Predictive modeling: guessing the NLP terms of tomorrow

Gil Francopoulo [1], Joseph Mariani [2], Patrick Paroubek [2]
1 LIMSI, CNRS, Université Paris-Saclay + Tagmatica (France)
2 LIMSI, CNRS, Université Paris-Saclay (France)

gil.francopoulo@wanadoo.fr, joseph.mariani@limsi.fr, pap@limsi.fr

Abstract

Predictive modeling, often called "predictive analytics" in a commercial context, encompasses a variety of statistical techniques that analyze historical and present facts to make predictions about unknown events. Often the unknown events are in the future, but prediction can be applied to any type of unknown whether it be in the past or future. In our case, we present some experiments applying predictive modeling to the usage of technical terms within the NLP domain.

Keywords: Predictive Modeling, Predictive Analytics, Term Extraction, Natural Language Processing

1. Introduction

Predictive modeling, often called "predictive analytics" in a commercial context, encompasses a variety of statistical techniques that analyze historical and present facts to make predictions about unknown events. Often the unknown events are in the future, but prediction can be applied to any type of unknown whether it be in the past or future. In our case, we present some experiments applying predictive modeling to the usage of technical terms within the NLP domain.

2. Context

Our work comes after the various studies initiated in the Workshop entitled: "Rediscovering 50 Years of Discoveries in Natural Language Processing" on the occasion of ACL's 50th anniversary in 2012 [Radev et al 2013] where a group of researchers studied the content of the corpus recorded in the ACL Anthology [Bird et al 2008]. Different studies were presented from reuse detection [Gupta et al 2012] to topic detection [Anderson et al 2012].

3. Corpus

Our research began by gathering a large corpus of NLP scientific articles covering documents produced from 1965 up to 2015. This corpus gathers a large content of our own research field, i.e. NLP, covering both written and spoken sub-domains and extended to a limited number of corpora, for which Information Retrieval and NLP activities intersect. This corpus was collected at LIMSI-CNRS (France) and is named NLP4NLP [Francopoulo et al 2015]. It contains currently 65,003 documents coming from various conferences and journals with either public or restricted access. This represents a large part of the existing published articles in our field, aside from the workshop proceedings and the published books. It should be noted, that most other studies in our domain are based on the ACL Anthology[1] which is dedicated to text processing, but as LIMSI and IMMI laboratories work both in the written and spoken language processing domains, we chose to address both. The ACL Anthology (although extremely valuable) is only approximately one third of our corpus, the majority of the other papers coming from the ISCA [2] and IEEE [3] archives. The ACL Anthology and ISCA archives are in open access, see the details of the 34 sub-corpora in table 1. Let's note that for a joint conference (which is a rather infrequent situation), the paper is counted once in each row within the table. So the sum of all cells is slightly more important than the total number of papers and venues.

4. Preprocessing

Our processes need two elements for each paper: the metadata and the content. The metadata are not obtained from the texts but from the BibTex record or the conference programs (see [Francopoulo et al 2015] for a justification). The metadata record comprises the corpus name, year, title and authors. The content is in PDF format and comprises the abstract, text body and reference section. We use Apache PDFBox[4] to identify the content type, i.e. whether the file is a sequence of images or an extractable text. For images, we use the Tesseract OCR[5] to produce the text material. For an extractable text, we reran PDFBox to produce the text material. In order to track difficult situations coming from bad PDF files whose extraction gives rubbish without breaking the PDFBox API, we adopted the strategy of computing a quality level for each paper. This quality is defined as the number of known words divided by the number of words. For this purpose, we use TagParser which is an industrial NLP pipeline (www.tagmatica.com). The motivation for using TagParser was that it is well-known to us, and rapidly usable. The TagParser pipeline [Francopoulo 2007] is used to compute the number of known words combining a morphological analysis with an LMF[6] formatted broad coverage lexicon. After a manual study of 500 "borderline" documents, a quality threshold of 91% has been experimentally set. Thus, a text whose quality is below 91% is ignored. Initially, the texts are in four languages: English, French, German and Russian. The number of texts in German and Russian is less than 0.5%. They are detected automatically and are ignored. The texts in French are a little bit more numerous (3%), so they are kept with the same status as the English ones. This is not a problem as our tool is able to process English and French. The content is rather clean, the remaining noise

