Structured Prediction for Speaker Identification in TV Series

Elena Knyazeva[12], *Guillaume Wisniewski*[12], *Hervé Bredin*[1], *François Yvon*[1]

[1] LIMSI – CNRS – Rue John Von Neumann, Orsay, France.
[2] Université Paris Sud – Orsay, France.

FirstName.LastName@limsi.fr

Abstract

Though radio and TV broadcast are highly structured documents, state-of-the-art speaker identification algorithms do not take advantage of this information to improve prediction performance: speech turns are usually identified independently from each other, using unstructured multi-class classification approaches. In this work, we propose to address speaker identification as a sequence labeling task and use two *structured prediction* techniques to account for the inherent temporal structure of interactions between speakers: the first one relies on Conditional Random Field and can take into account local relations between two consecutive speech turns; the second one, based on the SEARN framework, sacrifices exact inference for the sake of the expressiveness of the model and is able to incorporate rich structure information during prediction. Experiments performed on *The Big Bang Theory* TV series show that structured prediction techniques outperform the standard unstructured approach.

Index Terms: speaker identification, speaker diarization, sequence labeling, structured prediction

1. Introduction

Thanks to NIST Rich Transcription evaluation series started in 2002 [1] and to more recent initiatives such as ESTER [2], ETAPE and REPERE [3] evaluation campaigns, a significant amount of research has focused on speaker diarization and identification in conversational phone calls, radio and TV broadcast news or meetings – all with the same objective: answering the *who speaks when?* question.

Compiled in the recent review by *Anguera et al.* [4] and illustrated in the upper part of Figure 1, most (if not all) speaker diarization approaches share the same processing pipeline: speech activity detection, followed by two (sometimes merged into a single one) modules for temporal segmentation into homogeneous segments and their unsupervised clustering according to the identity of the speaker. Speaker identification is then addressed as a supervised multi-class classification problem.

Both clustering and classification modules consider their inputs (speech turns for the former, speaker clusters for the latter) as unordered, unstructured and independent from each other. For instance, hierarchical agglomerative clustering approaches (*e.g.* BIC clustering [5]) take a *bag of speech turns* as initial input and iteratively merge the two most similar clusters, update the similarity matrix and start again until a stopping criterion is met, completely disregarding the actual speech turns order.

1.1. Structure

As shown by the example in Figure 4, conversations within a typical TV series episode are highly structured: as episodes are usually divided into scenes involving only a subset of characters, speech turns of a given character are not uniformly distributed over the duration of the episode. Hence, knowing that *Penny* is speaking at a particular time t tells us a lot about the probability that she is also going to speak a few seconds later.

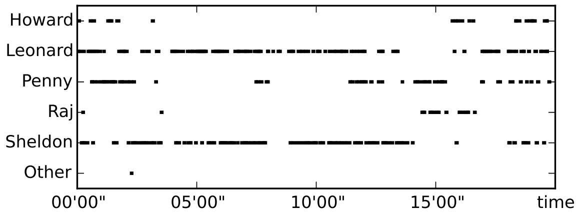

Figure 2: *Speaker identification for episode 2 of season 1 of* The Big Bang Theory.

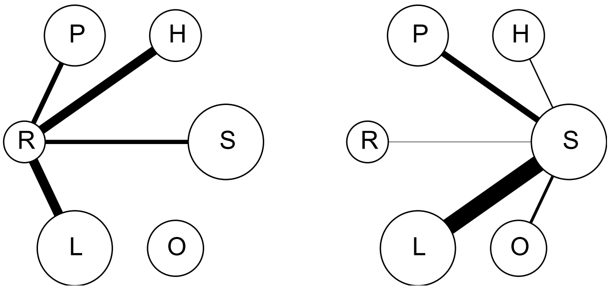

Figure 3: *Raj's (left) and Sheldon's (right) interactions with other characters*

Similarly, Figure 3 provides additional evidence – if need be – of the existing structure of the sequence of speakers. It depicts the amount of interaction (i.e. consecutive speech turns) between the main characters of the TV series. While *Raj* is the less talkative character, knowing that he is speaking at a particular time greatly increases the chance for the next speaker to be *Howard* or *Leonard*.

