Roberto Basili, Fabrizio Sebastiani, Giovanni Semeraro (Eds.)

Proceedings of the Fourth Italian Information Retrieval Workshop

IIR 2013

National Council of Research campus, Pisa, Italy
16 – 17 January 2013
http://iir2013.isti.cnr.it/

This volume is published and copyrighted by:

Roberto Basili

Fabrizio Sebastiani

Giovanni Semeraro

ISSN 1613-0073

Copyright © 2013 for the individual papers by the papers' authors. Copying permitted only for private and academic purposes. Re-publication of material from this volume requires permission by the copyright owners.

Table of Contents

SEMANTICS, NATURAL LANGUAGE AND APPLICATIONS

Preface

The purpose of the Italian Information Retrieval (IIR) workshop series is to provide a forum for stimulating and disseminating research in information retrieval, where Italian researchers (especially young ones) and researchers affiliated with Italian institutions can network and discuss their research results in an informal way. IIR 2013 took place in Pisa, Italy, at the National Council of Research campus on January 16-17, 2013, following the first three successful editions in Padua (2010), Milan (2011) and Bari (2012).

The contributions to IIR 2013 mainly address six relevant topics:

- theory
- query languages and operations
- image retrieval
- evaluation
- social media and information retrieval
- semantics, natural language and applications

Most submitted papers were from PhD students and early stage researchers. All the 24 submissions, both full and short original papers presenting new research results, as well as extended abstracts containing descriptions of ongoing projects or presenting already published results, were reviewed by two members of the Program Committee and 18 contributions were selected for presentation on the basis of originality, technical depth, style of presentation, and impact. Additionally to the presentations of these 18 submitted papers, IIR 2013 featured two special events. The first was an invited talk by Renato Soru, CEO of Tiscali SpA, in which the speaker addressed past, present, and future efforts by Tiscali to enter the Web search market. In particular, Soru highlighted some new features of "istella", the soon-to-be-announced Web search engine by Tiscali, mainly addressed at covering the Italian Web space, with a special emphasis on making Italy's cultural heritage digitally available to a wide audience. The second special event was a panel on EVALITA, an evaluation campaign which has been running biennially since 2007 and whose main goal is the evaluation of natural language processing tools for Italian. Several EVALITA task organizers have presented the main results obtained in the recent editions of the campaign and have discussed the unresolved challenges that still lie ahead of researchers, with the aim of generating awareness about the state-of-the-art in Italian NLP among IR researchers and of strengthening the relationships between the two communities.

The present proceedings include the papers that were presented at IIR 2013. We hope they represent an interesting contribution to IR research in Italy, and to IR research in general.

The Workshop Organisers

Roberto Basili
University of Roma "Tor Vergata" (Program co-Chair)

Fabrizio Sebastiani
ISTI-CNR (General Chair)

Giovanni Semeraro
University of Bari Aldo Moro (Program co-Chair)

Organization

General Chair

Fabrizio Sebastiani (ISTI-CNR)

Program Chairs

Roberto Basili (University of Rome “Tor Vergata”)

Giovanni Semeraro (University of Bari Aldo Moro)

IIR Steering Committee

Gianni Amati (Fondazione Ugo Bordoni)

Claudio Carpineto (Fondazione Ugo Bordoni)

Massimo Melucci (University of Padua)

Stefano Mizzaro (University of Udine)

Gabriella Pasi (University of Milano Bicocca)

Giovanni Semeraro (University of Bari Aldo Moro)

Program Committee

Giambattista Amati (Fondazione Ugo Bordoni)

Giuseppe Amodeo (Almawave srl)

Pierpaolo Basile (University of Bari Aldo Moro)

Giacomo Berardi (ISTI-CNR, Pisa)

Gloria Bordogna (IDPA-CNR Dalmine, Bergamo)

Claudio Carpineto (Fondazione Ugo Bordoni)

Fabio Crestani (Università della Svizzera Italiana)

Danilo Croce (University of Roma “Tor Vergata”)

Marco de Gemmis (University of Bari Aldo Moro)

Pasquale De Meo (VU University, Amsterdam)

Giorgio Maria Di Nunzio (University of Padua)

Giorgio Gambosi (University of Roma “Tor Vergata”)

Marco Gori (University of Siena)

Antonio Gulli (Microsoft)

Pasquale Lops (University of Bari Aldo Moro)

Marco Maggini (University of Siena)

Massimo Melucci (University of Padua)

Stefano Mizzaro (University of Udine)

Alessandro Moschitti (University of Trento)

Salvatore Orlando (University of Venezia)

Gabriella Pasi (University of Milano Bicocca)

Raffaele Perego (ISTI-CNR, Pisa)

Francesco Ricci (Free University of Bozen-Bolzano)

Fabrizio Silvestri (ISTI-CNR, Pisa)

Organizing Committee

Adriana Lazzaroni, IIT-CNR (Local Arrangements Chair)

Patrizia Andronico, IIT-CNR

Giacomo Berardi, ISTI-CNR (Webmaster)

Catherine Bosio, ISTI-CNR

Raffaella Casarosa, IIT-CNR

Giulio Galesi, ISTI-CNR

Additional Reviewers

Annalina Caputo (University of Bari Aldo Moro)

Piero Molino (University of Bari Aldo Moro)

Fedelucio Narducci (University of Milano Bicocca)

Are There New BM25 "Expectations"?

Emanuele Di Buccio and Giorgio Maria Di Nunzio

Dept. of Information Engineering – University of Padua
[dibuccio,dinunzio]@dei.unipd.it

Abstract. In this paper, we present some ideas about possible directions of a new interpretation of the Okapi BM25 ranking formula. In particular, we have focused on a full bayesian approach for deriving a smoothed formula that takes into account a-priori knowledge on the probability of terms. In fact, most of the efforts in improving the BM25 were done in capturing the language model (frequencies, length, etc.) but missed the fact that the constant equal to 0.5 used as a correction factor can be one of the parameters that can be modelled in a better way. This approach has been tested on a visual data mining tool and the initial results are encouraging.

1 Introduction

The relevance weighting model, also known as RSJ by the name of its creators (Roberston and Sparck-Jones), has been one of the most influential model in the history of Information Retrieval [1]. It is a probabilistic model of retrieval that tries to answer the following question:

> What is the probability that this document is relevant to this query?

'Query' is a particular instance of an information need, and 'document' a particular content description. The purpose of this question is to rank the documents in order of their probability of relevance according the Probability Ranking Principle [2]:

> If retrieved documents are ordered by decreasing probability of relevance on the data available, then the system's effectiveness is the best to be gotten for the data.

The probability of relevance is achieved by assigning weights to terms, the RSJ weight hereafter named as w_i, according to the following formula:

$$w_i = log \frac{p_i}{(1-p_1)} \frac{(1-q_i)}{q_i} \ , \tag{1}$$

where p_i is the probability that the document contains the term t_i given that the document is relevant, and q_i is the probability that the document contains the term t_i given that the document is not relevant. If the estimates of these

probabilities are computed by means of a maximum likelihood estimation, we obtain the following results:

$$p_i = \frac{r_i}{R} \tag{2}$$

$$q_i = \frac{n_i - r_i}{N - R} \tag{3}$$

where r_i is the number of relevant documents that contain term t_i, n_i the number of documents that contain term t_i, R and N the number of relevant documents and the total number of documents, respectively. However, this estimation leads to arithmetical anomalies; for example, if a term is not present in the set of relevant documents, its probability p_i is equal to zero and the logarithm of zero will return a minus infinity. In order to avoid this situation, a kind of smoothing is applied to the probabilities. By substituting Equation 2 and 3 in Equation 1 and adding a constant to smooth probabilities, we obtain:

$$w_i = log \frac{r_i + 0.5}{(R - r_i + 0.5)} \frac{(N - R - n_i + r_i + 0.5)}{n_i - r_i + 0.5} \ , \tag{4}$$

which is the actual RSJ score for a term. The choice of the constant 0.5 may resemble some Bayesian justification related to the binary independence model.[1] This idea is wrong, as Robertson and Sparck Jones explained in [3], and the real justification can be traced back to the work of Cox [4].

The Okapi BM25 weighting schema takes a step further and introduces the property of eliteness [5]:

> Assume that each term represent a concept, and that a given document is about that concept or not. A term is 'elite' in the document or not.

BM25 estimates the full eliteness weight for a term from the RSJ score, then approximates the term frequency behaviour with a single global parameter controlling the rate of approach. Finally, it makes a correction for document length. For a full explanation of how to interpret eliteness and integrate it into the BM25 formula read [6–9]. The resulting formula is summarised in the following way:

$$w'_i = f(tf_i) \cdot w_i \tag{5}$$

where w_i is the RSJ weight, and $f(tf_i)$ is a function of the frequency of the term t_i parametrized by global parameters.

In this paper, we concentrate on the RSJ weight and in particular to a full Bayesian approach for smoothing the probabilities and on a visual data analysis to assess the effectiveness of these new smoothed probabilities. In Section 2, we present the Bayesian framework, then in Section 3 we describe the visualisation approach; in Section 4, we describe the initial experiments on this approach. Some final remarks are given in Section 5.

[1] In this model; documents are represented as binary vectors: a term may be either present or not in a document and have a 'natural' a priori probability of 0.5.

2 Bayesian Framework

In Bayesian inference, a problem is described by a mathematical model M with parameters θ and, when we have observed some data D, we use Bayes' rule to determine our beliefs across different parameter values θ [10]:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)} \ , \tag{6}$$

the posterior distribution of our belief on θ is equal to a likelihood function $P(D|\theta,M)$, the mathematical model of our problem, multiplied by a prior distribution $P(\theta|M)$, our belief in the values of the parameters of the model, and normalized by the probability of the data $P(D|M)$. We control the prior by choosing its distributional form along with its parameters, usually called *hyper-parameters*. Since the product between $P(D|\theta,M)$ and $P(\theta|M)$ can be hard to calculate, one solution is to find a "conjugate" prior of the likelihood function [10].

In the case of a likelihood function which belongs to the exponential family, there always exists a conjugate prior. Naïve Bayes (NB) models have a likelihood of this type and, since the RSJ weight is related to the Binary Independence Model which is a multi-variate Bernoulli NB model, we can easily derive a formula to estimate the parameter θ. The multi-variate Bernoulli NB model represents a document d as a vector of V (number of words in the vocabulary) Bernoulli random variables $d = (t_1, ..., t_i, ..., t_V)$ such that:

$$t_i \sim Bern(\theta_{t_i}) \ . \tag{7}$$

We can write the probability of a document by using the NB assumption as:

$$P(d|\theta) = \prod_{k=1}^{V} t_i = \prod_{k=1}^{V} \theta_i^{x_k}(1-\theta_i)^{1-x_k} \ , \tag{8}$$

where x_i is a binary value that is equal either to 1 when the term t_i is present in the document or to 0 otherwise. With a Maximum Likelihood estimation, we would end up with the result shown in Equation 2 and 3; instead, we want to integrate the conjugate prior which in this case of a Bernoulli random variable is the *beta* function:

$$beta_i = \theta_i^{\alpha-1}(1-\theta_i)^{\beta-1} \ , \tag{9}$$

where i refers to the ith random variable t_i. Therefore, the new estimate of the probability of a term t_i that takes into account the prior knowledge is given by the posterior mean of Eq. 6 (see [10] for the details of this result). For the relevant documents we obtain:

$$\hat{\theta}_{t_i|rel} = \frac{r_i + \alpha}{R + \alpha + \beta} = \hat{p_i} \ , \tag{10}$$

where $\hat{p_i}$ is the new estimate of the probability p_i. Accordingly, the probability of a term in the non-relevant documents is:

$$\hat{\theta}_{t_i|\overline{rel}} = \frac{n_i - r_i + \alpha}{N - R + \alpha + \beta} = \hat{q_i} \ . \tag{11}$$

With this formula, we can recall different smoothing approaches; for example, with $\alpha = 0$ and $\beta = 0$ we obtain the Maximum Likelihood Estimation, with $\alpha = 1$, $\beta = 1$ the Laplace smoothing. We can even recall the RSJ score by assigning $\alpha = 0.5$ and $\beta = 0.5$.

3 Probabilistic Visual Data Mining

Now that we have new estimates for the probabilities p_i and q_i, we need a way to assess how the parameters α and β influence the effectiveness of the retrieval system. In [11, 12], we presented a visual data mining tool for analyzing the behavior of various smoothing methods, to suggest possible directions for finding the most suitable smoothing parameters and to shed the light into new methods of automatic hyper-parameters estimation. Here, we use the same approach for analyzing a simplified version of the BM25 (that is Equation 5 ignoring the term frequency function).

In order to explain the visual approach, we present the problem of retrieval in terms of a classification problem: classify the documents as relevant or non relevant. Given a document d and a query q, we consider d relevant if:

$$P(rel|d,q) > P(\overline{rel}|d,q) \ , \tag{12}$$

that is when the probability of being relevant is higher compared to the probability of not being relevant. By using Bayes rule, we can invert the problem and decide that d is relevant when:

$$P(d|rel,q)P(rel|q) > P(d|\overline{rel},q)P(\overline{rel}|q) \ . \tag{13}$$

Note that we are exactly in the same situation of Equation (2.2) of [9] where:

$$P(rel|d,q) \propto \frac{P(d|rel,q)P(rel|q)}{P(d|\overline{rel},q)P(\overline{rel}|q)} \ . \tag{14}$$

In fact, if we divide both members of Equation 13 by $P(d|\overline{rel},q)P(\overline{rel}|q)$ (we assume that this quantity is strictly greater than zero), we obtain:

$$\frac{P(d|rel,q)P(rel|q)}{P(d|\overline{rel},q)P(\overline{rel}|q)} > 1 \ , \tag{15}$$

where the ranking of the documents is given by the value of the ratio on the left (as in the BM25); moreover, we can classify a document as 'relevant' if this ratio is greater than one.

The main idea of the two-dimensional visualization of probabilistic model is to maintain the two probabilities separated and use the two numbers as two coordinates, X and Y, on the cartesian plane:

$$\underbrace{P(d|rel,q)P(rel|q)}_{X} > \underbrace{P(d|\overline{rel},q)P(\overline{rel}|q)}_{Y} \ . \tag{16}$$

If we take the logs, a monotonic transformation that maintains the order, and if we model the document as a multivariate binomial (as in the Binary Independence Model [1]), we obtain for the coordinate X:

$$\underbrace{\sum_{i \in V} x_i \log\left(\frac{\hat{p}_i}{1-\hat{p}_i}\right) + \sum_{i \in V} \log(1-\hat{p}_i)}_{P(d|rel,q)} + \underbrace{\log(P(rel|q))}_{P(rel|q)} \ . \tag{17}$$

Since we are using the Bayesian estimate $\hat{p}_i$, we can modulate it by adjusting the hyper parameters α and β of Equation 10. If we want to consider the terms that appear in the query, the first sum is computed over the terms $i \in q$, which corresponds to Equation (2.6) of [9].

We intentionally maintained explicit the two addends that are independent of the document, respectively $\sum_{i \in V} \log(1-\hat{p}_i)$ and $\log(P(rel|q))$. These two addends do not influence the ordering among documents (it is a constant factor independent of the document) but they can (and they actually do) affect the classification performance. If we rewrite the complete inequality and substitute these addends with constants we obtain: [2]

$$\sum_{i \in q} x_i \log\left(\frac{\hat{p}_i}{1-\hat{p}_i}\right) + c_1 > \sum_{i \in q} x_i \log\left(\frac{\hat{q}_i}{1-\hat{q}_i}\right) + c_2 \tag{18}$$

$$\sum_{i \in q} x_i \log\left(\frac{\hat{p}_i}{1-\hat{p}_i}\right) - \sum_{i \in q} x_i \log\left(\frac{\hat{q}_i}{1-\hat{q}_i}\right) > c_2 - c_1 \tag{19}$$

$$\sum_{i \in q} x_i \underbrace{\log\left(\frac{\hat{p}_i}{1-\hat{p}_i}\frac{1-\hat{q}_i}{\hat{q}_i}\right)}_{RSJ} > c_2 - c_1 \tag{20}$$

that is exactly the same formulation of the RSJ weight with new estimates for p_i and q_i, plus some indication about whether we classify a document as relevant or not.

3.1 A simple example

Let us consider a collection of 1,000 documents, suppose that we have a query with two terms, $q = \{t_1, t_2\}$, and the following estimates:

$$\hat{p}_1 = \frac{3+\alpha}{10+\alpha+\beta} \ , \ \hat{q}_1 = \frac{17+\alpha}{990+\alpha+\beta} \ ,$$
$$\hat{p}_2 = \frac{2+\alpha}{10+\alpha+\beta} \ , \ \hat{q}_2 = \frac{15+\alpha}{990+\alpha+\beta} \ ,$$

which means that we have

[2] Note that we need to investigate how this reformulation is related to Cooper's linked dependence assumption [13].

– 10 relevant document ($R = 10$) for this query;
– 20 documents that contain term t_1 ($n_1 = 20$) and three of them are known to be relevant ($r_1 = 3$);
– 17 documents that contain term t_2 ($n_2 = 17$) and two of them are known to be relevant ($r_2 = 2$).

For the log odds, we have:

$$\phi_1 = \log\left(\frac{\hat{p_1}}{1-\hat{p_1}}\right) = \log\left(\frac{3+\alpha}{7+\beta}\right) \ , \ \psi_1 = \log\left(\frac{\hat{q_1}}{1-\hat{q_1}}\right) = \log\left(\frac{17+\alpha}{973+\beta}\right) \ ,$$
$$\phi_2 = \log\left(\frac{\hat{p_2}}{1-\hat{p_2}}\right) = \log\left(\frac{2+\alpha}{8+\beta}\right) \ , \ \psi_2 = \log\left(\frac{\hat{q_2}}{1-\hat{q_2}}\right) = \log\left(\frac{15+\alpha}{975+\beta}\right) \ .$$

Suppose that we want to rank two document d_1 and d_2, where d_1 contains both terms t_1 and t_2, while d_2 contains only term t_1. Let us draw the points in the two-dimensional space, we assume the two constants c_1 and c_2 equal to zero:

$$\begin{aligned} X_{d_1} &= x_{1,d_1} * \phi_1 + x_{2,d_1} * \phi_2 = 1 * \phi_1 + 1 * \phi_2 \simeq -2.86, \\ Y_{d_1} &= x_{1,d_1} * \psi_1 + x_{2,d_1} * \psi_2 = 1 * \psi_1 + 1 * \psi_2 \simeq -11.77, \\ X_{d_2} &= x_{1,d_2} * \phi_1 + x_{2,d_2} * \phi_2 = 1 * \phi_1 + 0 * \phi_2 \simeq -1.10, \\ Y_{d_2} &= x_{1,d_2} * \psi_1 + x_{2,d_2} * \psi_2 = 1 * \psi_1 + 0 * \psi_2 \simeq -5.80 \end{aligned}$$

where $x_{i,d_j} = 1$ if term t_i occurs in document d_j, $x_{i,d_j} = 0$ otherwise.

In Figure 1, the two points (X_{d_1}, Y_{d_1}) and (X_{d_2}, Y_{d_2}) are shown. The line is a graphical help to indicate which point is ranked first: the closer the point, the higher the document in the rank. The justification of this statement is not presented in this paper for space reasons, refer to [14] for further details. What is important here is the possibility to assess the influence of the parameter α and β on the RSJ score. The objective is to study whether these two parameters can drastically change the ranking of the documents or not. In graphical terms, if we can "rotate" the points such that the closest to the line becomes the furthest.

Moreover, there are some considerations we want to address:

– when the number of terms in the query is small, it is very difficult to note any change in the ranking list. Remember that with 'n' query terms, we can only have 2^n points (or RSJ scores). In the event of a query constituted of a single term, all the documents that contain that query term collapse in one point.
– the Okapi BM25 weight 'scatters' the documents that are collapsed in one point in the space by multiplying the RSJ score with a scaling factor $f(tf_i)$ proportional to the frequency of the term in the document. Therefore, we expect this Bayesian approach to be more effective on the BM25 rather than on the simple RSJ score.

3.2 Visualization Tool

The visualisation tool was designed and developed in R [15]. It consists of three panels:

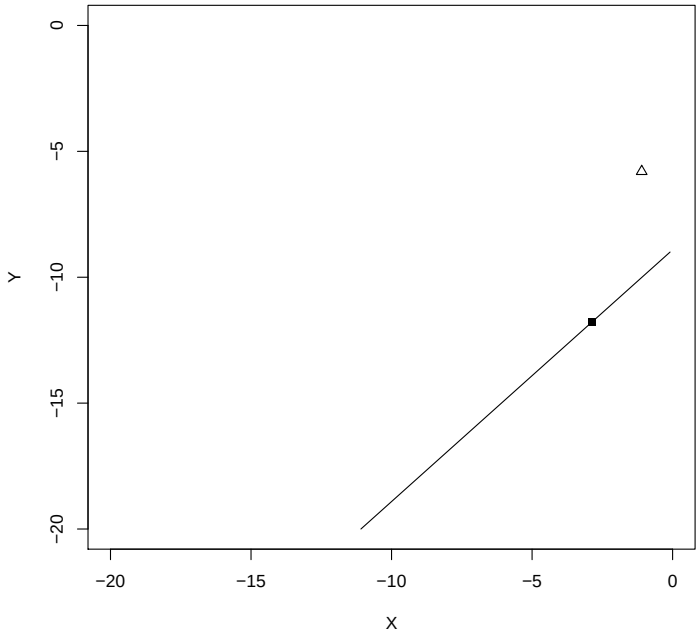


Fig. 1: Example for the documents d_1 and d_2 represented respectively by the points (X_{d_1}, Y_{d_1}) and (X_{d_2}, Y_{d_2}).

- *View Panel*: this displays the two-dimensional plot of the dataset according to the choices of the user.
- *Interaction Panel*: this allows for the interaction between the user and the parameters of the probabilistic models.
- *Performance Panel*: this displays the performance measures of the model.

Figure 2 shows the main window with the three panels. In the centre-right, there is the main view panel, the actual two-dimensional view of the documents as points, blue and red for relevant and non-relevant, respectively. The green line represents the ranking line, the closer the point the higher the rank in the retrieval list. At the top and on the left, there is the interaction panel where the user can choose different options: the type of the model (Bernoulli in our case), the type of smoothing (conjugate prior), the value of the parameters α and β. The bottom of the window is dedicated to the performance in terms of classification (not used in this experiment).

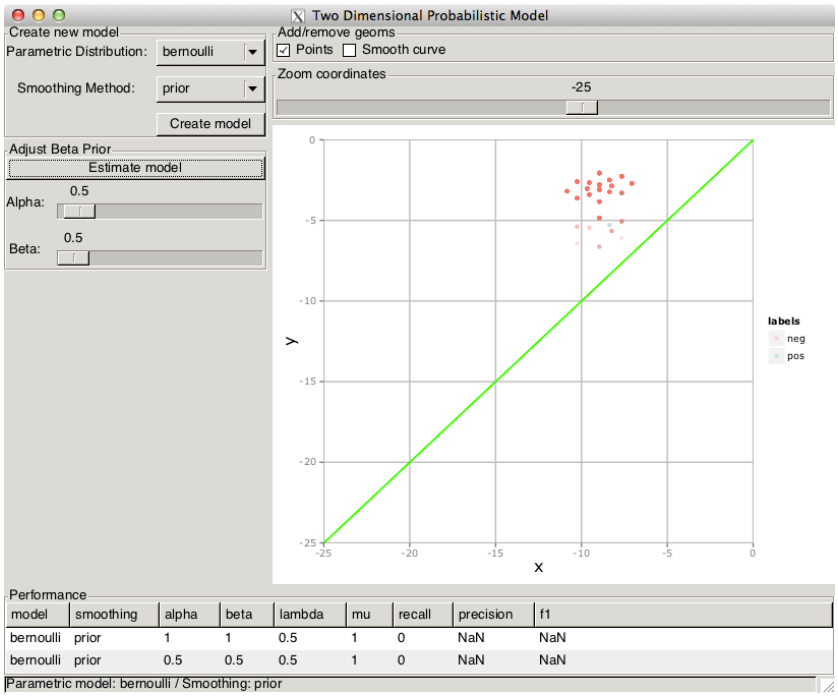


Fig. 2: Two-dimensional visualization tool: Main window.

4 Preliminary Experiments

Preliminary experiments were carried out on some topics of the TREC2001 Adhoc Web Track test collection.[3] The content of each document was processed during indexing except for the text contained inside the `<script></script>` and the `<style></style>` tags. When parsing, the title of the document was extracted and considered as the beginning of the document content. Stop words were removed during indexing.[4] For each topic we considered the set of documents in the pool, therefore those for which explicit assessment are available.

We considered two different experimental settings: (i) query-term based representation and (ii) collection vocabulary-based representation of the documents. In the former case, each document was represented by means of the descriptor extracted from the title of the TREC topics, used as queries: therefore V consisted of query terms; in the latter case V consisted of the entire collection vocabulary — both settings did not consider stopwords as part of V.

[3] http://trec.nist.gov/data/t10.web.html

[4] The stop words list is that available at the url http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words

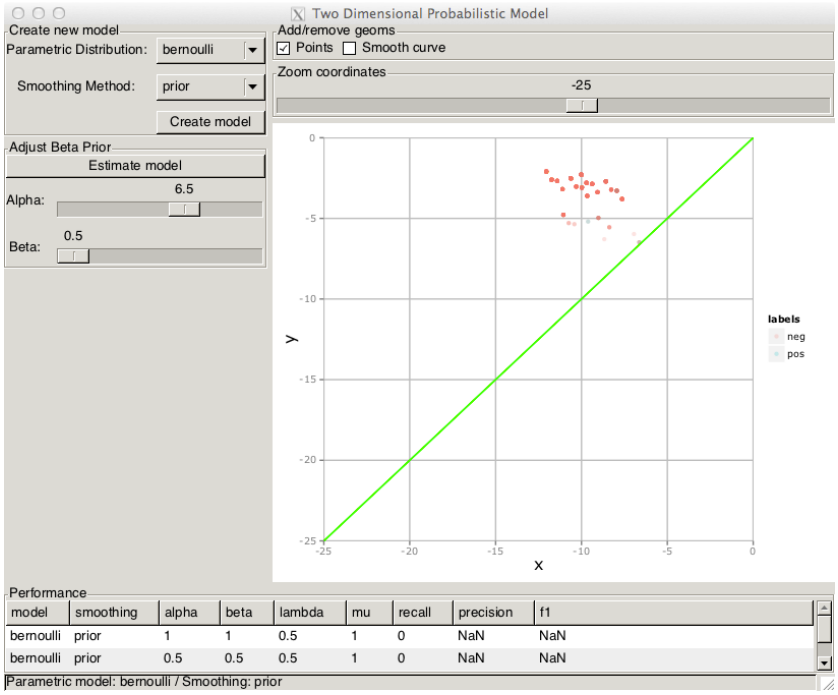


Fig. 3: Query 528: changed parameter alpha. Documents are stretched along the x-axis and rotate anti-clockwise.

In this paper, we report the experiments on topic 528. We selected this query because it contains five terms and it is easier to show the effect of the hyper-parameters. In Figure 2, the cloud of points generated by the two-dimensional approach is shown. Parameters α and β are set to the standard RSJ score constant 0.5. The line corresponds to the decision line of a classifier, and it also correspond to the 'ranking' line: imagine this line spanning the plane from right to left, each time the line touches a document, the document is added to the list of retrieved documents.

In Figure 3, the hyper-parameter α was increased and β was left equal to 0.5. When we increase α, the probability $\hat{p}_i$ tends to one, and the effect, in terms of the two dimensional plot, is that points rotate anti-clockwise. In Figure 4, the opposite effect is obtained by increasing β and leaving α equal to 0.5. In both situations, the list of ranked documents was significantly different from the original list produced by using the classical RSJ score.

