal{G}$, the goal of parsing consists of obtaining the parse tree t that best represents the relations between the structures of the sentence x according to $\mathcal{G}$. The probabilistic parsing can be formulated as:

$$\widehat{t} = \arg\max_{t \in \mathcal{T}} p_{\mathcal{G}}(t|x) \tag{1}$$

where $p_{\mathcal{G}}(t|x)$ is the probability of the parse tree t given the input string x using $\mathcal{G}$, and $\mathcal{T}$ is the set of all possible parse trees for x. In probabilistic parsing, a parse tree t that is associated to a string $x = x_1 \dots x_n$ can be decomposed into subtrees (or constituents) t_{ij}^A. A constituent t_{ij}^A is defined by the label of the root node, which is a nonterminal symbol A (either a *syntactic label* or a *part-of-speech tag*), and its span ij (the starting and ending indexes which delimit the part of the input sentence encompassed by the constituent $x_i \dots x_j$). Thus $t = t_{1n}^S$, where S is the axiom of the grammar. If $\mathcal{G}$ is in Chomsky Normal Form (CNF), then the maximization in Eq. (1) can be solved using a dynamic programming CKY-style algorithm.

In IP, the user amends a particular constituent in every interaction for some parse tree t. More precisely, he points out a particular node of the tree and amends the node label and/or its span. The formal framework for probabilistic IP can thus be defined as:

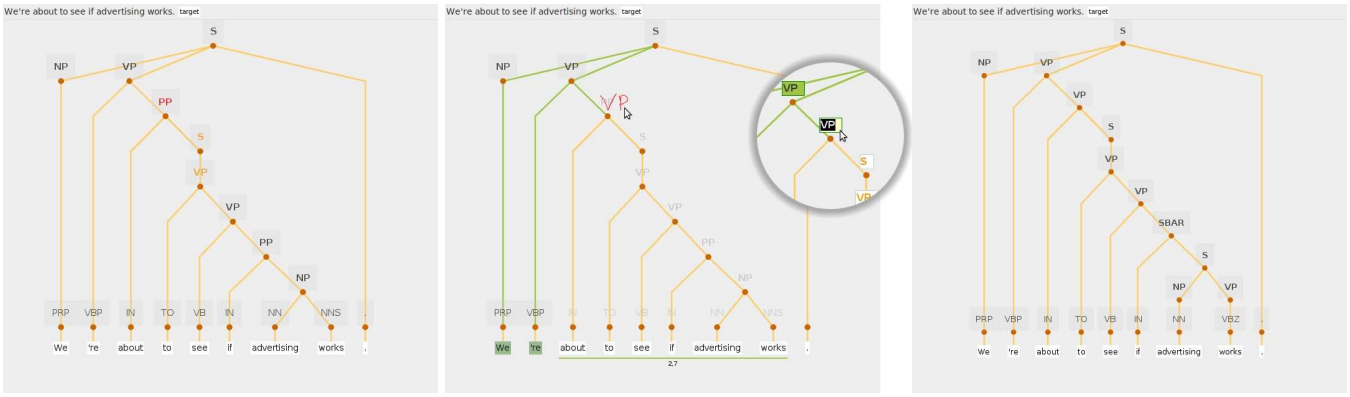
$$\widehat{t} = \arg\max_{t \in \mathcal{T}} p_{\mathcal{G}}(t|x, C, C_{ij}^A) \tag{2}$$

where $p_{\mathcal{G}}(t|x, C, C_{ij}^A)$ is the probability (according to $\mathcal{G}$) of a parse tree t given the input string x, the set of constituents previously validated by the user (history) C, and the constituent currently validated by the user (feedback) C_{ij}^A; and $\mathcal{T}$ is the set of all possible parse trees for x.

In such IP framework, the system suggests the (probably not error-free) best tree $\hat{t}$ in an automatic (unsupervised) way, and then reacts to the corrections introduced by the user over the constituents, proposing a new $\hat{t}'$ that takes into account the above-mentioned corrections; see Fig. 1 for a graphical example. This process can be summarized in the following 3-step protocol:

1. The system proposes a full parse tree t for the input sentence (see Fig.1a).
2. The user corrects the first erroneous tree constituent, implicitly validating a prefix tree t_p (see Fig.1b).
3. The system produces the most probable tree which is compatible with the validated prefix tree t_p and the user feedback (see Fig.1c).