A very few attempts have been made to take advantage of prior knowledge about the structure in the speaker identification process. For instance, local clustering (*a.k.a.* linear clustering) is usually applied as a pre-processing step to merge adjacent speech turns of a same speaker [4]. Though its objective is to reduce the size of the clustering problem and obtain more discriminant similarity matrices, this kind of approaches is also motivated by the non-uniform distribution of speech turns highlighted in Figure 4. In the same vein, recent variational Bayes approaches such as the “sticky HDP-HMM” proposed by *Fox et al.* [6] jointly constrain minimum speech turn duration (the

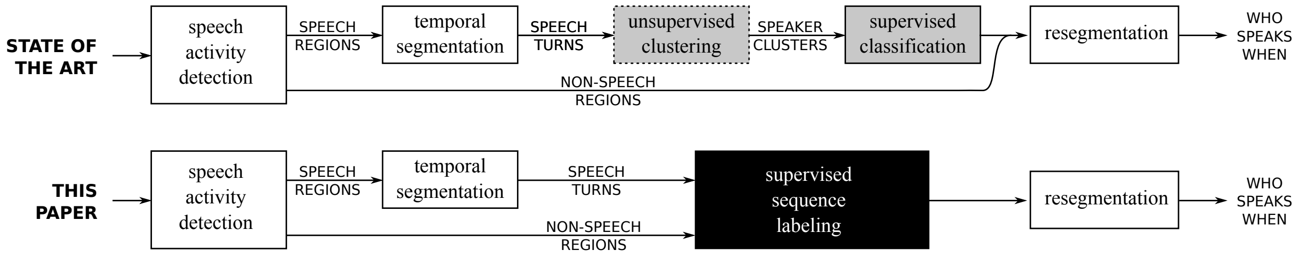

Figure 1: Speaker identification pipeline – structured (black) vs. unstructured (gray) prediction.

"sticky" part) and models the inter-speaker dynamics via transition probabilities (the "HMM" part). Attempts to step into the structured prediction bandwagon have been proposed recently, mostly through the use of Conditional Random Fields (CRF). For instance, [7] proposes a variational CRF to address the speaker tracking and detection tasks. However, they rely on a linear CRF that only accounts for dependencies between adjacent speech turns.

1.2. Sequence Labeling & Structured Prediction

Overall, existing approaches only take simple and local structure into account and completely overlook long-term (*did sheldon speak during the last 30 seconds?*) and higher-order structure (*is raj discussing with both howard and penny?*). In this paper, we propose to address speaker identification as a sequence labeling task and to use structured prediction techniques to account for the inherent temporal structure of interactions between speakers.

Sequence labeling consists in assigning a label to every element in a sequence of observations. Let $\mathbf{x} = (x_i)_{i=1}^n$ be a sequence of n observations and y_i be the label of the i^{th} element. The sequence of labels, denoted by $\mathbf{y} = (y_i)_{i=1}^n$, generally presents multiple dependencies. Because of the relations between the y_i, some combinations of labels will not be possible and some combinations will be more frequent. More formally, if Λ denotes the set of all possible labels (the domain of the y_i), and $\mathcal{Y}$ the domain of the *macro-label* $\mathbf{y}$, then the actual range of possible labelings $\mathcal{Y}$ is only a (tiny) subset of Λ^n. *Structured prediction* aims at developing models able to detect and exploit these dependencies so as to improve prediction performance.

1.3. Outline

The main contribution of this paper is to show that speaker identification can benefit from structured prediction, whose general theory is introduced in Section 2, along with the SEARN algorithm which provides an efficient way to take long-term structure into account. The experimental protocol and implementation details are presented in Section 3. Results are summarized and discussed in Section 4. Section 5 concludes the paper.