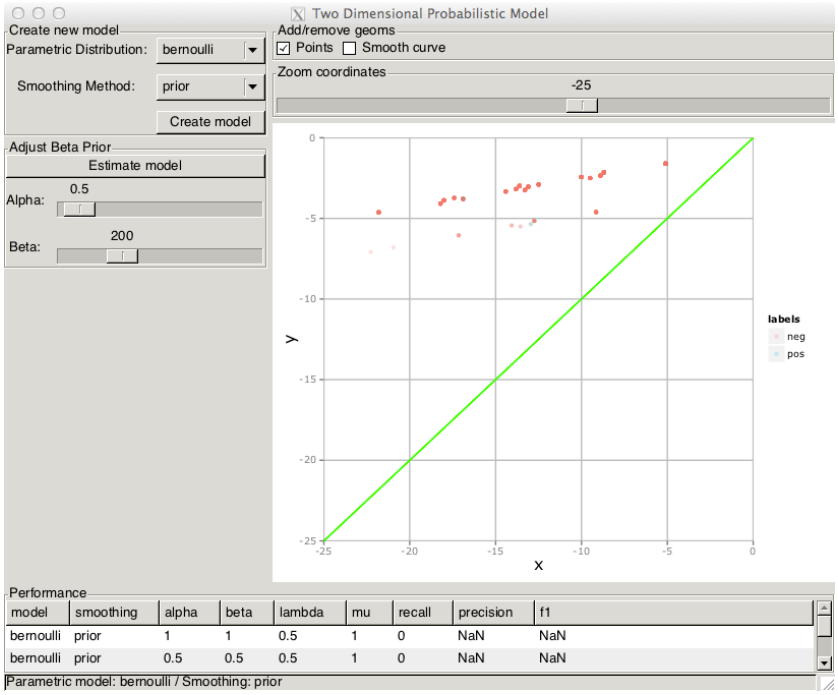


Fig. 4: Query 528: changed parameter beta. Documents are stretched along the x-axis and rotate clockwise.

5 Conclusions

This paper presents a new direction for the study of the Okapi BM25 model. In particular, we have focused on a full Bayesian approach for deriving a smoothed formula that takes into account our a-priori knowledge on the probability of terms. In fact, we think that many of the efforts in improving the BM25 were done mostly in capturing the language model (frequencies, length, etc.) but missed the fact that the 0.5 correction factor could be one of the parameters that can be modelled in a better way.

By starting from a slightly different approach, the classification of documents into relevant and non relevant classes, we derived the exact same formula of the RSJ weight but with more degrees of interaction. The two-dimensional visualization approach helped in understanding why some of the constants factors can be taken into account for the case of the classification and, more important, how the hyper-parameters can be tuned to obtain a better ranking.

After this preliminary experiment, we can draw some considerations: for the first time, it was possible to visualize the cluster of points that are generated by the RSJ scores; it was clear that very short queries tend to create a very small

number of points making it hard to perform a good retrieval; hyper-parameters do make a difference in both classification and retrieval.

There are still many open research questions we want to investigate in the future:

- so far, we have assumed that all the beta priors associated to each term use exactly the same values for hyper-parameters α and β. A more selective approach may be more effective;
- the coordinate of the points in the two-dimensional plot take into account the two constants of Equation 17. In particular, the addend $\sum_{i \in V} \log(1-\hat{p}_i)$ may be the cause of the 'rotation' of the points, hence the radical change of the ranking list;
- The current approach assumes that the value of R and r_i are known for each term in the query: indeed these values are adopted to estimate the coordinates of each document. A further research question is the effect of estimation based on feedback data on the capability of the probabilistic visual data mining approach adopted in this paper.

Acknowledgments. This work has been partially supported by the QONTEXT project under grant agreement N. 247590 (FP7/2007-2013).

References

1. Robertson, S.E., Sparck Jones, K.: Relevance weighting of search terms. In Willett, P., ed.: Document retrieval systems. Taylor Graham Publishing, London, UK, UK (1988) 143–160
2. Robertson, S.E.: The Probability Ranking Principle in IR. Journal of Documentation **33** (1977) 294–304
3. Jones, K.S., Walker, S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments. Inf. Process. Manage. **36** (2000) 779–808
4. Cox, D., Snell, D.: The Analysis of Binary Data. Monographs on Statistics and Applied Probability Series. Chapman & Hall (1989)
5. Robertson, S.: Understanding inverse document frequency: On theoretical arguments for idf. In: Journal of Documentation. Volume 60. (2004)
6. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In Croft, W.B., van Rijsbergen, C.J., eds.: SIGIR, ACM/Springer (1994) 232–241
7. Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M.: Okapi at TREC-3. In: Proceedings of the Third Text REtrieval Conference (TREC), Gaithesburg, USA (1994)
8. Robertson, S.E., Walker, S.: On relevance weights with little relevance information. SIGIR Forum **31** (1997) 16–24
9. Robertson, S.E., Zaragoza, H.: The probabilistic relevance framework: Bm25 and beyond. Foundations and Trends in Information Retrieval **3** (2009) 333–389
10. Kruschke, J.K.: Doing Bayesian Data Analysis: A Tutorial with R and BUGS. 1 edn. Academic Press/Elsevier (2011)

11. Di Nunzio, G., Sordoni, A.: How well do we know bernoulli? In: IIR. Volume 835 of CEUR Workshop Proceedings., CEUR-WS.org (2012) 38–44
12. Di Nunzio, G., Sordoni, A.: A visual tool for bayesian data analyisis: The impact of smoothing on naïve bayes text classifiers. In: Proceeding of the 35th International ACM SIGIR 2012. Volume 1002., Portland, Oregon, USA (2012)
13. Cooper, W.S.: Some inconsistencies and misnomers in probabilistic information retrieval. In: Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '91, New York, NY, USA, ACM (1991) 57–61
14. Di Nunzio, G.: Using scatterplots to understand and improve probabilistic models for text categorization and retrieval. Int. J. Approx. Reasoning **50** (2009) 945–956
15. Di Nunzio, G., Sordoni, A.: A Visual Data Mining Approach to Parameters Optimization. In Zhao, Y., Cen, Y., eds.: Data Mining Applications in R. Elsevier (2013, In Press)

The Bivariate 2-Poisson model for IR

Giambattista Amati[1] and Giorgio Gambosi[2]

[1] Fondazione Ugo Bordoni, Rome, Italy gba@fub.it
[2] Enterprise Engineering Department of University of Tor Vergata, Rome, Italy
giorgio.gambosi@uniroma2.it

1 Introduction

Harter's 2-Poisson model of Information Retrieval is a univariate model of the raw term frequencies, that does not condition the probabilities on document length [2]. A bivariate stochastic model is thus introduced to extend Harter's 2-Poisson model, by conditioning the term frequencies of the document to the document length. We assume Harter's hypothesis: the higher the probability $f(X = x|L = l)$ of the term frequency $X = x$ is in a document of length l, the more relevant that document is. The new generalization of the 2-Poisson model has 5 parameters that are learned term by term through the EM algorithm over term frequencies data.

We explore the following frameworks:

- We assume that the observation $\langle x, l\rangle$ is generated by a mixture of k Bivariate Poisson (k-BP) distributions (with $k \geq 2$) with or without some conditions on the form for the marginal of the document length, that can reduce the complexity of the model. We here reduce for the sake of simplicity to $k = 2$. In the case of the 2-BP we also assume the hypothesis that the marginal distribution of l is a Poisson. The elite set is generated by the BP of the mixture with higher value for the mean of term frequencies, λ_1.
- The covariate variable Z_3 of length and term frequency λ_3 could be learned from covariance [3, page 103]. Instead, we here consider Z_3 a latent random variable which is learned by extending the EM algorithm in a standard way.
- Our plan is to compare the effectiveness of the bivariate 2-Poisson model with respect to standard models of IR, and in particular with some additional baselines that are obtained in our framework as follows:
  - applying the Double Poisson Model, which is the 2-BP with the marginal distributions that are independent.
  - Reducing to the univariate case (standard 2-Poisson model) by normalizing the term frequency x to a smoothed value $\mathbf{tfn}$. For example, we can use the Dirichlet smoothing:

    $$\mathbf{tfn} = \frac{x + \mu \cdot \hat{p}}{l + \mu} \cdot \mu'$$

    where μ and μ' are parameters and $\hat{p}$ is the term prior.

2 The Bivariate 2-Poisson distribution

In order to define the bivariate 2-Poisson model we need first to remind the definition of a bivariate Poisson model, that can be introduced in several ways, for example as limit of a bivariate binomial, as a convolution of three univariate Poisson distributions, as a compounding of a Poisson with a bivariate binomial. We find that the trivariate reduction method of the convolution more convenient to easily extend Harter's 2-Poisson model to the bivariate case.

Let us consider the random variables Z_1, Z_2, Z_3 distributed according to Poisson distributions $P(\lambda_i)$, that is:

$$p(Z_i = x|\lambda_i) = e^{-\lambda_i}\frac{\lambda_i^x}{x!}$$

and the random variables $X = Z_1 + Z_3$ e $Y = Z_2 + Z_3$ distributed according to a bivariate Poisson distribution, $BP(\Lambda)$, where $\Lambda = (\lambda_1, \lambda_2, \lambda_3)$:

$$p(X = x, Y = y|\Lambda) = e^{-(\lambda_1+\lambda_2+\lambda_3)}\frac{\lambda_1^x}{x!}\frac{\lambda_2^y}{y!}\sum_{i=0}^{min(x,y)}\binom{x}{i}\binom{y}{i}i!\left(\frac{\lambda_3}{\lambda_1\lambda_2}\right)^i$$

The corresponding marginal distributions turn out to be Poisson

$$p(X = x|\Lambda) = \sum_{y=0}^{\infty} p(X = x, Y = y|\Lambda) = P(\lambda_1 + \lambda_3)$$
$$p(Y = y|\Lambda) = \sum_{x=0}^{\infty} p(X = x, Y = y|\Lambda) = P(\lambda_2 + \lambda_3)$$

with covariance $Cov(X, Y) = \lambda_3$.

Let us now consider the mixture $2BP(\Lambda_1, \Lambda_2, \alpha)$, where $\Lambda_1 = (\lambda_1^1, \lambda_2^1, \lambda_3^1)$ and $\Lambda_2 = (\lambda_1^2, \lambda_2^2, \lambda_3^2)$, of two bivariate Poisson distributions

$$p(x, y|\Lambda_1, \Lambda_2, \alpha) = \alpha \cdot BP(\Lambda_1) + (1 - \alpha) \cdot BP(\Lambda_2)$$

The corresponding marginal distributions are 2-Poisson

$$p(x|\Lambda_1, \Lambda_2, \alpha) = \alpha \cdot P(\lambda_1^1 + \lambda_3^1) + (1 - \alpha) \cdot P(\lambda_1^2 + \lambda_3^2) = 2P(\lambda_1^1 + \lambda_3^1, \lambda_1^2 + \lambda_3^2, \alpha)$$
$$p(y|\Lambda_1, \Lambda_2, \alpha) = \alpha \cdot P(\lambda_2^1 + \lambda_3^1) + (1 - \alpha) \cdot P(\lambda_2^2 + \lambda_3^2) = 2P(\lambda_2^1 + \lambda_3^1, \lambda_2^2 + \lambda_3^2, \alpha)$$

In our case, we consider the random variables *x*, number of occurrences of the term in the document, and $L^- = l - x$, document length out of the term occurrences, and set $X = x$ and $Y = L^- = l - x$ (hence, Y could possibly be 0): as a consequence, we have $x = X = Z_1 + Z_3$, $L^- = Y = Z_2 + Z_3$, and $l = X + Y = Z_1 + Z_2 + 2Z_3$.

Moreover, we want *x* to be distributed as a *2-Poisson* and L^- to be distributed as a *Poisson*. By assuming $\lambda_2^1 = \lambda_2^2 = \lambda_2$ and $\lambda_3^1 = \lambda_3^2 = \lambda_3$ we obtain

$$p(x|\Lambda_1, \Lambda_2, \alpha) = \alpha \cdot P(\lambda_1^1 + \lambda_3) + (1 - \alpha) \cdot P(\lambda_1^2 + \lambda_3) = 2P(\lambda_1^1 + \lambda_3, \lambda_1^2 + \lambda_3)$$
$$p(L^-|\Lambda_1, \Lambda_2, \alpha) = \alpha \cdot P(\lambda_2 + \lambda_3) + (1 - \alpha) \cdot P(\lambda_2 + \lambda_3) = P(\lambda_2 + \lambda_3)$$

This implies that, apart from α, we assume five latent variables in the model, $Z_1^1, Z_1^2, Z_2, Z_3, W$ each Z is Poisson distributed with parameters $\lambda_1^1, \lambda_1^2, \lambda_2, \lambda_3$ respectively and W is a binary random variable Bernoulli distributed with parameter α. The resulting bivariate distribution is

$$\begin{aligned} p(x, L^-|\Lambda_1, \Lambda_2, \alpha) &= \alpha \cdot p_1(x, L^-|\lambda_1, \lambda_2, \lambda_3) + (1-\alpha) \cdot p_2(x, L^-|\lambda_1^2, \lambda_2, \lambda_3) \\ &= \alpha \cdot BP(\lambda_1^1, \lambda_2, \lambda_3) + (1-\alpha) \cdot BP(\lambda_1^2, \lambda_2, \lambda_3) \end{aligned}$$

3 EM algorithm for the Bivariate Poisson

Given a set of observations $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, with $\boldsymbol{x}_i = (x_i, l_i)$, we wish to apply maximum likelihood to estimate the set of parameters Λ of a bivariate Poisson distribution $p(\boldsymbol{x}|\Theta)$ fitting such data. We wish to derive the value of Θ by maximizing the log-likelihood, that is computing

$$\Theta^* = \arg\max_{\Theta} \log \mathcal{L}(\Theta|\mathcal{X}) = \arg\max_{\Theta} \log \prod_{i=1}^{n} p(\boldsymbol{x}_i|\Theta)$$

In our case (see also [1]), we are interested to a mixture of 2 Bivariate Poisson with latent variables Z_1^1, Z_1^2, Z_2, Z_3, since with respect to the general case we have now $Z_2^1 = Z_2^2 = Z_2$ and $Z_3^1 = Z_3^2 = Z_3$. Then, for each observed pair of values $\boldsymbol{x}_i = (x_i, l_i)$, $w_i = 1$ if $\boldsymbol{x}_i$ is generated by the first component, and $w_i = 2$ if generated by the second one. Accordingly:

$$\boldsymbol{z}_i = (z_{i1}^1, z_{i1}^2, z_{i2}, z_{i3}) \text{ are such that } x_i = \begin{cases} z_{i1}^1 + z_{i3} \text{ if } w_i = 1 \\ z_{i1}^2 + z_{i3} \text{ if } w_i = 2 \end{cases}$$
$$\text{and } l_i = z_{i2} + z_{i3}$$

EM algorithm requires, in our case, to consider the complete dataset

$$(\mathcal{X}, \mathcal{Z}) = \{(\boldsymbol{x}_1, \boldsymbol{z}_1, w_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{z}_n, w_n)\}$$

and the set of parameters is $\Theta = \Lambda_1 \cup \Lambda_2 \cup \{\alpha\}$, with $\Lambda_k = \left\{\lambda_1^k, \lambda_2, \lambda_3\right\}$. Let also $\Lambda = \Lambda_1 \cup \Lambda_2$.

3.1 Maximization

Let us consider the k-th M-step for Θ. We can show the following estimates:

$$\alpha^{(k)} = \frac{1}{n} \sum_{i=1}^{n} p_i^{(k-1)} \text{ where } p_i^{(k-1)} = \frac{\alpha^{(k-1)} p(\boldsymbol{x}_i|\Lambda_1^{k-1})}{\alpha^{(k-1)} p(\boldsymbol{x}_i|\Lambda_1^{(k-1)}) + (1-\alpha^{(k-1)}) p(\boldsymbol{x}_i|\Lambda_2^{(k-1)})}$$

and p is the Bivariate Poisson with parameters Λ_i, and

$$\lambda_1^{1(k)} = \frac{\sum_{i=1}^{n} b_{1i}^{1\ (k-1)} p_i^{(k-1)}}{\sum_{i=1}^{n} p_i^{(k-1)}} \qquad \lambda_1^{2(k)} = \frac{\sum_{i=1}^{n} b_{1i}^{2\ (k-1)} (1 - p_i^{(k-1)})}{\sum_{i=1}^{n} (1 - p_i^{(k-1)})}$$

$$\lambda_2^{(k)} = \frac{1}{n} \sum_{i=1}^{n} b_{2i}{}^{(k-1)} \qquad \lambda_3^{(k)} = \frac{1}{n} \sum_{i=1}^{n} b_{3i}{}^{(k-1)}$$

where ${b_{hi}^{j}}^{(k)} = E[Z_h^j | W = j, \boldsymbol{x}_i, \Lambda^{(k)}]$ and ${b_{hi}}^{(k)} = E[Z_h | \boldsymbol{x}_i, \Lambda^{(k)}]$ with $h = 1, 2, 3$.

3.2 Expectation

We can show that the expectations ${b_{1i}^{j}}^{(k)}$ and ${b_{hi}}^{(k)}$ are:

$$
\begin{aligned}
{b_{3i}}^{(k)} &= \sum_{r=0}^{\min(x_i, l_i)} r \cdot p(Z_3 = r | \boldsymbol{x}_i, \Lambda) \text{ where } \Lambda^{(k)} = \Lambda_1^{(k)} \cup \Lambda_2^{(k)} \\
&= \sum_{r=0}^{\min(x_i, l_i)} r \frac{(1-\alpha) p(Z_3 = r, \boldsymbol{x}_i | W = 2, \Lambda^{(k)}) + \alpha p(Z_3 = r, \boldsymbol{x}_i | W = 1, \Lambda^{(k)})}{p(\boldsymbol{x}_i | \Lambda^{(k)})} \\
{b_{1i}^{1}}^{(k)} &= E[X_1 | W = 1, \boldsymbol{x}_i] - E[Z_3 | W = 1, \boldsymbol{x}_i] = x_i - {b_{3i}^{1}}^{(k)} \\
{b_{1i}^{2}}^{(k)} &= E[X | W = 2, \boldsymbol{x}_i, \Lambda^{(k)}] - E[Z_3 | W = 2, \boldsymbol{x}_i, \Lambda^{(k)}] = x_i - {b_{3i}^{2}}^{(k)} \\
{b_{2i}}^{(k)} &= E[Y | \boldsymbol{x}_i, \Lambda^{(k)}] - E[Z_3 | \boldsymbol{x}_i, \Lambda^{(k)}] = l_i - {b_{3i}}^{(k)}
\end{aligned}
$$

where

$$
p(Z_3 = r, \boldsymbol{x}_i | W = j, \Lambda^{(k)}) = P_0(r | {\lambda_3}^{(k)}) \cdot P_0(x - r | {\lambda_1^{j}}^{(k)}) \cdot P_0(l - r | {\lambda_2}^{(k)})
$$

and P_0 is the univariate Poisson, $p(\boldsymbol{x}_i | \Lambda^{(k)})$ is the mixture of the bivariate Poisson. Efficient implementation of the bivariate Poisson through recursion can be found in [4].

4 Conclusions

We have implemented the EM algorithm for the univariate 2-Poisson and we are currently extending the implementation to the bivariate case.

The implementation will be soon available together with the results of the experimentation at the web site `http://tinyurl.com/cfcm8ma`.

References

1. BRIJS, T., KARLIS, D., SWINNEN, G., VANHOOF, K., WETS, G., AND MANCHANDA, P. A multivariate poisson mixture model for marketing applications. *Statistica Neerlandica 58*, 3 (2004), 322–348.
2. HARTER, S. P. A probabilistic approach to automatic keyword indexing. part I: On the distribution of specialty words words in a technical literature. *Journal of the ASIS 26* (1975), 197–216.
3. KOCHERLAKOTA, S., AND KOCHERLAKOTA, K. *Bivariate discrete distributions*. Marcel Dekker Inc., New York, 1992.
4. TSIAMYRTZIS, P., AND KARLIS, D. Strategies for efficient computation of multivariate poisson probabilities. *Communications in Statistics-Simulation and Computation 33*, 2 (2004), 271–292.

A Query Expansion Method based on a Weighted Word Pairs Approach

Francesco Colace[1], Massimo De Santo[1], Luca Greco[1] and Paolo Napoletano[2]

[1] DIEM,University of Salerno, Fisciano,Italy,
desanto@unisa, fcolace@unisa.it, lgreco@unisa.it
[2] DISCo,University of Milano-Bicocca, Italy
napoletano@disco.unimib.it

Abstract. In this paper we propose a query expansion method to improve accuracy of a text retrieval system. Our technique makes use of explicit relevance feedback to expand an initial query with a structured representation called Weighted Word Pairs. Such a structure can be automatically extracted from a set of documents and uses a method for term extraction based on the probabilistic Topic Model. Evaluation has been conducted on TREC-8 repository and performances obtained using standard WWP and Kullback Leibler Divergency query expansion approaches have been compared.

Keywords: Text retrieval, query expansion, probabilistic topic model

1 Introduction

Over the years, several text retrieval models have been proposed: set-theoretic (including boolean), algebraic, probabilistic models [1], etc. Although each method has its own properties, there is a common denominator: the *bag of words* representation of documents.

The "bag of words" assumption claims that a document can be considered as a feature vector where each element indicates the presence (or absence) of a word, so that the information on the position of that word within the document is completely lost [1]. The elements of the vector can be weights and computed in different ways so that a document can be considered as a list of weighted features. The *term frequency-inverse document (tf-idf)* model is a commonly used weighting model: each term in a document collection is weighted by measuring how often it is found within a document (*term frequency*), offset by how often it occurs within the entire collection (*inverse document frequency*). Based on this model, also a query can be viewed as a document, so it can be represented as a vector of weighted words.

The relevance of a document to a query is the distance between the corresponding vector representations in the features space. Unfortunately, queries performed by users may not be long enough to avoid the inherent ambiguity of language (polysemy etc.). This makes text retrieval systems, that rely on the

bags of words model, generally suffer from low precision, or low quality document retrieval. To overcome this problem, scientists proposed methods to expand the original query with other topic-related terms extracted from *exogenous* (e.g. ontology, WordNet, data mining) or *endogenous* knowledge (i.e. extracted only from the documents contained in the collection) [2, 3, 1]. Methods based on *endogenous* knowledge, also known as relevance feedback, make use of a number of labelled documents, provided by humans (explicit) or automatic/semi-automatic strategies, to extract topic-related terms and such methods have demonstrated to obtain performance improvements of up to 40% [4]

In this paper we propose a new query expansion method that uses a structured representation of documents and queries, named *Weighted Word Pairs*, that is capable of reducing the effect of the inherent ambiguity of language so achieving better performance than a method based on a vector of weighted words. The *Weighted Word Pairs* representation is automatically obtained from documents, provided by a minimal explicit feedback, by using a method of *term extraction*[5][6][7] based on the *Latent Dirichlet Allocation* model [8] implemented as the *Probabilistic Topic Model* [9]. Evaluation has been conducted on TREC-8 repository: results obtained employing standard WWP and Kullback Leibler divergency have been compared.

This article is structured as follows: Section 2 gives an overview on related works and approaches to query expansion in text retrieval; in Section 3 a general framework for query expansion is discussed; Section 4 describes in detail our feature extraction method; in Section 5 performance evaluation is presented.

2 Related works

It is well documented that the query length in typical information retrieval systems is rather short (usually two or three words) [10] which may not be long enough to avoid the inherent ambiguity of language (polysemy etc.), and which makes text retrieval systems, that rely on a term-frequency based index, suffer generally from low precision, or low quality of document retrieval.

In turn, the idea of taking advantage of additional knowledge, by expanding the original query with other topic-related terms, to retrieve relevant documents has been largely discussed in the literature, where manual, interactive and automatic techniques have been proposed [2][1]. The idea behind these techniques is that, in order to avoid ambiguity, it may be sufficient to better specify "the meaning" of what the user has in mind when performing a search, or in other words "the main concept" (or a set of concepts) of the preferred topic in which the user is interested. A better specialization of the query can be obtained with additional knowledge, that can be extracted from *exogenous* (e.g. ontology, WordNet, data mining) or *endogenous* knowledge (i.e. extracted only from the documents contained in the repository) [3, 1].

In this paper we focus on those techniques which make use of the *Relevance Feedback* (in the case of endogenous knowledge) which takes into account the results that are initially returned from a given query and so uses the information

about the relevance of each result to perform a new expanded query. In the literature we can distinguish between three types of procedures for the assignment of the relevance: explicit feedback, implicit feedback, and pseudo feedback.

Most existing methods, due to the fact that the human labeling task is enormously annoying and time consuming [11], make use of the pseudo relevance feedback (top k retrieved are assumed to be relevant). Nevertheless, fully automatic methods suffer from obvious errors when the initial query is intrinsically ambiguous. As a consequence, in the recent years, some hybrid techniques have been developed which take into account a minimal explicit human feedback [4, 12] and use it to automatically identify other topic related documents.

However, whatever the technique that selects the set of documents representing the feedback, the expanded terms are usually computed by making use of well known approaches for term selection as Rocchio, Robertson, CHI-Square, Kullback-Lieber etc [13]. In this case the reformulated query consists in a simple (sometimes weighted) list of words. Although such term selection methods have proven their effectiveness in terms of accuracy and computational cost, several more complex alternative methods have been proposed, which consider the extraction of a structured set of words instead of simple list of them: a weighted set of clauses combined with suitable operators [14], [15], [16].

3 A general Query Expansion framework

A general query expansion framework can be described as a modular system including:

- the Information Retrieval (IR) module;
- the Feedback (F) module;
- the Feature Extraction (FE) module;
- the Query Reformulation (QR) module.

Such a framework is represented in Figure 1 and can be described as follows. The user initially performs a search task on the dataset $\mathcal{D}$ by inputting a query $\mathbf{q}$ to the IR system and obtains a set of documents $\mathcal{RS} = (\mathbf{d}_1, \cdots, \mathbf{d}_N)$ as a result. The module F, thanks to the explicit feedback of the user, identifies a small set of relevant documents (called *Relevance Feedback*) $\mathcal{RF} = (\mathbf{d}_1, \cdots, \mathbf{d}_M)$ from the hit list of documents $\mathcal{RS}$ returned by the IR system. Given the set of relevant document $\mathcal{RF}$, the module FE extracts a set of features $\mathbf{g}$ that must be added to the initial query $\mathbf{q}$. The extracted features can be weighted words or more complex structures such as weighted word pairs. So the obtained set $\mathbf{g}$ must be adapted by the QR module to be handled by the IR system and then added to the initial query. The output of this module is a new query $\mathbf{qe}$ which includes both the initial query and the set of features extracted from the $\mathcal{RF}$. The new query is then performed on the collection so obtaining a new result set $\mathcal{RS}' = (\mathbf{d'}_1, \cdots, \mathbf{d'}_N)$ different from the one obtained before. Considering the framework described above is possible to take into account any technique of feature extraction that makes use of the explicit relevant feedback and any IR

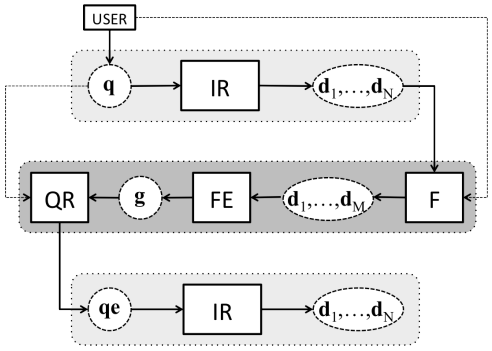


Fig. 1. General framework for Query Expansion.

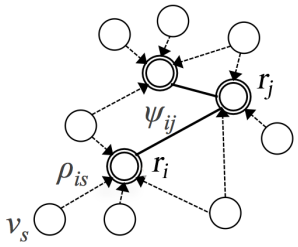


Fig. 2. Graphical representation of a *Weighted Word Pairs* structure.

systems suitable to handle the resulting expanded query **qe**. In this way it is possible to implement several techniques and make objective comparisons with the proposed one.

4 WWP feature selection method

The aim of the proposed method is to extract from a set of documents a compact representation, named *Weighted Word Pairs* (WWP), which contains the most discriminative word pairs to be used in the text retrieval task. The Feature Extraction module (FE) is represented in Fig. 3. The input of the system is the set of documents $\mathcal{RF} = (\mathbf{d}_1, \cdots, \mathbf{d}_M)$ and the output is a vector of weighted word pairs $\mathbf{g} = \{w'_1, \cdots, w'_{|\mathcal{T}_p|}\}$, where $\mathcal{T}_p$ is the number of pairs and w'_n is the weight associated to each pair (feature) $t_n = (v_i, v_j)$.

