Sentence Filtering for BioNLP: Searching for Renaming Acts

Pierre Warnier[1,2] **Claire Nédellec**[1]
[1]MIG INRA UR 1077, F78352 Jouy-en-Josas, France
[2]LIG Université de Grenoble, France
forename.lastname@jouy.inra.fr

Abstract

The Bacteria Gene Renaming (RENAME) task is a supporting task in the BioNLP Shared Task 2011 (BioNLP-ST'11). The task consists in extracting gene renaming acts and gene synonymy reminders in scientific texts about bacteria. In this paper, we present in details our method in three main steps: 1) the document segmentation into sentences, 2) the removal of the sentences exempt of renaming act (false positives) using both a gene nomenclature and supervised machine learning (feature selection and SVM), 3) the linking of gene names by the target renaming relation in each sentence. Our system ranked third at the official test with 64.4% of F-measure. We also present here an effective post-competition improvement: the representation as SVM features of regular expressions that detect combinations of trigger words. This increases the F-measure to 73.1%.

1 Introduction

The Bacteria Gene Renaming (Rename) supporting task consists in extracting gene renaming acts and gene synonymy reminders in scientific texts about bacteria. The history of bacterial gene naming has led to drastic amounts of homonyms and synonyms that are often missing in gene databases or even worse, erroneous (Nelson et al., 2000). The automatic extraction of gene renaming proposals from scientific papers is an efficient way to maintain gene databases up-to-date and accurate. The present work focuses on the recognition of renaming acts in the literature between gene synonyms that are recorded in the *Bacillus subtilis* gene databases. We assume that renaming acts do not involve unknown gene names. Instead, our system verifies the accuracy of synonymy relations as reported in gene databases by insuring that the literature attests these synonymy relations.

1.1 Example

This positive example of the training corpus is representative of the IE task:
*"Thus, a separate **spoVJ** gene as defined by the 517 mutation does not exist and is instead identical with **spoVK**."*

There are 2 genes in this sentence:

ID	Start	End	Name
T1	17	22	spoVJ
T2	104	109	spoVK

Table 1: Example of provided data.

There is also a renaming act: **R1** Renaming Former:T1 New:T2

Given all gene positions and identifications (Tn), the Rename task consists in predicting all renaming acts (Rn) between *Bacillus subtilis* genes in multi-sentence documents. The gene names involved are all acronyms or short names. Gene and protein names often have both a short and a long form. Linking short to long names is a relatively well-known task but linking short names together remains little explored (Yu et al., 2002). Moreover, specifying some of these synonymy relations as renaming appears quite rare (Weissenbacher, 2004). This task relates to the more general search of relations of

Proceedings of BioNLP Shared Task 2011 Workshop, pages 121–129,
Portland, Oregon, USA, 24 June, 2011. ©2011 Association for Computational Linguistics

synonymous nicknames, aliases or pseudonyms of proper nouns from definitory contexts in encyclopedia or dictionaries. For instance, in *Alexander III of Macedonia commonly known as Alexander the Great* the synonymy relation is supported by *commonly known as* between the proper noun *Alexander III of Macedonia* and the nickname *Alexander the Great*. Renaming act extraction differs from the search of coreferences or acronyms by the linguistic markers involved.

1.2 Datasets

The renaming corpus is a set of 1,648 PubMed references of bacterial genetics and genome studies. The references include the title and the abstract. The annotations provided are: the position and name of genes (see Table 1) for all sets and the renaming acts in the training and the development sets only.

	Train	**Dev.**	**Test**
Documents	1146	246	252
Genes	14372	3331	3375
Unique Genes	3415	1017	1126
New genes	0	480	73
Relations	308	65	88
Words / Doc	209	212	213
Genes / Doc	12.5	12.7	13.4
Unique Genes / Doc	3.0	4.1	4.5
Relations / Doc	0.27	0.26	0.35

Table 2: Datasets of the Rename task corpus.

2 Methods

An early finding is that renaming acts very seldom span several sentences (i.e. *former* and *new* are in the same sentence). For the training set, 95.4% of the relations verify this claim and in the development set, 96.1%. Thus, it is decided to first segment the documents into sentences and then to look for renaming acts inside independent sentences. Thus the maximum expected recall is then 96.1% on the development set. This is done by automatically filtering all the sentences out that do not contain evidence of a renaming act and then to relate the gene names occurring in the renaming sentences. The AlvisNLP pipeline (Nédellec et al., 2009) is used throughout this process (see Fig. 1).