[1] http://aclweb.org/anthology

[2] http://www.isca-speech.org/iscaweb

[3] http://www.signalprocessingsociety.org

[4] https://pdfbox.apache.org

[5] https://code.google.com/p/tesseract-ocr

[6] https://en.wikipedia.org/wiki/Lexical_Markup_Framework

being table contents, formula, variables and non English linguistic examples. See [Francopoulo et al 2015] for more details about the preprocessing as well as the solutions for some tricky problems like joint conferences management or abstract / body / reference sections detection.

short name	# docs	format	long name	language	access to content	period	# venues
acl	4264	conference	Association for Computational Linguistics Conference	English	open access *	1979-2015	37
acmtslp	82	journal	ACM Transaction on Speech and Language Processing	English	private access	2004-2013	10
alta	262	conference	Australasian Language Technology Association	English	open access *	2003-2014	12
anlp	278	conference	Applied Natural Language Processing	English	open access *	1983-2000	6
cath	932	journal	Computers and the Humanities	English	private access	1966-2004	39
cl	776	journal	American Journal of Computational Linguistics	English	open access *	1980-2014	35
coling	3813	conference	Conference on Computational Linguistics	English	open access *	1965-2014	21
conll	842	conference	Computational Natural Language Learning	English	open access *	1997-2015	18
csal	762	journal	Computer Speech and Language	English	private access	1986-2015	29
eacl	900	conference	European Chapter of the ACL	English	open access *	1983-2014	14
emnlp	2020	conference	Empirical methods in natural language processing	English	open access *	1996-2015	20
hlt	2219	conference	Human Language Technology	English	open access *	1986-2015	19
icassps	9819	conference	IEEE International Conference on Acoustics, Speech and Signal Processing - Speech Track	English	private access	1990-2015	26
ijcnlp	1188	conference	International Joint Conference on NLP	English	open access *	2005-2015	6
inlg	227	conference	International Conference on Natural Language Generation	English	open access *	1996-2014	7
isca	18369	conference	International Speech Communication Association	English	open access	1987-2015	28
jep	507	conference	Journées d'Etudes sur la Parole	French	open access *	2002-2014	5
lre	308	journal	Language Resources and Evaluation	English	private access	2005-2015	11
lrec	4552	conference	Language Resources and Evaluation Conference	English	open access *	1998-2014	9
ltc	656	conference	Language and Technology Conference	English	private access	1995-2015	7
modulad	232	journal	Le Monde des Utilisateurs de L'Analyse des Données	French	open access	1988-2010	23
mts	796	conference	Machine Translation Summit	English	open access	1987-2015	15
muc	149	conference	Message Understanding Conference	English	open access *	1991-1998	5
naacl	1186	conference	North American Chapter of the ACL	English	open access *	2000-2015	11
paclic	1040	conference	Pacific Asia Conference on Language, Information and Computation	English	open access *	1995-2014	19
ranlp	363	conference	Recent Advances in Natural Language Processing	English	open access *	2009-2013	3
sem	950	conference	Lexical and Computational Semantics / Semantic Evaluation	English	open access *	2001-2015	8
speechc	593	journal	Speech Communication	English	private access	1982-2015	34
tacl	92	journal	Transactions of the Association for Computational Linguistics	English	open access *	2013-2015	3
tal	177	journal	Revue Traitement Automatique du Langage	French	open access	2006-2015	10
taln	1019	conference	Traitement Automatique du Langage Naturel	French	open access *	1997-2015	19
taslp	6612	journal	IEEE/ACM Transactions on Audio, Speech and Language Processing	English	private access	1975-2015	41
tipster	105	conference	Tipster DARPA text program	English	open access *	1993-1998	3
trec	1847	conference	Text Retrieval Conference	English	open access	1992-2015	24
Total	67937					1965-2015	558

Table 1 Detail of NLP4NLP, with the convention that an asterisk indicates that the corpus is in ACL Anthology