A WWP structure can be suitably represented as a *graph* **g** of terms (Fig. 2). Such a graph is made of several clusters, each containing a set of words v_s (*aggregates*) related to an *aggregate root* (r_i), a special word which represents the centroid of the cluster. How aggregate roots are selected will be clear further. The weight ρ_{is} can measure how a word is related to an aggregate root and can be expressed as a probability: $\rho_{is} = P(r_i|v_s)$. The resulting structure is a subgraph rooted on r_i. Moreover, *aggregate roots* can be linked together

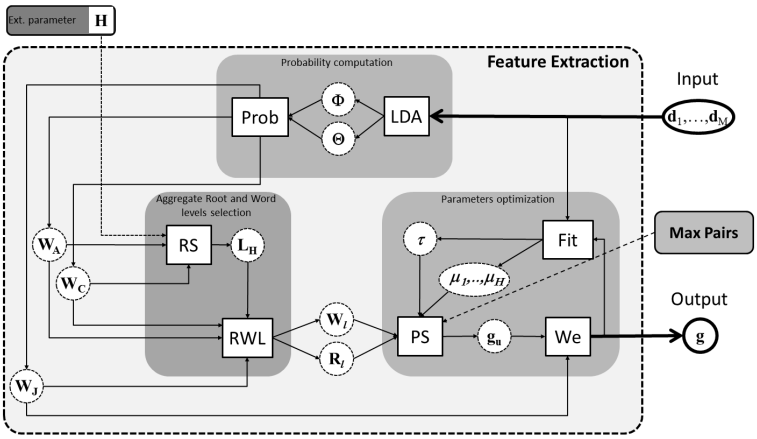


Fig. 3. Proposed feature extraction method. A *Weighted Word Pairs* **g** structure is extracted from a corpus of training documents.

building a centroids subgraph. The weight ψ_{ij} can be considered as the degree of correlation between two aggregate roots and can also be expressed as a probability: $\psi_{ij} = P(r_i, r_j)$. Being each aggregate root a special word, it can be stated that **g** contains directed and undirected pairs of features lexically denoted as words. Given the training set $\mathcal{RF}$ of documents, the term extraction procedure is obtained first by computing all the relationships between words and aggregate roots (ρ_{is} and ψ_{ij}), and then selecting the right subset of pairs $\mathcal{T}_{sp}$ from all the possible ones $\mathcal{T}_p$.

A WWP graph **g** is learned from a corpus of documents as a result of two important phases: the *Relations Learning* stage, where graph relation weights are learned by computing probabilities between word pairs (see Fig. 3); the *Structure Learning* stage, where an initial WWP graph, which contains all possible relations between aggregate roots and aggregates, is optimized by performing an iterative procedure. Given the number of aggregate roots H and the desired max number of pairs as constraints, the algorithm chooses the best parameter settings $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_H)$ and τ defined as follows:

1. μ_i: the threshold that establishes, for each aggregate root i, the number of *aggregate root/word* pairs of the graph. A relationship between the word v_s and the aggregate root r_i is relevant if $\rho_{is} \geq \mu_i$.
2. τ: the threshold that establishes the number of *aggregate root/aggregate root* pairs of the graph. A relationship between the aggregate root v_i and aggregate root r_j is relevant if $\psi_{ij} \geq \tau$.

4.1 Relations Learning

Since each aggregate root is lexically represented by a word of the vocabulary, we can write $\rho_{is} = P(r_i|v_s) = P(v_i|v_s)$, and $\psi_{ij} = P(r_i, r_j) = P(v_i, v_j)$. Considering that $P(v_i, v_j) = P(v_i|v_j)P(v_j)$, all the relations between words result from the computation of the joint or the conditional probability $\forall i, j \in \{1, \cdots, |\mathcal{T}|\}$ and $P(v_j)$ $\forall j$. An exact calculation of $P(v_j)$ and an approximation of the joint, or conditional, probability can be obtained through a smoothed version of the generative model introduced in [8] called Latent Dirichlet Allocation (LDA), which makes use of Gibbs sampling [9]. The original theory introduced in [9] mainly proposes a semantic representation in which documents are represented in terms of a set of probabilistic topics z. Formally, we consider a word u_m of the document $\mathbf{d}_m$ as a random variable on the vocabulary $\mathcal{T}$ and z as a random variable representing a topic between $\{1, \cdots, K\}$. A document $\mathbf{d}_m$ results from generating each of its words. To obtain a word, the model considers three parameters assigned: α, η and the number of topics K. Given these parameters, the model chooses θ_m through $P(\theta|\alpha) \sim Dirichlet(\alpha)$, the topic k through $P(z|\theta_m) \sim Multinomial(\theta_m)$ and $\beta_k \sim Dirichlet(\eta)$. Finally, the distribution of each word given a topic is $P(u_m|z, \beta_z) \sim Multinomial(\beta_z)$. The output obtained by performing Gibbs sampling on $\mathcal{RF}$ consists of two matrixes:

1. the *words-topics* matrix that contains $|\mathcal{T}| \times K$ elements representing the probability that a word v_i of the vocabulary is assigned to topic k: $P(u = v_i|z = k, \beta_k)$;
2. the *topics-documents* matrix that contains $K \times |\mathcal{RF}|$ elements representing the probability that a topic k is assigned to some word token within a document $\mathbf{d}_m$: $P(z = k|\theta_m)$.

The probability distribution of a word within a document $\mathbf{d}_m$ of the corpus can be then obtained as:

$$P(u_m) = \sum_{k=1}^{K} P(u_m|z = k, \beta_k)P(z = k|\theta_m). \tag{1}$$

In the same way, the joint probability between two words u_m and y_m of a document $\mathbf{d}_m$ of the corpus can be obtained by assuming that each pair of words is represented in terms of a set of topics z and then:

$$P(u_m, y_m) = \sum_{k=1}^{K} P(u_m, y_m|z = k, \beta_k)P(z = k|\theta_m) \tag{2}$$

Note that the exact calculation of Eq. 2 depends on the exact calculation of $P(u_m, y_m|z = k, \beta_k)$ that cannot be directly obtained through LDA. If we assume that words in a document are conditionally independent given a topic, an approximation for Eq. 2 can be written as [5, 6]:

$$P(u_m, y_m) \simeq \sum_{k=1}^{K} P(u_m|z = k, \beta_k)P(y_m|z = k, \beta_k)P(z = k|\theta_m). \tag{3}$$

Moreover, Eq. 1 gives the probability distribution of a word u_m within a document $\mathbf{d}_m$ of the corpus. To obtain the probability distribution of a word u independently of the document we need to sum over the entire corpus:

$$P(u) = \sum_{m=1}^{M} P(u_m)\delta_m \tag{4}$$

where δ_m is the prior probability for each document ($\sum_{m=1}^{|\mathcal{RF}|} \delta_m = 1$). If we consider the joint probability distribution of two words u and y, we obtain:

$$P(u, y) = \sum_{m=1}^{M} P(u_m, y_v)\delta_m \tag{5}$$

Concluding, once we have $P(u)$ and $P(u, y)$ we can compute $P(v_i) = P(u = v_i)$ and $P(v_i, v_j) = P(u = v_i, y = v_j)$, $\forall i, j \in \{1, \cdots, |\mathcal{T}|\}$ and so the relations learning can be totally accomplished.

4.2 Structure Learning

Once each ψ_{ij} and ρ_{is} is known $\forall i, j, s$, aggregate root and word levels have to be identified in order to build a starting WWP structure to be optimized as discussed later. The first step is to select from the words of the indexed corpus a set of aggregate roots $\mathbf{r} = (r_1, \ldots, r_H)$, which will be the nodes of the centroids subgraph. Aggregate roots are meant to be the words whose occurrence is most implied by the occurrence of other words of the corpus, so they can be chosen as follows:

$$r_i = \text{argmax}_{v_i} \prod_{j \neq i} P(v_i|v_j) \tag{6}$$

Since relationships' strenghts between aggregate roots can be directly obtained from ψ_{ij}, the centroids subgraph can be easily determined. Note that not all possible relationships between aggregate roots are relevant: the threshold τ can be used as a free parameter for optimization purposes. As discussed before, several words (aggregates) can be related to each aggregate root, obtaining H aggregates' subgraphs. The threshold set $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_H)$ can be used to select the number of relevant pairs for each aggregates' subgraph. Note that a relationship between the word v_s and the aggregate root r_i is relevant if $\rho_{is} \geq \mu_i$, but the value ρ_{is} cannot be directly used to express relationships' strenghts between aggregate roots and words. In fact, being ρ_{is} a conditional probability, it is always bigger than ψ_{is} which is a joint probability. Therefore, once pairs for the aggregates' subgraph are selected using ρ_{is}, relationships' strenght are represented on the WWP structure through ψ_{is}.
Given H and the maximum number of pairs as constraints (i.e. fixed by the user), several WWP structure $\mathbf{g}_t$ can be obtained by varying the parameters $\Lambda_t = (\tau, \mu)_t$. As shown in Fig.3, an optimization phase is carried out in order to search the set of parameters Λ_t which produces the best WWP graph

[6]. This process relies on a scoring function and a searching strategy that will be now explained. As we have previously seen, a $\mathbf{g}_t$ is a vector of features $\mathbf{g}_t = \{b_{1t}, \ldots, b_{|\mathcal{T}_{sp}|t}\}$ in the space $\mathcal{T}_{sp}$ and each document of the training set $\mathcal{RF}$ can be represented as a vector $\mathbf{d}_m = (w_{1m}, \ldots, w_{|\mathcal{T}_{sp}|m})$ in the space $\mathcal{T}_{sp}$. A possible scoring function is the cosine similarity between these two vectors:

$$\mathcal{S}(\mathbf{g}_t, \mathbf{d}_m) = \frac{\sum_{n=1}^{|\mathcal{T}_{sp}|} b_{nt} \cdot w_{nm}}{\sqrt{\sum_{n=1}^{|\mathcal{T}_{sp}|} b_{nt}^2} \cdot \sqrt{\sum_{n=1}^{|\mathcal{T}_{sp}|} w_{nm}^2}} \tag{7}$$

and thus the optimization procedure would consist in searching for the best set of parameters Λ_t such that the cosine similarity is maximized $\forall \mathbf{d}_m$. Therefore, the best $\mathbf{g}_t$ for the set of documents $\mathcal{RF}$ is the one that produces the maximum score attainable for each document when used to rank $\mathcal{RF}$ documents. Since a score for each document $\mathbf{d}_m$ is obtained, we have:

$$\mathbf{S}_t = \{\mathcal{S}(\mathbf{g}_t, \mathbf{d}_1), \cdots, \mathcal{S}(\mathbf{g}_t, \mathbf{d}_{|\mathcal{RF}|})\},$$

where each score depends on the specific set $\Lambda_t = (\tau, \mu)_t$. To compute the best value of Λ we can maximize the score value for each document, which means that we are looking for the graph which best describes each document of the repository from which it has been learned. It should be noted that such an optimization maximizes at the same time all $|\mathcal{RF}|$ elements of $\mathbf{S}_t$. Alternatively, in order to reduce the number of the objectives being optimized, we can at the same time maximize the mean value of the scores and minimize their standard deviation, which turns a multi-objective problem into a two-objective one. Additionally, the latter problem can be reformulated by means of a linear combination of its objectives, thus obtaining a single objective function, i.e., *Fitness* ($\mathcal{F}$), which depends on Λ_t,

$$\mathcal{F}(\Lambda_t) = E\left[\mathbf{S}_t\right] - \sigma\left[\mathbf{S}_t\right],$$

where E is the mean value of all the elements of $\mathbf{S}_t$ and σ_m is the standard deviation. By summing up, the parameters learning procedure is represented as follows, $\Lambda^* = \text{argmax}_t\{\mathcal{F}(\Lambda_t)\}$.

Since the space of possible solutions could grow exponentially, $|\mathcal{T}_{sp}| \leq 300$ [3] has been considered. Furthermore, the remaining space of possible solutions has been reduced by applying a clustering method, that is the *K-means* algorithm, to all ψ_{ij} and ρ_{is} values, so that the optimum solution can be exactly obtained after the exploration of the entire space.

5 Method validation

The proposed approach has been validated using IR systems that allow to handle structured queries composed of weighted word pairs. For this reason,the following open source tools were considered: Apache Lucene[4] which supports structured

[3] This number is usually employed in the case of Support Vector Machines.
[4] We adopted the version 2.4.0 of Lucene

query based on a weighted boolean model and Indri[5] which supports an extended set of probabilistic structured query operators based on INQUERY. The performance comparison was carried out testing the following FE/IR configurations:

- **IR only**. Unexpanded queries were performed using first Lucene and then Lemur as IR modules. Results obtained in these cases are referred as baseline.
- **FE(WWP) + IR**. Our WWP-based feature extraction method was used to expand initial query and feed Lucene and Lemur IR modules.
- **FE(KLD) + IR**. Kullback Leibler Divergency based feature extraction was used to expand initial query and feed Lucene and Lemur IR modules.

5.1 Datasets and Ranking Systems

The dataset from TREC-8 [17] collections (minus the Congressional Record) was used for performance evaluation. It contains about 520,000 news documents on 50 topics (no.401-450) and relevance judgements for the topics. Word stopping and word stemming with single keyword indexing were performed. Query terms for each topic's initial search (baseline) were obtained by parsing the title field of a topic. For the baseline and for the first pass ranking (needed for feedback document selection) the default similarity measures provided by Lucene and Lemur has been used. Performance was measured with TREC's standard evaluation measures: mean average precision (MAP), precision at different levels of retrieved results (P@5,10...1000), R-precision and binary preference (BPREF).

5.2 Parameter Tuning

The two most important parameters involved in the computation of WWP, given the number of documents for training, are the *number of aggregate roots H* and the *number of pairs*. The number of aggregate roots can be chosen as a trade off between retrieval performances and computational times, our choice was $H = 4$ since it seemed to be the best compromise (about 6 seconds per topic)[6]. However, we want to emphasize method effectiveness more than algorithm efficiency since algorithm coding has not been completely optimized yet.

Fig. 5.2 shows results of baseline and WWP method when changing *number of pairs* from 20 to 100 where the number of documents is fixed to 3: in this analysis, Lucene IR module is used . According to the graph, our system always provides better performances than baseline; the change in number of pairs has a great impact especially on precision at 5 where 60 pairs achieve the best results. Anyway, if we consider precision at higher levels together with map values, 50 pairs seem to be a better choice also for shorter computational times. Fig. 5.2 shows results of baseline and our method when changing *number of training documents* (Lucene IR Module used): here we can see that the overall behaviour of the system is better when choosing 3 relevant documents for training.

[5] We adopted the version 5... that is part of the Lemur Toolkit

[6] Results were obtained using an *Intel Core 2 Duo 2,40 GHz* PC with *4GB RAM* with no other process running.

Once again the system outperforms baseline especially at low precision levels. Discussed analysis led us to choose the following settings for the experimental stage: 4 aggregate roots, 50 pairs, 3 training documents.

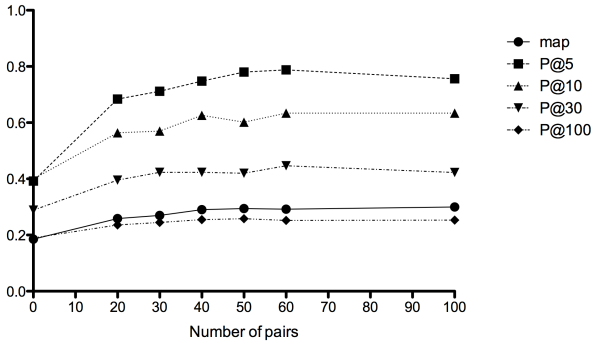


Fig. 4. *WWP* performance when changing number of pairs.

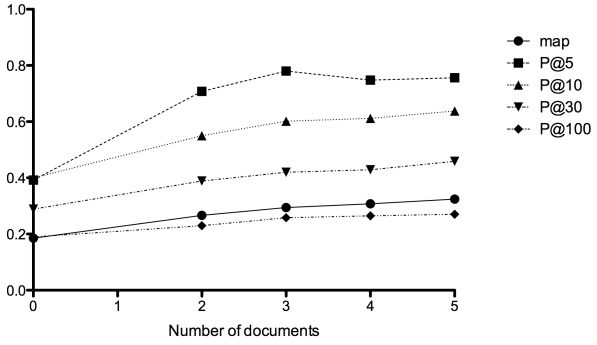


Fig. 5. *WWP* performance when changing number of training documents

5.3 Comparison with other methods

In Table 1 WWP method is compared with baseline and *Kullback-Leibler* divergence based method [13] when using both Lucene and Lemur as IR modules. Here we see that WWP outscores KLD, and baseline especially for low level precision while having good performances for other measures. However these results are obtained without removing feedback documents from the dataset so

IR	**Lucene**			**Lemur**		
FE	-	KLD	WWP	-	KLD	WWP
relret	2267	2304	3068	2780	2820	3285
map	0,1856	0,1909	0,2909	0,2447	0,2560	0,3069
Rprec	0,2429	0,2210	0,3265	0,2892	0,2939	0,3324
bpref	0,2128	0,2078	0,3099	0,2512	0,2566	0,3105
P@5	0,3920	0,5200	0,7600	0,4760	0,5720	0,7360
P@10	0,4000	0,4300	0,6020	0,4580	0,4820	0,5800
P@100	0,1900	0,1744	0,2612	0,2166	0,2256	0,2562
P@1000	0,0453	0,0461	0,0614	0,0556	0,0564	0,0657

Table 1. Results comparison for unexpanded query, KLD and WWP (FE) using Lucene and Lemur as IR modules.

IR	**Lucene**			**Lemur**		
FE	-	KLD	WWP	-	KLD	WWP
relret	2117	2178	2921	2630	2668	3143
map	0,1241	0,1423	0,2013	0,1861	0,1914	0,2268
Rprec	0,1862	0,1850	0,2665	0,2442	0,2454	0,2825
bpref	0,1546	0,1716	0,2404	0,1997	0,2044	0,2471
P@5	0,2360	0,3920	0,4840	0,3880	0,4120	0,5120
P@10	0,2580	0,3520	0,4380	0,3840	0,3800	0,4560
P@100	0,1652	0,1590	0,2370	0,1966	0,2056	0,2346
P@1000	0,0423	0,0436	0,0584	0,0526	0,0534	0,0629

Table 2. Results comparison for unexpanded query, KLD and WWP using Lucene or Lemur with RSD.

a big improvement in low level precision may appear a little obvious. Another performance evaluation was carried out using only the residual collection (RSD) where feedback documents are removed. Results for this evaluation are shown in table 2 where we see performance improvements also with residual collection.

6 Conclusions

In this work we have demonstrated that a Weighted Word Pairs hierarchical representation is capable of retrieving a greater number of relevant documents than a less complex representation based on a list of words. These results suggest that our approach can be employed in all those text mining tasks that consider matching between patterns represented as textual information and in text categorization tasks as well as in sentiment analysis and detection tasks. The proposed approach computes the expanded queries considering only endogenous

knowledge. It is well known that the use of external knowledge, for instance Word-Net, could clearly improve the accuracy of information retrieval systems and we consider this integration as a future work.

References

1. Christopher D. Manning, P.R., Schtze, H.: Introduction to Information Retrieval. Cambridge University (2008)
2. Efthimiadis, E.N.: Query expansion. In Williams, M.E., ed.: Annual Review of Information Systems and Technology. (1996) 121–187
3. Bhogal, J., Macfarlane, A., Smith, P.: A review of ontology based query expansion. Information Processing & Management **43**(4) (2007) 866 – 886
4. Okabe, M., Yamada, S.: Semisupervised query expansion with minimal feedback. IEEE Transactions on Knowledge and Data Engineering **19** (2007) 1585–1589
5. Napoletano, P., Colace, F., De Santo, M., Greco, L.: Text classification using a graph of terms. In: Complex, Intelligent and Software Intensive Systems (CISIS), 2012 Sixth International Conference on. (july 2012) 1030 –1035
6. Clarizia, F., Greco, L., Napoletano, P.: An adaptive optimisation method for automatic lightweight ontology extraction. In Filipe, J., Cordeiro, J., eds.: Enterprise Information Systems. Volume 73 of Lecture Notes in Business Information Processing. Springer Berlin Heidelberg (2011) 357–371
7. Clarizia, F., Greco, L., Napoletano, P.: A new technique for identification of relevant web pages in informational queries results. In: Proceedings of the 12th International Conference on Enterprise Information Systems: Databases and Information Systems Integration. (8-12 June 2010) 70–79
8. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3**(993–1022) (2003)
9. Griffiths, T.L., Steyvers, M., Tenenbaum, J.B.: Topics in semantic representation. Psychological Review **114**(2) (2007) 211–244
10. Jansen, B.J., Spink, A., Saracevic, T.: Real life, real users, and real needs: a study and analysis of user queries on the web. Inform. Proces. & Manag. **36**(2) (2000) 207–227
11. Ko, Y., Seo, J.: Text classification from unlabeled documents with bootstrapping and feature projection techniques. Inf. Process. Manage. **45** (2009) 70–83
12. Dumais, S., Joachims, T., Bharat, K., Weigend, A.: SIGIR 2003 workshop report: implicit measures of user interests and preferences. **37**(2) (2003) 50–54
13. Carpineto, C., de Mori, R., Romano, G., Bigi, B.: An information-theoretic approach to automatic query expansion. ACM Trans. Inf. Syst. **19** (2001) 1–27
14. Callan, J., Croft, W.B., Harding, S.M.: The inquery retrieval system. In: In Proceedings of the Third International Conference on Database and Expert Systems Applications, Springer-Verlag (1992) 78–83
15. Collins-Thompson, K., Callan, J.: Query expansion using random walk models. In: Proceedings of the 14th ACM international conference on Information and knowledge management. CIKM '05, New York, NY, USA, ACM (2005) 704–711
16. Lang, H., Metzler, D., Wang, B., Li, J.T.: Improved latent concept expansion using hierarchical markov random fields. In: Proceedings of the 19th ACM international conference on Information and knowledge management. CIKM '10, New York, NY, USA, ACM (2010) 249–258
17. Voorhees, E.M., Harman, D.: Overview of the eighth text retrieval conference (trec-8) (2000)

A flexible extension of XQuery Full-Text

Emanuele Panzeri and Gabriella Pasi

University of Milano-Bicocca
Viale Sarca 336, 20126 Milano, Italy
{panzeri,pasi}@disco.unimib.it

Abstract. This paper presents the implementation of an extension of the XQuery Full-Text language on top of the BaseX query engine. The proposed extension adds to the language two new flexible axes that allow users to express structural constraints that are evaluated in an approximate way with respect to a considered path; the constraints evaluation produces a scored set of elements. The implementation and the efficiency evaluations of the constraints are reported in this paper.

1 Introduction

Recent works have been dedicated at improving standard XML query languages, such as XQuery and XPath, by enriching their expressiveness in both content constraints [1, 8] and structural constraints [2, 6] evaluation. While the work reported in [1] has been adopted by W3C in the XQuery Full-Text extension[7], no approximate matching for structural-based constraints has been standardized by W3C yet. The adoption of structured query models (such as XQuery) to inquiry highly structured document repositories or XML databases forces users to be well aware of the underlying structure; non of the previous approaches allows users to directly specify flexible structural constraints the evaluation of which produces weighted fragments. The XQuery Full-Text language extension proposed in [5] was the first proposal to introduce a set of flexible constraints with an approximate matching. The extension allows users to formulate queries where the relative position of important nodes can be specified independently from an exact knowledge of the underlying structure. The extension gives to the user the ability to express structural constraints with approximate matching and to obtain a weighted set of fragments; users can also define a score combination using standard XQuery operators and obtain a customized element ranking.

In this work we present the implementation of the flexible constraints, as defined and motivated in [5], named `Near` and `Below`, that allow users to explicitly specify their tolerance to an approximate structural matching. The implementation, performed on top of the BaseX query engine [4], integrates and extends the fragment scoring introduced by the FullText extension by taking into account also the structural scores computed by the approximate constraint evaluation.

2 The XQuery Full-Text extension in a nutshell

For each element matched by the `Below` and `Near`, a score is computed by the approximate matching; the score is in the interval $]0, 1]$ where 1 represents a full

satisfaction of the constraint evaluation, while values less than 1 are assigned to target nodes *far* from the context node.

The constraint `Below` is defined as an XPath axis (like, for example, the `children`, `self`, etc axes) the evaluation of which is aimed at identifying elements that are direct descendants of a node. The `Below` constraint is specified as: `c/below::t`, where `c` is the *context* node, and `t` is the target node. The score computed by the `Below` axis evaluation, is computed by the formula: $w_{below}(c,t) = \frac{1}{|desc\_arcs(c,t)|}$. Where $desc\_arcs(c,t)$ is a function that returns the set of unique descending arcs from c to t.

The constraint `Near` is specified as a flexible axis of a path expression; it allows to identify XML elements connected through *any path* to the *context node*. The axis allows to define a maximum distance n that acts as a threshold on the number of arcs between the context node and the target node; nodes the distance of which is more then n arcs are filtered out from the possible results. The `Near` syntax is: `c/near(n)::t` and the score for its evaluation is computed as: $w_{near}(c,t,n) = \begin{cases} \frac{1}{|arcs(c,t)|} & \text{if } |arcs(c,t)| \leq n \\ 0 & \text{else.} \end{cases}$ where c is the context node, t is the current target node, n is the maximum allowed distance and $arcs(c,e)$ returns the set of arcs in the shortest path between c and t.

3 Implementation

The new axes have been integrated into the BaseX XQuery engine by extending both its language interpreter and its XQuery evaluation processor to include a new `score-structure` Score Variable definition. BaseX has been chosen for being the first system (and the only one, to the best of our knowledge) to implement the full XQuery Full-Text language. As described in [3], BaseX adopts an efficient indexing schema for XML documents. The XQuery FLOWR clauses have been made capable to identify the new structural score variable, and to allow its usage in sorting, ordering and results display. As an example the XQuery `for` clause has been extended as follows:

```
ForClause ::="for" "$" VarName TypeDeclaration? PositionalVar? FTScoreVar?
  StructScoreVar? "in" ExprSingle ("," "$" VarName TypeDeclaration?
  PositionalVar? FTScoreVar? "in" ExprSingle)*
StructScoreVar ::= "score-structure" "$" VarName
```

where `Varname` is a valid variable name; `TypeDeclaration` is a variable type declaration; and `ExprSingle` is the actual query for node selection as defined in the XQuery language. From the user point of view this approach offers unlimited possibilities of the usage of the new Structure Score Variable: the user can define aggregation functions using the default XQuery constructs.

Fig. 1a shows an example of the `Near` constraint application: the query `person//act/near::title` is evaluated and the three gray `title` nodes are matched with a score of 0.3, for the `act/movie/title` node, and 0.25 for the other two nodes. In Fig. 1b the evaluation of the query `person/below::name` is shown: three `name` nodes are retrieved; `person/name` node with a score of 1 and 0.3 for the other nodes.

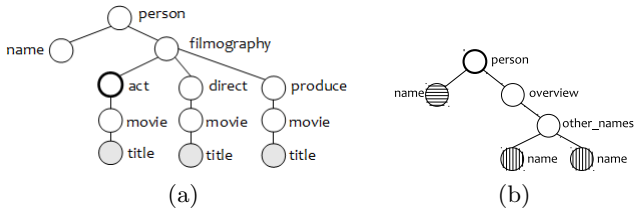


Fig. 1: Example of (a) `Near` and (b) `Below` constraint evaluation.

4 Evaluation

The performed evaluations compare the efficiency of all the new axes constraints with the *standard* XPath/XQuery counterparts (if available): in particular for the `Below` axis evaluation we executed each query using both the `Below` and the `descendant` axes. Concerning the `Near` axis evaluation, instead, no counterpart could be identified due to the innovative nature of the proposed axis.

The axes evaluations have been performed by using the IMDB INEX Data-Centric collection. Performance tests have been executed with an increasing size of the evaluated collection to verify the overhead introduced by the flexible axis evaluation in comparison with standard (if applicable) XPath axes constraints. Due to the nature of the BaseX indexing system that caches queries, result set, and opened databases, the evaluations have been performed by unloading the BaseX system between each run. All evaluation tests have been executed 5 times, and the average timings (removing the worst and the best results) are presented.