Steps 2 and 3 are iterated until the tree is fully validated, i.e., it contains no parsing errors. This way, a perfect output is guaranteed, since the user is tightly embedded into "the recognition loop" and therefore is able to modify the decisions of the system. This IP protocol has proved to be very effective in reducing



(a) System hypothesis (b) User feedback (c) System response

Fig. 1: Correcting constituent labels by means of on-line handwritten strokes.

the annotation effort, specially when confidence measures are used to help the user in detecting constituent errors [10], as shown in Fig.1a.

Sánchez-Sáez et al. [8] developed a web-based system[3] that implemented the previously described IP protocol. They built a lightweight client using standard web technologies that communicated with a CYK-Viterbi parsing engine via asynchronous HTTP connections. Our work draws on the same architecture, since the hardware requirements are very low on the client side, as the parsing load is carried out remotely.

3 Multimodal IP

Recent advances in input devices, like tablets and touchscreens, have favored the development of multimodal interfaces. With the purpose of easing the tree annotation process, we introduce a new feedback multimodality in the IP framework. This new feedback consists of annotating erroneous constituent labels by means of on-line handwritten text. Therefore, Handwritten Text Recognition (HTR) techniques are needed. The idea is to correct constituent labels by using an HTR system, while constituent spans should be corrected by using the mouse, touch, or a similar pointer-based device. In addition, the user can also use the keyboard to change the constituent label, if desired (see Fig. 1b).

It is important to notice that a non-deterministic annotation feedback is introduced with this new input modality, and also that the user may commit annotation errors. Notice also that the HTR process is a classification problem, where each constituent tag can be considered a class label. If we tackle this problem as an isolated handwritten word classification problem, then the posterior

[3] `http://cat.iti.upv.es/ipp/`

probability can be used for classification:

$$\widehat{D} = \arg\max_{D} \; p(f|M_D)\; p(D) \tag{3}$$

where f is the input signal (handwritten feedback), M_D is the model associated to class D, and $p(D)$ is the a-priori probability of class D.

Note that if Eq. (3) is used alone, then no contextual information can be considered for classification. At this point, we will make use of the information provided by the user interaction to help decoding non-deterministic feedback signals. Therefore, D becomes a *hidden variable* that represents the decoding of the input handwritten signal f associated to the syntactic label of the constituent C_{ij}^D modified by the user in the current interaction. From Eq. (2), marginalizing over D (in fact over C_{ij}^D), and approximating the sum with the value of the mode; applying basic probability rules; and ignoring terms that do not depend on the optimization variables (t and D), we can solve Eq. (2) as follows:

$$\begin{aligned}\widehat{t} &\approx \arg\max_{t\in\mathcal{T}} \max_{D} \; p_{\mathcal{G}}(t, C_{ij}^D|x, C, f) \\ &\approx \arg\max_{t\in\mathcal{T}} \max_{D} \; p_{\mathcal{G}}(t|C_{ij}^D, x, C, f)\; p_{\mathcal{G}}(f|C, C_{ij}^D, x)\; p_{\mathcal{G}}(C_{ij}^D|C, x)\end{aligned} \tag{4}$$

Note that the classification problem can be either decoupled from the tree annotation process or fully embedded in the tree annotation process, the last two terms of Eq. (4) are now needed to deal with the non-deterministic feedback, yielding:

$$\widehat{D} \approx \arg\max_{D} \; p_{\mathcal{G}}(f|C_{ij}^D)\; p_{\mathcal{G}}(C_{ij}^D|C) = \arg\max_{D} \; p(f|M_D)\; p_{\mathcal{G}}(C_{ij}^D|C) \tag{5}$$

