Term Ranking Adaptation to the Domain: Genetic Algorithm-Based Optimisation of the *C-Value*

Thierry Hamon[1,2], Christopher Engström[3], Sergei Silvestrov[3]

(1) LIMSI-CNRS, Orsay, France
thierry.hamon@limsi.fr
(2) University Paris 13, Sorbonne Paris Cité, France
(3) Mälardalen University, Sweden
christopher.engstrom@mdh.se
sergei.silvestrov@mdh.se

September 17, 2014

Plan

- Introduction
- Parametrised *C-Value* and optimisation method
- Term extraction and corpora
- Experiments and results
- Conclusion

Context

Text mining in scientific and technical fields (medicine, legal domain, energy production)

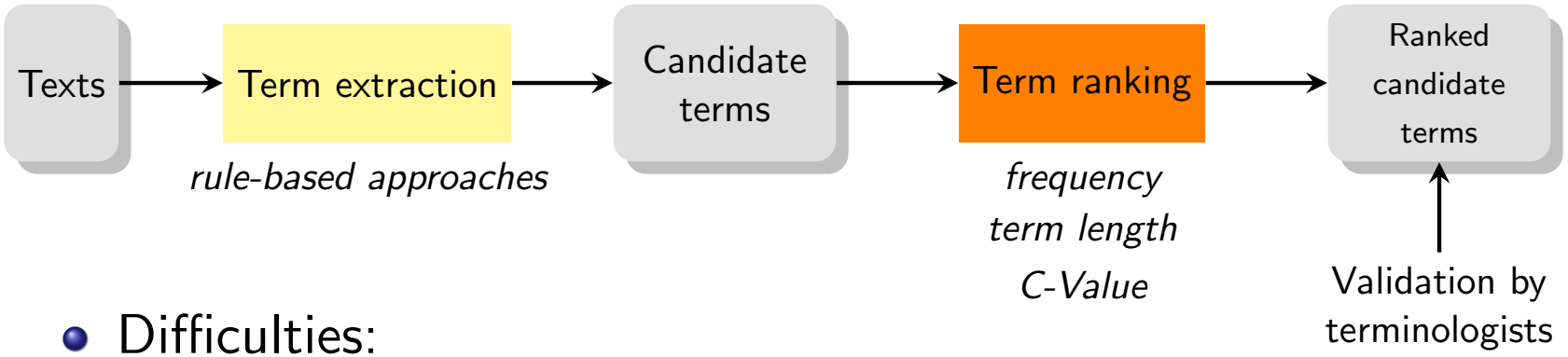
- Use of terminological resources providing terms (noun phrases that convey knowledge of the domain)
- But: low coverage of terminological resources when recognizing terms in texts

In *coimmunoprecipitation* **experiments**$_{C0681814}$ *using* *transfected* **COS cells**$_{C0376702}$, **GATA**$_{C1427248}$*-1* *and* **ER**$_{C0014239}$ *associate in a* **ligand**$_{C1749457}$*-dependent manner*.

bold: term issued from UMLS – box : extracted candidate term

Solution: Improving the coverage of terminological resources with automatic extraction of potential terminological entities (candidate terms)

Automatic term extraction framework



- Difficulties:
  - to catch the termhood of the extracted noun phrases
  - to identify terms of the domain

→ Need of automatically ranking the candidate terms according to their termhood

Objective: Proposition of improvement of the C-Value:

[Frantzi et al. 1997]

- Addition of parameters in the *C-Value*
- Optimization of the parameters with a genetic algorithm

Metrics for ranking extracted terms

- Frequency: *commonly considered as ranking metric*
  - Decrease either the recall as many candidate terms occur only once in the corpus or the precision

    [Justeson&Katz 1995, Frantzi et al. 2000, Dowdall et al. 2002]
- Length of the terms: *longer terms are less important*
  - Slight increase of the precision when combined to the frequency: single word candidate terms or short multi-word terms are preferred

    [Drouin 2002]
- C-Value: *Long multi-word terms which are not components of other terms are preferred*