`Below` axis evaluation: The `Below` axis has been compared with the standard `descendant` axis: both axes have been evaluated by executing the test without any query optimization introduced by BaseX. Five queries containing the `Below` axis have been evaluated against each collection by measuring its execution time. The same query, with the `Below` axis replaced by the `descendant` axis has then been executed and its timings compared. In Fig. 2 the evaluation results are sketched: not surprisingly the `Below` axis evaluation takes more time than the equivalent `descendant` axis to obtain the query results, due to the computation of the structural score. The `Below` axis evaluation takes in average 36% more time than the execution of the `descendant` counterpart.

`Near` axis evaluation: The `Near` axis evaluation has been performed by using the same IMDb collection used for the evaluation of the `Below` axis. The queries used during the evaluation process have been defined so as to require the BaseX engine to retrieve all the XML elements without neither adopting any optimization strategy nor any query re-writing; this aspect forced the BaseX system to perform a sequential analysis of the target nodes, and thus to provide a complete execution of the `Near` axis evaluation. Furthermore the BaseX Full-Text index has been avoided, further enforcing the complete iteration over any target node without using any BaseX pre-pruning strategy. These aspects allowed to measure the efficiency of the `Near` axis evaluation implementation.

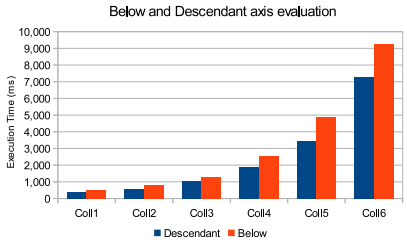


Fig. 2: Comparison between `Below` and `descendant` axis evaluation.

5 Conclusions and Future Work

The `Below` and `Near` axes, semantically and syntactically defined in [5] have been implemented and evaluated on top of the BaseX system, where both the query interpreter and the evaluation engine have been extended to identify and evaluate the new axes. The obtained results confirm that, although the flexible evaluation of both axes requires relatively longer times, the proposed flexible evaluation and the subsequent XML element ranking based on both textual and structural constraints can be successfully introduced into the XQuery language. Ongoing works are being conducted related to the definition, alongside the BaseX data structures, of ad-hoc indexes to better evaluate the new flexible constraints by adopting efficient pruning techniques during target node identification, thus further improving the axis evaluation performance.

References

1. S. Amer-Yahia, C. Botev, and J. Shanmugasundaram. TeXQuery: A Full-Text Search Extension to XQuery. In *WWW '04*, pages 583–594. ACM, 2004.
2. S. S. Bhowmick, C. Dyreson, E. Leonardi, and Z. Ng. Towards non-directional Xpath evaluation in a RDBMS. In *CIKM '09*, pages 1501–1504, 2009.
3. C. Grün. *Storing and Querying Large XML Instances*. PhD thesis, Universität Konstanz, December 2010.
4. C. Grün, S. Gath, A. Holupirek, and M. H. Scholl. XQuery Full Text Implementation in BaseX. In *XSym '09*, pages 114–128, 2009.
5. E. Panzeri and G. Pasi. An Approach to Define Flexible Structural Constraints in XQuery. In *AMT*, pages 307–317, 2012.
6. B. Truong, S. Bhowmick, and C. Dyreson. Sinbad:towards structure-independent querying of common neighbors xml databases. In *DASFAA'12*, pages 156–171. 2012.
7. W3C. XQuery/XPath FullText. www.w3.org/TR/xpath-full-text-10, March 2011.
8. C. Yu and H. V. Jagadish. Querying Complex Structured Databases. In *VLDB '07*, pages 1010–1021, 2007.

Towards a qualitative analysis of diff algorithms

Gioele Barabucci, Paolo Ciancarini, Angelo Di Iorio, and Fabio Vitali

Department of Computer Science and Engineering, University of Bologna

Abstract. This paper presents an ongoing research on the qualitative evaluation of diff algorithms and the deltas they produce. Our analysis focuses on qualities that are seldom studied: instead of evaluating the speed or the memory requirements of an algorithm, we focus on how much *natural*, compact and fit for use in a certain context the produced deltas are. This analysis started as a way to measure the naturalness of the deltas produced by JNDiff, a diff algorithm for XML-based literary documents. The deltas were considered natural if they expressed the changes in a way similar to how a human expert would do, an analysis that could only be carried out manually. Our research efforts have expanded into the definition of a set of metrics that are, at the same time, more abstract (thus they capture a wider range of information about the delta) and completely objective (so they can be computed by automatic tools without human supervision).

1 Challenges in evaluating diff algorithms and deltas

The diff algorithms have been widely studied in literature and applied to very different domains (source code revision, software engineering, collaborative editing, law making, etc.) and data structures (plain text, trees, graphs, ontologies, etc.). Their output is usually expressed as *edit script*s, also called *deltas* or *patch*es. A delta is a set of operations that can be applied to the older document in order to obtain the newer one. Deltas are hardly ever unique, since multiple different sequences of operations can be devised, all capable of generating the newer document from the older one.

Each algorithm uses its own strategies and data-structures to calculate the "best" delta. Some of them are very fast, others use a limited amount of memory, others are specialized for use in a specific domain and data format. Surprisingly enough, the evaluation of the quality of the deltas has received little attention.

The historical reason is that most algorithms have been proposed by the database community focusing more on efficiency rather than quality. Another reason is that the produced deltas are not easily comparable: not only each algorithm choose different sequences of changes, but they even use their own internal model and recognize their own set of changes. For example, some algorithms detect moves while others do not, or the same name is used for different operations. Given this degree of heterogeneity, it is hard to evaluate the quality of these algorithms in an automatic and objective way.

Nonetheless, we believe that such an evaluation is essential for the final users and can effectively support them in selecting the best algorithm for their needs.

It is important, for instance, that the operations contained in a delta reflect meaningful changes to the documents, i.e., the detected changes are as close as possible to the editing operations that were actually performed by the author on the original document. Our research started by studying a way to make this quality more explicit.

2 A first manual approach: *naturalness* in JNDiff

In [2] we proposed an explicit metric useful in the evaluation, design and implementation of algorithms for diffing literary XML documents: the *naturalness*. Naturalness indicates how much an edit script resembles the changes effectively performed by the author on a document.

In relation to naturalness, in [2] we:

- discussed an extensible set of natural operations that diff algorithms should be able to detect, focusing on text-centric documents;
- presented an algorithm (NDiff) that detects many of these natural changes;
- described JNDiff, a Java implementation of the NDiff algorithm together with tools to apply the delta produced and highlight modifications;
- presented a case study in detecting changes in XML-encoded legislative bills, and described the benefits of natural deltas in improving the editing and publishing workflow of such documents.

In our view, naturalness is a property connected to the human application and interpretation of document editing, i.e., it can be fully validated only by people that know the editing process. That is why we started researching into other more objective ways to indirectly measure the naturalness.

The first approximation was to examine how close was the generated delta to the description of the changes given by an expert. The first step to calculate such approximation is to create a *gold standard* (an ideal edit script) by comparing pairs of document versions, both visually and structurally. The second step is to identify *clusters of changes* in the delta of each algorithm that correspond to the changes in the gold standard, and to assign a similarity value to each one, depending on a number of parameters. Most of these operations are manual: we manually generate the gold standard, then we manually link the changes in the delta to the clusters, dealing with different output formats.

The key part of the second step consists in assigning a score to each cluster of edit operations to assess its naturalness, taking into account these aspects:

1. *Minimality of the cluster*: we consider a cluster composed of a few sophisticated changes more natural than a cluster composed of many basic changes.
2. *Minimality of the number of nodes*: we rate as more natural clusters that affect fewer nodes, either elements or text characters: this penalizes imprecise edit scripts and rewards scripts in which only the needed nodes are modified.
3. *Minimality of the length of text nodes*: we regard as more natural the edit scripts in which the basic unit for text modifications are the single words, not whole paragraphs; the insertion/removal of big chunks of text where only few characters have been changed is considered verbose and not natural.

The score of the i-th cluster of the delta is calculated as the inverse of a weighted sum of the above mentioned parameters. Further coefficients m_e, m_n and m_c are needed to balance the weights (because, for instance, the number of characters is on average much higher than the number of edit actions or of affected nodes).

$$nat\,(\Delta, i) = (w_e \times m_e \times EDITS_i + w_n \times m_n \times NODES_i + w_c \times m_c \times CHARS_i)^{-1}$$

Eventually, after various experiments on real-world documents, we instantiated the general formula as:

$$nat(\Delta, i) = (0.2 \times EDITS_i + 0.1 \times NODES_i + 0.0077 \times CHARS_i)^{-1}$$

3 Automated analysis

The JNDiff naturalness formula as presented has two main issues. First, its evaluation is impossible to automate as it requires the identification of a gold standard for each cluster of each delta and the ability to match the generated changes to the corresponding changes in the gold standard. Second, different users have different requests and expectations for a diff system and a single metric is not enough to show how well these requests are matched. Not only these requests are different, often they also conflict with each other: for example in certain cases a cursory summary of what has changed is enough, while in others an extreme level of detail is needed.

The idea of measuring objective indicators on the delta, on the other hand, is promising. The same idea of naturalness could be generalized and seen as one of the many qualities of a delta.

In order to define metrics for analyzing deltas under multiple aspects and in an objective way, one has to rely on properties that can be extracted and elaborated by automatic tools without resorting to human evaluations. To reach this goal we elaborated a universal delta model that works on linear texts, trees and graphs. This universal delta model is based on the concept of iterative recognition of more meaningful changes starting from simple changes. Using this model we extracted the properties shown in table 1.

By themselves, the values of these properties say little about the various qualities of the delta. However, once these properties have been extracted they form the base upon which we build several metrics, each of which focused on a single aspect of the analyzed delta. For instance the measure of how much redundant information has been included in a delta (the so called *conciseness* metric) uses two properties of deltas: the number of modified elements and the number of referenced-yet-not-modified elements. We derived four key metrics, described in table 2.

Preliminary experiments showed that these metrics are useful to characterize diff algorithms. In particular, we compared the deltas produced by three well-known XML diff tools (JNDiff [2], XyDiff [1] and Faxma [3]) on a small dataset of real documents. The metrics highlighted, for example, the tendency of some

Property	Definition
population	The total number of changes of which a change is composed of, including itself.
depth	The maximum number of encapsulation layers that must be crossed to reach an atomic change.
width	The number of distinct changes encapsulated inside the change.
touched elements	The number of distinct pieces of information that are included as part of the change or of the encapsulated changes.
modified elements	The minimum number of pieces of information that must be modified by the change to fulfill its purpose.
number of top-level	The number of changes that are not encapsulated in any other change.

Table 1. Properties of changes and deltas

Metric	Definition	Formula
Precision	How many non modified elements have been included in the delta.	$\frac{modified-elements(\delta)}{touched-elements(\delta)}$
Conciseness	How much the changes found in the delta have been grouped into bigger changes.	$1 - \frac{\#top-level(\delta)}{population(\delta)}$
Meaningfulness	How much of the delta conciseness is due to the use of complex changes.	$\frac{\#top-level_{complex}(\delta)}{\#top-level(\delta)}$
Aggregation	How much of the inner parts of the delta is expressed using complex changes instead of atomic changes.	$1 - \frac{\#top-level_{atomic}(\delta)}{population_{atomic}(\delta)}$

Table 2. Metrics

algorithms to detect many localized small changes instead of fewer big changes, or to aggregate changes, or to produce verbose output. In the future we plan to further investigate new metrics (together with their related qualities) and new applications to discover information about the editing process of documents.

References

1. Cóbena, G., Abiteboul, S., Marian, A.: Detecting changes in XML documents. In: Agrawal, R., Dittrich, K.R. (eds.) Proceedings of the 18th International Conference on Data Engineering, San Jose, CA, USA, February 26 - March 1, 2002. pp. 41–52. IEEE Computer Society (2002)
2. Di Iorio, A., Schirinzi, M., Vitali, F., Marchetti, C.: A natural and multi-layered approach to detect changes in tree-based textual documents. In: Filipe, J., Cordeiro, J. (eds.) Enterprise Information Systems, 11th International Conference, ICEIS 2009, Milan, Italy, May 6-10, 2009. Proceedings. Lecture Notes in Business Information Processing, vol. 24, pp. 90–101. Springer (2009)
3. Lindholm, T., Kangasharju, J., Tarkoma, S.: Fast and simple XML tree differencing by sequence alignment. In: Bulterman, D.C.A., Brailsford, D.F. (eds.) Proceedings of the 2006 ACM Symposium on Document Engineering, Amsterdam, The Netherlands, October 10-13, 2006. pp. 75–84. ACM (2006)

On Suggesting Entities as Web Search Queries

Extended Abstract

Diego Ceccarelli[1,2,3], Sergiu Gordea[4], Claudio Lucchese[1], Franco Maria Nardini[1], and Raffale Perego[1]

[1] ISTI-CNR, Pisa, Italy – {firstname.lastname}@isti.cnr.it
[2] IMT Institute for Advanced Studies Lucca, Lucca, Italy
[3] Dipartimento di Informatica, Università di Pisa, Pisa, Italy
[4] AIT GmbH, Wien, Austria – sergiu.gordea@ait.ac.at

Abstract. The Web of Data is growing in popularity and dimension, and named entity exploitation is gaining importance in many research fields. In this paper, we explore the use of entities that can be extracted from a query log to enhance query recommendation. In particular, we extend a state-of-the-art recommendation algorithm to take into account the semantic information associated with submitted queries. Our novel method generates highly related and diversified suggestions that we assess by means of a new evaluation technique. The manually annotated dataset used for performance comparisons has been made available to the research community to favor the repeatability of experiments.

1 Semantic Query Recommendation

Mining the past interactions of users with the search system recorded in query logs is an effective approach to produce relevant query suggestions. This is based on the assumption that *information* searched by past users can be of interest to others. The typical interaction of a user with a Web search engine consists in *translating her information need in a textual query made of few terms.* We believe that the "*Web of Data*" can be profitably exploited to make this process more user-friendly and alleviate possible vocabulary mismatch problems.
We adopt the *Search Shortcuts* (SS) model proposed in [1, 2]. The SS algorithm aims to generate suggestions containing only those queries appearing as final in successful sessions. The goal is to suggest queries having a high potentiality of being useful for people to reach their initial goal. The SS algorithm works by efficiently computing similarities between partial user sessions (the one currently performed) and historical successful sessions recorded in a query log. Final queries of most similar successful sessions are suggested to users as **search shortcuts**.

A virtual document is constructed by merging successful session, i.e., ending with a clicked query. We annotate virtual documents to extract relevant named entities. Common annotation approaches on query logs consider a single query and try to map it to an entity (if any). If a query is ambiguous, the risk is to always map it to the most popular entity. On the other hand, in case of

ambiguity, we can select the entity with the highest likelihood of representing the semantic context of a query.

We define *Semantic Search Shortcuts* (S^3) the query recommender system exploiting this additional knowledge. Please note that S^3 provides a list of related entities, differently from traditional query recommenders as SS that for a given query produce a flat list of recommendations. We assert that entities can potentially deliver to users much more information than raw queries.

In order to compute the entities to be suggested, given an input query q, we first retrieve the top-k most relevant virtual documents by processing the query over the SS inverted index built as described above. The result set R_q contains the top-k relevant virtual documents along with the entities associated with them. Given an entity e in the result set, we define two measures:

$$score(e, \mathcal{VD}) = \begin{cases} conf(e) \times score(\mathcal{VD}), & \text{if } e \in \mathcal{VD}.entities \\ 0 & \text{otherwise} \end{cases}$$

$$score(e, q) = \sum_{\mathcal{VD} \in R_q} score(e, \mathcal{VD})$$

where $conf(e)$ is the confidence of the annotator in mapping the entity e in the virtual document $\mathcal{VD}$, while $score(\mathcal{VD})$ represents the similarity score returned by the information retrieval system. We rank the entities appearing in R_q using their score *w.r.t.* the query.

2 Experimental Evaluation

We used a large query log coming from the Europeana portal[1], containing a sample of users' interactions covering two years (from August 27, 2010 to January, 17, 2012). We preprocessed the entire query log to remove noise (e.g., queries submitted by software robots, mispells, different encodings, etc). Finally, we obtained 139,562 successful sessions. An extensive characterization of the query log can be found in [3]. To assess our methodology we built a dataset consisting of 130 queries split in three disjoint sets: 50 short queries (1 term), 50 medium queries (on average, 4 terms), 30 long terms (on average, 9 terms). For each query in the three sets, we computed the top-10 recommendations produced by the SS query recommender system and we manually mapped them to entities by using a simple interface providing an user-friendly way to associate entities to queries[2].

We are interested in evaluating two aspects of the set of suggestions provided. These are our main research questions:

[1] We acknowledge the Europeana Foundation for providing us the query logs used in our experimentation. http://www.europeana.eu/portal/

[2] Interested readers can download the dataset from: http://hpc.isti.cnr.it/~ceccarelli/doku.php/sss.

Relatedness : How much information related to the original query a set of suggestions is able to provide?

Diversity : How many different aspects of the original query a set of suggestions is able to cover?

To evaluate these aspects, we borrow from the annotators the concept of *semantic relatedness* between two entities proposed by Milne and Witten [4]:

$$rel(e_1, e_2) = 1 - \frac{log(max(|I_L(e_1)|, |I_L(e_2)|)) - log(|I_L(e_1) \cup I_L(e_2)|)}{log(|KB|) - log(min(|I_L(e_1)|, |I_L(e_2)|))}$$

where e_1 and e_2 are the two entities of interest, the function $I_L(e)$ returns the set of all entities that link to the entity e in Wikipedia, and KB is the whole set of entities in the knowledge base. We extend this measure to compute the similarity between two set of entities (the function I_L gets a set of entities and returns all the entities that link *at least* on entity in the given set). At the same time, given two sets of entities E_1, E_2, we define the diversity as $div(E_1, E_2) = 1 - rel(E_1, E_2)$. Given a query q, let E_q be the set of entities that have been manually associated with the query. We define the relatedness and the diversity of a list of suggestions S_q as:

Definition 1 *The average relatedness of a list of suggestions is computed as:*

$$rel(S_q) = \frac{\sum_{s \in S_q} rel(E_s \setminus E_q, E_q)}{|S_q|}$$

where E_s represents the set of entities mapped to a suggestion s (could contain more than one entity in the manual annotated dataset). Please note that we remove the entities of the original query from each set of suggestions as we are not interested in suggesting something that do not add useful content *w.r.t.* the starting query ($E_s \setminus E_q$).

Definition 2 *The average diversity of a list of suggestions is defined as:*

$$div(S_q) = \frac{\sum_{s \in S_q} div(E_s, E_{S_q \setminus s})}{|S_q|}$$

For each suggestion, we intend to evaluate how much information it adds *w.r.t* the other suggestions. $E_{S_q \setminus s}$ denotes the union of the entities belonging to all the suggestions except the current suggestion s.

Experimental Results: For each set of queries in the dataset described above (*short*, *medium* and *long*), we compared the average relatedness and the average diversity of the recommendations generated by SS and by S^3.

Figure 1 shows the average relatedness computed for each query q belonging to a particular set of queries. Results confirm the validity of our intuition as, for all the three sets, the results obtained by S^3 are always better than the results obtained by considering the SS suggestions. It is worth to observe that the longer the queries the more difficult the suggestion of related queries. This

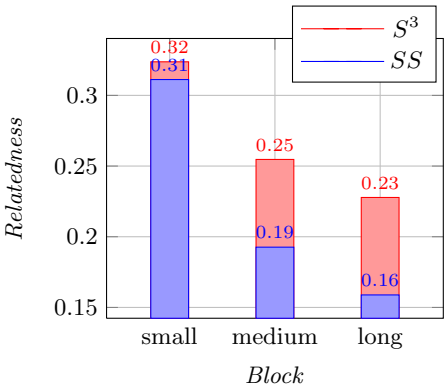


Fig. 1: Per-set average relatedness computed between the list of suggestions and the given query.

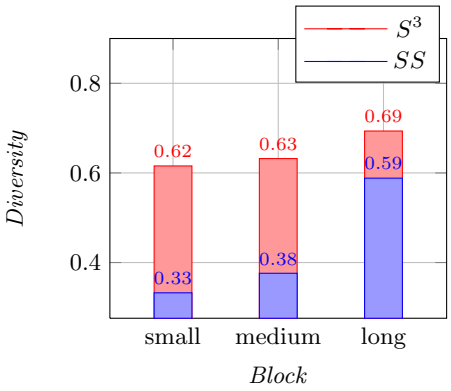


Fig. 2: Per-set average diversity computed between the list of suggestions and the given query.

happens because long queries occur less frequently in the log and then we have less information to generate the suggestions. If we consider single sets, the highest gain of S^3 in terms of average relatedness is obtained for medium and long queries: this means that relying on entities allows to mitigate the sparsity of user data.

Figure 2 reports the average diversity of the suggestions over the queries of each set. Here, we observe an opposite trend, due to the fact that the longer the queries, the more terms/entities they contain, and the more different the suggestions are. Furthermore, we observe that, for the most frequent queries, SS has a very low performance *w.r.t.* S^3. This happens because for frequent queries SS tends to retrieve popular reformulations of the original query, thus not diversifying the returned suggestions. S^3 does not suffer for this problem since it works with entities thus diversifying naturally the list of suggestions. We leave as future work the study of a strategy for suggesting entities aiming at maximizing the diversity on a list of suggestions.

References

1. Baraglia, R., Cacheda, F., Carneiro, V., Fernandez, D., Formoso, V., Perego, R., Silvestri, F.: Search shortcuts: a new approach to the recommendation of queries. In: Proc. RecSys'09. ACM, New York, NY, USA (2009)
2. Broccolo, D., Marcon, L., Nardini, F.M., Perego, R., Silvestri, F.: Generating suggestions for queries in the long tail with an inverted index. IP&M
3. Ceccarelli, D., Gordea, S., Lucchese, C., Nardini, F.M., Tolomei, G.: Improving europeana search experience using query logs. In: Proc. TPDL'11. pp. 384–395
4. Milne, D., Witten, I.: Learning to link with wikipedia. In: Proc. CIKM'08. pp. 509–518. ACM (2008)

Visual Features Selection

Giuseppe Amato, Fabrizio Falchi, and Cladio Gennaro

ISTI-CNR, via G. Moruzzi 1, 56124 Pisa, Italy
{giuseppe.amato,fabrizio.falchi,claudio.gennaro}@isti.cnr.it

Abstract. The state-of-the-art algorithms for large visual content recognition and content based similarity search today use the "Bag of Features" (BoF) or "Bag of Words" (BoW) approach. The idea, borrowed from text retrieval, enables the use of inverted files. A very well known issue with the BoF approach is that the query images, as well as the stored data, are described with thousands of words. This poses obvious efficiency problems when using inverted files to perform efficient image matching. In this paper, we propose and compare various techniques to reduce the number of words describing an image to improve efficiency.

Keywords: bag of features, bag of words, local features, content based image retrieval, landmark recognition

1 INTRODUCTION

During the last decade, the use of local features, as for instance SIFT [Lowe, 2004], has obtained an increasing appreciation for its good performance in tasks of image matching, object recognition, landmark recognition, and image classification. The total number of local features extracted from an image depends on its visual content and size. However, the average number of features extracted from an image is in the order of thousands. The BoF approach [Sivic and Zisserman, 2003] quantizes local features extracted from images representing them with the closest local feature chosen from a fixed visual vocabulary of local features (visual words). Matching of images represented with the BoF approach is performed with traditional text retrieval techniques.

However a query image is associated with thousands of visual words. Therefore, the search algorithm on inverted files has to access thousands of different posting lists. As mentioned in [Zhang et al., 2009], "a fundamental difference between an image query (e.g. 1500 visual terms) is largely ignored in existing index design. This difference makes the inverted list inappropriate to index images." From the very beginning [Sivic and Zisserman, 2003] some words reduction techniques were used (e.g. removing 10% of the more frequent images).

To improve efficiency, many different approaches have been considered including GIST descriptos [Douze et al., 2009], Fisher Kernel [Zhang et al., 2009] and Vector of Locally Aggregated Descriptors (VLAD) [Jégou et al., 2010]. However, their usage does not allow the use of traditional text search engine which has actually been another benefit of the BoF approach.

In order to mitigate the above problems, this paper proposes, discusses, and evaluates some methods to reduce the number of visual words assigned to images. This paper is a summary of a longer paper that will presented at VISAPP 2013 [Amato et al., 2013].

2 PROPOSED APPROACH

The goal of the BoF approach is to substitute each description of the region around an interest points (i.e., each local features) of the images with visual words obtained from a predefined vocabulary in order to apply traditional text retrieval techniques to content-based image retrieval. At the end of the process, each image is described as a set of visual words. The retrieval phase is then performed using text retrieval techniques considering a query image as disjunctive text-query. Typically, the *cosine* similarity measure in conjunction with a term weighting scheme is adopted for evaluating the similarity between any two images.

The proposed words reduction criteria are: *random*, *scale*, *tf*, *idf*, *tf*id*. Each proposed criterion is based on the definition of a score that allows us to assign each local feature or word, describing an image, an estimate of its importance. Thus, local features or words can be ordered and only the most important ones can be retained. The percentage of information to discard is configurable through a score threshold, allowing trade-off between efficiency and effectiveness. The *random* criterion was used as a baseline. It assigns random score to features. The *scale* criterion is based on the information about the size of the region from which the local features was extracted: the larger the region, the higher the score.

The retrieval engine used in the experiments is built as following:

1. For each image in the dataset the SIFT local features are extracted for the identified regions around interest points.
2. A vocabulary of words is selected among all the local features using the *k-means* algorithm.
3. The *Random* or *Scale* reduction techniques are performed (if requested).
4. Each image is described following the BoF approach, i.e., with the ID of the nearest word in the vocabulary to each local feature.
5. The *tf*, *idf*, or *tf*idf* reduction technique are performed (if requested).
6. Each image of the test set is used as a query for searching in the training set. The similarity measure adopted for comparing two images is the Cosine between the query vector and the image vectors corresponding to the set of words assigned to the images. The weight assigned to each word of the vectors are calculated using *tf*idf* measure.
7. In case the system is requested to identify the content of the image, the landmark of the most similar image in the dataset (which is labeled) is assigned to the query image.

3 Experimental results

The quality of the retrieved images is typically evaluated by means of precision and recall measures. As in many other papers, we combined these information by means of the mean Average Precision (mAP), which represents the area below the precision and recall curve.

For evaluating the performance of the various reduction techniques approaches, we use the Oxford Building datasets that was presented in [Philbin et al., 2007] and has been used in many other papers. The dataset consists of 5,062 images of 55 buildings in Oxford. The ground truth consists of 55 queries and related sets of results divided in best, correct, ambiguous and not relevant. The vocabulary used has one million words.

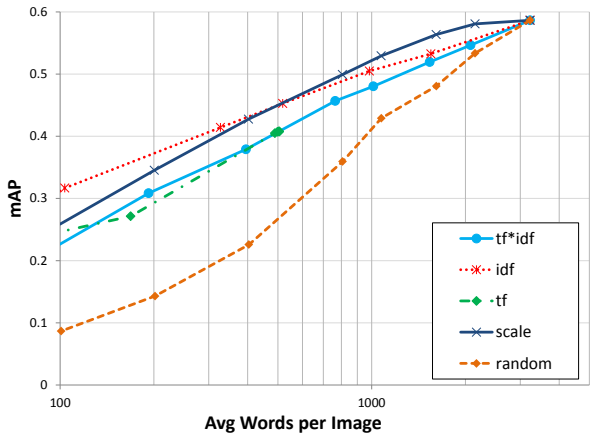


Fig. 1. Mean average precision of the various selection criteria obtained on the Oxford Buildings 5k dataset.