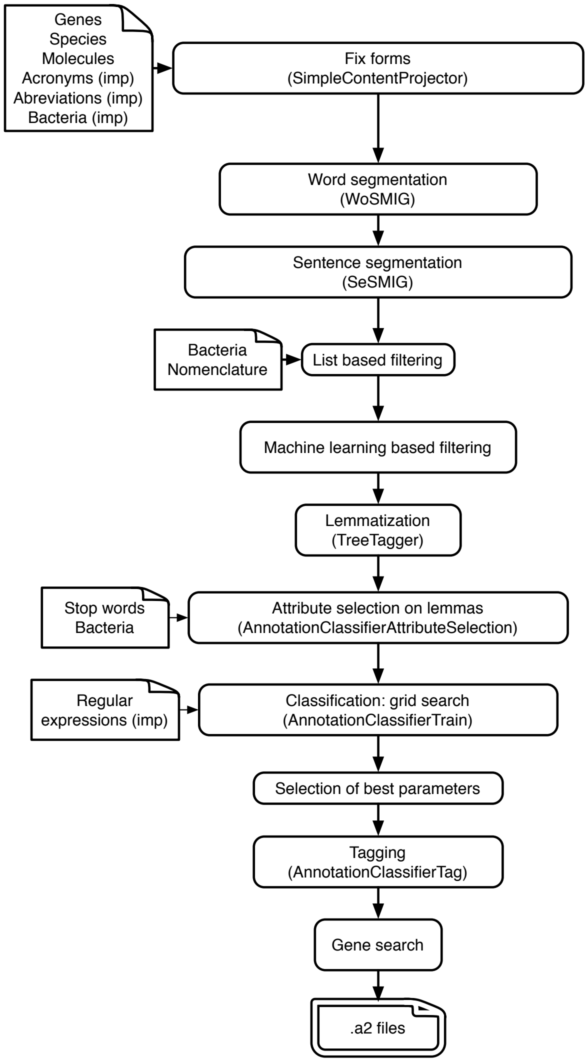

Figure 1: Flowchart: Notes represent the resources used and (imp) represent later improvements not used for the official submission.

2.1 Word and sentence segmentation

Word and sentence segmentation is achieved by the Alvis NLP pipeline. Named entity recognition supplements general segmentation rules.

2.1.1 Derivation of boundaries from named entities

Named entities often contains periods that should not be confused with sentence ends. Species abbreviations with periods are specially frequent in the task corpus. First, dictionaries of relevant named entities from the molecular biology domain (e.g.

genes, species and molecules) are projected onto the documents before sentence segmentation, so that periods that are part of named entities are disambiguated and not interpreted as sentence ends. Moreover, named enties are frequently multi-word. Named entity recognition prior to segmentation prevents irrelevant word segmentation. For example, the projection of named entity dictionaries on the excerpt below reveals the framed multi-word entities:
"Antraformin, a new inhibitor of Bacillus subtilis transformation. [...] *During this screening program, Streptomyces sp. 7725-CC1 was found to produce a specific inhibitor of B. subtilis transformation."*

2.1.2 Word segmenter

The word segmenter (WosMIG in Fig. 1) has the following properties: 1) primary separator: space, 2) punctuation isolation: customized list, 3) custom rules for balanced punctuation, 4) fixed words: not splittable segments The following list of terms is obtained from the example:
['Antraformin' , ',', 'a', 'new', 'inhibitor', 'of', 'Bacillus subtilis', 'transformation', '.', [...], 'During', 'this', 'screening', 'program', ',', 'Streptomyces sp.', '7725-CC1', 'was', 'found', 'to', 'produce', 'a', 'specific', 'inhibitor', 'of', 'B. subtilis', 'transformation', '.']

2.1.3 Sentence segmenter

The sentence segmenter (SeSMIG in Fig. 1) has the following properties: 1) strong punctuation: customized list; 2) tokens forcing the end of a sentence (e.g. *etc...*); 3) an upper case letter must follow the end of a sentence. The system works very well but could be improved with supervised machine learning to improve the detection of multi-word named entities. Finally, the list of words is split into sentences:
[['Antraformin' , ',', 'a', 'new', 'inhibitor', 'of', 'Bacillus subtilis', 'transformation', '.'],
[...],
['During', 'this', 'screening', 'program', ',', 'Streptomyces sp.', '7725-CC1', 'was', 'found', 'to', 'produce', 'a', 'specific', 'inhibitor', 'of', 'B. subtilis', 'transformation', '.']]