5. Term extraction

The aim is to extract the domain terms from the abstract and bodies of the texts. We follow the approach called "contrastive strategy" in the same line as TermoStat [Drouin 2004]. **The main idea is to reject words or sequence of words of the non-specialized (or "ordinary") language which are considered as not interesting, and to retain the remaining terms which are considered as the domain terms.** To this end, one large non specialized corpus[7] was parsed with TagParser and the results were filtered with fifteen syntactic patterns (like N of N), excluding names of authors, and finally a large statistical matrix was recorded. Afterwards, we proceeded in three steps: first, we made a manual detection of noise upon the 2,000 most frequent words in order to eliminate non semantic words such as "Cj" which is a mathematical variable and not a term of the domain. We found 180 words which were recorded manually in a stop-list. Secondly, we studied the remaining frequent terms in order to manually merge a small amount of synonyms (25) which were not in the parser dictionary. Thirdly, we reran the system.

Concerning the results, 20% of the extracted terms are single terms, the rest being multi-word expressions, but, as shown in Table 2, the single terms are frequently the abbreviation of a multi-word expression. The number of (different) extracted terms is 5.1M and the number of occurrences of these terms is 400M. Because we will run complex computations, we cannot consider the 5.1M terms: we took the 200 most frequent terms of the collection. Among these terms, the 20 most used terms are presented in table 2.

The pros and cons of the contrastive strategy have already been studied, especially with respect to the specific level of the term. Other approaches like the one implemented in Saffron are oriented towards the construction of a domain model based on internal domain coherence [Bordea et al 2013] and are more focused on discovering intermediate or generic terms. Our strategy favors the leaves of the hierarchy, and is less sensitive to generic terms that can be used in other domains as these terms may be encountered in the non-specialized corpus. Our objective being to study the relation of the specialized terms with respect to the accurate time line, the contrastive strategy is more adequate.

6. Building time-series on past events

The core of predictive modeling relies on capturing relationships between some known explanatory variables and some unknown predicted variables. In our context, the explanatory variables are frequencies of the domain specific terms from the past events whose position in the time-line is important and such data are called "time-series". Each instance of a past event represents a different time step and the attributes give values associated with that time step, in our case term frequencies. It should be added that in other applications than ours, time-series could be difficult to manage with respect to periodicity and irregular time samples which need to be converted to comparable time stamps. But these difficulties do not apply to our computation because we do not make the hypothesis that there is any periodicity and our time intervals are of equal size, namely one year each.

Headword	Variants of all sorts : inflections, synonyms and case variants	Occurrences#	Rank
HMM	HMMs, Hidden Markov Model, Hidden Markov Models, Hidden Markov model, Hidden Markov models, hidden Markov Model, hidden Markov Models, hidden Markov model, hidden Markov models	1941666	1
SR	ASR, ASRs, Automatic Speech Recognition, SRs, Speech Recognition, automatic speech recognition, speech recognition	1905633	2
NP	NPs, noun phrase, noun phrases	1889393	3
LM	LMs, Language Model, Language Models, language model, language models	1849106	4
POS	POSs, Part Of Speech, Part of Speech, Part-Of-Speech, Part-of-Speech, Parts Of Speech, Parts of Speech, Pos, part of speech, part-of-speech, parts of speech, parts-of-speech	1845879	5
parser	parsers	1758609	6
annotation	annotations	1697676	7
classifier	classifiers	1637323	8
segmentation	segmentations	1176050	9
dataset	data-set, data-sets, datasets	1101115	10
parsing	parsings	1081910	11
MT	MTs, Machine Translation, Machine Translations, machine translation, machine translations	958254	12
neural network	ANN, ANNs, Artificial Neural Network, Artificial Neural Networks, NN, NNs, Neural Network, Neural Networks, NeuralNet, NeuralNets, neural networks	861226	13
predicate	predicates	850768	14
ngram	ngrams	836350	15
metric	metrics	824732	16
SVM	SVMs, Support Vector Machine, Support Vector Machines, support vector machine, support vector machines	806432	17
GMM	GMMs, Gaussian Mixture Model, Gaussian Mixture Models, Gaussian mixture model, Gaussian mixture models	800952	18
iteration	iterations	755354	19
SNR	SNRs, Signal Noise Ratio, Signal Noise Ratios, signal noise ratio, signal noise ratios	744811	20

Table 2 Most frequent English terms in the collection

[7] The "ordinary" corpus is made of the British National Corpus, the Open American National Corpus, the Suzanne corpus release-5 and the English EuroParl archives (years 1999 until 2009) totalizing 200M words.