We first report the results obtained in a content based image retrieval scenario using the Oxford Building dataset using the ground truth given by the authors [Philbin et al., 2007]. In Figure 1 we report the mAP obtained On the x-axis we reported the average words per image obtained after the reduction. Note that the x-axis is logarithmic. We first note that all the reduction techniques significantly outperform naive *random* approach and that both the *idf* and *scale* approaches are able to achieve very good mAP results (about 0.5) while reducing the average number of words per image from 3,200 to 800. Thus, just taking the 25% of the most relevant words, we achieve the 80% of the effectiveness. The comparison between the *idf* and *scale* approaches reveals that *scale* is preferable for reduction up to 500 words per image. Thus, it seems very important to discard small regions of interest up to 500 words.

While the average number of words is useful to describe the length of the image description, it is actually the number of distinct words per image that have

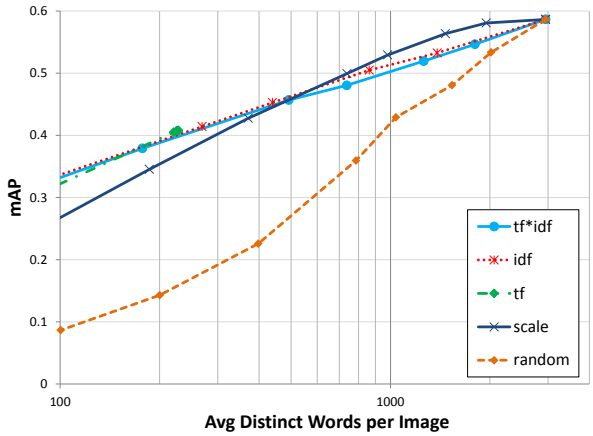


Fig. 2. Mean average precision of the various selection criteria obtained on the Oxford Buildings 5k dataset.

more impact on the efficiency of searching using inverted index. Thus, in Figure 2 we report mAP with respect to the average number of distinct words. In this case the results obtained by *tf*idf* and *tf* are very similar to the ones obtained by *idf*. In fact, considering *tf* in the reduction results in a smaller number of average distinct words per image for the same vales of average number of words.

References

[Amato et al., 2013] Amato, G., Falchi, F., and Gennaro, C. (2013). On reducing the number of visualwords in the bag-of-features representation. In *VISAPP 2013 - Proceedings of the International Conference on Computer Vision Theory and Applications*.

[Douze et al., 2009] Douze, M., Jégou, H., Sandhawalia, H., Amsaleg, L., and Schmid, C. (2009). Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '09, pages 19:1–19:8, New York, NY, USA. ACM.

[Jégou et al., 2010] Jégou, H., Douze, M., and Schmid, C. (2010). Improving bag-of-features for large scale image search. *Int. J. Comput. Vision*, 87:316–336.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

[Philbin et al., 2007] Philbin, J., Chum, O., Isard, M., Sivic, J., and Zisserman, A. (2007). Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[Sivic and Zisserman, 2003] Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 1470–, Washington, DC, USA. IEEE Computer Society.

[Zhang et al., 2009] Zhang, X., Li, Z., Zhang, L., Ma, W.-Y., and Shum, H.-Y. (2009). Efficient indexing for large scale visual search. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1103 –1110.

Experimenting a Visual Attention Model in the Context of CBIR systems

Franco Alberto Cardillo, Giuseppe Amato, and Fabrizio Falchi

Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Richerche, Pisa, Italy
franco.alberto.cardillo@isti.cnr.it, giuseppe.amato@isti.cnr.it,
fabrizio.falchi@isti.cnr.it

Abstract. Many novel applications in the field of object recognition and pose estimation have been built relying on local invariant features extracted from selected key points of the images. Such keypoints usually lie on high-contrast regions of the image, such as object edges. However, the visual saliency of the those regions is not considered by state-of-the art detection algorithms that assume the user is interested in the whole image. Moreover, the most common approaches discard all the color information by limiting their analysis to monochromatic versions of the input images. In this paper we present the experimental results of the application of a biologically-inspired visual attention model to the problem of local feature selection in landmark and object recognition tasks. The model uses color-information and restricts the matching between the images to the areas showing a strong saliency. The results show that the approach improves the accuracy of the classifier in the object recognition task and preserves a good accuracy in the landmark recognition task when a high percentage of visual features is filtered out. In both cases the reduction of the average numbers of local features result in high efficiency gains during the search phase that typically requires costly searches of candidate images for matches and geometric consistency checks.

1 Introduction

Given an image as query, a Content-Based Image Retrieval (CBIR) system returns a list of images ranked according to their visual similarity with the query image. When queried, it extracts the same features from the query image and compare their values with those stored in the index, choosing the most similar images according to a specified similarity measure. Many CBIR systems support general visual similarity searches using global features such as color and edge histograms. The adoption of descriptions based on local features based (e.g., SIFT and SURF), from the computer vision field, provided multimedia information systems with the possibility to build applications for different tasks, like, e.g., object recognition and pose estimation.

However, the number of local visual features extracted from cluttered, real-world images is usually in the order of thousands. When the number is 'too' large,

the overall performance of a CBIR system may decline. If too many features are extracted from 'noise', i.e., regions that are not relevant, not only the CBIR becomes slower in its computations, but also its matching accuracy declines due to many false matches among the features. The reduction of the number of visual features used in the image descriptions can thus be considered a central point in reaching a good overall performance in a CBIR system. If only the keypoints extracted from relevant regions are kept a great improvement might be reached both in the timings and the accuracy of the system.

In this work we present an approach concerning the application of a biologically-inspired visual attention model for filtering out part of the features in the images. The human visual system is endowed with attentional mechanisms able to select only those areas in the field of view that are likely to contain relevant information. The basic assumption of our experimental work is that the user chooses the query image according to its most salient areas and expects the CBIR system to return images with a similar appearance in their salient areas. The model we implemented has a strong biological inspiration: it uses an image encoding that respects what is known about the early visual system by mimicking the biological processes producing the neural representation of the image formed by our brain. Since the biological inspiration does not bias the system towards specific features, the approach can be used in generic image recognition tasks.

In order to assess quantitatively the performance of the visual attention model, we tested it on two tasks: a landmark recognition task and an object recognition task using two publicly available datasets. The results show that the filtering of the features based on the image saliency is able to drastically reduce the number of keypoints used by the system with an improvement or just a slightly decrease in the accuracy of the classifier in, respectively, the object recognition task and the landmark recognition task.

The rest of this paper is organized as follows. The next section briefly discusses the biological inspiration of our model. Section 4 describes the model of visual attention and its relationships with the biological facts introduced in section 3. Section 5 presents the datasets we used in the current experimentation and the results we obtained. The last section discusses the pros and the cons of our approach and briefly delineates some research lines we will follow in the future.

2 Previous Works

Visual attention has been used to accomplish different tasks in the context of Content Based Image Retrieval. For example, some works used attention as a mean to re-rank the images returned after a query. However, since our focus is on image filtering, we will restrict our analysis to two recent approaches that introduce an attentional mechanism for reducing the number of features used by a CBIR system with the goal of improving both its speed and its accuracy.

[Marques et al., 2007] proposes a segmentation method that exploits visual attention in order to select regions of interest in a CBIR dataset with the idea of

using only those regions in the image similarity function. They use the saliency map produced by the Itti-Koch model [Itti et al., 1998] for the selection of the most salient points of an image. The selected points are then used for segmenting the image using a region growing approach. The segmentation algorithm is guided by the saliency computed by the Stentiford model of visual attention, whose output allows an easier and more precise segmentation than Itti-Koch's model. They experimented their methods on a dataset containing 110 images of road signs, red soda cans, and emergency triangles. Since that dataset is well known and used in other published experimentations, we used it in order to test our filtering approach.

[Gao and Yang, 2011] propose a method for filtering SIFT keypoints using saliency maps. The authors use two different algorithms for computing the image saliency, the Itti-Koch model (for local-contrast analysis) and a frequency-based method (for global-contrast analysis) that analyzes the Fourier spectrum of the image [Hou and Zhang, 2007]. The final saliency, corresponding to the simple sum of the saliency maps computed by the two methods, is used to start a segmentation algorithm based on fuzzy growing. They experimented their method on a dataset composed by 10 classes with more than 10 images per class, extracted from the ALOI image dataset and the Caltech 256 photo gallery. The original images were modified using various transformations. The authors show that their method has a precision that is lower than standard SIFT and comparable to PCA-SIFT (a filtering approach based on Principal Component Analysis). Even if the accuracy is not improved by the filtering, their approach is much faster than the other two and is thus suitable for use in CBIR systems.

In this work we experiment and evaluate a model of visual attention both on the dataset described above and on a more complex dataset. The harder dataset contains a large number of real, cluttered photographies of monuments located in Pisa. The dataset contains pictures downloaded from Internet (e.g., flickr images) that have not undergone any modification.

3 Biological Inspiration

When we open our eyes we see a colorful and meaningful three-dimensional world surrounding us. Such visual experience results from a sequence of transformations performed on the light stimuli that starts in our eyes. The light is focused on the retinal surface, then processed and transferred to our thalamus, and finally routed to the cerebral cortex. The initial transformation is accomplished by the retina, starting from the photoreceptors. Photoreceptors are connected to bipolar cells in the middle retinal layer, which are then connected to the third and final layer (the retinal output), populated by ganglion cells. Ganglion cells have a structured receptive field, i.e., they are connected to well-defined areas in the retina and do not react to the simple presence of a light stimulus. In particular, ganglion cells have a receptive field with a *center–surround* organization [Kuffler, 1953]. For example, an *on–center, off–surround* ganglion cell reaches its maximum activity level when the light hits and fills the central part

of the receptive-field and no light stimuli are present in the surround area. The output of the ganglion cells reaches the striate cortex or *area V1*. In this layer, the cells start computing complex features; for example, V1 cells show preference for specific orientations. Colors are the results of a complex processing that takes place at various stages in the processing pipeline described above. Cells described by the opponent process theory can be found as early as in the last retinal layers: it is possible to find bipolar, ganglion (and later LGN cells) that have a preferred wavelength with a center-surround organization. In particular, there are (R+, G-) cells, excited by a red centre and inhibited by a green surround, and (G+, R-), (B+, Y-), (Y+, B-), where 'Y' stands for yellow and 'B' for blue. These cells, together with the achromatic channel composed by (Wh+, Bl-) and (Wh-, Bl+) cells (where 'Wh' stands for White and 'Bl' stands for Black), allow our visual system to represent million of colors by combining the activation patterns of the photoreceptors. Furthermore, this antagonism in color processing makes the visual system responsive to discontinuities, as edges, that are what best describe the shape of an object.

3.1 Visual Attention

The visual stimuli we receive contain a overwhelming amount of visual information, that is simply too 'large' for our brain to process. Evolution has endowed humans with a series of filters able to reduce the large amount of the incoming information. A recent definition of visual attention can be found in [Palmer, 1999]. Visual attention is defined as those processes that enable an observer to recruit resources for processing selected aspects of the retinal image more fully than non-selected aspects. Evidence gathered in several psychological experiments shows that our attentional system can be roughly subdivided into two main components that operate very differently and at different stages. The first system, called preattentive, starts operating as soon as the light strikes the retinal photoreceptors. It processes basic visual features, like color, orientation, size or movements, in parallel and over the entire field of view. This system is responsible of the visual pop-out effect, i.e., the situations where an image area attracts our attention due to its differences with the rest of the other image parts. The second system, called attentive, correspond to focused attention. When the target is not recognized by the preattentive system, the attentive processing starts and uses information computed by the preattentive system in order to select spatial regions that might contain the target object. It necessarily operates sequentially since it needs to focus several spatial regions looking for specific object features.

According to the "Feature Integration Theory" [Treisman and Gelade, 1980] (FIT), the parallel, preattentive processes build an image representation with respect to a single feature and encode the information in feature maps (color, orientation, spatial frequency, ...). The maps are combined and the their peaks of activity are inspected guided by a global map that summarize the information computed in the various dimensions. One of the most influential detailed models was proposed in [Koch and Ullman, 1985]. Such model is similar to FIT in the description of the preattentive and attentive stages, but proposes some

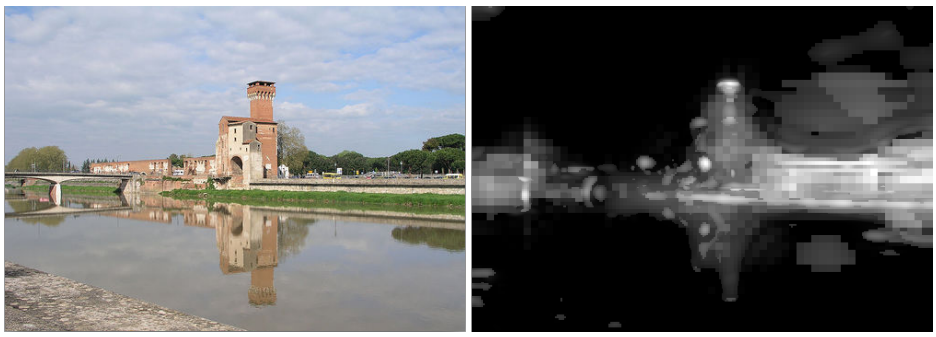


Fig. 1. Example of the application of the visual attention model. Left: original image; Right: saliency map computed by the model: the brighter the pixel the more salient the area surrounding it is.

intermediate structures able to give a plausible answer to the attentional shifts, both in visual pop-out and in conjunctive search.

4 The Computational Model

In this experimentation we implemented a bottom-up model of Visual Attention that extends [Itti et al., 1998]. It is part of larger model that includes top-down attentional mechanisms for object learning and mechanisms. The model performs a multiresolution analysis of an input image and produces a saliency map assigning a weight to each image pixel (area) according to the computed saliency. The model is biologically-inspired: it encodes the image according to what is known about the retinal and early cortical processing and elaborates the channels with algorithms that resemble the biological processes, even if only at a functional level. Biologically-inspired models use a less sophisticated image encoding and processing than other approaches, but are not biased towards any specific visual feature. Less general approaches, that focus on specific features or measures for computing the saliency, are well suited for application domains characterized by a low variability in object appearance, but may fail when the content of the images is not restricted to any specific category. The bottom-up model performs a multiresolution image analysis by using in each processing step a pyramidal representation of the input image. After encoding the input values using five different channels for intensity and colors and four channels for the oriented features, it builds feature maps using a center-surround organization and computes the visual conspicuity of each level in every pyramid. For each level of the conspicuity pyramids, the model builds a local saliency map that shows the saliency of the image areas at a given scale. The level saliency maps are then merged into a unique, low-resolution global saliency map encoding the overall saliency of image areas.

The input images are encoded using the Lab color space, where for each pixel the channels L, a, and b corresponds, respectively, to the dimensions intensity

(luminance), red-green, and blue-yellow. The Lab values are then split into five different channels: intensity, red, green, blue, and yellow. Each channel extracted from the image is then encoded in an image pyramid following to the algorithm described in [Adelson et al., 1984,Greenspan et al., 1994].

4.1 Visual Features

The set of feature used by the model includes intensity and color, computed according to the center-surround receptive-field organization characterizing ganglion and LGN cells, and oriented lines, computed in area V1. The raw l, a, b values are used to extract the color channels $\mathcal{I}_{\mathtt{I}}$, $\mathcal{I}_{\mathtt{R}}$, $\mathcal{I}_{\mathtt{G}}$, $\mathcal{I}_{\mathtt{B}}$, and $\mathcal{I}_{\mathtt{Y}}$ that correspond, respectively, to intensity, red, green, blue, and yellow. Local orientation maps are computed on the intensity pyramid by convolving the intensity image in each layer with a set of oriented Gabor filters at four different orientations $\theta \in \left\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3}{2}\pi\right\}$. Such filters provide a good model of the receptive fields characterizing cortical simple cells [Jones and Palmer, 1987], as discussed in the previous section. The filters used in the model implementation are expressed as follows [Daugman, 1985]:$F(x, y, , \theta, \psi) = \exp(-\frac{x_o^2+\gamma^2 y_o^2}{2\sigma^2}) \cos(\frac{2\pi}{\lambda} x_o + \psi)$ where $x_o = x\cos\theta + y\sin\theta$ $y_o = -x\sin\theta + y\cos\theta$. Each image in the intensity image is convolved with Gabor filters of fixed size, in the current implementation they are 15×15 pixels wide. The rest of the parameters is set as follows: $\gamma = 0.3,\ \sigma = 3.6,\ \lambda = 4.6$, since those values are compatible with actual measurements taken from real cells [Serre et al., 2007].

The model uses the center-surround organization as found in the ganglion cells for color and intensity information. The channel for intensity, for example, is encoded in two different contrast maps, the first one for the on-center/off-surround receptive fields, the second one for the off-centre/on-surround opponency. Both types of cells present a null response on homogeneous areas, where the stimuli coming from the centre and the surround of the receptive field compensate each other.

The original model [Itti et al., 1998] uses double-opponent channels, meaning that the red-green and green-red image encoding are represented by a same map. We used single-opponent channels since such choice allows us to distinguish, for example, strong dark stimuli from strong light ones. In order to respect the biological inspiration we use radial symmetric masks and we do not perform across-scale subtraction as in the original model. Basically, given two pyramids of two different features f and $f^\star$, corresponding to the excitatory and the inhibitory features of the contrast map, the feature corresponding to the center of the receptive field is convolved with a Gaussian kernel G_0 that provides the excitatory response. The feature corresponding to the surround of the receptive field is convolved with two different Gaussians G_1, G_2 with different sizes, that virtually provide the response of ganglion cells with different sizes of their receptive fields. The results of the convolutions correspond to the inhibitory part of the receptive field.

The feature maps are computed for the following couples of ordered opponent features: (R, G) and (G, R), encoding, respectively, red-on/green-off cells

and green-on/red-off opponencies; (B, Y) and (Y, B), encoding, respectively, blue-on/yellow-off and yellow-on/blue-off opponencies. Furthermore, we encode center-surround differences for intensity in separate feature maps: $I_{on,off}$, $I_{off,on}$. The two maps encode, respectively, on-centre/off-surround and off-centre/on-surround cells for intensity. The feature maps are hereafter denoted with RG, GR, BY, YB, $I_{on,off}$, and $I_{off,on}$. Since the oriented features are extracted using differential operators, they do not need to be processed as the other maps.

Before building the saliency maps for each level of the image pyramid, we need to merge the feature contrast maps in the same dimension: color, intensity, and orientation. This step is inspired by the FIT model, where parallel separable features are computed in parallel, each one competing with features in the same dimension. For example, in order to build the feature conspicuity map for color, we need to merge in a single map the two contrast maps RG (obtained by merging the R-G and G-R opponent channels) and BY. Simple summation or the creation of a map with the average values among the various contrast maps are not suited for the goal of creating a saliency map. For example, a red spot among many green spot should be given a higher saliency value than the green ones: with a merging algorithm based on simple summation or on the average red and green spot would receive the same weight. There are several strategies that could be used for modifying a map according to its relevance. Each strategy tries to decrease the values in maps that contain many peaks of activation and to enhance the values in maps that have few regions of activity. We implemented a merging step based on Summed Area Tables (SATs). Each pixel (r, c) in a SAT contains the sum of the pixel values in the subimage with corners located at image coordinates $(0, 0)$ and (r, c), where the origin is the upper left corner.

In order to enhance maps with small spots of activity, for each pixel (r, c), we read the SAT value for a squared box centered at (r, c) with size equal to 1% the minimum dimension of the feature map and the SAT value for the entire image. Then we set the value for the feature conspicuity map using the following formula: $FCM(r, c) = c_{SAT} + 2 \cdot c_{SAT} \cdot \tanh(c_{SAT} - s_{SAT})$, where r and c are the coordinates in the feature contrast map FCM, c_{SAT} and s_{SAT} are, respectively, the sum of the values in the box representing the center and the surround values read from the SAT. This normalization procedure is repeated several times in order to inhibit weak regions while enhancing peaks of activity.

4.2 Saliency map

The final saliency map is created at the lowest resolution of the pyramid. Several options are available and we chose to set the value of each pixel p with the maximum value of the areas in the image pyramid that are mapped onto p by the subsampling procedure. With respect to other solutions (average over the maps, summation) the max pooling operation allows us to keep and highlight in the global saliency map also areas that are very salient at only a single scale. By looking at pixels in the saliency map with high values, we can navigate through the pyramidal hierarchy to access the level where the maximum activation is present and analyze the salient region. However, in this paper we limit

our experimentation to the bottom-up part that limits its computations to the bottom-up part.

5 Experimentations

We tested the proposed VA-based filtering approach on one landmark recognition and one objection recognition tasks using two different datasets:

- the publicly available dataset containing 1227 photos of 12 landmarks (object classes) located in Pisa (also used in the works [Amato et al., 2011], and [Amato and Falchi, 2011], [Amato and Falchi, 2010]), hereafter named PISA-DATASET. The dataset is divided in a *training set* (Tr) consisting of 226 photos (20% of the dataset) and a *test set* (Te) consisting of 921 photos (80% of the dataset).
- The publicly available dataset containing 258 photos belonging to three classes (cans, road signs, and emergency triangles), hereafter named STIM-DATASET. The dataset is similarly split into a training and a test set containing, respectively, 206 and 52 photos.

The experiments were conducted using the Scale Invariant Feature Transformation (SIFT) [Lowe, 2004] algorithm that represents the visual content of an image using scale-invariant local features extracted from regions around selected keypoints. Such keypoints usually lie on high-contrast regions of the image, such as object edges. Image matching is performed by comparing the description of the keypoints in two images searching for matching pairs. The candidate pairs for matches are verified to be consistent with a geometric transformation (e.g., affine or homography) using the RANSAC algorithm [Fischler and Bolles, 1981]. The percentage of verified matches is used to argue whether or not the two images contain the very same rigid object.

The number of local features in the description of the images is typically in the order of thousands. This results in efficiency issues on comparing the content of two images described with the SIFT descriptors. For this reason we applied a filtering strategy selecting only the SIFT keypoints extracted from regions with a high saliency. Each image in the dataset was processed by the VA model producing a saliency map. Since the resolution of the saliency map is very low, each saliency map has been resized to the dimension of the input image.

5.1 PISA-DATASET

In order to study how many SIFT keypoints could be filtered out by the index, we applied several thresholds on the saliency levels stored in the saliency map. The thresholds range from 0.3 to 0.7 the maximum saliency value (normalized to 1). The 0.3 threshold did not modify at all any of the saliency maps, meaning that all of the saliency maps had values larger than 0.3. SIFT keypoints were filtered out only when they corresponded to points in the saliency map with a value below the given threshold. In order to see how effective the filtering by the VA

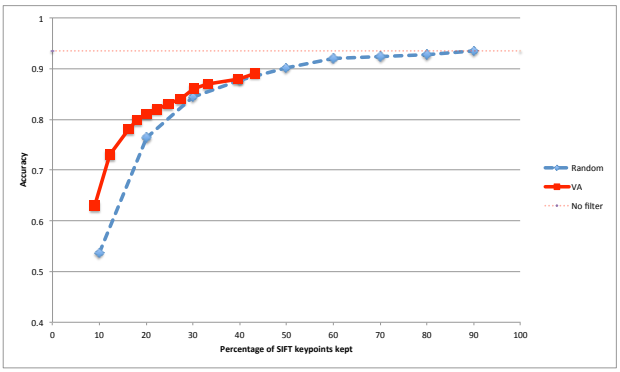


Fig. 2. Accuracy obtained after the application of the VA and random filtering on the PISA-DATASET. Solid line: accuracy after filtering features using the saliency map; dashed line: accuracy obtained after random filtering. The maximum accuracy obtained by not applying any filter is shown by the horizontal dotted line.

model was, we compared it against random filtering, in this second case, we kept from 10% to 90% of the original SIFT keypoints by incrementally removing some chosen randomly ones. Notice that standard feature selection algorithms cannot be directly applied since in our approach the keypoints cannot be considered object features.

We used *accuracy* in assigning the correct landmark to the test images (in the previously mentioned dataset) as the measure of performance. For each test image, the best candidate match between the training images is selected using the SIFT description and verifying the matches searching for an affine transformation using the RANSAC algorithm.

The results of the experimentation are shown in figure 2. The x-axis shows the percentage of SIFT keypoints kept after filtering. The y-axis corresponds to the accuracy reached by the classifier after the filtering. The maximum accuracy is reached by not removing any keypoint and is equal to 0.935. The accuracy does not vary much till a 40% filtering, when it starts decreasing.

When all the saliency values are used, the filtering performed using the visual saliency maps reaches a 0.89 accuracy when it removes almost 57% of the original keypoints. The performance of the VA-based filter is very similar to the random-based one when 30% keypoints are kept. However, when the percentages of removed keypoints inreases, the VA-based filtering algorithm outperforms the random filtering.

The results of the model when on aggressive filtering levels are quite encouraging. The model is in fact able to preserve regions that are significant for the recognition of the specific object. There is a decrease in the overall accuracy with respect to the SIFT classifiers, but the time needed to perform the classification is significantly lower. In fact, when the classification uses 100% of the SIFT keypoints (no filtering), the average time for classifying a single test

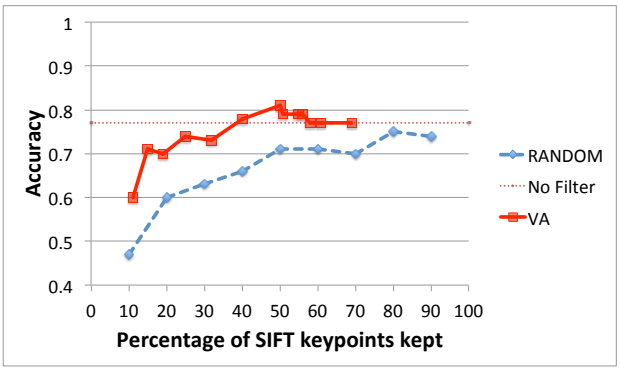


Fig. 3. Accuracy obtained after the application of the VA and random filtering on the STIM-DATASET. Solid line: accuracy after filtering features using the saliency map; dashed line: accuracy obtained after random filtering. The maximum accuracy obtained by not applying any filter is shown by the horizontal dotted line.

images is 7.2 seconds. When we use only 30% or 20% of the original SIFT keypoints (VA-based filtering) the time needed for the classification of an image is, respectively, 0.78 and 0.6 seconds per image on average. Even when the random filter and the VA-based filter have the same accuracy, the use of saliency provides 'better keypoints. When only a 40% percentage of the original keypoints is kept, the average time needed to classify a single image is 1.07 and 0.97 seconds for, respectively, images preprocessed using the random filter and the VA-based filter.

However, this experimentation has also shown a relevant limitation of filtering approaches based on bottom-up visual attention. In fact, many test images misclassified by the classifier contain salient regions that are radically different from the other images in the same category. For example, since many pictures contain people in front of monuments, the visual attention filter is prone to remove (i.e., assign a low saliency to) the monument in the background and preserve the people as the most salient areas. This behaviour is particularly evident on very aggressive filtering levels, where only the most salient regions are kept. In many cases the monument simply disappears in the saliency map.

5.2 STIM-DATASET

In the case of the STIM-DATASET the saliency maps were thresholded using values ranging from 0.1 to 0.9 the maximum value in the map. By applying that set of thresholds, the percentage of SIFT keypoints kept and used by the classifier ranges from 11% to 77% (on average) the number of keypoints originally extracted from images. In this dataset the relevant objects are well-separated by the background in almost every image. Furthermore, since they never fill the entire frame, their features are not considered too 'common' to be salient and

are not suppressed by the attentional mechanism. From the graph shown in Fig. 3 it is clear that the VA-based filtering is able both to improve the accuracy and to decrease the time needed for the classification. By using only half the keypoints selected by the VA model, the classifier reaches 81% accuracy much greater than that obtained using 100% of the original keypoints or 90% randomly selected, that are equal to, respectively, 0.77 and 0.74.

6 Conclusions

In this paper we have presented a filtering approach based on a visual attention model that can be used to improve the performance of large-scale CBIR systems and object recognition algorithms. The model uses a richer image representation than other common and well-known models and is able to process a single image in a short time thanks to many approximations used in various processing steps.

The results show that a VA-based filtering approach allows to reach a better accuracy on object recognition tasks where the objects stand out clearly from the background, like in the STIM-DATASET. In these cases a VA-based filtering approach reduces significantly the number of keypoints to be considered in the matching process and allows to reach a greater number of correct classifications. The results on the PISA-DATASET are encouraging: a faster response in the classification step is obtained with only a minor decrease in accuracy. However, the results need a deeper inspection in order to gain a better understanding of the model on cluttered scene where the object (or landmark) to be detected does not correspond to the most salient image areas.