2.2 Sentence filtering

Once the corpus is segmented into sentences, the system filters out the numerous sentences that most likely do not contain any renaming act. This way, the further relation identification step focuses on relevant sentences and increases the precision of the results (Nedellec et al., 2001). Before the filtering, the recall is maximum (not 100% due to few renaming acts spanning two sentences), but the precision is very low. The sentence filters aim at keeping the recall as high as possible while gradually increasing the precision. It is composed of two filters. The first filter makes use of an a priori knowledge in the form of a nomenclature of known synonyms while the second filter uses machine learning to filter the remaining sentences. In the following, the term Bacillus subtilis gene nomenclature is used in the sense of an exhaustive inventory of names for Bacillus subtilis genes.

2.2.1 Filtering with a gene nomenclature

We developed a tool for automatically building a nomenclature of *Bacillus subtilis* gene and protein names. It aggregates the data from various gene databases with the aim of producing the most exhaustive nomenclature. The result is then used to search for pairs of synonyms in the documents. Among various information on biological sequences or functions, the entries of gene databases record the identifiers of the genes and proteins as asserted by the biologist community of the species. *Bacillus subtilis* community as opposed to other species has no nomenclature committee. Each database curator records unilateral naming decisions that may not reported elsewhere. The design of an exhaustive nomenclature require the aggregation of multiple sources.

Databases Our sources for the *Bacillus subtilis* nomenclature are six publicly available databases plus an in-house database. The public databases are generalist (1 to 3) or devoted to *Bacillus subtilis* genome (4 to 6) (see Table 3):

GenBank The genetic sequence database managed by the National Center for Biotechnology Information (NCBI) (Benson et al., 2008). It contains the three official versions of the annotated genome of *B. subtilis* with all gene canonical names;

UniProt the protein sequence database managed by the Swiss Institute of Bioinformatics (SIB), the European Bioinformatics Institute (EBI) and the Protein Information Resource (PIR) (Bairoch et al., 2005). It contains manual annotated protein sequences (Swiss-Prot) and automatically annotated protein sequences (TrEMBL (Bairoch and Apweiler, 1996)). Its policy is to conserve a history of all information relative to these sequences and in particular all names of the genes that code for these sequences.

Genome Reviews The genome database managed by the European Bioinformatics Institute (EBI) (Sterk et al., 2006). It contains the re-annotated versions of the two first official versions of the annotated genome of *B. subtilis*;

BSORF The Japanese *Bacillus subtilis* genome database (Ogiwara et al., 1996);

Genetic map the original genetic map of *Bacillus subtilis*;

GenoList A multi-genome database managed by the Institut Pasteur (Lechat et al., 2008). It contains an updated version of the last official version of the annotated genome of *B. subtilis*;

SubtiWiki A wiki managed by the Institute for Microbiology and Genetics in Göttingen (Flórez et al., 2009) for *Bacillus subtilis* reannotation. It is a free collaborative resource for the *Bacillus* community;

EA_List a local lexicon manually designed from papers curation by Anne Goelzer and Élodie Marchadier (MIG/INRA) for Systems Biology modeling (Goelzer et al., 2008).

Nomenclature merging We developed a tool for periodically dumping the content of the seven source databases through Web access. With respect to gene naming the entries of all the databases contain the same type of data per gene:

- a unique identifier (required);
- a canonical name, which is the currently recommended name (required);
- a list of synonyms considered as deprecated names (optional).

The seven databases are handled one after the other. The merging process follows the rules:

- the dump of the first database (SubtiWiki, see Table 3 for order) in the list is considered the most up-to-date and is used as the reference for the integration of the dumps of the other databases;
- for all next dumps, if the unique gene identifier is new, the whole entry is considered as new and the naming data of the entry is added to the current merge;
- else, if the unique identifier is already present into the merge, the associated gene names are compared to the names of the merge. If the name does not exist in the merge, it is added to the merge as a new name for this identifier and synonym of the current names. The synonym class is not ordered.

Order	Databases	AE	AN
1	SubtiWiki	4 261	5920
2	GenoList	0	264
3	EA_List	33	378
4	BSORF	0	42
5	UniProt	0	74
6	Genome Reviews	0	0
7	GenBank	0	7
8	Genetic Map	0	978
	Total	4 294	7 663

Table 3: Database figures. AE: number of added entries, AN: number of added names.

