Méthodologie de la recherche

Cours de Traitement Automatique des Langues
M1 DAC
UPMC

Anne-Laure Ligozat

2015/2016

À rendre

- Synthèse bibliographique
- Application

- un thème à choisir (cf dernière section du cours)
- en binôme

Objectifs du cours

- Scientifiques

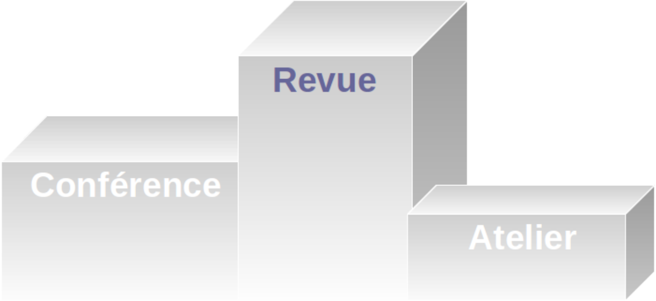
  Découverte d'un **thème du TAL**
- Techniques

  Manipulation d'**outils de TAL**
- Méthodologiques

  Rédaction d'une **synthèse bibliographique**

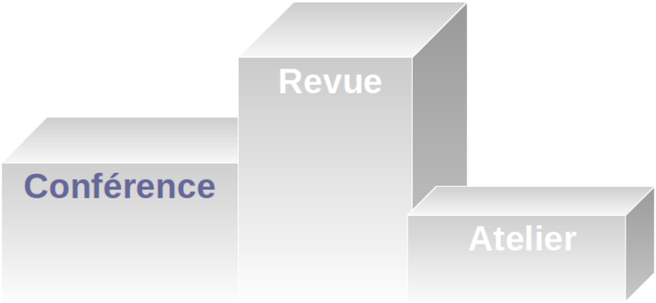
Nature des articles

- **Revue** (*journal*) internationale ou nationale
  - processus de sélection (plus ou moins) strict et long
  - environ 20/30 pages



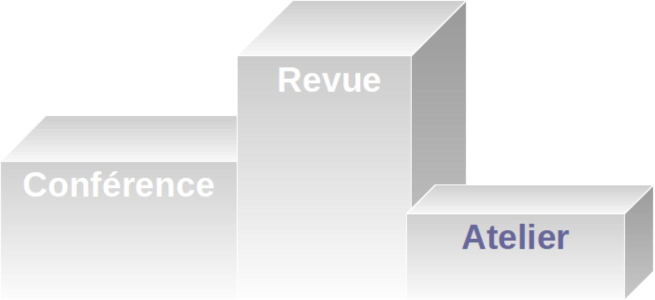
Nature des articles

- **Conférence**
  - comité de lecture
  - annuelles ou bisannuelles
  - environ 10 pages



Nature des articles

- **Atelier** (*workshop*), working notes de campagnes d'évaluation
    - intérêt : rencontre des spécialistes du domaines
    - description précise de systèmes



Principales revues et conférences en TAL

Revues	Conférences
Computational Linguistics NATURAL LANGUAGE ENGINEERING	acl hlt Coling NAACL HLT ACL-IJCNLP EACL LANGUAGE RESOURCES EVALUATION CONFERENCE LREC
t.a.l.	TALN CORIA CONFÉRENCE EN RECHERCHE D'INFORMATION ET APPLICATIONS

+ workshops comme BioNLP

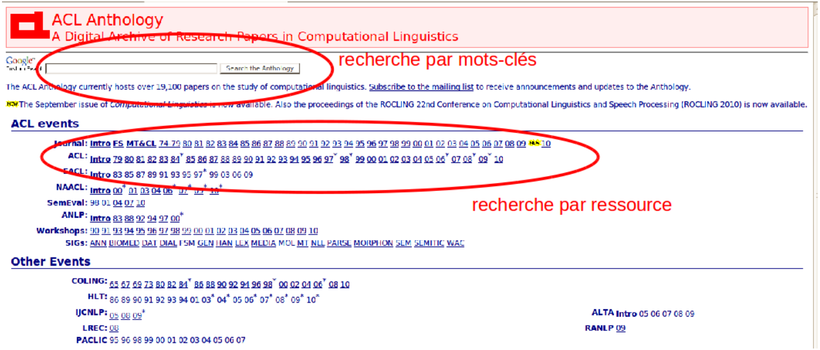
+ campagnes d'évaluation (CLEF, SemEval...)