2. Structured Prediction

We will now present two frameworks that have been proposed for sequence labeling.

2.1. Generalizing Multi-Class Classification

Many machine learning models like CRF [8] or SVM$^{\text{struct}}$ [9] have been proposed to take advantage of the information conveyed by relations between the labels. They all adopt the same approach which can be seen as a generalization of multi-class classification: given a $\mathbf{w}$-parametrized scoring function F that measures the compatibility between a sequence of observations $\mathbf{x}$ and a sequence of labels $\mathbf{y}$, sequence labeling amounts at finding the most compatible output among all possible labelings in Λ^n:

$$\mathbf{y}^* = \operatorname*{arg\,max}_{\mathbf{y}\in\Lambda^n} F(\mathbf{x}, \mathbf{y}; \mathbf{w}) \tag{1}$$

The $\arg\max$ operator denotes the search in the space of all possible outputs that takes place during inference. Existing methods differ by the way they estimate the parameter vector: in a CRF, $\mathbf{w}$ is chosen by maximizing the conditional likelihood of $\mathbf{y}$ given $\mathbf{x}$, in SVM$^{\text{struct}}$ by maximizing a margin criterion.

The scoring function used to discriminate the expected solution among all possible solutions is generally defined as a dot product between a parameter vector $\mathbf{w}$ and a feature function $\phi(\mathbf{x}, \mathbf{y})$. ϕ can account for any relevant relation between labels, and between input data and labels. In the case of speaker identification, features can be the time elapsed since the last speech turn of a particular speaker, the list of the m speakers who last spoke, or even the number of time a character spoke up since the beginning. All these features can be directly inferred from the sequence of labels.

In their general formulation, structured prediction methods can use arbitrary relations. However, in practice, solving the $\arg\max$ in Equation (1) is a combinatorial optimization problem (the number of label sequences grows exponentially with the sequence length) for which exact solutions can be found efficiently only when very specific feature functions are considered. More precisely, if the scoring function is assumed to be decomposable (i.e. it can be expressed as a product of *local* scoring functions), the $\arg\max$ can be solved efficiently thanks to the Viterbi algorithm [10]. This is why existing sequence labeling methods usually rely on the following decomposition:

$$F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \sum_{i=1}^{n} f(y_{i-1}, y_i, \mathbf{x}; \mathbf{w}) \tag{2}$$

where f is a local scoring function that only takes local relations between consecutive labels into account.

2.2. SEARN (Search-Learn)

SEARN [11] is another structured prediction method that proposes an alternative framework for sequence labeling. It relies on the intuition that solving the $\arg\max$ problem can be performed as a series of local decisions. In this approach, the output label sequence is built by choosing, at each position i, the label y_i of the current observation x_i using features that describe the full observation sequence $\mathbf{x}$ and the past decisions from y_1 to y_{i-1}. This approach *reduces* structured prediction to a sequence of multiclass classification problems. Inference then consists in predicting labels one after the other – using a linear model, for instance:

$$y_i^* = g(\mathbf{x}, i, h_i) = \arg\max_{y \in \Lambda} \langle \mathbf{w} | \phi(\mathbf{x}, i, y, h_i) \rangle \qquad (3)$$

where y_i^* is the predicted label at position i, $\mathbf{w}$ the parameter vector, $h_i = y_1^*, ..., y_{i-1}^*$ the history of past decisions and ϕ a joint feature map that can incorporate any features related to the sequence of observations $\mathbf{x}$ and the labels history h_i.

Sketched in Algorithm 1[1], the training procedure performs inference on each input sequence (lines #5 and #6) while keeping track of both actual y_i^* and expected decisions y_i. The parameter vector $\mathbf{w}$ can then be estimated using a standard multiclass learning algorithm (line #15). The main challenge faced by SEARN is that the previous predictions influence the distribution of examples S upon which the learn will be tested, violating the crucial assumption that examples are i.i.d.. To avoid this problem, SEARN relies on an iterative procedure that progressively takes a larger amount of past decisions into consideration (lines #8 to #11), thus ensuring that the distribution of examples will become more and more similar at train and test times.