Synonym pair dictionary: The aggregated nomenclature is used to produce a dictionary of all combinations of pairs in the synonym classes.

Sentence filtering by gene cooccurrence: For each sentence in the corpus, if a pair of gene synonyms according to the lexicon is found inside then the sentence is kept for the next stage. Otherwise, it is definitively discarded. The comparison is a case-insensitive exact match preserving non alphanumeric symbols. The recall at this step is respectively 90.9% and 90.2% on the train and development sets. The recall loss is due to typographic errors in gene names in the nomenclature. The precision at this stage is respectively 38.9% and 38.1% on the train and development sets. There are still many false positives due to gene homologies or renaming acts concerning other species than *Bacillus subtilis* for instance.

2.2.2 Sentence filtering by SVM

Feature selection The second filtering step aims at improving the precision by machine learning classification of the remaining sentences after the first filtering step. Feature selection is applied to enhance the performances of the SVM as it is shown to suffer from high dimensionality (Weston et al., 2001). Feature selection is applied to a bag-of-word representation using the Information Gain metrics of the Weka library (Hall et al., 2009). Words are lemmatized by TreeTagger (Schmid, 1994). A manual inspection of the resulting sorting highly ranks words such as *formerly* or *rename* and parentheses while ranking other words such as *cold* or *encode* surprisingly certainly due to over-fitting. Although the feature selection is indeed not particularly efficient compared to the manual selection of relevant features but does help filtering out unhelpful words and then drastically reducing the space dimension from 1919 to 141 for the best run.

Sentence classification and grid search: A SVM algorithm (LibSVM) with a RBF kernel is applied to the sentences encoded as bag of words. The two classes are: "contains a renaming act" (True) or not (False). There are 4 parameters to tune: 1) the number of features to use ($N \in 1, 5, 10, ..., 150$) meaning the N first words according to the feature selection, 2) the weight of the classes: True is fixed to 1 and False is tuned ($W \in 0.2, 0.4, ..., 5.0$), 3) the errors weight ($C \in 2^{-5,-7,...,9}$), 4) the variance of the Gaussian kernel ($G \in 2^{-11,-9,...,1}$). Thus, to find the best combination of parameters for this problem, $\#N * \#W * \#C * \#G = 31 * 25 * 8 * 7 = 43,400$ models are trained using 10-fold cross-validation on the training and development sets together (given the relatively small size of the training set) and ranked by F-measure. This step is mandatory because the tuning of C and G alone yield variations of F-measure from 0 to the maximum. The grid search is run on a cluster of 165 processors and takes around 30 minutes. The best model is the model with the highest F-measure found by the grid search.

Test sentence filtering: Finally the test set is submitted to word and sentence segmentation, feature filtering and tagged by the best SVM model (AnnotationClassifierTag in Fig. 1). The sentences that are assumed to contain a renaming act are kept and the others are discarded (see Fig. 2).

2.3 Gene position searching

At this step, all remaining sentences are assumed to be true positives. They all contain at least one pair of genes that are synonymous according to our gene nomenclature. The other gene names are not considered. The method for relating gene candidates by a renaming relation, relies on the assumption that all gene names are involved in at least one relation. Most of the time, sentences contain only two genes. We assume in this case that they are related by a renaming act. When there are more than two genes in a sentence, the following algorithm is applied: 1) compute all combinations of couples of genes; 2) look-up the lexicon for those couples and discard those that are not present; 3) if a given gene in a couple has multiple occurrences, take the nearest instance from the other gene involved in the renaming act.