Accès aux articles : Google scholar

Travaux de recherche de tous domaines

Accès aux articles : anthologie ACL

Conférences et revues internationales en TAL



Organisation d'un article

- Introduction
  - problème dans son **contexte** général
  - à quelle **problématique** les auteurs veulent-ils répondre ?
  - très rapide **état de l'art**
  - **apport** de l'article
  - **organisation** de l'article
- Corps de l'article
  - **état de l'art**
  - sinon, pas de format "type" en informatique, mais en général
    - **hypothèse**
    - algorithme/**méthode**
    - **résultats** et analyse d'erreurs
    - **comparaison** à l'existant
- Conclusion
  - **rappel** des idées fortes et résultats principaux
  - **discussion**
  - proposition de futures **directions** de recherche

Exemple d'article

À vous de jouer : annotez un article

- Event Extraction as Dependency parsing
- David McClosky, Mihai Surdeanu, and Christopher D. Manning
- ACL
- 2011

Event Extraction as Dependency Parsing

David McClosky, Mihai Surdeanu, and Christopher D. Manning
Department of Computer Science
Stanford University
Stanford, CA 94305
{mcclosky,mihais,manning}@stanford.edu

Résumé de l'état de l'art

Abstract

Nested event structures are a common occurrence in both open domain and domain specific extraction tasks, e.g., a "crime" event can cause a "investigation" event, which can lead to an "arrest" event. However, most current approaches address event extraction with highly local models that extract each event and argument independently. We propose a simple approach for the extraction of such structures by taking the tree of event-argument relations and using it directly as the representation in a reranking dependency parser. This provides a simple framework that captures global properties of both nested and flat event structures. We explore a rich feature space that models both the events to be parsed and context from the original supporting text. Our approach obtains competitive results in the extraction of biomedical events from the BioNLP'09 shared task with a F1 score of 53.5% in development and 48.6% in testing.

Contexte

1 Introduction

Event structures in open domain texts are frequently highly complex and nested: a "crime" event can cause an "investigation" event, which can lead to an "arrest" event (Chambers and Jurafsky, 2009). The same observation holds in specific domains. For example, the BioNLP'09 shared task (Kim et al., 2009) focuses on the extraction of nested biomolecular events, where, e.g., a REGULATION event causes a TRANSCRIPTION event (see Figure 1a for a detailed example). Despite this observation, many state-of-the art supervised event extraction models still extract events and event arguments independently, ignoring their underlying structure (Björne et al., 2009; Miwa et al., 2010b).

Problématique

In this paper, we propose a new approach for supervised event extraction where we take the tree of relations and their arguments and use it directly as the representation in a dependency parser (rather than conventional syntactic relations). Our approach is conceptually simple: we first convert the original representation of events and their arguments to dependency trees by creating dependency arcs between *event anchors* (phrases that anchor events in the supporting text) and their corresponding arguments.[1] Note that after conversion, only event anchors and entities remain. Figure 1 shows a sentence and its converted form from the biomedical domain with four events: two POSITIVE REGULATION events, anchored by the phrase "acts as a costimulatory signal," and two TRANSCRIPTION events, both anchored on "gene transcription." All events take either protein entity mentions (PROT) or other events as arguments. The latter is what allows for nested event structures. Existing dependency parsing models can be adapted to produce these semantic structures instead of syntactic dependencies. We built a global reranking parser model using multiple decoders from MSTParser (McDonald et al., 2005; McDonald et al., 2005b). The main contributions of this paper are the following:

Apports

1. We demonstrate that parsing is an attractive approach for extracting events, both nested and otherwise.

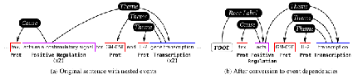
[1] While our approach only works on trees, we show how we can handle directed acyclic graphs in Section 5.