  [Frantzi et al. 1997]

$$C - Value(t) = \begin{cases} log_2(|t|+1) \cdot f(t) \text{ if } t \text{ is not included in a term} \\ log_2(|t|+1) \cdot (f(t) - \frac{1}{P(T_t)} \sum_{t' \in T_t} f(t')) \text{ otherwise} \end{cases}$$

  - Mix improvement: precision increases by 31% for the candidate terms only appearing as nested, but only by 1% for all the candidate terms

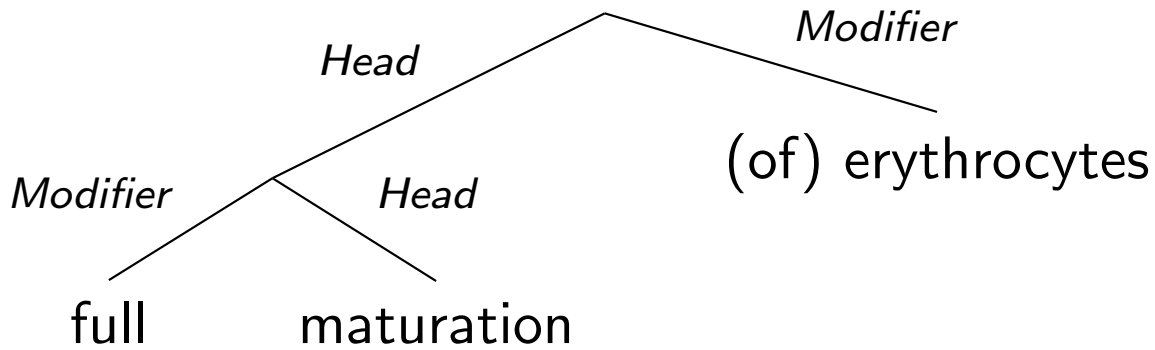
    [Frantzi et al. 2000]

Term extraction

Extraction of candidate terms by YaTeA :

[Aubin et Hamon 2006]

- Shallow parsing of the POS-tagged and lemmatized texts
- Identification of single-word and multi-word candidate terms
- Representation of the terms as a syntactic tree (with the syntactic role to each term components)



- Association of statistical measures (frequency, *C-Value*)

Parametrised *C-Value*: *C-Value**

- Taking into account terminological practice of each domain:
  - *influence to the length of the terms ($|t|$) may depend on the domain:* $log_2(\frac{|t|+1}{|t|^{\alpha}})$
- Taking into account the syntactic role of the terms and their nestedness in the weight of the term length:
  - Root terms which are not nested in any other term: α_R
  - Head terms of another term: α_H
  - Modifier terms of another term: α_M

Parametrised *C-Value*: *C-Value**

- Taking into account the frequency distribution of the nested terms (β_H, β_M):

  use of a β-norm to give more penalty to a term nested in several terms with unbalanced frequency distribution

- Considering the influence of the nesting terms (c_H, c_M):

  higher c gives a higher penalty if the term is included in others

$$C - Value^* = \begin{cases} log_2\left(\frac{|t|+1}{|t|^{\alpha_R}}\right) \cdot f(t), \text{ if } t \text{ is not included in a term (Root)} \\ log_2\left(\frac{|t|+1}{|t|^{\alpha_H}}\right) \cdot \left(f(t) - c_H\left(\sum_{t' \in T_t} f(t')^{\beta_H}\right)^{1/\beta_H}\right), \\ \qquad\qquad \text{if } t \text{ is a Head term} \\ log_2\left(\frac{|t|+1}{|t|^{\alpha_M}}\right) \cdot \left(f(t) - c_M\left(\sum_{t' \in T_t} f(t')^{\beta_M}\right)^{1/\beta_M}\right), \\ \qquad\qquad \text{if } t \text{ is a Modifier term} \end{cases}$$