Algorithm 1: SEARN-inspired learning algorithm

```
   Input : labeled sequences $\mathcal{T} = (\mathbf{x}_i, \mathbf{y}_i)_{i=1}^{l}$, the number
           of iterations $N$, $\beta = 0$, $\beta_{\text{step}} \in [0, 1]$, the
           uniform random number generator $\text{rand}(0, 1)$
1  for $t \in [\![1, N]\!]$ do
2      $S \leftarrow \emptyset$ ;   ▷ Set of collected examples
3      for $\mathbf{x}, \mathbf{y} \in \mathcal{T}$ do
4          $h \leftarrow \emptyset$;
5          for $i \in [\![0, n]\!]$ do
6              $y_i^* = \arg\max_{y \in \Lambda} \langle \mathbf{w} | \phi(\mathbf{x}, i, y, h) \rangle$;
7              $S \leftarrow S \cup \{\phi(\mathbf{x}, i, h), y_i\}$;
8              if $\text{rand}(0, 1) > \beta$ then
9                  $h \leftarrow h \cup \{y_i\}$;
10             else
11                 $h \leftarrow h \cup \{y_i^*\}$;
12             end
13         end
14     end
       ▷ Train classifier
15     $\mathbf{w}^* = \arg\min_{\mathbf{w}} \mathbb{E}_{(x,y) \sim S} [\mathbb{1}\{y \neq g(x)\}]$ ;
16     $\beta \leftarrow \min(\beta + \beta_{\text{step}}, 1)$
17 end
```

While inference is exact in the CRF model, SEARN relies on a greedy search in the space of all allowed labelings Λ^n. Trading-off the global optimality of inference for the additional flexibility in the design of features and long range dependencies between labels has proved useful for many sequence labeling tasks in natural language processing [12, 13, 14]. However, it is hindered by error propagation. Past decisions are never questioned: an error will make future decisions more difficult.

To alleviate this problem, we propose an *easy-first sequence labeling strategy*. The classifier starts by labeling the observations it is the most confident about and makes the most difficult decisions at the end, minimizing their impact on subsequent decisions. Easy-first strategies can readily be implemented within the SEARN framework by generalizing the definition of decisions: the label sequence will be built by successively (a) choosing an 'open' position for which the label has not been predicted yet and (b) labeling it. Inference then amounts to solving:

$$i^*, y^* = \arg\max_{i, y \in \mathcal{O} \times \Lambda} \langle \mathbf{w} | \phi(\mathbf{x}, i, y, h_i) \rangle \qquad (4)$$

where $\mathcal{O}$ is the set of positions that must still be labeled. One complication arises during learning as easy-first oracles are not deterministic: there are as many oracle decisions in Equation 4, as open positions. This problem has already been described in natural language processing for dependency parsing [15] and PoS tagging [16]. The chosen solution is to randomly select one oracle decision (i^*, y^*) among all oracle positions, in order to avoid any bias while moving from the oracle to the real case decisions distribution.

3. Experiments

We now turn to the experimental part, where we show how structure can help identifying the speaker of a segment.

3.1. Corpus and protocol

Experiments are conducted on the first season of *The Big Bang Theory* TV series and results are reported using standard *Identification error rate* (IER):

$$\text{IER} = \frac{\text{miss} + \text{fa} + \text{confusion}}{\text{speech}} \qquad (5)$$

where speech is the total duration of *speech* according to the reference annotation, miss (respectively fa) is the total duration of segments incorrectly classified as *non-speech* (resp. *speech*) and confusion is the total duration of *speech* segments whose detected label is incorrect.