1626

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 1626–1635,
Portland, Oregon, June 19-24, 2011. ©2011 Association for Computational Linguistics

Apports (suite)

État de l'art



(a) Original sentence with nested events (b) After conversion to event dependencies

Figure 1: Nested events in the text fragment: "…the *HTLV-1 transactivator protein, tax*, **acts as a costimulatory signal** for *GM-CSF* and *IL-2* **gene transcription** …" Throughout this paper, **bold text** indicates instances of event anchors and *italicized text* denotes entities (PROTEINs in the BioNLP'09 domain). Note that in (a) there are two copies of each type of event, which are merged to single nodes in the dependency tree (Section 3.1).

2. We propose a wide range of features for event extraction. Our analysis indicates that features which model the global event structure yield considerable performance improvements, which proves that modeling event structure jointly is beneficial.
3. We evaluate on the biomolecular event corpus from the the BioNLP'09 shared task and show that our approach obtains competitive results.

2 Related Work

The pioneering work of Miller et al. (1997) was the first, to our knowledge, to propose parsing as a framework for information extraction. They extended the syntactic annotations of the Penn Treebank corpus (Marcus et al., 1993) with entity and relation mentions specific to the MUC-7 evaluation (Chinchor et al., 1997) — e.g., EMPLOYEE OF relations that hold between person and organization named entities — and then trained a generative parsing model over this combined syntactic and semantic representation. In the same spirit, Finkel and Manning (2009) merged the syntactic annotations and the named entity annotations of the OntoNotes corpus (Hovy et al., 2006) and trained a discriminative parsing model for the joint problem of syntactic parsing and named entity recognition. However, both these works require a unified annotation of syntactic and semantic elements, which is not always feasible, and focused only on named entities and binary relations. On the other hand, our approach focuses on event structures that are nested and have an arbitrary number of arguments. We do not need a unified syntactic and semantic representation (but we can and do extract features from the underlying syntactic structure of the text).

Finkel and Manning (2009b) also proposed a parsing model for the extraction of nested named entity mentions, which, like this work, parses just the corresponding semantic annotations. In this work, we focus on more complex structures (events instead of named entities) and we explore more global features through our reranking layer.

In the biomedical domain, two recent papers proposed joint models for event extraction based on Markov logic networks (MLN) (Riedel et al., 2009; Poon and Vanderwende, 2010). Both works propose elegant frameworks where event anchors and arguments are jointly predicted for all events in the same sentence. One disadvantage of MLN models is the requirement that a human expert develop domain-specific predicates and formulas, which can be a cumbersome process because it requires thorough domain understanding. On the other hand, our approach maintains the joint modeling advantage, but our model is built over simple, domain-independent features. We also propose and analyze a richer feature space that captures more information on the global event structure in a sentence. Furthermore, since our approach is agnostic to the parsing model used, it could easily be tuned for various scenarios, e.g., models with lower inference overhead such as shift-reduce parsers.

Our work is conceptually close to the recent CoNLL shared tasks on semantic role labeling, where the predicate frames were converted to se-

1627

Méthode

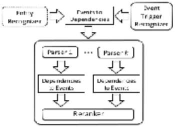


Figure 2: Overview of the approach. Rounded rectangles indicate domain independent components; regular rectangles mark domain-specific modules; blocks in dashed lines surround components not necessary for the domain presented in this paper.

mantic dependencies between predicates and their arguments (Surdeanu et al., 2008; Hajic et al., 2009). In this representation the dependency structure is a directed acyclic graph (DAG), i.e., the same node can be an argument to multiple predicates, and there are no explicit dependencies between predicates. Due to this representation, all joint models proposed for semantic role labeling handle semantic frames independently.