7. Evaluation

We faced three questions: the first is how to choose the right algorithm. The second question deals with the size of the Past (i.e. the number of years) we need to take into account, starting backwards from today. We build the time-series with records dating from the 60's, but the old documents are not very numerous and are of bad quality, due to the fact that most of them are PDF files with scanned material. Another point deals with the relevance of using 50-year-old documents to predict semantic clues within a technical domain which changed drastically over the last decades. The third question is how to evaluate the prediction. These three questions are not independent: if we set an evaluation benchmark, we will then be able to compare the various algorithms with a certain size of the Past and finally to take the best parameters for our needs.

Obviously, aside from waiting until next year, a prediction is hard to verify. But, it is possible, to make an evaluation of the past events, with the hypothesis that an algorithm which was proved to be good in the past will be a good one in the future. Let's recall that we know the frequencies of each term from the 60's until the current year (i.e. 2015, as the present article was first submitted in 2015). The benchmark is as follows: as a first step, we willingly restrict our knowledge to the events from the 60's until last year (i.e. 2014 included). As a second step, we call a given algorithm to predict the present, i.e. the frequencies of the current year (i.e. 2015). **As a third step, we compare the predicted value with the factual value of the current year (i.e. 2015). We then compute a score from the difference between the prediction and the factual observation with the following formula:**

$$1 - \left(\sum_{on\ terms} \frac{|\ predicted\ freq - factual\ freq\ |}{factual\ freq} \right) /\ terms\#$$

We repeat this process to all algorithms with all values for the size of the Past, starting from 2 years until 50 years, in order to confront all these pairs of algorithm / size of Past.

We use Weka[8] because this software environment has a large spectrum of algorithms [Witten et al 2011]. The number of algorithms for predictive modeling of numeric variables is 25 in the last version for developers (version 3.7.13 as of March 2016) installed with the Time Series plugin. We set a one-hour guard time for each run because some algorithms are too slow for our experiments. For instance the Multilayer Perceptron is known to be very slow (see the comparison made by Ian H Witten[9]). Thus, the number of algorithms is 21 instead of 25.
We started with a size of Past of 2 until 50 included, that means that we ran 1029 sessions (i.e. 21*(50-2+1)). We call these algorithms with their default parameters which give the results with only the best result of each algorithm presented in Table 3.

We may notice that the difference between the various algorithms is rather small for the best runs, but let's recall that these are the 21 best runs among 1029 ones. One additional point could be said about the closeness of the figures in the comparison: Weka proposes 25 algorithms but some of them belong to the same family, for instance the family of regression algorithms like SMOreg or Additive Regression (see a comprehensive tutorial in [Smola et al 1998] and also [Shevade et al 1999]).

Algorithm name	Best size of Past	Correct Prediction Score	Computation time	Rank
GaussianProcesses	18	0.7226	1 s	1
SMOreg	16	0.7165	1 s	2
RandomizableFilteredClassifier	30	0.7041	1 s	3
KStar	3	0.6860	1 s	4
DecisionStump	3	0.6859	1 s	5
LWL	3	0.6859	1 s	6
AdditiveRegression	3	0.6859	1 s	7
IBk	3	0.6859	2 s	8
DecisionTable	3	0.6853	8 s	9
RandomForest	3	0.6839	7 s	10
MultiScheme	3	0.6741	1 s	11
M5P	3	0.6741	1 s	12
Vote	3	0.6741	1 s	13
ZeroR	3	0.6741	1 s	14
RegressionByDiscretization	8	0.6737	1 s	15
RandomTree	3	0.6732	1 s	16
RandomCommittee	3	0.6732	2 s	17
Bagging	4	0.6650	1 s	18
RandomSubSpace	4	0.6488	3 s	19
CVParameterSelection	11	0.5090	1 s	20
Stacking	11	0.5090	1 s	21

Table 3 Comparison of 21 algorithms

The algorithm labeled as "GaussianProcesses" appears to be the best algorithm. This algorithm implements Gaussian processes for regression without hyperparameter-tuning. To make choosing an appropriate noise level easier, this implementation applies normalization to the target attribute as well. Missing values are replaced by the global mean-mode. Nominal attributes are converted to binary ones (from the Weka documentation). This algorithm is called with its default parameters. Table 4 presents the detail of the best run with a ranking according to the frequency in a given year.