Manual annotations are available for episodes 1 to 6 with labels $\Lambda = \{$non-speech, howard, leonard, penny, raj, sheldon, other$\}$ where other englobes all characters but the main five [17, 18]. Due to the limited size of this test set ($\approx$ 2 hours), we use leave-one-episode-out cross-validation and report IER values averaged over the six rotations.

3.2. Baseline

Coarse annotations for the remaining episodes 7 to 17 are obtained from the publicly available TVD corpus [19] via the automatic alignment of subtitles and transcripts procedure described in [20]. They are used to train and tune the various modules of the baseline system described at the top of Figure 1. Table 1 lists the acoustic features used for each module. Speech activity detection relies on a 2-states HMM with 64 Gaussians per state (speech vs. non-speech). Temporal segmentation relies on standard BIC segmentation with full covariance and 500ms minimum duration constraint [5]. Speech turns are then passed directly to the supervised classification module implemented using the standard GMM/UBM open-set speaker identification approach [21], with one 64-component Gaussian mixture for each main character. Finally, HMM resegmentation is applied using 256 Gaussians per state.

3.3. Structured prediction

In our experiments, we consider 4 different sequence labeling methods. **Linear SVM** is a multi-class classifier based on the linear Support Vector Machine (SVM) implementation of the

[1] For the sake of efficiency, we used a stochastic approximation of SEARN. Refer to [11] for a more general description.

	ZCR	Energy	Δ	ΔΔ	MFCC	Δ	ΔΔ
SAD	✓				14	✓	
Seg.		✓			12		
SID			✓	✓	11	✓	✓

Table 1: *Acoustic features for speech activity detection (SAD), segmentation/resegmentation (Seg.) and speaker identification (SID) modules are computed every* 16*ms on* 32*ms windows.*

SCIKITLEARN library [22] – it serves as an additional unstructured prediction baseline. **Chain CRF** is a linear CRF that takes into account local structure through interactions between consecutive labels (Equation (2)) – we use the PYSTRUCT library [23] for that purpose. **Left-to-right SEARN** is the greedy method described by Equation (3) and **Easy-first SEARN** the free-order method described in Equation (4). Parameter β is set to 0.1 in all experiments, as proposed in [11]. All other hyperparameters are chosen by cross-validation.

The 14-dimensional observation vector x_i is made of the concatenation of $p(\text{speech}|s_i)$ and $p(\text{non-speech}|s_i)$ estimated by the SAD module, $p(\text{C}|s_i)$ for each of the 5 main characters C as estimated by the SID module, and 7 binary features (one per possible label) encoding the output of the baseline system. GMM/UBM log-likelihood ratios are calibrated into probabilities using isotonic regression [24].

The left-to-right SEARN approach relies on an additional set of binary features derived from the last $K = 4$ predictions. It includes simple history features such as *is prediction* y_{i-k} *non-speech? penny? sheldon?* and higher order features made of the conjunction of the previous pairs (and triplets) of predictions such as *are predictions* y_{i-k} $=$ *non-speech and* y_{i-k-1} $=$ *raj* ? We also consider a feature describing the number of times a label was selected in the last K predictions. The easy-first SEARN approach relies on a simpler set of binary features derived from the previous predictions within a ±2 neighborhood, such that both SEARN approaches rely on a context of $K = 4$ predictions.

4. Results

Results achieved by the different methods are presented in Table 2. To assess the impact of error propagation, we also report "oracle" experiments for the two SEARN models, in which the classifier relies on the true history to perform its predictions.

PREDICTION	IER
Baseline	30.5%
Linear SVM	29.8%
Chain CRF	29.3%
Left-to-right SEARN	30.0%
with oracle history	23.9%
Easy-first SEARN	29.1%
with oracle neighborhood	23.4%

Table 2: *Performance comparison.*

Results show that the three structured prediction methods (chain CRF, left-to-right and easy-first SEARN) outperform both unstructured approaches (baseline and multi-class SVM), highlighting the relevance of the episode structure for identifying speakers. It also appears that the best performance are obtained with chain CRF and easy-first SEARN based on a small ±1 or ±2 neighborhood. This questions the interest of considering non-local dependencies.