3 Approach

Figure 2 summarizes our architecture. Our approach converts the original event representation to dependency trees containing both event anchors and entity mentions, and trains a battery of parsers to recognize these structures. The trees are built using event anchors predicted by a separate classifier. In this work, we do not discuss entity recognition because in the BioNLP'09 domain used for evaluation entities (PROTEINs) are given (but including entity recognition is an obvious extension of our model). Our parsers are several instances of MSTParser[2] (McDonald et al., 2005; McDonald et al., 2005b) configured with different decoders. However, our approach is agnostic to the actual parsing models used and could easily be adapted to other dependency parsers. The output from the reranking parser is converted back to the original event representation and passed to a reranker component (Collins, 2000; Charniak and Johnson, 2005), tailored to optimize the task-specific evaluation metric.

Note that although we use the biomedical event domain from the BioNLP'09 shared task to illustrate our work, the core of our approach is almost domain independent. Our only constraints are that each event mention be activated by a phrase that serves as an event anchor, and that the event-argument structures be mapped to a dependency tree. The conversion between event and dependency structures and the reranker metric are the only domain dependent components in our approach.

3.1 Converting between Event Structures and Dependencies

As in previous work, we extract event structures at sentence granularity, i.e., we ignore events which span sentences (Björne et al., 2009; Riedel et al., 2009; Poon and Vanderwende, 2010). These form approximately 5% of the events in the BioNLP'09 corpus. For each sentence, we convert the BioNLP'09 event representation to a graph (representing a labeled dependency tree) as follows. The nodes in the graph are protein entity mentions, event anchors, and a virtual ROOT node. Thus, the only words in this dependency tree are those which participate in events. We create edges in the graph in the following way. For each event anchor, we create one link to each of its arguments labeled with the slot name of the argument (for example, connecting **gene transcription** to *IL-2* with the label THEME in Figure 1b). We link the ROOT node to each entity that does not participate in an event using the ROOT-LABEL dependency label. Finally, we link the ROOT node to each top-level event anchor, (those which do not serve as arguments to other events) again using the ROOT-LABEL label. We follow the convention that the source of each dependency arc is the head while the target is the modifier.

The output of this process is a directed graph, since a phrase can easily play a role in two or more events. Furthermore, the graph may contain self-referential edges (self-loops) due to related events sharing the same anchor (example below). To guarantee that the output of this process is a tree, we must post-process the above graph with the follow-

[2] http://sourceforge.net/projects/mstparser

1628

	Unreranked			Reranked		
Decoder(s)	R	P	F1	R	P	F1
1P	65.6	76.7	70.7	68.0	77.6	72.5
2P	67.4	77.1	71.9	67.9	77.3	72.3
1N	67.5	76.7	71.8	—	—	—
2N	68.9	77.1	72.7	—	—	—
1P, 2P, 2N	—	—	—	68.5	78.2	73.1

(a) Gold event anchors

	Unreranked			Reranked		
Decoder(s)	R	P	F1	R	P	F1
1P	44.7	62.2	52.0	47.8	59.6	53.1
2P	45.9	61.8	52.7	48.4	57.5	52.5
1N	46.0	61.2	52.5	—	—	—
2N	38.6	66.6	48.8	—	—	—
1P, 2P, 2N	—	—	—	48.7	59.3	53.5

(b) Predicted event anchors

Table 1: BioNLP recall, precision, and F1 scores of individual decoders and the best decoder combination on development data with the impact of event anchor detection and reranking. Decoder names include the features order (1 or 2) followed by the projectivity (P = projective, N = non-projective).

decoder, number of different decoders producing the parse (when using multiple decoders).

- **Event path:** Path from each node in the event tree up to the root. Unlike the **Path** features in the parser, these paths are over event structures, not the syntactic dependency graphs from the original English sentence. Variations of the **Event path** features include whether to include word forms (e.g., "binds"), types (BINDING), and/or argument slot names (THEME). We also include the path length as a feature.
- **Event frames:** Event anchors with all their arguments and argument slot names.
- **Consistency:** Similar to the parser **Consistency** features, but capable of capturing larger classes of errors (e.g., incorrect number or types of arguments). We include the number of violations from four different classes of errors.