[8] www.cs.waikato.ac.nz/ml/weka
[9] cs.waikato.ac.nz/ml/weka/mooc/moredataminingwithweka/slides/Class5-MoreDataMiningWithWeka-2014.pdf

Factual value for 2013	Factual value for 2014	Factual value for 2015	(Simulated) prediction for 2015	Rank
classifier (0.00576)	annotation (0.00792)	dataset (0.00886)	dataset (0.00653)	1
LM (0.00565)	dataset (0.00639)	DNN (0.00613)	annotation (0.00626)	2
dataset (0.00548)	POS (0.00600)	classifier (0.00491)	POS (0.00549)	3
POS (0.00536)	LM (0.00513)	POS (0.00485)	LM (0.00479)	4
annotation (0.00509)	classifier (0.00507)	neural network (0.00455)	classifier (0.00466)	5
SR (0.00507)	SR (0.00449)	LM (0.00454)	DNN (0.00437)	6
HMM (0.00478)	parser (0.00388)	SR (0.00439)	SR (0.00429)	7
parser (0.00404)	DNN (0.00369)	parser (0.00436)	HMM (0.00365)	8
GMM (0.00367)	HMM (0.00352)	annotation (0.00414)	neural network (0.00345)	9
segmentation (0.00298)	neural network (0.00326)	HMM (0.00384)	tweet (0.00312)	10

Table 4 Details of the best run for evaluation

8. Reliability estimation over all terms

It is difficult to compute an evaluation of the reliability concerning a predicted event, the reliability being defined as the gap between the predicted and factual frequency of the extracted terms. The aim is to have an estimation of the drift (or the absence of drift) for the prediction for the first next year compared to the other four years, and so on, over time. The only concrete option seems to use the same strategy as the benchmarking process as presented in the section dedicated to evaluation (i.e. section 7). The period of time being the last 18 years, we need to restrict the times series to the first 13 years and to predict the events for the last 5 years. We then compare individually the factual term frequencies with the predicted ones over the 200 terms to compute a gap. We observe that there is an important drift as presented in figure 1. **This is in line with the intuition that the further a prediction ranges into the future, the greater the probability of error.** The reliability drastically collapses after four years.

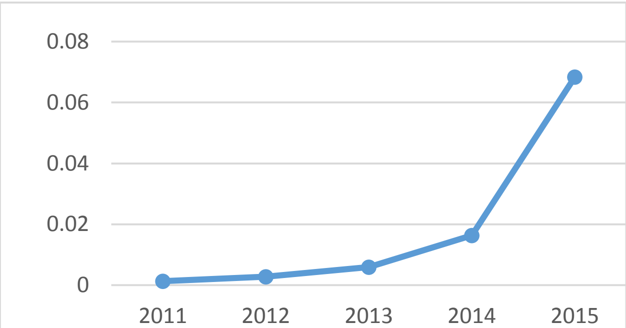


Figure 1 Average gap between prediction and observation on a time scale of 5 years

9. Estimation of surprises

Another use of prediction is to compare in the past years the gap between what would have been predicted and what actually happened. This may provide an analysis of the surprises that we lived: the difference between a continuous "research-as-usual" flow and the sudden uprising of new scientific paradigms, the detection of ruptures in research.

In order to do so, we used a slightly modified version of the reliability computation algorithm presented in section 8. We considered the same set of 200 terms, and we computed the prediction for the years 2011-2015 of the frequency for each of those terms based on the past years, from 1998 to the year preceding the one of the prediction. We then compare individually the factual term frequencies with the predicted ones over the 200 terms to compute a gap, that we will call the "surprise" and we sum up the individual differences to obtain a global measure of the difference between the prediction and the observed reality.

We observe that the "surprise" was larger in 2011 and 2012 than in the following years (Fig. 2).