After this experimentation, we still think that bottom-up attention might be useful in the context of image similarity computations. In the context of landmark recognition, Better results could be obtained if the bottom-up processes receive a kind of top-down modulation signal able to modify the computation of the image saliency according the searched object. In fact, without such kind of modulation, if a query image contains only a single object, that same object might not be salient in any other image in the dataset.

The experimentation suggests at least two research lines. The short term goal is to evaluate the model for searching and retrieving images visually similar to a given query image. However, such goal requires the construction of a good dataset enabling a quantitative evaluation of the results. Except in very simple cases, it is not very clear when and how to consider two images visually similar. The long term goal is to introduce a form of top-down attentional modulation that enables object searches in very large datasets. Since CBIR systems usually relies upon an image index, it is far from clear how the most common index structures might be modified for allowing the introduction of that modulation.

References

[Adelson et al., 1984] Adelson, E., Anderson, C., Bergen, J., Burt, P., and Ogden, J. (1984). Pyramid methods in image processing. *RCA Engineer*, 29(6):33–41.

[Amato and Falchi, 2010] Amato, G. and Falchi, F. (2010). kNN based image classification relying on local feature similarity. In *SISAP '10: Proceedings of the Third International Conference on SImilarity Search and APplications*, pages 101–108, New York, NY, USA. ACM.

[Amato and Falchi, 2011] Amato, G. and Falchi, F. (2011). Local feature based image similarity functions for kNN classfication. In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence (ICAART 2011)*, pages 157–166. SciTePress. Vol. 1.

[Amato et al., 2011] Amato, G., Falchi, F., and Gennaro, C. (2011). Geometric consistency checks for knn based image classification relying on local features. In *SISAP '11: Fourth International Conference on Similarity Search and Applications, SISAP 2011, Lipari Island, Italy, June 30 - July 01, 2011*, pages 81–88. ACM.

[Daugman, 1985] Daugman, J. (1985). Uncertainty relations for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A*, 2:1160–1169.

[Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

[Gao and Yang, 2011] Gao, H.-p. and Yang, Z.-q. (2011). Integrated visual saliency based local feature selection for image retrieval. In *Intelligence Information Processing and Trusted Computing (IPTC), 2011 2nd International Symposium on*, pages 47 –50.

[Greenspan et al., 1994] Greenspan, H., Belongie, S., Perona, P., Goodman, R., Rakshit, S., and Anderson, C. (1994). Overcomplete steerable pyramid filters and rotation invariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR94)*, pages 222–228.

[Hou and Zhang, 2007] Hou, X. and Zhang, L. (2007). Saliency detection: A spectral residual approach. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –8.

[Itti et al., 1998] Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259.

[Jones and Palmer, 1987] Jones, J. and Palmer, L. (1987). An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58:1233–1258.

[Koch and Ullman, 1985] Koch, C. and Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4:219–227.

[Kuffler, 1953] Kuffler, W. (1953). Discharge patterns and functional organization of mammalian retina. *Journal of Neurophysiology*, 16:37–68.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

[Marques et al., 2007] Marques, O., Mayron, L. M., Borba, G. B., and Gamba, H. R. (2007). An attention-driven model for grouping similar images with image retrieval applications. *EURASIP J. Appl. Signal Process.*, 2007(1):116–116.

[Palmer, 1999] Palmer, S. (1999). *Vision Science, Photons to phenomenology.* The MIT Press.

[Serre et al., 2007] Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., and Poggio, T. (2007). Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):411–426.

[Treisman and Gelade, 1980] Treisman, A. and Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136.

Cumulated Relative Position: A Metric for Ranking Evaluation (Extended Abstract)*

Marco Angelini[3], Nicola Ferro[1], Kalervo Järvelin[2], Heikki Keskustalo[2], Ari Pirkola[2], Giuseppe Santucci[3], and Gianmaria Silvello[1]

[1] University of Padua, Italy
{ferro,silvello}@dei.unipd.it
[2] University of Tampere, Finland
{kalervo.jarvelin,heikki.keskustalo,ari.pirkola}@uta.fi
[3] "La Sapienza" University of Rome, Italy
{angelini,santucci}@dis.uniroma1.it

Abstract. The development of multilingual and multimedia information access systems calls for proper evaluation methodologies to ensure that they meet the expected user requirements and provide the desired effectiveness. In this paper, we propose a new metric for ranking evaluation, the CRP.

1 Introduction and Motivations

The development of information access systems calls for proper evaluation methodologies in particular for what is concerned with the evaluation of rankings. A range of evaluation metrics, such as MAP and nDCG, are widely used and they are particularly suitable to the evaluation of *Information Retrieval (IR)* techniques in terms of the quality of the output ranked lists, and often to some degree suitable to the evaluation of user experience regarding retrieval. Unfortunately, the traditional metrics do not take deviations from optimal document ranking sufficiently into account. We think that a proper evaluation metric for ranked result lists in IR should: (a) explicitly handle graded relevance including negative gains for unhelpful documents, and (b) explicitly take into account document misplacements in ranking either too early or too late given their degree of relevance and the optimal ranking. In the present paper, we propose such a new evaluation metric, the *Cumulated Relative Position (CRP)*.

We start with the observation that a document of a given degree of relevance may be ranked too early or too late regarding the ideal ranking of documents for a query. Its relative position may be negative, indicating too early ranking, zero indicating correct ranking, or positive, indicating too late ranking. By cumulating these relative rankings we indicate, at each ranked position, the net effect of document displacements, the CRP. CRP explicitly handles: (a) graded

* The extended version of this abstract has been published in [1].

relevance, and (b) document misplacements either too early or too late given their degree of relevance and the ideal ranking. Thereby, CRP offers several advantages in IR evaluation: (i) at any number of retrieved documents examined (rank) for a given query, it is obvious to interpret and it gives an estimate of ranking performance; (ii) it is not dependent on outliers since it focuses on the ranking of the result list; (iii) it is directly user-oriented in reporting the deviation from ideal ranking when examining a given number of documents; the effort wasted in examining a suboptimal ranking is made explicit.

2 Definition of Cumulated Relative Position

We define the set of *relevance degrees* as $(REL, \leq)$ such that there is an order between the elements of REL. For example, for the set $REL = \{\texttt{nr}, \texttt{pr}, \texttt{fr}, \texttt{hr}\}$, `nr` stands for "non relevant", `pr` for "partially relevant", `fr` for "fairly relevant", `hr` stands for "highly relevant", and it holds $\texttt{nr} \leq \texttt{pr} \leq \texttt{fr} \leq \texttt{hr}$.

We define a function $\mathrm{RW} : REL \to \mathbb{Z}$ as a monotonic function which maps each relevance degree ($rel \in REL$) into an *relevance weight* ($w_{rel} \in \mathbb{Z}$), e.g. $\mathrm{RW}(\texttt{hr}) = 3$. This function allows us to associate an integer number to a relevance degree.

We define with D the set of documents we take into account, with $N \in \mathbb{N}$ a natural number, and with D^N the set of all possible vectors of length N containing different orderings of the documents in D. We can also say that a vector in D^N represents a ranking list of length N of the documents D retrieved by an IR system. Let us consider a vector $\mathbf{v} \in D^N$, a natural number $j \in [1, N]$, and a relevance degree $rel \in REL$, then the *ground truth* function is defined as:

$$\begin{aligned} \mathrm{GT} : & D^N \times \mathbb{N} \to REL \\ & \mathbf{v}[j] \mapsto rel \end{aligned} \tag{1}$$

Equation 1 allows us to associate a relevance degree to the document $d \in D$ retrieved at position j of the vector $\mathbf{v}$, i.e. it associates a relevance judgment to each retrieved document in a ranked list.

In the following, we define with $\mathbf{r} \in D^N$ the vector of documents retrieved and ranked by a run r, with $\mathbf{i} \in D^N$ the ideal vector containing the best ranking of the documents in the pool (e.g. all highly relevant documents are grouped together in the beginning of the vector followed by fairly relevant ones and so on and so forth), and with $\mathbf{w} \in D^N$ the worst-case vector containing the worst rank of the documents retrieved by the pool (e.g. all the relevant documents are put in the end of the vector in the inverse relevance order).

From function GT we can point out a set called *relevance support* defined as:

$$RS(\mathbf{v}, rel) = \{j \in [1, N] \mid \mathrm{GT}(\mathbf{v}, j) = rel\} \tag{2}$$

which, given a vector $\mathbf{v} \in D^N$ – it can be a run vector $\mathbf{r}$, the ideal vector $\mathbf{i}$, or the worst-case vector $\mathbf{w}$ – and a relevance degree rel, contains the indexes j

of the documents of $\mathbf{v}$ with which the given relevance degree (rel) relevance is associated.

Given the ideal vector $\mathbf{i}$ and a relevance degree rel, we can define the *minimum rank* in $\mathbf{i}$ as the first position in which we find a document with relevance degree equal to rel. In the same way, we can define the *maximum rank* in $\mathbf{i}$ as the last position in which we find a document with relevance degree equal to rel. In formulas, they become:

$$\begin{aligned} \min_{\mathbf{i}}(rel) &= \min\big(RS(\mathbf{i}, rel)\big) \\ \max_{\mathbf{i}}(rel) &= \max\big(RS(\mathbf{i}, rel)\big) \end{aligned} \tag{3}$$

Given a vector $\mathbf{v}$ and a document at position $j \in [1, N]$, we can define the *Relative Position (RP)* as:

$$\mathrm{RP}(\mathbf{v}, j) = \begin{cases} 0 & \text{if } \min_{\mathbf{i}}\big(\mathrm{GT}(\mathbf{v}, j)\big) \leq j \leq \max_{\mathbf{i}}\big(\mathrm{GT}(\mathbf{v}, j)\big) \\ j - \min_{\mathbf{i}}(\mathrm{GT}\big(\mathbf{v}, j)\big) & \text{if } j < \min_{\mathbf{i}}\big(\mathrm{GT}(\mathbf{v}, j)\big) \\ j - \max_{\mathbf{i}}(\mathrm{GT}\big(\mathbf{v}, j)\big) & \text{if } j > \max_{\mathbf{i}}\big(\mathrm{GT}(\mathbf{v}, j)\big) \end{cases} \tag{4}$$

RP allows for pointing out misplaced documents and understanding how much they are misplaced with respect to the ideal case $\mathbf{i}$. Zero values denote documents which are within the ideal interval, positive values denote documents which are ranked below their ideal interval, and negative values denote documents which are above their ideal interval. Note that the greater the absolute value of $\mathrm{RP}(\mathbf{v}, j)$ is, the bigger is the distance of the document at position j from its ideal interval. From equation 4, it follows that $\mathrm{RP}(\mathbf{i}, j) = 0$, $\forall j \in [1, N]$.

Given a vector $\mathbf{v}$ and a document at position $j \in [1, N]$, we can define the *Cumulated Relative Position (CRP)* as:

$$\mathrm{CRP}(\mathbf{v}, j) = \sum_{k=1}^{j} \mathrm{RP}(\mathbf{v}, k) \tag{5}$$

For each position j, CRP sums the values of RP up to position j included. From equation 5, it follows that $\mathrm{CRP}(\mathbf{i}, j) = 0$, $\forall j \in [1, N]$.

We can point out the following properties for CRP:

- CRP can only be zero or negative before reaching the rank of the recall base (R);
- the faster the CRP curve goes down before R, the worse the run is;
- after R the CRP curve is non-decreasing;
- after that the last relevant document has been encountered, CRP remains constant;
- the sooner we reach the x-axis (balance point: $b_{\mathbf{r}}$), the better the run is.

In Figure 1 we can see a sketch of the CRP for a topic of a run. For a given topic there are two fixed values which are the rank of recall base (R) and the

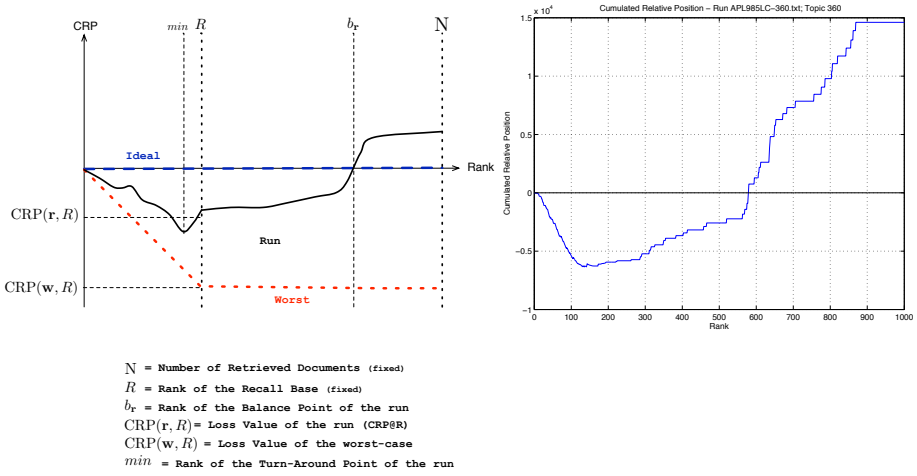


Fig. 1. Cumulative Relative Position sketch for a topic of a given run (on the left) and the CRP curve of a real run taken from TREC7.

number of retrieved documents (N); this allows us to compare systems on the R basis.

The principal indicator describing the CRP curve of a topic for a given run which is the *recovery value* (ρ) defined as the ratio between R and $b_{\mathbf{r}}$: $\rho = \frac{R}{b_{\mathbf{r}}}$.

The recovery-value is always between 0 and 1 ($0 < \rho \leq 1$) where $\rho = 1$ indicates a perfect ranking and $\rho \to 0$ a progressively worse ranking. Please note that $\rho \to 0$ when $b_{\mathbf{r}} \to \infty$.

3 Final Remarks

We think that the CRP offers several advantages in IR evaluation because (a) it is obvious to interpret and it gives an estimate of ranking performance as a single measure; (b) it is independent on outliers since it focuses on the ranking of the result list; (c) it directly reports the effort wasted in examining suboptimal rankings; (d) it is based on graded relevance.

Acknowledgements The work reported in this paper has been supported by the PROMISE network of excellence (contract n. 258191) project as a part of the 7th Framework Program of the European commission (FP7/2007-2013).

References

1. M. Angelini, N. Ferro, K. Järvelin, H. Keskustalo, A. Pirkola, G. Santucci, and G. Silvello. Cumulated Relative Position: A Metric for Ranking Evaluation. In *Information Access Evaluation meets Multilinguality, Multimodality, and Visual Analytics. Proc. of the 3rd Int. Conf. of the CLEF Initiative (CLEF 2012)*. Lecture Notes in Computer Science (LNCS) 7488, Springer, Heidelberg, Germany, 2012.

Visual Interactive Failure Analysis: Supporting Users in Information Retrieval Evaluation (Extended Abstract)*

Marco Angelini[2], Nicola Ferro[1], Giuseppe Santucci[2], and Gianmaria Silvello[1]

[1] University of Padua, Italy
{ferro,silvello}@dei.unipd.it
[2] "La Sapienza" University of Rome, Italy
{angelini,santucci}@dis.uniroma1.it

Abstract. Evaluation has a crucial role in *Information Retrieval (IR)* and developing tools to support researchers and analysts when analyzing results and investigating strategies to improve IR system performance can help make the analysis easier and more effective. To this purpose we present a Visual Analytics-based approach to support the analyst in performing failure and what-if analysis.

1 Introduction

Designing, developing, and testing an IR system is a challenging task, especially when it comes to understanding and analysing the behaviour of the system under different conditions in order to tune or to improve it as to achieve the level of effectiveness needed to meet the user expectations.

Failure analysis is especially resource demanding in terms of time and human effort, since it requires inspecting, for several queries, system logs, intermediate output of system components, and, mostly, long lists of retrieved documents which need to be read one by one in order to try to figure out why they have been ranked in that way with respect to the query at hand.

Considering this, it is important to define new ways to help IR researchers, analysts and developers to understand the limits and strengths of the IR system under investigation. Visual analytics techniques can give assistance to this process by providing graphic tools which interacting with IR techniques may ease the work of the users.

The goal of this paper is to exploit a visual analytics approach to design a methodology and develop an interactive visual system which support IR researchers and developers in conducting experimental evaluation and improving their systems by: (i) reducing the effort needed to conduct failure analysis; (ii) allowing them to anticipate what the impact of a modification to their system could be before needing to actually implement it.

* The extended version of this abstract has been published in [1].

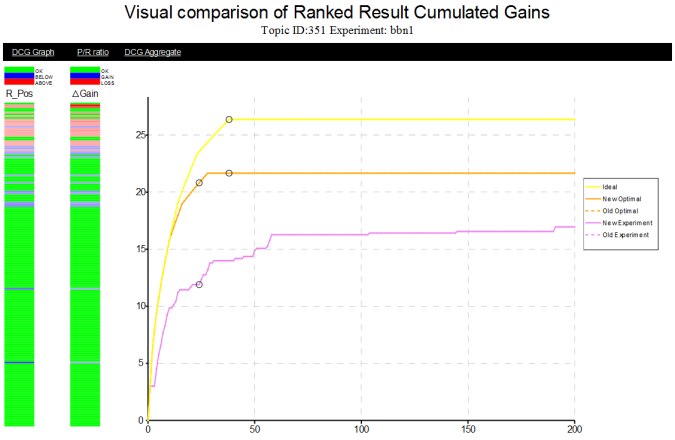


Fig. 1. The Visual Analytics prototype.

2 Failure Analysis

As far as the failure analysis is concerned, we introduce a ranking model that allows us to understand what happens when you misplace documents with different relevance grades in a ranked list. The proposed ranking model is able to quantify, rank by rank, the gain/loss obtained by an IR system with respect to both the ideal ranking, i.e. the best ranked list that can be produced for a given topic, and the optimal ranking, i.e. the best ranked list that can be produced using the documents actually retrieved by the system.

Starting from the *Discounted Cumulative Gain (DCG)* measures, we introduce two functions: the relative position, which quantifies how much a document has been misplaced with respect to its ideal (optimal) position, and the delta gain, which quantifies how much each document has gained/lost with respect to its ideal (optimal) DCG. On top of this ranking model, we propose a visualization, see Figure 1, where the DCG curves for the experiment ranking, the ideal ranking, and the optimal ranking are displayed together with two bars, on the left, representing the relative position and the delta gain. Please note that an equivalent graph can be obtained by using nDCG in the place of DCG.

The proposed ranking model and the related visualization are quite innovative because, usually, information visualization and visual analytics are exploited to improve the presentation of the results of a system to the end user, rather than applying them to the exploration and understanding of the performances and behaviour of an IR system. Secondly, comparisons are usually made with respect to ideal ranking only while our method allows user to compare a system also which respect to the optimal ranking produced with the system results, thus giving the possibility of better interpreting the obtained results [2].

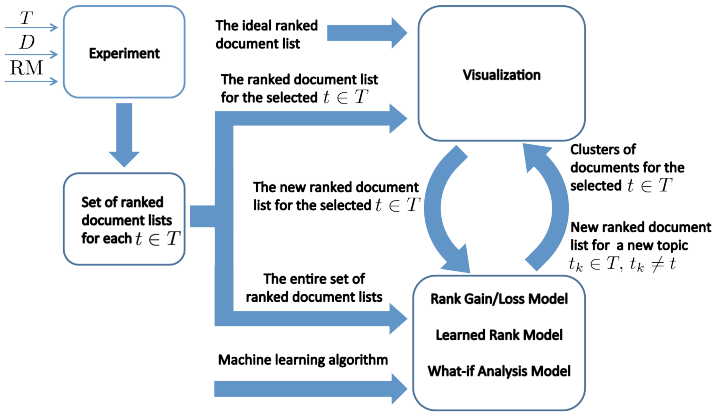


Fig. 2. Data pipeline.

3 What-If Analysis

When it comes to the what-if analysis, i.e. allowing users to anticipate the impact of a modification, we allow them to simulate what happens when you change the ranking of a given document for a certain topic not only in terms of which other documents will change their rank for that topic but also in terms of the effect that this change has on the ranking of the other topics. In other terms, we try to give the user an estimate of the "domino effect" that a change in the ranking of a single document can have. Moreover, when you simulate the move of a single document (and all the related documents), you produce a new ranking for a given topic which corresponds to a new version of your system, in our case a bug fixing in a component of the system. However, this new version of the system will now behave differently when ranking documents for the other topics in your experimental collection. Therefore, a change in the system which positively affects the performances on topic t_1 may have the side-effect to be detrimental for the performances on topic t_2 and we would like to give users an estimate also of this kind of "domino effect".

Therefore, the overall goal is to have an initial raw estimate of the effect of a planned modification before actually implementing it in terms of effect both for the topic under examination and for the other topics. This gives researchers and developers the possibility of exploring several alternatives before having to implement them and of determining a reasonable trade-off between the effort and costs for given modifications and the expected improvements.

Figure 2 shows the block diagram describing the pipeline of the data exchanged in the whole process. We consider the general-purpose IR scenario composed by a set of topics T, a collection of documents D, and a ranking model RM; an IR system for a given topic $t_k \in T$ retrieves a set of documents $D_j \subseteq D$.

The ranking model RM generates for each topic $t_k \in T$ a ranked document list RL_j. The whole set of ranked lists constitute the input for building the Clustering via Learning to Rank Model that is in charge of generating, for each document, a similarity cluster. The Visualization deals with one topic t at time: it takes as input the ranked document list for the topic t and the ideal ranked list, obtained choosing the most relevant documents in the collection D for the topic t and ordering them in the best way. While visually inspecting the ranked list, it is possible to simulate the effect of interactively reordering the list, moving a target document d and observing the effect on the ranking while this shift is propagated to all the documents of the cluster containing the documents similar to d. This cluster of documents simulates the "domino effect" within the given topic t.

When the analyst is satisfied with the results, i.e. when he has produced a new ranking of the documents that corresponds to the effect that is expected by modifications that are planned for the system, he can feed the Clustering via Learning to Rank Model with the newly produced ranked list, obtain a new model which takes into account the just introduced modifications, and inspecting the effects of this new model for other topics. This re-learning phase simulates the "domino effect" on the other topics different from t caused by a possible modification in the system.

4 Final Remarks

This paper presented a fully-fledged analytical and visualization model to support interactive exploration of IR experimental results. The overall goal of the paper has been to provide users with tools and methods to investigate the performances of a system and explore different alternatives for improving it avoiding a continuous iteration of trials-and-errors to see if the proposed modifications actually provide the expected improvements.

Acknowledgements The work reported in this paper has been supported by the PROMISE network of excellence (contract n. 258191) project as a part of the 7th Framework Program of the European commission (FP7/2007-2013).

References

1. M. Angelini, N. Ferro, G. Santucci, and G. Silvello. Visual Interactive Failure Analysis: Supporting Users in Information Retrieval Evaluation. In J. Kamps, W. Kraaij, and N. Fuhr, editors, *Proc. 4th Symposium on Information Interaction in Context (IIiX 2012)*. ACM Press, New York, USA, 2012.
2. E. Di Buccio, M. Dussin, N. Ferro, I. Masiero, G. Santucci, and G. Tino. To Rerank or to Re-query: Can Visual Analytics Solve This Dilemma? In *Multilingual and Multimodal Information Access Evaluation. Proc. of the 2nd Int. Conf. of the Cross-Language Evaluation Forum (CLEF 2011)*, pages 119–130. LNCS 6941, Springer, Heidelberg, Germany, 2011.

Myusic: a Content-based Music Recommender System based on eVSM and Social Media

Cataldo Musto[1], Fedelucio Narducci[2], Giovanni Semeraro[1], Pasquale Lops[1], and Marco de Gemmis[1]

[1] Department of Computer Science
University of Bari Aldo Moro, Italy
name.surname@uniba.it
[2] Department of Information Science, Systems Theory, and Communication
University of Milano-Bicocca, Italy
narducci@disco.unimib.it

Abstract. This paper presents Myusic, a platform that leverages social media to produce content-based music recommendations. The design of the platform is based on the insight that user preferences in music can be extracted by mining Facebook profiles, thus providing a novel and effective way to sift in large music databases and overcome the cold-start problem as well. The content-based recommendation model implemented in Myusic is eVSM [4], an enhanced version of the vector space model based on distributional models, Random Indexing and Quantum Negation. The effectiveness of the platform is evaluated through a preliminary user study performed on a sample of 50 persons. The results showed that 74% of users actually prefer recommendations computed by social media-based profiles with respect to those computed by a simple heuristic based on the popularity of artists, and confirmed the usefulness of performing user studies because of the different outcomes they can provide with respect to offline experiments.

1 Introduction and Related Work

One of the main issues of the so-called *personalization pipeline* is preference acquisition and elicitation. That step has always been considered the *bottleneck* in recommendation process since classical approaches for gathering user preferences are usually time consuming or intrusive. The widespread diffusion of social networks in the age of Web 2.0 offers a new interesting chance to overcome that problem, since users spend 22% of their time on social networks[3] and 30 billion pieces of content are shared on Facebook every month [3]. In this scenario, to harvest social media is a recent trend in the area of Recommender Systems (RSs): it can merge the un-intrusiveness of implicit user modeling with the accuracy of explicit techniques, since the information left by users is freely provided and actually reflects real preferences.

[3] http://blog.nielsen.com/nielsenwire/social/

This paper presents Myusic, a tool that provides users with music recommendations. The goal of the system is to catch user preferences in music and filter the huge amount of data stored in platforms such as iTunes or Amazon in order to produce personalized suggestions about artists users could like. The filtering model behind Myusic is eVSM, an enhanced extension of VSM based on distributional models, Random Indexing and Quantum Negation. As introduced in [4], eVSM provides a lightweight semantic representation based on distributional models, where each artist (and the user profile, as well) is modeled as a vector in a semantic vector space, according to the tags used to describe her and the co-occurrences between the tags themselves. The model is based on the assumption that a user profile can be built by combining the tag-based representation (obtained by crawling Last.fm platform) of the artists she is interested in. Next, classical similarity measures can be exploited to match item descriptions with content-based user profiles. A prototype version of Myusic was made available online for two months in order to design a user study and evaluate the effectiveness of the model as well as its impact on real users.

Generally speaking, this work concerns to the area of music recommendation. The commonly used technique for providing recommendations is collaborative filtering, implemented in very well known services, such as MyStrands[4], Last.fm[5] or iTunes Genius. An early attempt to recommend music using collaborative filtering was done by Shardanand [8]. Another trend is to use content-based recommendation strategies, which analyze diverse sets of low-level features (e.g. harmony, rhythm, melody), or high-level features (metadata or content-based data available in social media) [2] to provide recommendations. The use of Linked Data for music recommendation is investigated in [6]. Recently, Bu et al. [1] followed the recent trend of harvesting information coming from social media for personalization tasks and proposed its application for music recommendation. Finally, Wang et al. [9] showed the usefulness of tags with respect to other content-based sources.

The paper is organized as follows: the architecture of the systems is sketched in Section 2; Section 3 focuses on the results of a preliminary experimental evaluation and finally Section 4 contains conclusions and directions for future research.

2 Myusic: content-based music recommendations

The general architecture of Myusic is sketched in Figure 1. We can identify four main components:

Crawler. The Crawler module queries Last.fm through its public APIs to build a corpus of available artists. For each artist, the name, a picture, the title of the most popular tracks, their playcount and a set of tags that describe that artist are crawled. All the crawled data are locally stored.

[4] http://www.mystrands.com
[5] http://www.last.fm

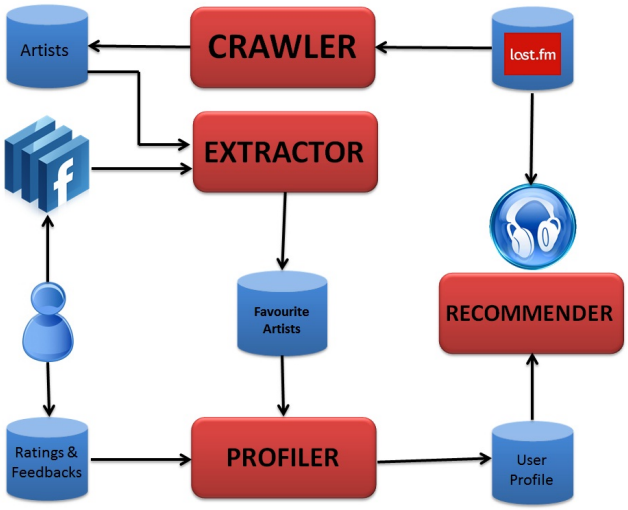


Fig. 1. Myusic architecture

Extractor. The EXTRACTOR module connects to Facebook, extracts artists the user likes (*Favourite Music* section in the Facebook profile, see Figure 2), and maps them to the data gathered from Last.fm in order to build a preliminary set of artists the user likes. This information is locally modeled in her own profile to let her receive recommendations even in her first interaction with Myusic, thus avoiding the cold-start. Implicit information coming from the links posted by the user and the events she attended are extracted, as well.