To improve performance and robustness, features are pruned as in Charniak and Johnson (2005): selected features must distinguish a parse with the highest F1 score in a n-best list, from a parse with a suboptimal F1 score at least five times.

Rerankers can also be used to perform model combination (Toutanova et al., 2008; Zhang et al., 2009; Johnson and Ural, 2010). While we use a single parsing model, it has multiple decoders.[10] When combining multiple decoders, we concatenate their n-best lists and extract the unique parses.

[10]We only have n-best versions of the projective decoders. For the non-projective decoders, we use their 1-best parse.

4 Experimental Results

Our experiments use the BioNLP'09 shared task corpus (Kim et al., 2009) which includes 800 biomedical abstracts (7,449 sentences, 8,597 events) for training and 150 abstracts (1,450 sentences, 1,809 events) for development. The test set includes 260 abstracts, 2,447 sentences, and 3,182 events. Throughout our experiments, we report BioNLP F1 scores with approximate span and recursive event matching (as described in the shared task definition). For preprocessing, we parsed all documents using the self-trained biomedical McClosky-Charniak-Johnson reranking parser (McClosky, 2010). We bias the anchor detector to favor recall, allowing the parser and reranker to determine which event anchors will ultimately be used. When performing n-best parsing, n = 50. For parser feature pruning, $\alpha = 0.001$.

Table 1a shows the performance of each of the decoders when using gold event anchors. In both cases where n-best decoding is available, the reranker improves performance over the 1-best parsers. We also present the results from a reranker trained from multiple decoders which is our highest scoring model.[11] In Table 1b, we present the output for the predicted anchor scenario. In the case of the 2P decoder, the reranker does not improve performance, though the drop is minimal. This is because the reranker chose an unfortunate regularization constant during crossvalidation, most likely due to the small size of the training data. In later experiments where more

[11]Interestingly, the 1N decoder [illegible] gains, possibly because [illegible] by the 2N decoder.

1652

Résultats

Analyse d'erreurs

Event Class	Count	R	P	F1
Gene Expression	722	68.6	75.8	72.0
Transcription	137	42.3	51.3	46.4
Protein Catabolism	14	64.3	75.0	69.2
Phosphorylation	135	80.0	82.4	81.2
Localization	174	44.8	78.8	57.1
Binding	347	42.9	51.7	46.9
Regulation	291	23.0	36.6	28.3
Positive Regulation	983	28.4	42.5	34.0
Negative Regulation	379	29.3	43.5	35.0
Total	3,182	42.6	56.6	48.6

Table 4: Results in the test set broken by event class; scores generated with the main official metric of approximate span and recursive event matching.

dividual sentences) by using a representation with a unique ROOT node for all event structures in a document. This representation has the advantage that it maintains cross-sentence events (which account for 5% of BioNLP'09 events), and it allows for document-level features that model discourse structure. We plan to explore these ideas in future work.

Discussion

One current limitation of the proposed model is that it constrains event structures to map to trees. In the BioNLP'09 corpus this leads to the removal of almost 5% of the events, which generate DAGs instead of trees. Local event extraction models (Björne et al., 2009) do not have this limitation, because their local decisions are blind to (and hence not limited by) the global event structure. However, our approach is agnostic to the actual parsing models used, so we can easily incorporate models that can parse DAGs (Sagae and Tsujii, 2008).

Pistes

Additionally, we are free to incorporate any new techniques from dependency parsing. Parsing using dual-decomposition (Rush et al., 2010) seems especially promising in this area.

6 Conclusion

Rappel

In this paper we proposed a simple approach for the joint extraction of event structures: we converted the representation of events and their arguments to dependency trees with arcs between event anchors and event arguments, and used a reranking parser to parse these structures. Despite the fact that our approach has very little domain-specific engineering, we obtain competitive results. Most importantly, we showed that the joint modeling of event structures is beneficial: our reranker outperforms parsing models without reranking in five out of the six configurations investigated.