3 Discussion

The system ranks 3rd/3 among three participants in the Rename task official evaluation with a F-measure of 64.4% (see Fig. 4), five points behind the second. The general approach we used for this task is pragmatic: 1) simplify the problem by focusing on sentences instead of whole documents for a minimal loss, 2) then use a series of filters to improve the precision of the sentence classification while keeping the recall to its maximum, 3) and finally relate gene names known to be synonymous inside sentences for a minimal loss (around 2% of measure). As opposed to what is observed in Gene Normalization tasks (Hirschman et al., 2005), the Rename task is characterised by the lack of morphological resemblance of gene synonyms. The gene synonyms are not typographic variants and the recognition of renaming act requires gene context analysis. The clear bottleneck of our system is the sentence filtering part and in particular the feature selection that brings a lot of noise by ranking statistically spurious terms. On the plus side, the whole system is fully automated to the exception of the resources used for the word segmentation that were designed manually for other tasks. Moreover, our strategy does not assume that the gene pairs from the nomenclature may be mentioned for other reasons than renaming, it then tends to overgeneralize. However, many occurrences of the predicted gene pairs are not involved in renaming acts because the reasons for mentioning synonyms may be different than renaming. In particular, equivalent genes of other species (orthologues) with high sequence similarities may have the same name as in *Bacillus subtilis*. An obvious improvement of our method would consists in first relating the genes to their actual species before relating the only *Bacillus subtilis* gene synonyms by the renaming relation.

Team	Pre.	Rec.	F-M.
U. of Turku	95.9	79.6	87.0
Concordia U.	74.4	65.9	69.9
INRA	**57.0**	**73.9**	**64.4**

Table 4: Official scores in percentage on the test set.

3.1 Method improvement by IE patterns

After the official submission and given the result of our system compared to competitors, a simple modification of the feature selection was tested with significant benefits: the addition of regular expressions as additional features. Intuitively there are words or patterns that strongly appeal to the reader as important markers of renaming acts. For example, variations of *rename* or adverbs such *originally* or *formerly* would certainly be reasonable candidates. Fifteen such shallow patterns were designed (see Table 5) supplemented by six more complex ones, originally designed to single out gene names. In appendix A, one of them is presented, the precision of which is 95.3% and recall 27.5%. That is, more than a quarter of renaming acts in the training and development sets together. Interestingly, in table 5 the word *formerly* (3rd in feature selection ranking) alone recalls 10.7% of the renaming acts with a precision of 96.9%. In contrast, the words *originally* and *reannotated* although having 100% precision are respectively ranked 33rd and 777th. In total, 21 patterns are represented as boolean features of the classification step in addition to the ones selected by feature selection. Unsurprisingly, the best classifiers, according to the cross-validation F-measure after the grid search, only used the regular expressions as features neglecting the terms chosen by feature selection. A significant improvement is achieved: +8.7% of F-measure on the test set (see Fig. 2).

Pattern	Pre.	Rec.	F-M.
(reannotated)	100.0	0.4	0.7
(also called)	100.0	0.4	0.7
(formerly)	96.9	10.7	19.2
(originally)	100.0	1.4	2.8
((also)? known as)	100.0	1.8	3.4
(were termed)	100.0	0.4	0.7
(identity of)	100.0	0.7	1.4
(be referred (to|as)?)	100.0	0.4	0.7
(new designation)	100.0	0.4	0.7
(allel\w+)	80.0	2.8	5.4
(split into)	100.0	0.4	0.7
(rename)	83.4	1.8	3.4
(renamed)	88.5	8.0	14.6
(renaming)	100.0	0.4	0.7
(E(\.|scherichia) coli)	11.3	4.5	6.4

Table 5: Handwritten patterns. Scores are in percentage on the training and development sets together **after** the gene nomenclature filtering step. A very low precision means the pattern could be used to filter out rather than in.

3.2 Error analysis

The false positive errors of the sentence filtering step, using hand-written patterns can be classified as follows: 1) omission: *Characterization of **abn2** (**yxiA**), encoding a Bacillus subtilis GH43 arabinanase, Abn2, and its role in arabino-*

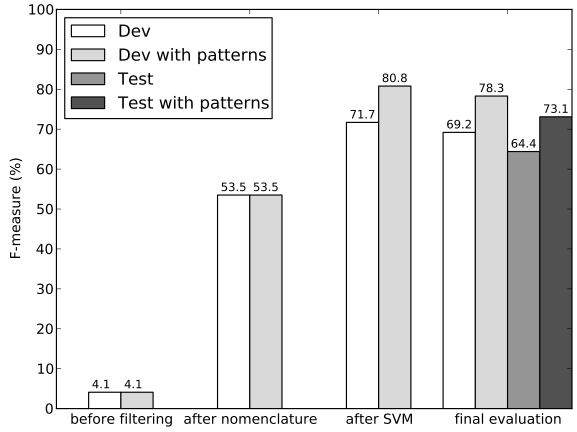

Figure 2: Evolution of F-measure at different measure points for the Rename task. Dev: training on train set and testing on dev set. Test: training on train + dev sets and testing on test set (no intermediary measure). 64.4% is the official submitted score. 73.1% is the best score achieved by the system on the test set.