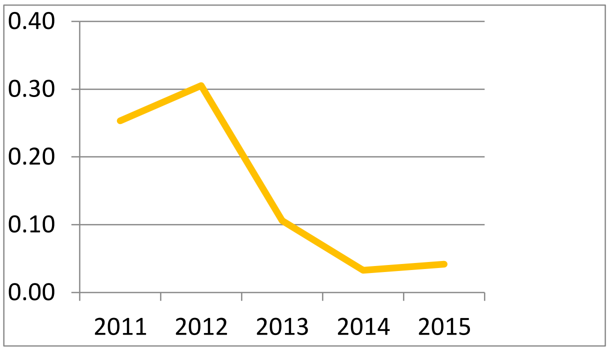


Figure 2 Estimation of surprises for the 200 terms

We then considered individual terms: HMM, SVM, Neural Networks (NN) and DNN (Fig. 3 and 4).

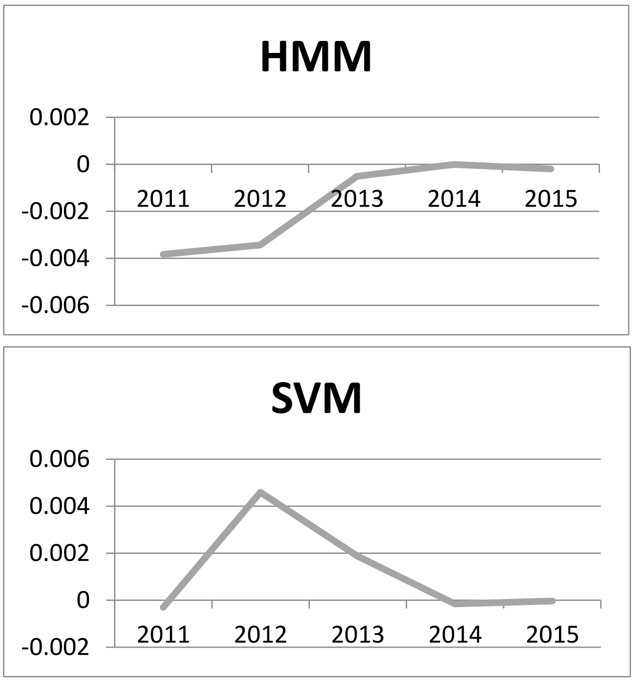
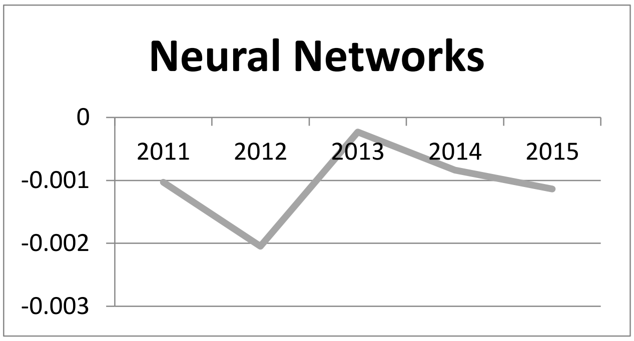


Figure 3 Estimation of surprises for HMM and SVM

We observe that HMM was under-predicted in 2011 and 2012, before rejoining a fluent use. SVM was over-predicted in 2012 and 2013 and is getting normal since 2014. Neural Networks was under-predicted in 2011 and 2012, before getting "normal" in 2013 and again slightly under-predicted in 2014 and 2015. DNN started its somehow unexpected extension in 2013 that it kept in 2014. It's now rejoining research mainstream.



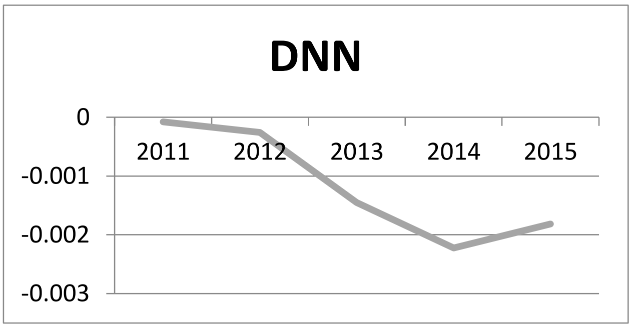


Figure 4 Estimation of surprises for NN and DNN

10. Final computation

We are now ready to compute the prediction for the next five years, provided that we take into account the reliability of the prediction estimation as presented in section 8. We therefore use the Gaussian Processes algorithm on 18 years backwards. The full precise numeric values are given on the corpus web site.