Acknowledgments

The authors would like to thank Mark Johnson for helpful discussions on the reranker component and the BioNLP shared task organizers, Sampo Pyysalo and Jin-Dong Kim, for answering questions. We gratefully acknowledge the support of the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0181. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of DARPA, AFRL, or the US government.

References

Jari Björne, Juho Heimonen, Filip Ginter, Antti Airola, Tapio Pahikkala, and Tapio Salakoski. 2009. *Extracting Complex Biological Events with Rich Graph-Based Feature Sets*. Proceedings of the Workshop on BioNLP: Shared Task.

Nate Chambers and Dan Jurafsky. 2009. *Unsupervised Learning of Narrative Schemas and their Participants*. Proceedings of ACL.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 2005 Meeting of the Association for Computational Linguistics (ACL)*, pages 173–180.

Nancy Chinchor. 1997. *Overview of MUC-7*. Proceedings of the Message Understanding Conference (MUC-7).

Michael Collins. 2000. Discriminative reranking for natural language parsing. In *Machine Learning: Proceedings of the Seventeenth International Conference (ICML 2000)*, pages 175–182.

Jenny R. Finkel and Christopher D. Manning. 2009. *Joint Parsing and Named Entity Recognition*. Proceedings of NAACL.

Jenny R. Finkel and Christopher D. Manning. 2009b. *Nested Named Entity Recognition*. Proceedings of EMNLP.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria A. Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen

Objectif d'une synthèse

Idée

Présenter le domaine en donnant les **principaux axes** des travaux effectués

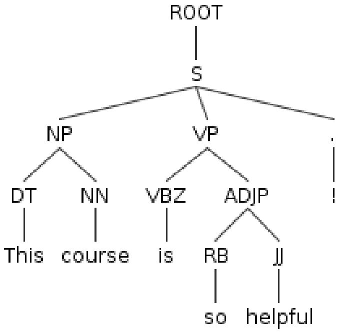
- = **état de l'art** ($\neq$ analyse de l'existant)
- présenter également les **limites** des méthodes actuelles

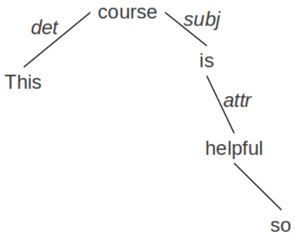
Attention: théorique vs technique

Ex : formalisme d'analyse syntaxique

- structure en dépendances ou en constituants = théorique
- arbre ou format XML ou parenthésé = technique

Aspects techniques vs théoriques : exemple






```
(ROOT
  (S
    (NP (DT This) (NN course))
    (VP (VBZ is)
      (ADJP (RB so) (JJ helpful)))
    (. !)))
```

det(course-2, This-1)
subj(course-2, is-3)
attr(helpful-5, is-3)
advmod(helpful-5, so-4)

Plan d'une synthèse (1/2)

Introduction

- présenter le **sujet**
  - Ex : Cette synthèse aborde la problématique de l'extraction de relations.
- donner les **définitions** des termes principaux
  - Ex : extraction d'information, extraction de relations
- expliquer les **difficultés**
  - ambiguïtés de rattachement, variations...
- présenter l'**historique** du domaine et les articles fondateurs

Plan d'une synthèse (2/2)

Corps

- organiser en fonction des **axes de recherche** actuels (et non du type d'application, des équipes...)
- citer des **travaux représentatifs** de chaque axe
- **évaluation** dans le domaine

Conclusion

- **pistes** de recherche futures

En détails

- Attention au **vocabulaire** utilisé: reprendre les termes du domaine (quitte à se répéter)
  - traduire les termes : tokenization = segmentation en mots
  - être précis : terme ≠ mot ≠ entité
- Donner des **exemples**
- Donner des **résultats** de systèmes (beaucoup de campagnes d'évaluation en TAL)
- Attention au format des **références**