However, the results achieved when considering an oracle history show that non-local dependencies are, in fact, highly relevant. Models simply fall far short of taking advantage of them. This might be due (i) to error propagation or (ii) to a poor estimate of the weights given to each structure features. The best performance obtained by the easy-first strategy tends to show that efforts to reduce error propagation might indeed be beneficial. For instance, the proportion of other labels (i.e. secondary characters) in episode 4 is a lot larger than for the other episodes (30% vs. $\leq$ 10% on average). IER for this particular episode drops from 40.8% to 21.1% when relying on the oracle neighborhood. Error propagation has a huge impact in this case.

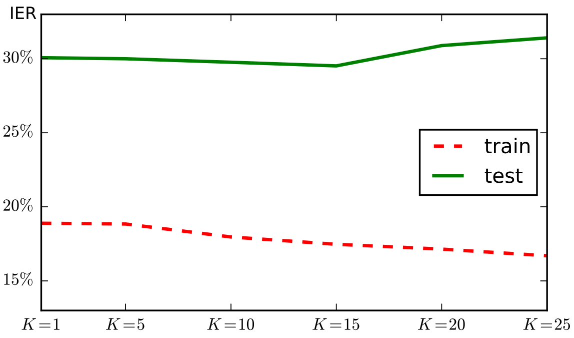

Figure 4: *Learning curve of the Left-to-Right* SEARN *model.*

Figure 4 represents the learning curve of the left-to-right SEARN model. It appears that the model quickly overfits when the history size K increases. While a large history helps identifying the speaker on the training set, structure information does not generalize well to other episodes.

5. Conclusion and Future Work

In this paper, we proposed to model the speaker identification problem as a sequence labeling task. Based on the observation that TV series episodes are highly structured documents, we show that structured prediction techniques lead to improved identification performance.

Oracle experiments suggest that error propagation is the main issue in the approaches based on SEARN. Ways to reduce the influence of errors will be investigated in the future (for instance by weighing each decision by a confidence score). We have also showed that structured prediction approaches are prone to overfitting – maybe due to the relatively high dimension of the structured features space. One way of reducing this dimension would be to delexicalize structure features (for instance by switching from *"is prediction* y_{i-i} *non-speech? penny? raj?"* to *"is prediction* y_{i-i} *similar to current label* y_i*?"*. Providing a better initial segmentation into speech turns would also definitely help to achieve better performance. As a matter of fact, we ran the same experiments on perfect speech turn segmentation and readily gained an absolute 10% improvement in terms of IER.

6. Acknowledgements

This work benefited from the support of the CAMOMILE project (CHISTERA) and the MetaDaTV project (ANR).

7. References

[1] J. S. Garofolo, J. G. Fiscus, A. F. Martin, D. S. Pallett, and M. A. Przybocki, "NIST rich transcription 2002 evaluation: A preview," in *Proceedings of the Third International Conference on Language Resources and Evaluation Spain*, Las Palmas, Canary Islands, May 2002.

[2] S. Galliano, G. Gravier, and L. Chaubard, "The ESTER 2 Evaluation Campaign for the Rich Transcription of French Radio Broadcasts," in *Proceedings of the Tenth Annual Conference of the International Speech Communication Association*, Brighton, United Kingdom, September 2009, pp. 2583–2586.

[3] J. Kahn, O. Galibert, L. Quintard, M. Carr, A. Giraudel, and P. Joly, "A Presentation of the REPERE Challenge," in *International Workshop on Content-Based Multimedia Indexing*, 2012, pp. 1–6.

[4] X. Anguera Miro, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker Diarization: A Review of Recent Research," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 2, pp. 356–370, February 2012.

[5] S. S. Chen and P. Gopalakrishnan, "Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion," in *DARPA Broadcast News Transcription and Understanding Workshop*, Virginia, USA, 1998.