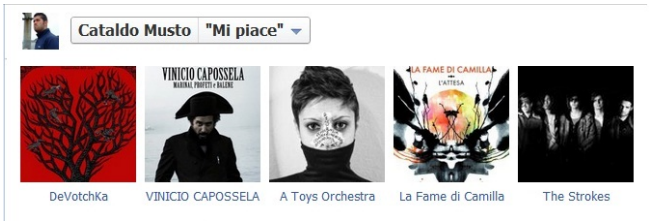


Fig. 2. User Preferences from a Facebook profile

Profiler. The process of building user profiles is performed in two steps. First, a weight is assigned to each artist returned by the EXTRACTOR. The weight of a specific artist is defined according to a simple heuristic: if a user posted a song, that information can be considered as a *light* evidence of her preference for that artist, while the fact that she explicitly clicked on "Like" on her Facebook page can be considered as a *strong* evidence. For example,

on a 5-point Likert scale, a score equal to 3 is assigned to the artists whose name appear among the links posted by the user, while a score equal to 4 is assigned to those occurring in her favorite Facebook pages. If an artist occurs in both lists (that is to say, the user likes it and posted a song, as well), 5 out of 5 is assigned as score. Next, a profiling model has to be chosen. The eVSM framework provides four different profiling models [5]: a basic profile (referred to as *RI*), a simple variant that exploits negative user feedbacks (called *QN*), and two weighted counterparts which give greater weight to the artists a user liked the most (respectively, *W-RI* and W-QN). Regardless the profiling model, in eVSM user profiles are defined in eVSM by means of two vectors, $\boldsymbol{p}_{+u}$ and $\boldsymbol{p}_{-u}$, which represent user preferences and negative feedbacks, respectively. They are defined as follows:

$$\boldsymbol{p}_{+u} = \sum_{i=1}^{|I_u^+|} \boldsymbol{a}_i * r(u, a_i) \tag{1}$$

$$\boldsymbol{p}_{-u} = \sum_{i=1}^{|I_u^-|} \boldsymbol{a}_i * (MAX - r(u, a_i)) \tag{2}$$

where I_u^+ is the set of user favorite artists, I_u^- is the set of artists the user dislikes, MAX is the highest rating that can be assigned to an item, $r(u, a_i)$ is the score assigned to the artist a_i and $\boldsymbol{a}_i$ is the vector space representation of the artist. Since each artist is described through a set of tags $t_1 \dots t_n$ extracted from Last.fm, the vector space representation is a weighted vector $\boldsymbol{a}_i = (w_{t_1}, \dots, w_{t_n})$ where w_{t_i} is the weight of the tag t_i. Generally speaking, W-QN model combines $\boldsymbol{p}_{+u}$ with $\boldsymbol{p}_{-u}$ through a Quantum Negation operator implemented in eVSM framework, while W-RI model exploits only the information coming from $\boldsymbol{p}_{+u}$ and does not take into account negative feedback. Finally, RI and QN follow the same insight of their weighted counterpart with the difference that they do not exploit the user rating $r(u, a_i)$, thus a uniform weight is given to each artist.

Recommender. Given a semantic vector space representation based on distributional models for both artists and user profiles, through similarity measures it is possibile to produce as output a ranked list of suggested artists. The cosine similarity for all the possible couples $(\boldsymbol{p}_u, \boldsymbol{a})$ is computed, where $\boldsymbol{p}_u$ is the vector space representation of user u, while $\boldsymbol{a}$ is the vector describing the artist a. Figure 3 shows an example of recommendation list. The platform allows the user to express feedbacks on recommendations. Positive and negative feedbacks are used to respectively update positive and negative profile vectors and to trigger the recommendation process again.

3 Experimental Evaluation

The goal of the experimental evaluation is to validate the design of the platform by carrying out a user study whose goal is to analyze the impact and the effective-

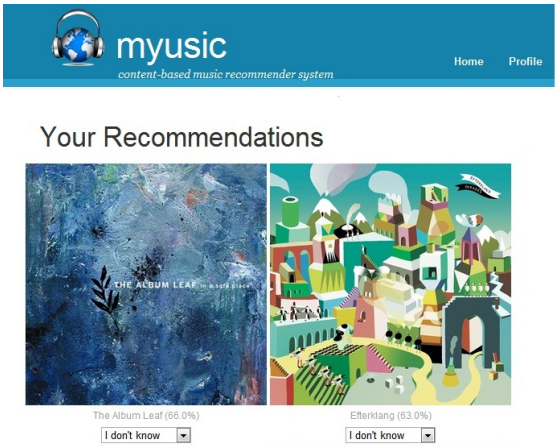


Fig. 3. An example of recommendation list in Myusic platform

ness of the different configurations of eVSM implemented in Myusic. Specifically, a user study involving 50 users under 30, heterogeneously distributed by sex, education and musical knowledge (according to the availability sampling strategy) has been performed. They interacted for two months with the online version of Myusic. A crawl of Last.fm was performed at the end of November, 2011 and data about 228,878 artists were extracted. Each user explicitly granted the access to her Facebook profile to extract data about favourite artists. At the end of the Extraction step, a set of 980 different artists the 50 users like were extracted from Facebook pages. Generally speaking, 1,720 feedbacks were collected: 1,495 of them came from Facebook profiles, while 225 were explicitly provided by the users (for example, expressing a feedback on their recommendations). The collected feedbacks were highly unbalanced since only 116 (6.71%) on 1,720 were negative. Last.fm APIs were exploited to extract the most popular tags associated to each artist. The less expressive and meaningful ones (such as *seenlive*, *cool*, and so on) were considered as noisy and filtered out. The design of the user study was oriented to answer to the following questions:

- **Experiment 1**: Does the cold-start problem can be mitigated by modeling user profiles which integrate information coming from social media?
- **Experiment 2**: Do the users actually perceive the utility of adopting weighting schemes and negation when user profiles are represented?
- **Experiment 3**: How does the platform perform in terms of novelty, serendipity and diversity of the proposed recommendations?

In the first experiment, users were asked to login and to extract their data from their own Facebook page. Next, a user profile was built according to a profiling model *randomly* chosen among the 4 described above and a preliminary set of recommendations was proposed to the target user. In order to evaluate

the effectiveness of the EXTRACTOR we compared the recommendation list generated through eVSM to a baseline represented by a list produced by simply ranking the most popular artists. Next, we asked users to tell which list they preferred. Obviously, they were not aware about which list was the baseline and which one was built through eVSM. A plot that summarizes users' answers is provided in Figure 4-a. It is straightforward to note that users actually prefer social media-based recommendations, since 74% of them preferred that strategy with respect to a simple heuristic based on popularity of the artists stored in database. However, even if the results gained by this profiling technique were outstanding, it is necessary to understand why 26% of the users simply preferred the most popular artists. Probably, there is a correlation between users' knowledge in music and the list they choose. It is likely that users with very *generic* tastes prefer a list of popular singers. Similarly, it is likely that users with a poor knowledge in music might prefer a list of well-known singers with respect to a list where most of the artists, even if related to their tastes, were unknown. A larger evaluation with users, split according to their musical knowledge, may be helpful to understand the dynamics behind users' choices. Similarly, it would be good to investigate the impact of the amount of the information extracted from Facebook profiles with the accuracy of the recommendations. The second experiment was performed in two steps. In the first step users were asked to login and to extract their data from their own Facebook page, as in Experiment 1. Next, two profiles were built by following the RI and the W-RI profiling models, respectively. Finally, recommendations were generated from both profiles, and users were asked to choose the configuration they preferred. As in Experiment 1, they were not aware about which recommendations were generated by exploiting their weighted profile and which ones were produced through its unweighted counterpart. Results of this experiments are shown in Figure 4-b. Differently from the results obtained from an *in-vitro* experiment performed in a scenario of movie recommendation [5], users did not perceive as useful the introduction of a weighting scheme designed to give higher significance to the artists the user likes the most. On the contrary, the RI profiling model was the preferred one for 70% of the users involved in the experiment. Similarly, in the second step of the experiment the RI profiling model was compared to the QN one, in order to evaluate the impact on user perception of modeling negative preferences. Also in this case the results were conflicting with the outcomes that emerged from the *in-vitro* experiment since 65% of the users preferred the recommendations generated through the profiling technique that does not model negative preferences. Even if the results of Experiment 2 did not confirmed the outcomes of the offline evaluation of eVSM they are actually interesting. First, they confirmed the usefulness of combining offline experiments with user studies thanks to the different outcomes they can provide. Indeed, in user-centered applications such as content-based recommender systems, user perception and user feedbacks play a central role and these factors need to be taken into account. In general, further investigation is needed because most of these results may be due to a specific *bias* of the designed experiment. As stated above, the extraction of data

from Facebook pages crawls information about what a specific user likes, so very few negative feedback were collected (less than 7%). Consequently, the negative part of the user profile was very poor and this might justify the results. It is likely that collecting more negative feedbacks would be enough to confirm the usefulness of negative information. Finally, in Experiment 3 users were asked to express their preference on the recommendations produced through the RI profiling model (since it emerged as the best one from the previous experiment) in terms of novelty, accuracy and diversity. The results of this experiment are sketched in Figure 4-c. In general, the results are encouraging since most of the users expressed a positive opinion about the system. Specifically, Myusic has a positive impact on final users in terms of trust, since the opinion of 92% of the users ranges from *Good* to *Very Good*. This is likely due to the good accuracy of the recommendations produced by the system. Indeed, more than 80% of the users considered as accurate or very accurate the suggestions of the system. Similarly, also the outcomes concerning diversity were positive, since more than 60% labeled the level of diversity among the recommendations as *Very Good*. The only aspect that needs improvements regards the novelty of recommendations since 34% of the users labeled as not novel the suggestions produced by the system. This outcome was somehow expected since overspecialization it is a typical problem of content-based recommender systems (CBRS). However, even if these results lead us to carry on this research, they have to be considered as preliminary since this evaluation needs to be extended by comparing results of eVSM with other state of the art models, such as LSI, VSM or collaborative filtering.

4 Conclusions and Future Directions

In this paper we proposed Myusic, a music recommendation platform. It implements a content-based recommender system based on eVSM, an enhanced version of classical VSM. The most distinguishing aspect of Myusic is the exploitation of Facebook profiles for acquiring user preferences. An experimental evaluation carried out by involving real users demonstrated that leveraging social media is an effective way for overcoming the cold-start problem of CBRS. On the other hand, the exploitation of relevance feedback and user ratings generally did not improve the predictive accuracy of Myusic. Users showed to trust the system, and Myusic also achieved good results in terms of accuracy and diversity of recommendations. Those results encouraged keeping on this research. In the future we will investigate the adoption of recommendation strategies tailored on the music background of each user, even by learning accurate interaction models in order to classify users [7]. Furthermore, we will try to introduce more unexpected suggestions. Experiments showed that novelty needs to be improved.

References

1. S. Bu, J.and Tan, C. Chen, C. Wang, H. Wu, L. Zhang, and X. He. Music recommendation by unified hypergraph: combining social media information and music

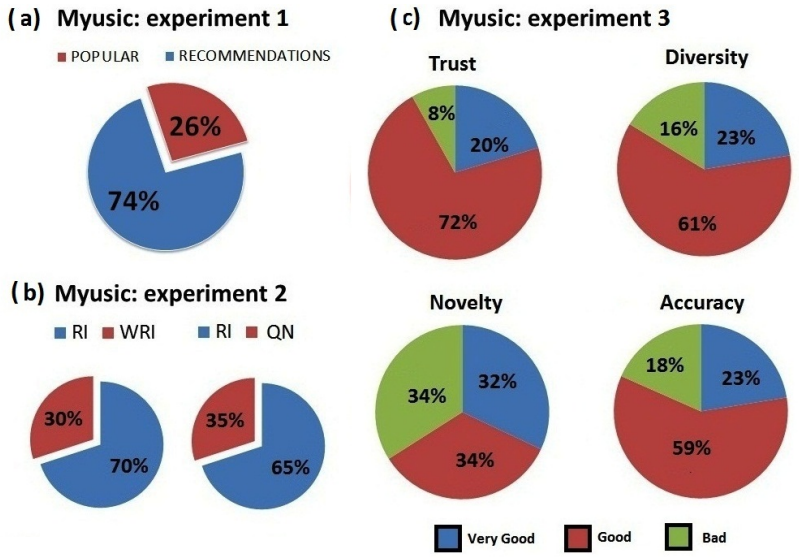


Fig. 4. Results of Experiments

content. In *Proceedings of the international conference on Multimedia*, MM '10, pages 391–400, New York, NY, USA, 2010. ACM.

2. C. Hahn, S. Turlier, T. Liebig, S. Gebhardt, and C. Roelle. Metadata Aggregation for Personalized Music Playlists. *HCI in Work and Learning, Life and Leisure*, pages 427–442, 2010.
3. J. Manyka, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers. Big data: The next frontier for innovation, competition, and productivity. Technical report, McKinsey Global Institute, 2011.
4. C. Musto. Enhanced vector space models for content-based recommender systems. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 361–364. ACM, 2010.
5. C. Musto, G. Semeraro, P. Lops, and M. de Gemmis. Random indexing and negative user preferences for enhancing content-based recommender systems. In *EC-Web*, pages 270–281, 2011.
6. A. Passant and Y. Raimond. Combining Social Music and Semantic Web for Music-Related Recommender Systems. In *Social Data on the Web, Workshop of the 7th International Semantic Web Conference*, Karlsruhe, Deutschland, Oktober 2008.
7. Giovanni Semeraro, Stefano Ferilli, Nicola Fanizzi, and Fabio Abbattista. Learning interaction models in a digital library service. In Mathias Bauer, Piotr J. Gmytrasiewicz, and Julita Vassileva, editors, *User Modeling*, volume 2109 of *Lecture Notes in Computer Science*, pages 44–53. Springer, 2001.
8. U. Shardanand. Social information filtering for music recommendation. Bachelor thesis, Massachusetts Institute of Technology, Massachusetts, 1994.
9. D. Wang, T. Li, and M. Ogihara. Are tags better than audio? the effect of joint use of tags and audio content features for artistic style clustering. In *ISMIR*, pages 57–62, 2010.

A Preliminary Study on a Recommender System for the Million Songs Dataset Challenge

Fabio Aiolli

University of Padova, Italy, email: aiolli@math.unipd.it

Abstract. In this paper, the preliminary study we have conducted on the Million Songs Dataset (MSD) challenge is described. The task of the competition was to suggest a set of songs to a user given half of its listening history and complete listening history of other 1 million people. We focus on memory-based collaborative filtering approaches since they are able to deal with large datasets in an efficient and effective way. In particular, we investigated on *i)* defining suitable similarity functions, *ii)* studying the effect of the "locality" of the collaborative scoring function, that is, how many of the neirest neighboors (and how much) they influence the score computation, and *iii)* aggregating multiple ranking strategies to define the overall recommendation. Using this technique we won the MSD challenge which counted about 150 registered teams.

1 Introduction

The Million Song Dataset Challenge [9] was a large scale, music recommendation challenge, where the task was the one to predict which songs a user will listen to, provided the listening history of the user. The challenge was based on the Million Song Dataset (MSD), a freely-available collection of meta data for one million of contemporary songs (e.g. song titles, artists, year of publication, audio features, and much more) [4]. About one hundred and fifty teams participated to the challenge. The subset of data actually used in the challenge was the so called Taste Profile Subset that consists of more than 48 million triplets *(user,song,count)* gathered from user listening histories. Data consists of about 1.2 million users and covers more than 380,000 songs in MSD. The user-item matrix is very sparse as the fraction of non-zero entries (the density) is only 0.01%.

The task of the challenge was to recommend the most appropriate songs for a user given half of her listening history and the complete history of another 1 million users. Thus, the challenge focused on the ordering of the songs on the basis of the relevance for a given user, and this makes the particular problem different from the more classical problem of predicting rates a user will give to unseen items [6, 11]. For example, popular tasks like the Netflix [3] and Movielens fall in this last case. A second important characteristic of the MSD problem is that we do not have explicit or direct feedback about what users like and how much they like it. In fact, we only have information of the form "user u listened to song i" without any knowledge about wether user u actually liked song i or not. A third important aspect of the MSD data is the presence of meta data concerning songs including title, artist, year of publication, etc. An interesting question then was wether this additional information could help or not. Finally,

given the huge size of the datasets involved, time and memory efficiency of the method used turned out to be another very important issue in the challenge.

Collaborative Filtering (CF) is a technology that uses the item by user matrix to discover other users with similar tastes as the active user for which we want to make the prediction. The intuition is that if other users, similar to the active user, already purchased a certain item, then it is likely that the active user will like that item as well. A similar (dual) consideration can be made by changing the point of view. If we know that a set of items are often purchased together (they are similar in some sense), then, if the active user has bought one of them, probably he/she will be interested to the other as well. In this paper, we show that, even if this second view has been far more useful to win the MSD competition, the first view also brings useful and diverse information that can be aggregated in order to boost the performance of the recommendation.

In Section 2, collaborative filtering is described and proposed as a first approach to solve the problem of MSD. In particular, we briefly discuss the most popular state-of-the-art techniques: model based and memory based CF methods. In the same section, we propose a variant of memory based CF particularly suitable to tasks with implicit feedback and binary ratings, and we propose a new parameterized similarity function that can be adapted to different applicative domains. Finally, in Section 3, empirical results of the proposed techniques are presented and discussed.

2 A Collaborative Filtering approach to the MSD task

Collaborative Filtering techniques use a database in the form of a user-item matrix R of preferences. In a typical Collaborative Filtering scenario, a set $\mathcal{U}$ of n users and a set $\mathcal{I}$ of m items exist and the entries of $R = \{r_{ui}\} \in \mathbb{R}^{n \times m}$ represent how much user u likes item i. In this paper, we assume $r_{ui} \in \{0, 1\}$ as this was the setting of the MSD challenge[1]. Entries r_{ui} represent the fact that user u have listened to (or would like to listen to) the song i. In the following we refer to items or songs interchangeably. The MSD challenge task has been more properly described as a *top-τ recommendation* task. Specifically, for any *active user* u, we want to identify a list of τ ($\tau = 500$ in the challenge) items $I_u \subseteq \mathcal{I}$ that he/she will like the most. Clearly, this set must be disjoint with the set of items already rated (purchased, or listened to) by the active user.

2.1 Model-based Collaborative Filtering

Model-based CF techniques construct a model of the information contained in the matrix R. There are many proposed techniques of this type, including Bayesian models, Clustering models, Latent Factor models, and Classification/Regression models.

In recent literature about CF, matrix factorization techniques [8] have become a very popular and effective choice to implement the CF idea. In this kind of models one tries to learn a linear embedding of both users and items into a smaller dimensional

[1] Note that in this definition we neglet the information given by the *count* attribute of the triplets indicating how many times the song has been listened to by a user. In fact, at the start of the competition, the organizers warned us on the fact that this attribute could be unreliable and absolutely not correlated with likings.

space. More formally, in its basic form, one needs to find two matrices $P \in \mathbb{R}^{n\times k}$ and $Q \in \mathbb{R}^{k\times m}$, such that $R = PQ$, in such a way to minimize a loss over training data. A common choice for this loss is the root mean square error (RMSE).

Despite the fact that matrix factorization is recognized as a state-of-the-art technique in CF, we note that it has some drawbacks that make it unsuitable for the MSD task. First of all, learning the model is generally computationally very expansive and this is a problem when the size of the matrix R is very large as it was in our case. Second, since it is tipically modelled as a regression problem, it does not seem very good for implicit feedback tasks. In this cases we only have binary values of relevance and the value 0 cannot properly be considered the same as unrelevant since the no-action on an item can be due to many other reasons beyond not liking it (the user can be unaware of the existence of the item, for example). Finally, baseline provided by the organizers of the challenge and other teams entries, both based on matrix factorization techniques, have shown quite poor results for this particular task, thus confirming our previous claims.

2.2 Memory-based Collaborative Filtering

In memory-based Collaborative Filtering algorithms, also known as Neighborhood Models, the entire user-item matrix is used to generate a prediction. Generally, given a new user for which we want to obtain the prediction, the set of items to suggest are computed looking at similar users. This strategy is typically referred to as *user-based recommendation*. Alternatively, in the *item-based recommendation* strategy, one computes the most similar items for the items that have been already purchased by the active user, and then aggregates those items to form the final recommendation. There are many different proposal on how to aggregate the information provided by similar users/items (see [11] for a good survey). However, most of them are tailored to classical recommendation systems and they are not promptly compliant with the implicit feedback setting where only binary relevance values are available. More importantly, computing the neirest neighbors requires the computation of similarities for every pair of users or songs. This is simply infeasible in our domain given the huge size of the datasets involved. So, we propose to use a simple weighted sum strategy the considers positive information only. A deeper analysis of this simple strategy will allows us to highlight an interesting duality which exists between user-based and item-based recommendation algorithms.

In the *user-based* type of recommendation, the scoring function, on the basis of which the recommendation is made, is computed by

$$h^U_{ui} = \sum_{v\in\mathcal{U}} f(w_{uv}) r_{vi} = \sum_{v\in\mathcal{U}(i)} f(w_{uv}),$$

that is, the score obtained on an item for a target user is proportional to the similarities between the target user u and other users v that have purchased the item i ($v \in \mathcal{U}(i)$). This score will be higher for items which are often rated by similar users.

On the other hand, within a *item-based* type of recommendation [5, 10], the target item i is associated with a score

$$h^S_{ui} = \sum_{j\in\mathcal{I}} f(w_{ij}) r_{uj} = \sum_{j\in\mathcal{I}(u)} f(w_{ij}),$$

and hence, the score is proportional to the similarities between item i and other items already purchased by the user u ($j \in \mathcal{I}(u)$).

Note that, the two formulations above do not have a normalization factor. A normalization with the sum of the similarities with the neighbors is tipically performed in neighboorhood models for tasks with explicit rates. In our case, we wanted to consider positive information only in the model. As we see in the following, an effect similar to the normalization is given by the function $f(w)$. The proposed strategy seems appropriate in our setting and makes the prediction much faster as we only need to compute pair similarities with only a few other (in the order of tens in our task) users/items.

The function $f(w)$ can be assumed monotonic not decreasing and its role is to emphasize/deemphasize similarity contributions in such a way to adjust the *locality* of the scoring function, that is how many of the nearest users/items really matter in the computation. As we will see, a correct setting of this function turned out to be very useful with the challenge data.

Interestingly, in both cases, we can decompose the user and item contributions in a linear way, that is, we can write $h^U_{ui} = \mathbf{w}_u^\top \mathbf{r}_i$, $\mathbf{w}_u \in \mathbb{R}^n$, and $h^S_{ui} = \mathbf{w}_i^\top \mathbf{r}_u$, $\mathbf{w}_i \in \mathbb{R}^m$. In other words, we are defining an embedding for items (in user based recommendation systems) and for users (in item based recommendation systems). In the specific case above, this corresponds to choose the particular vector $\mathbf{r}_i$ as the vector with n entries in $\{0, 1\}$, where $\mathbf{r}_i^{(u)} = r_{ui}$. Similarly, for the representation of users in item-based scoring, we choose $\mathbf{r}_u$ as the vector with m entries in $\{0, 1\}$, such that $\mathbf{r}_u^{(i)} = r_{ui}$. In the present paper we mainly focus on exploring how we can learn the vectors $\mathbf{w}_i$ and $\mathbf{w}_u$ in a principled way by using the entire user-item preference matrix on-the-fly when a new recommendation has to be done. Alternatively, we could also try to learn the weight vectors from data by noticing that a recommendation task can be seen as a multilabel classification problem where songs represent the labels and users represent the examples. We have performed preliminary experiments in this sense using the preference learning approach described in [1]. The results were promising but the problem in this case was the computational requirements of a *model-based* paradigm like this. For this reason we decided to postpone a further analysis of this setting to future works.

2.3 User-based and Song-based similarity

In large part of CF literature the cosine similarity is the standard measure of correlation and not much work has been done until now to adapt the similarity to a given problem. Our opinion is that it cannot exist a single similarity measure that can fit all possible domains where collaborative filtering is used. With the aim to bridge this gap, in this section, we try to define a parametric family of user-based and item-based similarities that can fit different problems.

In the challenge, we have not relevance grades since the ratings are binary values. This is a first simplification we can exploit in the definition of the similarity functions. The similarity function that is commonly used in this case, both for the user-based case and the item-based case, is the cosine similarity. In the case of binary grades the cosine similarity can be simplified as in the following. Let $\mathcal{I}(u)$ be the set of items rated by a

generic user u, then the cosine similarity between two users u, v is defined by

$$w_{uv} = \frac{|\mathcal{I}(u) \cap \mathcal{I}(v)|}{|\mathcal{I}(u)|^{\frac{1}{2}} |\mathcal{I}(v)|^{\frac{1}{2}}}$$

and, similarly for items, by setting $\mathcal{U}(i)$ the set of users which have rated item i, we obtain:

$$w_{ij} = \frac{|\mathcal{U}(i) \cap \mathcal{U}(j)|}{|\mathcal{U}(i)|^{\frac{1}{2}} |\mathcal{U}(j)|^{\frac{1}{2}}}.$$

The cosine similarity has the nice property to be symmetric but, as we show in the experimental section, it might not be the better choice. In fact, especially for the item case, we are more interested in computing how likely it is that an item will be appreciated by a user when we *already* know that the same user likes another item. It is clear that this definition is not symmetric. As an alternative to the cosine similarity, we can resort to the conditional probability measure which can be estimated with the following formulas:

$$w_{uv} = P(u|v) = \frac{|\mathcal{I}(u) \cap \mathcal{I}(v)|}{|\mathcal{I}(v)|}$$

and

$$w_{ij} = P(i|j) = \frac{|\mathcal{U}(i) \cap \mathcal{U}(j)|}{|\mathcal{U}(j)|}$$

Previous works (see [7] for example) pointed out that the conditional probability measure of similarity, $P(i|j)$, has the limitation that items which are purchased frequently tend to have higher values not because of their co-occurrence frequency but instead because of their popularity. In our opinion, this might not be a limitation in a recommendation setting like ours. Perhaps, this could be an undesired feature when we want to cluster items. In fact, this correlation measure has not to be thought of as a real similarity measure. As we will see, experimental results seem to confirm this hypothesis, at least in the item-based similarity case.

Now, we are able to propose a parametric generalization of the above similarity measures. This parametrization permits ad-hoc optimizations of the similarity function for the domain of interest. For example, this can be done by validating on available data. Specifically, we propose to use the following combination of conditional probabilities:

$$w_{uv} = P(v|u)^{\alpha} P(u|v)^{1-\alpha} \quad w_{ij} = P(j|i)^{\alpha} P(i|j)^{1-\alpha} \tag{1}$$

where $\alpha \in [0, 1]$ is a parameter to tune. As above, we estimate the probabilities by resorting to the frequencies in the data and derive the following:

$$w_{uv} = \frac{|\mathcal{I}(u) \cap \mathcal{I}(v)|}{|\mathcal{I}(u)|^{\alpha} |\mathcal{I}(v)|^{1-\alpha}} \quad w_{ij} = \frac{|\mathcal{U}(i) \cap \mathcal{U}(j)|}{|\mathcal{U}(i)|^{\alpha} |\mathcal{U}(j)|^{1-\alpha}}. \tag{2}$$

It is easy to note that the standard similarity based on the conditional probability $P(u|v)$ (resp. $P(i|j)$) is obtained setting $\alpha = 0$, the other inverted conditional $P(v|u)$ (resp. $P(j|i)$) is obtained setting $\alpha = 1$, and, finally, the cosine similarity case is obtained when $\alpha = \frac{1}{2}$. This analysis also suggests an interesting interpretation of the cosine similarity on the basis of conditionals.