Parameter optimisation

- Optimisation of the parameters α (α_R, α_H, α_M), β (β_H, β_M) and c (c_H, cM)
- Use of a real-coded genetic algorithm [Wright 1991]
- Fitness function: $f = \sum_{i \in I} r(i)$ where
  - $I = \{N/6,\ 2N/6,\ 3N/6,\ 4N/6,\ 5N/6\}$
  - N: total number of terms
  - $r(i)$: number of annotated terms among the i highest ranked terms
- Algorithm configuration:
  - Population of 200 individuals
  - Selection of the parents with a tournament selection scheme
  - Creation of new samples: BLX-0.5 blend crossover scheme [Herrera et al. 2003]
  - Mutation rate: 20% (a old parameter is replaced with a new randomly generated one)

Corpora

- **Genia** Corpus: [JinDong et al. 2003]
  - 1,999 Medline abstracts (transcription factors in human cells)
  - 436,967 words, 36,607 annotated terms
  - 49,249 extracted candidate terms
- PennBioIE Corpus [Kulick&al2004]
  - **CYP450** sub-corpus:
    - 1,100 Medline abstracts (cytochrome P450 protein modif.)
    - 298,843 words, 42,337 annotated terms
    - 47,168 candidate terms
  - **Oncology** sub-corpus
    - 1,157 Medline abstracts (cancer genomics)
    - 276,161 words, 6,704 annotated term
    - 39,542 extracted candidate terms
- Pre-processing in the Ogmios platform: [Hamon et al. 2007]
  - Word and sentence segmentation
  - Part-of-speech tagging and lemmatisation with the Genia Tagger [Tsuruoka et al. 2005]

Experiments

- Ranking of the terms extracted from the Genia corpus with the *C-Value** (60/40% random split)
- Each corpora: 10-fold cross-validation for ranking the extracted terms
- Recycling Genia parameters on the CYP450 and Oncology corpora

Definition of several model configurations

Model	Parameters
M_1	$\alpha = \beta = c = 1$
$M_{\beta c}$	$\alpha = 1, \beta_H = \beta_M, c_H = c_M$
$M_{\alpha^3 c}$	$\alpha_R, \alpha_H, \alpha_M, \beta = 1, c_H = c_M$
$M_{\alpha^3 \beta}$	$\alpha_R, \alpha_H, \alpha_M, \beta_H = \beta_M, c = 1$
$M_{\alpha\beta c}$	$\alpha_R = \alpha_H = \alpha_M, \beta_H = \beta_M, c_H = c_M$
$M_{\alpha^3 \beta c}$	$\alpha_R, \alpha_H, \alpha_M, \beta_H = \beta_M, c_H = c_M$
$M_{\alpha^3 \beta^2 c^2}$	$\alpha_R, \alpha_H, \alpha_M, \beta_H, \beta_M, c_H, c_M$

Evaluation

- Comparison with the terms annotated in the corpora
- Evaluation measures:
  - Precision: $\frac{\text{number of annotated candidate terms}}{\text{number of ranked candidate terms}}$
  - Recall: $\frac{\text{number of annotated candidate terms}}{\text{number of annotated terms}}$
  - F-measure: $\frac{2 \times Precision \times Recall}{Precision + Recall}$
  - average precision (MAP)

    [Buckley&Voorhees 2005]
  - R-precision: the precision at the rank R (number of annotated terms)
    $\rightarrow$ Point where the precision should be optimal and more suitable for terminology building
- Comparison to the baselines: ranking with the frequency and with the *C-Value*

Results

Random split: 60% for the training, 40% for the test

Model	R-prec$_{train}$	R-prec$_{test}$	avg Prec$_{train}$	avg Prec$_{test}$
frequency	0.4590	0.4671	0.4338	0.4441
C-Value	0.3344	0.3594	0.3935	0.4147
M_1	0.5091	0.5090	0.5088	0.5124
$M_{\beta c}$	0.4974	0.5084	0.4910	0.5002
$M_{\alpha^3 c}$	0.5259	**0.5285**	**0.5416**	**0.5407**
$M_{\alpha^3 \beta}$	**0.5293**	0.5272	0.5387	0.5363
$M_{\alpha \beta c}$	0.5144	0.5139	0.5266	0.5269
$M_{\alpha^3 \beta c}$	0.5197	0.5207	0.5386	0.5360
$M_{\alpha^3 \beta^2 c^2}$	0.5222	0.5233	0.5330	0.5262