where $p(f|M_D)$ is a feedback likelihood model for recognizing f, and $p_{\mathcal{G}}(C_{ij}^D|C)$ is a history-conditioned decoding feedback prior probability. This prior probability is similar to the second term of Eq. (3). In Eq. (5), C_{ij}^D shows that the class label D must be compatible with previously validated constituents C and must account for the span ij of the input string x according to the grammar $\mathcal{G}$. Since we are considering PCFG as parsing models, this means that the search can be carried out efficiently by taking into account only those Ds that satisfy the restrictions imposed by constituents previously validated by the user. Thus, the classification problem is carried out only with the labels that account for the span ij in the analysis table. Notice that the first term of Eq. (5) corresponds to the posterior probability of the constituent and that can be efficiently computed [10]. Taking also into account that both terms in Eq. (5) need to be efficiently combined, we approach the expression as follows:

$$\widehat{D} = \arg\max_{D} \; p(f|M_D)^{\alpha}\; p_{\mathcal{G}}(C_{ij}^D|C)^{1-\alpha} \tag{6}$$

where $\alpha \in [0, 1]$ is a fusion percentage, in such a way that when $\alpha = 0$ no HTR is considered (just contextual information from the parse tree) and vice versa for $\alpha = 1$.

4 Experiments

We evaluate to what extent on-line handwriting can be a useful interaction modality for IP parsing, and whether late fusion can contribute to disambiguate user intent. We emphasize on the fact that non-deterministic feedback decoding will never be error-free. In other words, system performance has to be sacrificed to some extent for the sake of a potential improvement in ergonomics and/or usage experience. To this end, we acknowledge that assessing the performance of an MIP system from a user interaction point of view should ultimately require human supervision and judgment. However, the cost of a formal field study of this kind of systems is exceedingly high, since it typically involves expensive work by a panel of linguistic experts. Therefore, the solution we adopted is based on the tried-and-true pattern recognition assessment paradigm based on labeled corpora. This way, we simulated the user interaction in the same way other authors have made before [6,7]. We assumed for simplicity that the cost of correcting an on-line decoding error is equally similar to the one provided by another user interaction.

Learning parsing models. The sentences for training the parsing models were taken from the UPenn Treebank [5]. Sections 02–21 were used for training and section 23 for testing. The open source Natural Language Toolkit (NLTK)[4] was used to obtain a right-factored binary grammar from the training set. After parsing the test set, 2,723 erroneous syntactic labels were detected. In our experiments, an erroneous syntactic label must be handwritten to be corrected.

Learning HTR models. We used Hidden Markov Models (HMM) for recognizing on-line handwritten labels. Character-level HMMs for the HTR recognizer were trained with 17 different writers from the UNIPEN dataset [3]. Three writers not included in the training set were used to build syntactic labels for testing. Sixty syntactic labels were composed by concatenating single characters. We used the HTK toolkit[5] for HMM training and recognition. For on-line handwriting recognition, 6 features, described in[7], were used.

Experimental framework. We study the multimodal fusion of pen strokes and contextual information, stated in Eq. (6). To this end, we simulated handwriting degradation by randomly rotating the handwritten samples of each syntactic label. Then, we carried out a classification task using the 2,723 erroneous syntactic labels with 3 writers. Hence, 8,169 samples were eventually used for testing. The results are shown in Table 1.

As previously pointed out, when $\alpha = 1.0$ only the HTR classifier was used for classifying, and when $\alpha = 0.0$ only the contextual information was used for classifying. We have explored α values between 1.0 and 0.4. Row 0 corresponds to the ideal situation in which there is no distortion. In this row the best results were obtained for $\alpha = 1.0$ and $\alpha = 0.9$ in optimal conditions, i.e., when labels are not distorted; which means that the classification could be carried out considering

[4] `http://www.nltk.org`
[5] `http://htk.eng.cam.ac.uk`

Table 1: Classification error rate when using contextual information. θ represents the degree of distortion (rotation angle). Other columns correspond to different values of the fusion value α. In each row, the best result is marked in bold.