polysaccharide degradation. (PMID 18408032). In this case the sentence has been filtered out by the SVM and then the couple **abn2/yxia** was not annotated as a renaming act, 2) incorrect information in the nomenclature: *These results substantiate the view that **sigE** is the distal member of a 2-gene operon and demonstrate that the upstream gene (**spoIIGA**) is necessary for sigma E formation.* (PMID 2448286). Here, the integration of the Genetic Map to the nomenclature has introduced a wrong synonymy relation between **spoIIGA** and **sigE**, 3) homology with another species: *We report the cloning of the wild-type allele of **divIVB1** and show that the mutation lies within a stretch of DNA containing two open reading frames whose predicted products are in part homologous to the products of the Escherichia coli minicell genes **minC** and minD.* (PMID 1400224). The name pair actually exists in the nomenclature but here, **divIVB1** is a gene of *B. subtilis* and **minC** is a gene of *E. Coli*, 4) another problem linked to the lexicon is the fact the synonym classes are not disjoint. Some deprecated names of given genes are reused as canonical names of other genes. For example, **purF** and **purB** referred to two different genes of *B. subtilis* but **purB** was also formerly known as **purF**: *The following gene order has been established: pbuG-**purB**-**purF**-purM-purH-purD-tre* (PMID 3125411). Hence, **purF** and **purB** are uncorrectly recognized as synonyms while they refer to two different genes in this context. Possible solutions for improving the system could be: 1) the inclusion of species names as SVM features, 2) the removal of some couples from the nomenclature (**PurF/purB** for instance), 3) evaluate the benefits of each resource part of the nomenclature.

4 Conclusion

Our system detects renaming acts of *Bacillus subtilis* genes with a final F-measure of 64.4%. After sentence segmentation, the emphasis is on sentence filtering using an exhaustive nomenclature and a SVM. An amelioration of this method using patterns as features of the machine learning algorithm was shown to improve significantly (+8.7%) the final performance. It was also shown that the bag of words representation is sub-optimal for text classification experiments (Fagan, 1987; Caropreso and Matwin, 2006) With the use of such patterns, the filtering step is now very efficient. The examination of the remaining errors showed the limits of the current shallow system. A deeper linguistic approach using syntactic parsing seems indicated to improve the filtering step further.

Acknowledgments

The authors would like to thank Julien Jourde for granting them the permission to use the *Bacteria subtilis* synonym nomenclature that he is currently building and Philippe Veber for his insightful advices on text classification. This research is partly founded by the French Oseo QUAERO project.

References

A. Bairoch and R. Apweiler. 1996. The SWISS-PROT protein sequence data bank and its new supplement TREMBL. *Nucleic Acids Research*, 24(1):21.

A. Bairoch, R. Apweiler, C.H. Wu, W.C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, and Others. 2005. The universal protein resource (UniProt). *Nucleic Acids Research*, 33(suppl 1):D154.

D.A. Benson, I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, and D.L. Wheeler. 2008. GenBank. *Nucleic acids research*, 36(suppl 1):D25.

M. Caropreso and S. Matwin. 2006. Beyond the Bag of Words: A Text Representation for Sentence Selection. *Advances in Artificial Intelligence*, pages 324–335.

J.L. Fagan. 1987. Experiments in automatic phrase indexing for document retrieval: a comparison of syntactic and non-syntactic methods.

LA Flórez, SF Roppel, A.G. Schmeisky, C.R. Lammers, and J. Stülke. 2009. A community-curated consensual annotation that is continuously updated: the Bacillus subtilis centred wiki SubtiWiki. *Database: The Journal of Biological Databases and Curation*, 2009.

A Goelzer, B Brikci, I Martin-Verstraete, P Noirot, P Bessières, S Aymerich, and V Fromion. 2008. Reconstruction and analysis of the genetic and metabolic regulatory networks of the central metabolism of Bacillus subtilis. *BMC systems biology*, 2(1):20.

M Hall, E Frank, G Holmes, B Pfahringer, P Reutemann, and I H Witten. 2009. The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

Lynette Hirschman, Alexander Yeh, Christian Blaschke, and Alfonso Valencia. 2005. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC bioinformatics*, 6 Suppl 1:S1, January.