- Training and test sets: very similar results
- Models based on the *C-Value** outperformed the baselines
- Usefulness of M_1 when there are no annotated terms for training
- Strong influence of α (negative effect when set to 1 or equal values)
- $M_{\alpha^3 \beta^2 c^2}$: difficulties for genetic algorithm to find global optimal parameter values

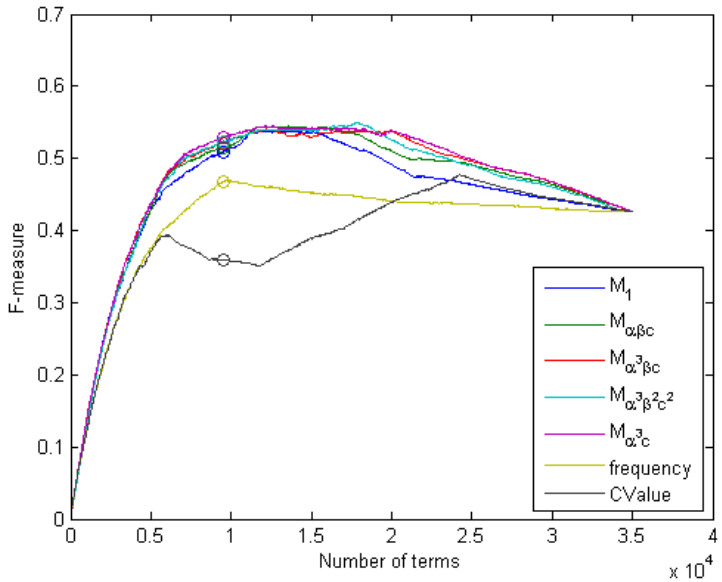
Analyzis of the parameter values

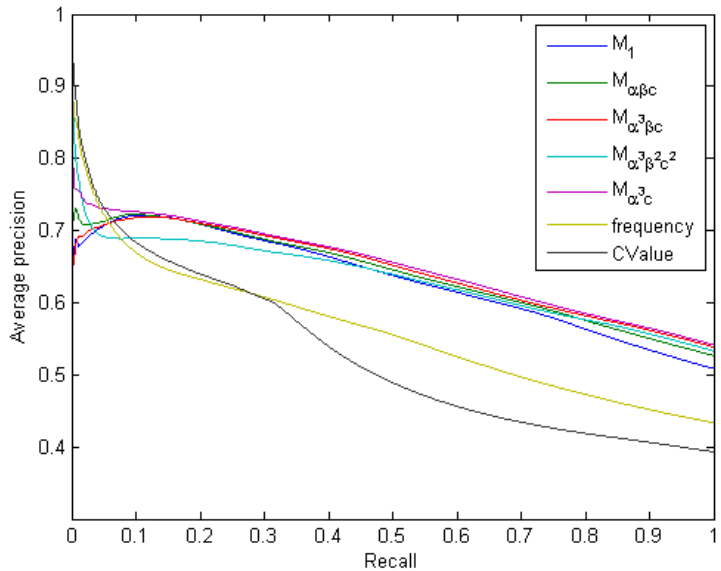
Estimation on the 60% training part of the Genia corpus

Model	α_R	α_H	α_M	β_H	β_M	c_H	c_M
M_1	1	1	1	1	1	1	1
$M_{\beta c}$	1	1	1	0.997	0.997	0.7095	0.7095
$M_{\alpha^3 c}$	1.1014	1.0344	0	1	1	0.91	0.91
$M_{\alpha^3 \beta}$	1.1622	1.1445	0.075	1.0132	1.0132	1	1
$M_{\alpha \beta c}$	1.1604	1.1604	1.1604	1.0140	1.014	0.9953	0.9953
$M_{\alpha^3 \beta c}$	1.1067	1.0961	0.0857	0.9538	0.9538	0.8316	0.8316
$M_{\alpha^3 \beta^2 c^2}$	1.3005	0.6093	0.7381	1.5085	1.1307	1.5224	1.17