Exemple de plan de synthèse

Extraction de relations

- Introduction
- Approche à base de patrons
  - surfaciques
  - syntaxiques
  - acquisition automatique de patrons
- Approche par apprentissage
  - avec attributs vectoriels
  - sur structure arborescente
- Évaluation
- Conclusion

Bibliographie : références

- Citer ses sources !
- Les citer correctement
  - Éléments indispensables dans la référence :
    - **titre** de l'article
    - noms des **auteurs**
    - titre de la **ressource** : nom de la conférence ou de la revue
    - **année** de publication
  - Exemple :
    - Anja Belz, Michael White, Dominic Espinosa, Eric Kow, Deirdre Hogan, and Amanda Stent. 2011. The first surface realisation shared task: Overview and evaluation results. In Proceedings of the 13th European Workshop on Natural Language Generation (ENLG), Nancy, France

Bibliographie : format ACL

- Citations dans le texte
  - Citations within the text appear in parentheses as (Gusfield, 1997) or, if the author's name appears in the text itself, as Gusfield (1997). Append lowercase letters to the year in cases of ambiguity. Treat double authors by using both authors' last names (e.g., (Aho and Ullman, 1972), but use et al. when more than two authors are involved. (e.g. (Chandra et al., 1981)) Collapse multiple citations (e.g., (Gusfield, 1997; Aho and Ullman, 1972).)
- Dans la partie Références
  - Alfred V. Aho and Jeffrey D. Ullman. 1972. The Theory of Parsing, Translation and Compiling, volume 1. Prentice-Hall, Englewood Cliffs, NJ.
  - Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. Journal of the Association for Computing Machinery, 28(1):114–133.
  - Dan Gusfield. 1997. Algorithms on Strings, Trees and Sequences. Cambridge University Press, Cambridge, UK.

BibTex : format des références

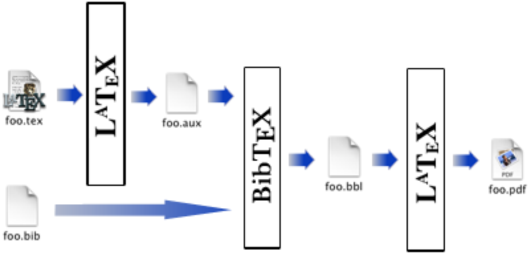
BibTex : gestion et traitement des données bibliographiques

forme : clé = valeur

```
@inproceedings{LanglaisEACL2009,
  author = {Langlais, Philippe and Yvon, François and
            Zweigenbaum, Pierre},
  title = {Improvements in Analogical Learning:
           Application to Translating Multi-Terms
           of the Medical Domain},
  booktitle = {Proceedings of the 12th Conference of
               the European Chapter of the Association
               for Computational Linguistics (EACL 2009)},
  publisher = {Association for Computational Linguistics},
  year = {2009},
  pages = {487-495},
  url = {http://www.aclweb.org/anthology/E09-1056}
}
```

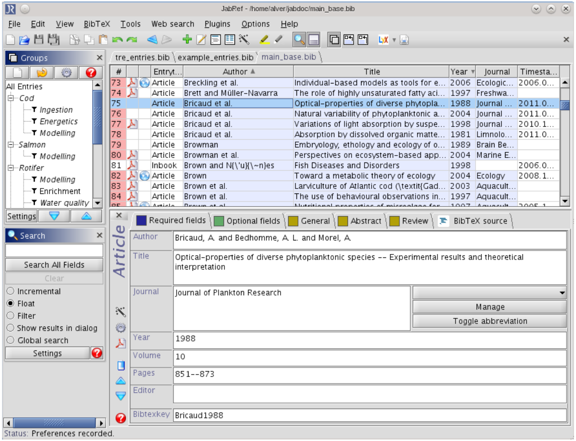
BibTex : utilisation dans LaTeX (pdflatex)

- dans le .tex :
  \cite{LanglaisEACL2009} montrent qu'il est également possible de traduire...
- dans le .pdf :
  (Langlais et al., 2009) montrent qu'il est également possible de traduire..