	α						
θ	1.0	0.9	0.8	0.7	0.6	0.5	0.4
0	**3.6**	**3.6**	4.2	5.4	7.3	9.6	13.2
±10	4.3	**3.7**	4.2	5.4	7.4	10.0	13.8
±20	**4.0**	**4.0**	4.6	5.7	7.5	9.6	14.6
±30	16.9	9.6	**8.8**	9.7	11.4	13.6	17.1
±40	16.2	8.5	**7.7**	8.9	10.7	12.9	16.5
±50	23.9	16.9	14.5	**13.8**	15.4	17.7	20.8
±60	33.4	24.7	18.3	**15.6**	16.7	18.8	22.7
±70	51.7	38.4	26.1	22.2	**21.2**	22.4	24.1
±80	45.4	39.8	32.7	30.5	28.8	**28.6**	29.0
±90	58.2	49.3	38.3	34.1	**33.1**	**33.1**	33.2

the HTR classifier alone, discarding thus contextual information. However, notice that when the distortion θ increased, the best α decreased, which means that the context is really important in reducing the classification error rate.

These results therefore suggest that including multimodality in an IP framework has positive benefits. First, we found an interesting balance between user effort and system accuracy. This reinforced our initial guesses depicted in Section 1. Second, our approach allows for a comfortable way of interacting with a parsing application, since pen-based devices (e.g., styli, wands, or touchscreens) are common-place today and users are thus accustomed to using these input devices. Third, the system can decode the submitted user feedback with really good precision, so it can leverage this information to improve its output whenever the user interacts with the system. Finally, MIP allows us to advance the state of the art on parsing, specifically in the interactive pattern recognition domain.

5 Conclusion and Future Work

Our results have shown that MIP can ease the annotating task for the human, at the expense of introducing an non-deterministic feedback signal for the system. In our experiments, such a non-deterministic feedback comes from pen-based on-line handwriting strokes, but other input modalities may fit in our framework. This poses new challenges for researching novel ways to improve recognition accuracy. As demonstrated, when the (on-line) user feedback is somewhat degraded, incorporating contextual (off-line) information allows to significantly obtain better classification rates. We have found it beneficial in the parsing domain, though we believe this notion should enhance other interactive pattern recognition systems.

Future work includes assessing the MIP framework with real users through a formal evaluation. Previous informal tests with a non-expert audience have suggested that ours is a realistic approach. However, we would need to recruit experienced linguistics to draw strong conclusions about the feasibility of the system in the long term.

Acknowledgments

This research has received funding from the EC's 7th Framework Programme (FP7/2007-13) under grant agreement No.287576-CasMaCat; from the Spanish MEC under the STraDA project (TIN2012-37475-C02-01) and the MITTRAL project (TIN2009-14633-C03-01); from the GV under the Prometeo project; and from the *Universidad del Cauca* (Colombia).

References

1. S. Afonso, E. Bick, R. Haber, and D. Santos. Floresta sintá(c)tica: a treebank for portuguese. In *Proc. LREC*, pp. 1698–1703, 2002.
2. T. Brants and O. Plaehn. Interactive corpus annotation. In *Proc. LREC*, 2000.
3. I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet. UNIPEN project of on-line data exchange and recognizer benchmarks. In *Proc. ICPR*, pp. 29–33, 1994.
4. M. Lease, E. Charniak, M. Johnson, and D. McClosky. A look at parsing and its applications. In *Proc. AAAI*, pp. 1642–1645, 2006.
5. M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.
6. D. Ortiz, L. A. Leiva, V. Alabau, and F. Casacuberta. Interactive machine translation using a web-based architecture. In *Proc. IUI*, pp. 423–425, 2010.
7. V. Romero, L. A. Leiva, A. H. Toselli, and E. Vidal. Interactive multimodal transcription of text images using a web-based demo system. In *Proc. IUI*, pp. 477–478, 2009.
8. R. Sánchez-Sáez, L. A. Leiva, J. A. Sánchez, and J. M. Benedí. Interactive predictive parsing using a web-based architecture. In *Proc. NAACL-HLT*, pp. 37–40, 2010.
9. R. Sánchez-Sáez, J. A. Sánchez, and J. M. Benedí. Interactive predictive parsing. In *Proc. IWPT*, pp. 222–225, 2009.
10. R. Sánchez-Sáez, J. A. Sánchez, and J. M. Benedí. Confidence measures for error discrimination in an interactive predictive parsing framework. In *Proc. COLING*, pp. 1220–1228, 2010.