- α_M close to 0 and significantly smaller than the other two α: shorter modifier candidate terms are penalised
- α_R and $\alpha_H > 1$: shorter Head or Root candidate terms are preferred
- β close to 1: low influence of frequency distribution
- $0 <= c <= 1$: low impact, slight decrease of R-precison when $c = 1$

Evolution of the average precision and F-measure





- Frequency and the *C-Value*: better average precision for the very first terms
- Then, all the *C-Value** models outperform the frequency and the *C-Value* ranking
- Also: F-measure: After one hundred terms and until 70% of candidate terms: better ranking with the *C-Value** models
- Similar ranking for all the *C-Value** models

10-fold cross-validation
on the three corpora

- Objective: analyzis of the (in)dependence of *C-Value** parameters on corpora on the three first models

	Genia		CYP450		Oncology	
Model	R-prec	avgPrec	R-prec	avgPrec	R-prec	avgPrec
Frequency	0.3882	0.3589	0.3079	0.2434	0.1017	0.0615
C-Value	0.3055	0.3509	0.2711	0.2013	0.1019	0.0606
$M_{\alpha^3 c}$	0.4318	0.4212	0.3540	0.3028	0.0962	0.0643
$M_{\alpha^3 \beta}$	0.4324	0.4098	0.3595	0.3051	0.0959	0.0620
$M_{\alpha^3 \beta c}$	0.4323	0.4133	0.3621	0.3074	0.0961	0.0643

- *C-Value**-based ranking on **Genia** and **CYP450**: better R-precision and average precision than those obtained with the baselines
  $\rightarrow$ Parameter optimisation can be successfully applied to various text collections
- **Oncology**:
  - Similar ranking whatever the parameters
  - but lower R-precision and slightly better average precision
    $\rightarrow$ disappointing results may be due to the reference (few terms are annotated)

Recycling Genia parameters on other corpora

Use the parameter values estimated on Genia to rank the terms extracted from the CYP450 and Oncology corpora

	CYP450		Oncology	
	R-prec.	avgPrec.	R-prec.	avgPrec.
Frequency	0.3315	0.2596	0.1498	0.0849
CValue	0.2960	0.2042	0.1450	0.0800
M_1	0.3677	0.3484	0.1441	0.0774
$M_{\alpha\beta c}$	**0.4517**	**0.3837**	0.1355	0.0771
$M_{\alpha^3\beta c}$	0.3959	0.3515	0.1508	0.0917
$M_{\alpha^3\beta^2 c^2}$	0.4074	0.3445	0.1450	0.0817
$M_{\alpha^3 c}$	0.3885	0.3410	0.1498	**0.0939**
$M_{\beta c}$	0.3906	0.3314	0.1412	0.0861
$M_{\alpha^3\beta}$	0.3885	0.3552	**0.1517**	0.0879

- **CYP450**:
  - all the *C-Value** models outperform the baselines and M_1 model
  - better model: $M_{\alpha\beta c}$ (syntactic role of the nested terms is not taken into account)

$\rightarrow$ Parameters set on a training corpus achieve good results on other corpora

- **Oncology**: results difficult to interpret (similar results with any model)

Possible reason: low variation in the number of terms given the term length?

Conclusion

- Ranking of candidate terms extracted from specialized corpora
- Proposition of an improved and parametrised *C-Value*, the *C-Value**
- Optimization of the parameters with a standard genetic algorithm
- Evaluation on the three corpora
  - Increase of the R-precision and average precision
  - Improvement of the ranking even when no annotated terms are available (M_1)
  - Parameters set on a training corpus achieve good results on other corpora

Future works

- Analyse the behaviour of *C-Value** according to the domain
- Study the parameter estimation method on smaller training set
- Evaluate the impact of the *C-Value** on other domains