We rank the terms according to their frequency in a given year. Table 5 shows the first 30 terms. It appears in Table 5 that we foresee that for 2016, the terms "dataset", "DNN", "POS" and "neural network" will stay popular at the same ranking. The terms "annotation", "parser", "tweet", "annotator" will become slightly more popular while the terms "classifier", "LM", "SR", "metric" will become less popular.

11. Further developments

In the next round of experiments, we plan to go from term analysis to topic analysis in order to free ourselves from the need to have an explicit term associated to a topic to be able to detect its apparition. Thus, we will be able to more emphasis on investigating the emergence of new topics by using Latent Dirichlet Allocation [Blei et al 2003] and in particular its supervised variant [Blei et al 2007], and to study topic dynamics [Blei et al 2006].

12. Conclusion

The experiments presented here deal with the record of a large set of term usages over a period of time of 50 years. Various algorithms have been evaluated and compared in order to select the one which provides the best guessing of the frequencies of the most popular terms for the next five year. These experiments can be applied to any other domain. The only elements which are specific concern the term extraction, namely the stop-list and the synonyms, whose creation has been made manually, taking advantage of the fact that we have a good knowledge of the NLP domain ourselves. If our method was to be applied to another domain, an expert in this given domain would therefore be needed for this task.

13. Acknowledgements

We'd like to thank Wolfgang Hess for the ISCA archive, Douglas O'Shaughnessy, Denise Hurley, Rebecca Wollman and Casey Schwartz for the IEEE data, Nicoletta Calzolari, Helen van der Stelt and Jolanda Voogd for the LRE Journal articles, Olivier Hamon and Khalid Choukri for the LREC proceedings, Nicoletta Calzolari, Irene Russo, Riccardo Del Gratta, Khalid Choukri for the LRE Map, Min-Yen Kan for the ACL Anthology, Florian Boudin for the TALN proceedings and Ellen Voorhees for the TREC proceedings.

A part of this experiment has been done in the context of the research project REQUEST 018062-25005 FSN-AAP-Big Data n.3, funding from the French Ministry of Industry ("Ministère du Redressement Productif" and "Commissariat Général à l'Investissement"), fund for developing Digital Economy ("Investissements d'Avenir Développement de l'Economie Numérique"), Cloud Computing – Call for Tenders N°3 Big Data.

Factual 2014	Factual 2015	Prediction for 2016	Prediction for 2017	Prediction for 2018	Prediction for 2019	Prediction for 2020	Rank
annotation	dataset	dataset	dataset	dataset	dataset	dataset	1
dataset	DNN	DNN	DNN	DNN	DNN	DNN	2
POS	classifier	annotation	neural network	neural network	neural network	neural network	3
LM	POS	POS	SR	RNN	RNN	RNN	4
classifier	neural network	neural network	classifier	POS	parser	parser	5
SR	LM	classifier	LM	parser	SR	SR	6
parser	SR	parser	POS	annotation	LM	metric	7
DNN	parser	SR	RNN	classifier	classifier	POS	8
HMM	annotation	LM	parser	SR	metric	parsing	9
neural network	HMM	HMM	HMM	metric	POS	classifier	10
ngram	metric	RNN	metric	LM	parsing	LM	11
annotator	RNN	metric	parsing	parsing	HMM	tweet	12
GMM	parsing	parsing	GMM	tweet	MT	MT	13
metric	GMM	GMM	annotation	MT	tweet	SNR	14
SVM	MT	tweet	MT	annotator	GMM	kernel	15
segmentation	ngram	MT	tweet	HMM	SNR	annotation	16
tweet	SVM	annotator	SNR	GMM	kernel	WER	17
parsing	segmentation	ngram	WER	SNR	WER	GMM	18
MT	NP	segmentation	SVM	kernel	optimization	LDA	19
WER	SNR	SVM	ngram	SVM	LDA	subset	20
NP	iteration	SNR	segmentation	predicate	SVM	HMM	21
predicate	annotator	subset	kernel	ngram	subset	optimization	22
iteration	tweet	WER	subset	subset	Bleu	predicate	23
subset	LSTM	iteration	iteration	NLP	ngram	NLP	24
Wikipedia	subset	predicate	annotator	WER	iteration	annotator	25
NLP	WER	kernel	optimization	segmentation	regularization	regularization	26
SNR	kernel	NP	LDA	optimization	normalization	ngram	27
LDA	predicate	LDA	normalization	LDA	segmentation	semantic	28
Bleu	optimization	optimization	Bleu	CRF	NLP	CRF	29
normalization	Bleu	Bleu	predicate	iteration	predicate	SVM	30