P. Lechat, L. Hummel, S. Rousseau, and I. Moszer. 2008. GenoList: an integrated environment for comparative analysis of microbial genomes. *Nucleic Acids Research*, 36(suppl 1):D469.

C. Nedellec, M. Abdel Vetah, and Philippe Bessières. 2001. Sentence filtering for information extraction in genomics, a classification problem. *Principles of Data Mining and Knowledge Discovery*, pages 326–337.

C Nédellec, A Nazarenko, and R Bossy. 2009. Information Extraction. *Handbook on Ontologies*, pages 663–685.

K E Nelson, I T Paulsen, J F Heidelberg, and C M Fraser. 2000. Status of genome projects for nonpathogenic bacteria and archaea. *Nature biotechnology*, 18(10):1049–54, October.

A. Ogiwara, N. Ogasawara, M. Watanabe, and T. Takagi. 1996. Construction of the Bacillus subtilis ORF database (BSORF DB). *Genome Informatics*, pages 228–229.

H Schmid. 1994. Probabilistic part-of-speech tagging using decision trees.

P. Sterk, P.J. Kersey, and R. Apweiler. 2006. Genome reviews: standardizing content and representation of information about complete genomes. *Omics: a journal of integrative biology*, 10(2):114–118.

Davy Weissenbacher. 2004. La relation de synonymie en Génomique. *RECITAL*.

J. Weston, S. Mukherjee, O Chapelle, M. Pontil, T. Poggio, and V. Vapnik. 2001. Feature selection for SVMs. *Advances in neural information processing systems*, pages 668–674.

Hong Yu, Vasileios Hatzivassiloglou, Carol Friedman, Andrey Rzhetsky, and W.J. Wilbur. 2002. Automatic extraction of gene and protein synonyms from MEDLINE and journal articles. In *Proceedings of the AMIA Symposium*, page 919. American Medical Informatics Association.

A Gene or operon couple matching pattern

Pattern that uses bacteria gene naming rules (3 lower case + 1 upper case letters), short genes (3 lower case letters), long gene names, factorized operons (3 lower case + several upper case letters), gene names including special and/or numerical characters in presence or not of signal words such as *named*, *renamed*, *formerly*, *formally*, *here*, *herein*, *hereafter*, *now*, *previously*, *as*, *designated*, *termed* and/or *called*, only if the pattern does not begin with *and* or *orf*. Although this pattern could be used to directly filter in sentences containing a renaming act, its recall is too low thus it is used as a feature of the classifier instead.

```
and|orf\
GENE|OPERON-fact\
[|((now|as|previously|formerly|formally|here(in|after))\
((re)named|called|designated|termed) (now|as|previously|formerly|formally|here(in|after))\
GENE|OPERON-fact)|]
```

Table 6: Long pattern used for gene pair matching.

Terms matched	Pattern	PMID
short-GENE (short-GENE)	***cotA*** *(formerly* ***pig****)*	8759849
long-GENE (long-GENE)	***cotSA*** *(****ytxN****)*	10234840
fact-OPERON (fact-OPERON)	***ntdABC*** *(formally* ***yhjLKJ****)*	14612444
spe-GENE (spe-GENE)	***lpa-8*** *(****sfp****)*	10471562
GENE (GENE)	***cwlB*** [***lytC***]	8759849
GENE (now designated GENE)	***yfiA*** *(now designated* ***glvR****)*	11489864
GENE (previously GENE)	***nhaC*** *(previously* ***yheL****)*	11274110
GENE (formerly called GENE)	***bkdR*** *(formerly called* ***yqiR****)*	10094682
GENE (now termed GENE)	***yqgR*** *(now termed* ***glcK****)*	9620975
GENE (GENE) other forms	***fosB****(****yndN****)*	11244082
GENE (hereafter renamed GENE)	***yhdQ*** *(hereafter renamed* ***cueR****)*	14663075
GENE (herein renamed GENE)	***yqhN*** *(herein renamed* ***mntR****)*	10760146
GENE (formally GENE)	***ntdR*** *(formally* ***yhjM****)*	14612444
GENE (formerly GENE)	***mtnK*** *(formerly* ***ykrT****)*	11545674
GENE (renamed GENE)	***yfjS*** *(renamed* ***pdaA****)*	12374835
GENE (named GENE)	***yvcE*** *(named* ***cwlO****)*	16233686
GENE (GENE)	***pdaA*** *(****yfjS****)*	14679227

Table 7: Examples matched with the long pattern.