JabRef

- gestion graphique des références bibliographiques
- exports BibTeX, texte, OpenOffice (plugin)...



Caractéristiques des références

- références anciennes si besoin (articles fondateurs) et récentes (cinq dernières années)
- articles d'auteurs, laboratoires et pays variés

Objectif de l'application

- Objectif : aborder le domaine du TAL de la synthèse d'un point de vue pratique
- Application = mise en œuvre d'un algorithme, évaluation/comparaison d'outils/méthodes...

Choix à faire

- **Corpus de test**
  - quelle source ? quel format ? prétraitement nécessaires ? $\rightarrow$ cette année, corpus unique
- **Entrée/sorties** du système
  - quel format en entrée, quel format en sortie ?
- **Tests** de l'application
  - cas nominal, limites, erreurs
- Mode d'**évaluation**
  - choisir dès le départ une évaluation standard (ex : rappel, précision, f-mesure)

Rapport sur l'application

- **Expliciter** et **justifier** les choix
- Donner des **exemples** précis des résultats du système

Analyse morphologique

- notions : lemmatisation, stemming, familles morphologiques
- articles : étiquetage du français et construction de familles morphologiques
- application : Comparer stemming, lemmatisation et familles morphologiques (en utilisant les derivationally related forms de Wordnet ou avec un outil de segmentation morphologique comme Morfessor) pour de la sélection de passages répondant à des questions

Terminologie : extraction de termes et de collocations

- notions : termes (Multi Word Unit - MWU ou MultiWord Expression - MWE), mesures de cooccurences/collocations
- articles : reconnaissance d'acronymes, reconnaissance de termes
- application : Constitution automatique d'une base d'acronymes avec leur signification et annotation des acronymes dans les textes; évaluation de l'apport à la recherche de passages

Terminologie, variations de termes

- notions : termes, expressions, variations linguistiques
- articles : reconnaissance de variantes, validation de relations entre termes
- application : Validation en contexte de variations morpho-sémantiques de mots (en utilisant les informations de synonymie et morphologiques de Wordnet) à partir des cooccurrents (issus par exemple de la base de cooccurrences Wortschatz)

Variations, paraphrase

- notions : paraphrase, implication textuelle
- articles : reconnaissance de paraphrases, implication textuelle
- application : Utilisation d'un outil d'implication textuelle (exemple : EDITS, ou BIUTEE) ou une banque de paraphrases (exemple : PPDB) et évaluation sur QA4MRE

Entités nommées

- notions : reconnaissance d'entités nommées, désambiguïsation et résolution d'entités nommées
- articles : typage non supervisé d'entités; résolution d'entités nommées
- application : Suivi d'entités nommées

Analyse syntaxique

- notions : analyse syntaxique en constituants et en dépendances, arbres syntaxiques
- articles : génération de questions, analyse syntaxique du français, correction d'analyse morpho-syntaxique
- application 1 : Génération d'hypothèses et validation
- application 2 : Correction d'analyse syntaxique de questions

Sémantique : synonymie, structuration des connaissances

- notions : sens, ambiguité, relation sémantiques, évaluation, construction de ressources
- articles : synonymie, construction de ressource

Anaphore et coréférence

- notions : coréférence
- article : apprentissage et coreference; anaphore
- application : Évaluation d'un système d'annotation de coréférence et analyse du corpus

Analyse thématique

- notions : cohésion lexicale, distribution des mots dans le texte et dans blocs, segmentation thématique
- article : segmentation par ressources segmentation thématique
- application : Étude de la segmentation thématique pour la sélection de passages (2 segmenteurs)

Résumé automatique

- notions : résumé par extraction, critères de sélection de phrases importantes
- articles : résumé par ordonnancement résumé multidocuments
- application : Identification de thèmes

Analyse du discours

- notions : relations du discours, structure logique
- articles : analyse par règles apprentissage de relations implicites
- application 1 : Segmentation automatique en phrases et reconnaissance des titres
- application 2 : Reconnaissance de relations du discours