Table 5 Final computation of the prediction over the next 5 years

14. Bibliographical References

Anderson Ashton, McFarland Dan, Jurafsky Dan (2012). Towards a Computational History of the ACL: 1980-2008. Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries, Jeju, Republic of Korea.

Bird Steven, Dale Robert, Dorr Bonnie J, Gibson Bryan, Joseph Mark T, Kan Min-Yen, Lee Dongwon, Powley Brett, Radev Dragomir R, Tan Yee Fan (2008). The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics, Proceedings of LREC, Marrakech, Morocco.

Blei David, Ng Andrew Y, Jordan Michael I (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research 3-2003, 993-1022.

Blei David, Lafferty John D (2006). Dynamic Topic Models, Proceedings of the International Conference on Machine Learning, ACM, pp113-120.

Blei David, McAuliffe John D (2007). Supervised Topic Models, Neural Information Processing Systems 21.

Bordea Georgeta, Buitelaar Paul, Polajnar Tamara (2013). Domain-independent term extraction through domain modelling, in Proceedings of the 10th International Conference on Terminology and Artificial Intelligence, Villetaneuse, France.

Drouin Patrick (2004). Detection of Domain Specific Terminology Using Corpora Comparison, in Proceedings of LREC 2004, 26-28 May 2004, Lisbon, Portugal.

Francopoulo Gil (2007). TagParser: well on the way to ISO-TC37 conformance. ICGL (International Conference on Global Interoperability for Language Resources), Hong Kong, PRC.

Francopoulo Gil, Mariani Joseph, Paroubek Patrick (2015). NLP4NLP: The Cobbler's Children Won't Go Unshod, in D-Lib Magazine: The magazine of Digital Library Research[10].

[10] www.dlib.org/dlib/november15/francopoulo/11francopoulo.html

Gupta Parth, Rosso Paolo (2012), Text Reuse with ACL: (Upward) Trends, Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries, Jeju, Republic of Korea.

Mariani Joseph, Paroubek Patrick, Francopoulo Gil, Delaborde Marine (2013). Rediscovering 25 Years of Discoveries in Spoken Language Processing: a Preliminary ISCA Archive Analysis, in Proceedings of Interspeech 2013, 26-29 August 2013, Lyon, France.

Mariani Gil, Paroubek Patrick, Francopoulo Gil, Hamon Olivier (2014). Rediscovering 15 years of Discoveries in Language Resources and Evaluation: The LREC Anthology Analysis, in Proceedings of LREC 2014, 26-31 May 2014, Reykjavik, Island.

Radev Dragomir R, Muthukrishnan Pradeep, Qazvinian Vahed, Abu-Jbara, Amjad (2013). The ACL Anthology Network Corpus, Language Resources and Evaluation 47: 919–944, Springer.

Shevade S.K., Keerthi S.S., Bhattacharyya C., Murthy K.R.K. (1999). Improvements to the SMO Algorithm for SVM Regression. IEEE Transactions on Neural Networks.

Smola Alex J, Schölkopf Bernhard (1998). A tutorial on Support Vector Regression. NeuroCOLT2 Technical Report Series.

Witten Ian H, Frank Eibe, Hall Mark A (2011). Data Mining: practical machine learning tools and techniques. Third Edition. Morgan Kaufmann, Burlington, USA.