[6] E. Fox, E. Sudderth, M. Jordan, and A. Willsky, "A sticky HDP-HMM with application to speaker diarization," *Annals of Applied Statistics*, vol. 5, no. 2A, pp. 1020–1056, 2011.

[7] M. Moattar and M. Homayounpour, "Variational conditional random fields for online speaker detection and tracking," *Speech Communication*, vol. 54, no. 6, pp. 763 – 780, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016763931200012X

[8] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289. [Online]. Available: http://dl.acm.org/citation.cfm?id=645530.655813

[9] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proceedings of the Twenty-first International Conference on Machine Learning*, ser. ICML '04. New York, NY, USA: ACM, 2004, pp. 104–. [Online]. Available: http://doi.acm.org/10.1145/1015330.1015341

[10] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theor.*, vol. 13, no. 2, pp. 260–269, Sep. 2006. [Online]. Available: http://dx.doi.org/10.1109/TIT.1967.1054010

[11] H. Daumé, Iii, J. Langford, and D. Marcu, "Search-based structured prediction," *Mach. Learn.*, vol. 75, no. 3, pp. 297–325, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1007/s10994-009-5106-x

[12] J. Kazama and K. Torisawa, "A new perceptron algorithm for sequence labeling with non-local features," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, ser. EMNLP-CoNLL'07, 2007, pp. 315–324.

[13] L. Ratinov and D. Roth, "Design challenges and misconceptions in named entity recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, ser. CoNLL '09. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009, pp. 147–155.

[14] Y. Tsuruoka, Y. Miyao, and J. Kazama, "Learning with lookahead: Can history-based models rival globally optimized models?" in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, ser. CoNLL'11. Portland, Oregon, USA: Association for Computational Linguistics, June 2011, pp. 238–246.

[15] Y. Goldberg and M. Elhadad, "An efficient algorithm for easy-first non-directional dependency parsing," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010, pp. 742–750. [Online]. Available: http://aclweb.org/anthology/N10-1115

[16] J. Ma, T. Xiao, J. Zhu, and F. Ren, "Easy-first chinese pos tagging and dependency parsing," in *Proceedings of COLING 2012*. The COLING 2012 Organizing Committee, 2012, pp. 1731–1746. [Online]. Available: http://aclweb.org/anthology/C12-1106

[17] M. Tapaswi, M. Bäuml, and R. Stiefelhagen, ""Knock! Knock! Who is it?" Probabilistic Person Identification in TV-Series," in *International Conference on Computer Vision and Pattern Recognition*, 2012.

[18] M. Bäuml, M. Tapaswi, and R. Stiefelhagen, "Semi-supervised Learning with Constraints for Person Identification in Multimedia Data," in *International Conference on Computer Vision and Pattern Recognition*, 2013.

[19] A. Roy, C. Guinaudeau, H. Bredin, and C. Barras, "TVD: a Reproducible and Multiply Aligned TV Series Dataset," in *LREC 2014, 9th Language Resources and Evaluation Conference*, 2014.

[20] H. Bredin, A. Roy, N. Pêcheux, and A. Allauzen, ""Sheldon speaking, bonjour!" - Leveraging Multilingual Tracks for (Weakly) Supervised Speaker Identification," in *ACM Multimedia 2014, The 22nd ACM International Conference on Multimedia*, 2014.

[21] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[23] A. C. Müller and S. Behnke, "pystruct - learning structured prediction in python," *Journal of Machine Learning Research*, vol. 15, pp. 2055–2060, 2014. [Online]. Available: http://jmlr.org/papers/v15/mueller14a.html

[24] A. Niculescu-Mizil and R. Caruana, "Predicting Good Probabilities with Supervised Learning," in *Proceedings of the 22nd International Conference on Machine Learning*, ser. ICML '05. New York, NY, USA: ACM, 2005, pp. 625–632. [Online]. Available: http://doi.acm.org/10.1145/1102351.1102430