2.4 Locality of the Scoring Function

In Section 2 we have seen how the final recommendation is computed by a scoring function that aggregates the scores obtained using individual users or items. So, it is important to determine how much each individual scoring component influences the overall scoring. This is the role of the function $f(w)$. In the following experiments we use the exponential family of functions, that is $f(w) = w^q$ where $q \in \mathbb{N}$. The effect of this exponentiation is the following. When q is high, smaller weights drop to zero while higher ones are (relatively) emphasized. At the other extreme, when $q = 0$, the aggregation is performed by simply adding up the ratings. We can note that, in the user-based type of scoring function, this corresponds to take the popularity of an item as its score, while, in the case of item-based type of scoring function, this would turn out in a constant for all items (the number of ratings made by the active user).

2.5 Ranking Aggregation

There are many sources of information available regarding songs. For example, it could be useful to consider the additional meta-data which are also available and to construct alternative rankings based on that. It is always difficult to determine a single strategy which is able to correctly rank the songs. An alternative is to use multiple strategies, generate multiple rankings, and finally combine those rankings. Typically, these different strategies are individually *precision oriented*, meaning that each strategy is able to correctly recommend a few of the correct songs with high confidence but, it may be that, other songs which the user likes, cannot be suggested by that particular ranker. Hopefully, if the rankers are different, then the rankers can recommend different songs. If this is the case, a possible solution is to predict a final recommendation that contains all the songs for which the single strategies are more confident. The stochastic aggregation strategy that we used in the challenge can be described in the following way. We assume we are provided with the list of songs, not yet rated by the active user, given in order of confidence, for all the basic strategies. On each step, the recommender randomly choose one of the lists according to a probability distribution p_i over the predictors and recommends the best scored item of the list which has not yet been inserted in the current recommendation. In our approach the best p_i values are simply determined by validation on training data.

3 Experiments and Results

In the MSD challenge we have: *i)* the full listening history for about 1M users, *ii)* half of the listening history for 110K users (10K validation set, 100K test set), and we have to predict the missing half. Further, we also prepared a "home-made" validation subset (*HV*) of the original training data of about 900K users of training (*HVtr*, with full listening history). The remaining 100K user's histories has been split in two halves (*HVvi* the visible one, *HVhi* the hidden one).

The experiments presented in this section are based on this *HV* data and compare different similarities and different approaches. The baseline is represented by the simple

popularity based method which recommends the most popular songs not yet listened to by the user. Besides the baseline, we report experiments on both the user-based and song-based scoring functions, and an example of the application of ranking aggregation. Given the size of the datasets involved we do not stress on the significance of the presented results. This is confirmed by the fact that the presented results do not differ significantly from the results obtained over the indipendent set of users used as the test set in the challenge.

3.1 Taste Profile Subset Stats

For completeness, in this section, we report some statistics about the original training data. In particular, the following table shows the minimum, maximum, and average, number of users per song and songs per user. The median value is also reported.

Data Statistics	min	max	ave	median
users per song	1	110479	125.794	13
songs per user	10	4400	47.45681	27

We can see that the large majority of songs have only few users which listened to it (less than 13 users for half of the songs) and the large majority of users have listened to few songs (less than 27 for half of the users). These characteristics of the dataset make the top-τ recommendation task quite challenging.

3.2 Truncated Mean Average Precision

Conformingly to the challenge, we used the truncated mAP (mean average precision) as the evaluation metric [9]. Let y denote a ranking over items, where $y(p) = i$ means that item i is ranked at position p. The mAP metric emphasizes the top recommendations. For any $k \leq \tau$, the *precision at k* (π_k) is defined as the proportion of correct recommendations within the top-k of the predicted ranking (assuming the ranking y does not contain the visible songs),

$$\pi_k(u, y) = \frac{1}{k} \sum_{p=1}^{k} r_{uy(p)}$$

For each user the (truncated) average precision is the average precision at each recall point:

$$AP(u, y) = \frac{1}{\tau_u} \sum_{p=1}^{\tau} \pi_k(u, y) r_{uy(p)}$$

where τ_u is the smaller between τ and the number of user u's positively associated songs. Finally, the average of $AP(u, y_u)$'s over all users gives the mean average precision (mAP).

3.3 Results

The result obtained on the HV data with the baseline (recommendation by popularity) is presented in Table 1(a). With this strategy, each song i simply gets a score proportional to the number of users $|\mathcal{U}(i)|$ which listened to the song.

In Table 1, we also report on experiments that show the effect of the locality parameter q for different strategies: item based and user based (both conditional probability and cosine versions). As we can see, beside the case IS with cosine similarity (Table 1c), a correct setting of the parameter q drammatically improves the effectiveness on HV data. We can clearly see that the best performance is reached with the conditional probability on an item based strategy (Table 1b).

Method	mAP@500
Baseline (Recommendation by Popularity)	0.02262

(a)

IS ($\alpha = 0$)	mAP@500
q=1	0.12224
q=2	0.16581
q=3	**0.17144**
q=4	0.17004
q=5	0.16830

(b)

IS ($\alpha = \frac{1}{2}$)	mAP@500
q=1	**0.16439**
q=2	0.16214
q=3	0.15587
q=4	0.15021
q=5	0.14621

(c)

US ($\alpha = 0$)	mAP@500
q=1	0.08030
q=2	0.10747
q=3	0.12479
q=4	0.13298
q=5	**0.13400**
q=6	0.13187
q=7	0.12878

(d)

US ($\alpha = \frac{1}{2}$)	mAP@500
q=1	0.07679
q=2	0.10436
q=3	0.12532
q=4	0.13779
q=5	0.14355
q=6	**0.14487**
q=7	0.14352

(e)

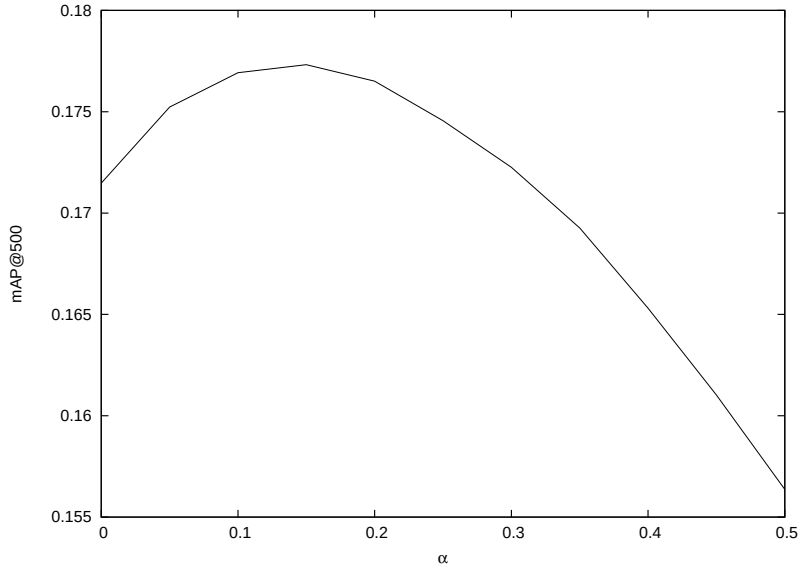
Table 1: Results obtained by the baseline, item-based (IS) and user-based (US) CF methods varying the locality parameter (exponent q) of the similarity function.

In Figure 1, results obtained fixing the parameter q and varying the parameter α for both user-based and item-based recommendation strategies are given. We see that, in the item-based case, the results improve when setting a non-trivial α. In fact, the best result has been obtained for $\alpha = 0.15$.

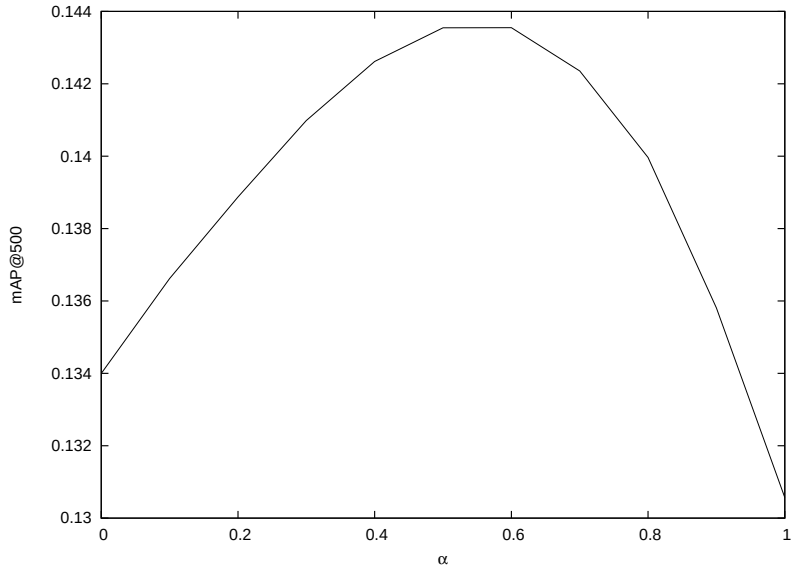
Finally, in Table 2, two of the best performing rankers are combined, and their recommendation aggregated, by using the stochastic algorithm described in Section 2.5. In particular, in order to maximize the diversity of the two rankers, we aggregated an item-based ranker with a user-based ranker. We can see that the combined performance improves further on validation data. Building alternative and effective rankers based on available meta-data is not a trivial task and it was not the focus of our current study. For this we decided to postpone this additional analysis to a near future.

3.4 Comparison with other approaches

We end this section by comparing our with other approaches that have been used in the challenge. Best ranked teams all used variants of memory based CF, besides the



(a) IS with $0 \leq \alpha \leq 0.5$, $q = 3$, best-mAP@500: $0.177322(\alpha = 0.15)$



(b) US with $0 \leq \alpha \leq 1$, $q = 5$, best-mAP@500: $0.143551(\alpha = 0.6)$

Fig. 1: Results obtained by item-based (IS) and user-based (US) CF methods varying the α parameter.

(IS, $\alpha = 0.15$, $q = 3$)	(US, $\alpha = 0.3$, $q = 5$)	mAP@500
0.0	1.0	0.14098
0.1	0.9	0.14813
0.2	0.8	0.15559
0.3	0.7	0.16248
0.4	0.6	0.16859
0.5	0.5	0.17362
0.6	0.4	0.17684
0.7	0.3	0.17870
0.8	0.2	**0.17896**
0.9	0.1	0.17813
1.0	0.0	0.17732

(a)

Table 2: Results obtained aggregating the rankings of two different strategies, item-based (IS, $\alpha = 0.15$, $q = 3$) and user-based (US, $\alpha = 0.3$, $q = 5$), with different combinations.

5-th ranked team that used the Absorption algorithm by YouTube [2] which is a graph based method that performs a random walk on the rating graph to propagate preferences information over the graph. On the other side, matrix factorization based techniques showed a very poor performance on this task and people working on that faced serious memory and time efficiency problems. Finally, some teams tried to inject meta data information in the prediction process with scarse results. In our opinion, this can be due to the fact that there is a lot of implicit information contained in the user's history and this is much more than explicit information one can get from metadata. We conclude that meta data information can be more effectively used in a *cold start* setting.

4 Conclusion

In this paper we have presented the technique we used to win the MSD challenge. The main contributions of the paper are: a novel scoring function for memory based CF that results particularly effective (and efficient) on implicit rating settings and a new similarity measure that can be adapted to the problem at hand. In the near future we want to investigate on the possibility of using metadata information to boost the performance and in a more solid way to aggregate multiple predictions.

5 Acknowledgments

This work was supported by the Italian Ministry of Education, University, and Research (MIUR) under Project PRIN 2009 2009LNP494_005. We would like to thank the referees for their comments, which helped improve this paper considerably.

References

1. Fabio Aiolli and Alessandro Sperduti. A preference optimization based unifying framework for supervised learning problems. In Johannes Fürnkranz and Eyke Hüllermeier, editors, *Preference Learning*, pages 19–42. Springer-Verlag, 2010.
2. Shumeet Baluja, Rohan Seth, D. Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 895–904, New York, NY, USA, 2008. ACM.
3. James Bennett, Stan Lanning, and Netflix Netflix. The netflix prize. In *In KDD Cup and Workshop in conjunction with KDD*, 2007.
4. Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.
5. Mukund Deshpande and George Karypis. Item-based top-*n* recommendation algorithms. *ACM Trans. Inf. Syst.*, 22(1):143–177, 2004.
6. Christian Desrosiers and George Karypis. A comprehensive survey of neighborhood-based recommendation methods. In *Recommender Systems Handbook*, pages 107–144. 2011.
7. George Karypis. Evaluation of item-based top-n recommendation algorithms. In *CIKM*, pages 247–254, 2001.
8. Yehuda Koren and Robert M. Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 145–186. 2011.
9. Brian McFee, Thierry Bertin-Mahieux, Daniel P.W. Ellis, and Gert R.G. Lanckriet. The million song dataset challenge. In *Proceedings of the 21st international conference companion on World Wide Web*, WWW '12 Companion, pages 909–916, New York, NY, USA, 2012. ACM.
10. Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW*, pages 285–295, 2001.
11. Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, January 2009.

Distributional models vs. Linked Data: exploiting crowdsourcing to personalize music playlists

Cataldo Musto[1], Fedelucio Narducci[2], Giovanni Semeraro[1], Pasquale Lops[1], and Marco de Gemmis[1]

[1] Department of Computer Science
University of Bari Aldo Moro, Italy
name.surname@uniba.it
[2] Department of Information Science, Systems Theory, and Communication
University of Milano-Bicocca, Italy
narducci@disco.unimib.it

Abstract. This paper presents Play.me, a system that exploits social media to generate personalized music playlists. First, we extracted user preferences in music by mining Facebook profiles. Next, given this preliminary playlist based on explicit preferences, we enriched it by adding new artists related to those the user already likes. In this work two different enrichment techniques are compared: the first one relies on knowledge stored on DBpedia while the latter is based on the similarity calculations between semantic descriptions of the artists. A prototype version of the tool was made available online in order to carry out a preliminary user study to evaluate the best enrichment strategy. This paper summarizes the results presented in EC-Web 2012 [3].

1 Introduction and Related Work

According to a recent study[3], 31,000 hours of music (and 28 million songs) are currently available on iTunes Store. As a consequence, the problem of information overload is currently felt for online music libraries and multimedia content, as well. However, the recent spread of social networks provides researchers with a rich source to draw to overcome the typical bottleneck represented by user preferences elicitation.

Given this insight, in this work we propose Play.me, a system that leverages social media for personalizing music playlists. The filtering model is based on the assumption that information about music preferences can be gathered from Facebook profiles. Next, explicit Facebook preferences may be enriched with new artists related to those the user already likes. In this paper we compare two different enrichment techniques: the first leverages the knowledge stored on DBpedia while the second is based on similarity calculations between semantics descriptions of artists. The final playlist is then ranked and finally presented

[3] http://www.digitalmusicnews.com/permalink/2012/120425itunes

to the user that can express her feedback. A prototype version of Play.me was made available online and a preliminary user study to detect the best enrichment technique was performed. Generally speaking, this work can be placed in the area of music recommendation (MR), a topic that has been widely covered in literature: an early attempt of handling MR problem is due to Shardanand [5], who proposed collaborative filtering to provide music recommendations. Similarly to our work, in [1] Lamere analyzed the use of tags as source for music recommendation, while the use of Linked Data is investigated in [4].

2 Play.me: personalized playlists generator

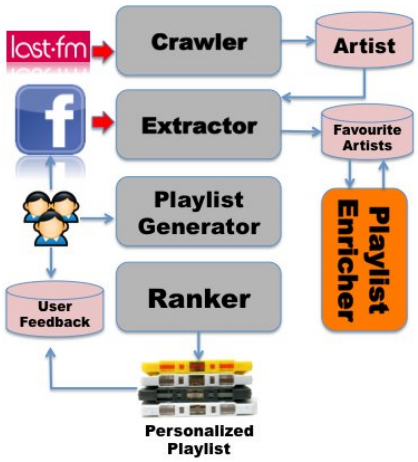


Fig. 1. Play.me architecture

The general architecture of Play.me is depicted in Figure 1. The generation is directly triggered by the user, who invokes the PLAYLIST GENERATOR module. The set of her favourite artists is built by mapping her preferences gathered from her own Facebook profile (specifically, by mining the links she posted as well as the pages she likes) with a set of artists extracted from Last.fm. Given this preliminary set, the PLAYLIST ENRICHER adds new artists by using different enrichment strategies. Finally, for each artist in that set, the most popular tracks are extracted and the final playlist is shown to the target user, who can express her feedback. A working implementation of Play.me has been made available online (Figure 2). For a complete description of the system it is possible to refer to [3], while in this paper we just focus on the enrichment algorithms.

Enrichment based on Linked Data. The first technique for enriching user preferences extracted from Facebook relies on the exploitation of DBpe-

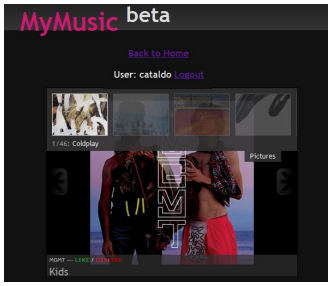


Fig. 2. Play.me screenshot

dia[4]. Our approach is based on the assumption that each artist can be mapped to a DBpedia node. The inceptive idea is that the similarity between two artists can be computed according to the number of properties they share (e.g. two Italian bands playing rock music are probably similar). Thus, we decided to use DBPEDIA-OWL:GENRE (describing the genre played by the artist) and DC-TERMS:SUBJECT, that provides information about the musical category. Operationally, we queried a SPARQL endpoint to extract the artists that share as many properties as possible with the target one. Finally, we ranked them according to their playcount in Last.fm. The first m artists returned by the endpoint are considered as related and added to the set of the favourite artists.

Enrichment based on Distributional Models. Each artist in Play.me is described through a set of tags (extracted from Last.fm), where each tag provides information about the genre played by the artist or describes features typical of her songs (e.g. *melanchonic*). According to the insight behind distributional models [2], each artist can be modeled as a point in a semantic vector space, and the position depends on the tags used to describe her and the co-occurrences between the tags themselves. The rationale behind this strategy is that the relatedness between two artists can be calculated by comparing their vector-space representation through the classical cosine similarity. So, we compute the cosine similarity between the target artist and all the other ones in the dataset, and the m with the highest scores are added to the list of favourite ones.

3 Experimental Evaluation

In the experimental evaluation we tried to identify the technique able to generate the most relevant playlists. We carried out an experiment by involving 30 users against a Last.fm crawl containing data on 228k artists. In order to identify the best enrichment technique, we asked users to use the application for three weeks. In the first two weeks the system was set with a different enrichment technique, while in the last a simple baseline based on the most popular artists was used. Given the playlist generated by the system, users were asked to express their

[4] http://dbpedia.org

feedback only on the tracks generated by the enrichment process. Results are reported in Table 1. The parameter m refers to the number of artists added by the enrichment algorithm for each one extracted from Facebook. It is worth to notice

Table 1. Results: each score represent the ratio of positive feedbacks.

	Artists		
Strategy	m=1	m=2	m=3
Linked Data	65.9%	64.6%	63.2%
Distributional Models	**76.3%**	**75.2%**	**69.7%**
Popularity	58%		

that both enrichment strategies outperform the baseline. This means that the social network data actually reflect user preferences. The enrichment technique that gained the best performance is that based on *distributional models*. However, even though this technique gained the best results, a deeper analysis can provide different outcomes. Indeed, with *m=3* the gap between the approaches drops down: this means that a pure content-based representation introduces more noise than DBpedia, whose effectiveness stays constant. The good results obtained by the baseline can be justified by the low diversity of the users involved in the evaluation. More details about the experimental settings are reported in [3].

4 Conclusions and Future Work

In this work we presented Play.me, a system for building music playlists based on social media. Specifically, we compared two techniques for enriching the playlists, the first based on DBpedia and the second based on similarity calculations in vector spaces. From the experimental session it emerged that the approach based on distributional models was able to produce the best playlists. Generally speaking, there is still space for future work since the enrichment might be tuned by analyzing different DBpedia properties or different tags. Furthermore, context-aware personalized playlists could be a promising research direction.

References

1. P. Lamere. Social Tagging and Music Information Retrieval. *Journal of New Music Research*, 37(2):101–114, 2008.
2. A. Lenci. Distributional approaches in linguistic and cognitive research. *Italian Journal of Linguistics*, (1):1–31, 2010.
3. C. Musto, G. Semeraro, P. Lops, M. de Gemmis, and F. Narducci. Leveraging social media sources to generate personalized music playlists. In *EC-Web 2012*.
4. A. Passant and Y. Raimond. Combining Social Music and Semantic Web for Music-Related Recommender Systems. In *Social Data on the Web, ISWC Workshop*, 2008.
5. U. Shardanand. Social information filtering for music recommendation. Bachelor thesis, Massachusetts Institute of Technology, Massachusetts, 1994.

Opinion and Factivity Analysis of Italian political discourse

Rodolfo Delmonte[1], Daniela Gîfu[2], Rocco Tripodi[1]

[1] Ca'Foscari University, Department Language Science,
Ca' Bembo, dd. 1075, 30123, Venice
delmont@unive.it, rocco.trip@gmail.com
[2]„Alexandru Ioan Cuza" University, Faculty of Computer Science,
16, General Berthelot St., 700483, Iaşi
daniela.gifu@info.uaic.ro

Abstract. The success of a newspaper article for the public opinion can be measured by the degree in which the journalist is able to report and modify (if needed) attitudes, opinions, feelings and political beliefs. We present a symbolic system for Italian, derived from GETARUNS, which integrates a range of natural language processing tools with the intent to characterise the print press discourse from a semantic and pragmatic point of view. This has been done on some 500K words of text, extracted from three Italian newspapers in order to characterize their stance on a deep political crisis situation. We tried two different approaches: a lexicon-based approach for semantic polarity using off-the-shelf dictionaries with the addition of manually supervised domain related concepts; another one is a feature-based semantic and pragmatic approach, which computes propositional level analysis with the intent to better characterize important component like factuality and subjectivity. Results are quite revealing and confirm the otherwise common knowledge about the political stance of each newspaper on such topic as the change of government that took place at the end of last year, 2011.

Keywords: journalist opinion, sentiment analysis, political discourse, lexical-semantic, syntax, print press, Government of Italy.

1 Introduction

In this paper, we discuss paradigms for evaluating linguistic interpretation of discourses as applied by a light scaled version the system for text understanding called GETARUNS. We focus on three aspects critical to a successful evaluation: creation of large quantities of reasonably good training data, lexical-semantic and syntactic analysis. Measuring the polarity of a text is usually done by text categorization methods which rely on freely available resources. However, we assume that in order to properly capture opinion and sentiment [6,10,11,17] expressed in a text or dialog, any system needs a linguistic text processing approach that aims at producing semantically viable representation at propositional level. In particular, the idea that the task may be solved by the use of Information Retrieval tools like Bag of

Words Approaches (BOWs) is insufficient. BOWs approaches are sometimes also camouflaged by a keyword based Ontology matching and Concept search [10], based on SentiWordNet (*Sentiment Analysis and Opinion Mining with WordNet*) [2]– more on this resource below -, by simply stemming a text and using content words to match its entries and produce some result [16]. Any search based on keywords and BOWs is fatally flawed by the impossibility to cope with such fundamental issues as the following ones, which Polanyi and Zaenen [12] named contextual valence shifters:
- presence of negation at different levels of syntactic constituency;
- presence of lexicalized negation in the verb or in adverbs;
- presence of conditional, counterfactual subordinators;
- double negations with copulative verbs;
- presence of modals and other modality operators.

It is important to remember that both PMI and LSA analysis [16] systematically omit function or stop words from their classification set of words and only consider content words. In order to cope with these linguistic elements we propose to build a propositional level analysis directly from a syntactic constituency or chunk-based representation. We implemented these additions on our system called GETARUNS (*General Text And Reference Understanding System)* which has been used for semantic evaluation purposes in the challenge called RTE and other semantically heavy tasks [1,4]. The output of the system is an xml representation where each sentence of a text or dialog is a list of attribute-value pairs. In order to produce this output, the system makes use of a flat syntactic structure and a vector of semantic attributes associated to the verb compound at propositional level and memorized. Important notions required by the computation of opinion and sentiment are also the distinction of the semantic content of each proposition into two separate categories: objective vs. subjective.

This distinction is obtained by searching for factivity markers again at propositional level [14]. In particular we take into account: modality operators like intensifiers and diminishes, modal verbs, modifiers and attributes adjuncts at sentence level, lexical type of the verb (from ItalWordNet classification, and our own), subject's person (if 3rd or not), and so on.

As will become clear below, we are using a lexicon-based [9,15] rather than a classifier-based approach, i.e. we make a fully supervised analysis where semantic features are associated to lemma and concept of the domain by creating a lexicon out of frequency lists. In this way the semantically labelled lexicon is produced in an empirical manner and fits perfectly the classification needs.

The paper is structured as follows. Section 2 comments on the role of print press discourse; Section 3 describes the system for multi-dimensional political discourse analysis. Section 4 presents comparative analysis of print press discourses collected during the Berlusconi's resignation in favour of Monti's nominating the President of Italian Government (October 12 – December 12, 2011). Finally, section 5 highlights interpretations anchored in our analysis and presents a conclusion.

2 Print press discourse

Mirror of contemporary society, located in permanent socio-cultural revaluation, the texts of print press can disrupt or use a momentary political power. In contemporary society, the struggles stake is no longer the social use of technology, but it is the huge production and dissemination of representations, informations and languages.

At present, the legitimacy of competence and credibility or reputation of political authority is increasingly in competition with mediatic credibility and the charisma already confirmed in public space. In political life we see how „heavy" actors are imposed, benefiting preferential treatment in their publicity and/or how insignificant actors, with reduced visibility, are ignored, even marginalized, notwithstanding their possibly higher reputation. Most of the times, launching the new actors is accompanied by changing others, intermediate body, the militants, condemned not only to mediatic silence, but simply silenced: in this way, the role of opinion leaders is drastically reduced.

Print press, in its various forms, assigns political significance to institutional activities and events in their succession; it forms the political life of a nation, from objective information to become the subject of public debate. In this case, the role of print press is double:

1. secure information as a credible discourse to end a rumor;
2. enter politics in language forms, so they become consistently interpretable in a symbolic system of representations.

The press is designed to legitimize the actions of politicians, attending their visibility efforts, confirming or increasing their reputation. Print press includes essentially political discourses, containing both a specific orientation and a political commitment. The reader has the possibility to choose what and when to read, leaving time to reflection, too. Disproportionality is a risk to the reality described.

No wonder why the people in power, if they intend to govern in peace, try to curb the enthusiasm of the media. Most of the times, through excellence in the elections, the print press is focused on topical issues, leading topics of public interest and events of internal and external social life. However, the perception of social reality depends on how it is presented. So the newspaper, like any commercial product, is dependent on aesthetic presentations that may distort any event-selection alternative to news items which are sensational and, often, negative (i.e. our comparative study).

3 The System GETARUNS

In this section we will present a detailed description of the symbolic system for Italian that we used in this experiment. The system is derived from GETARUNS, a multilingual system for deep text understanding with limited domain dependent vocabulary and semantics, that works for English, German and Italian and has been documented in the past 20 years or so with lots of publications and conference presentations[3,5]. The deep version of the system has been scaled down in the last ten years to a version that can be used with unlimited text and vocabulary, again for English and Italian. The two versions can work in sequence in order to prevent

failures of the deep version. Or they work separately to produce less constrained interpretations of the text at hand.

The "shallow" scaled version of GETARUNS has been adapted for the Opinion and Sentiment analysis and results have already been published for English [6]. Now, the current version which is aimed at Italian has been made possible by the creation of the needed semantic resources, in particular a version of SentiWordNed adapted to Italian and heavily corrected and modified. This version (see 3.0) uses weights for the English WordNet and the mapping of sentiment weights has been done automatically starting from the linguistic content of WordNet glosses. However, this process has introduced a lot of noise in the final results, with many entries totally wrong. In addition, there was a need to characterize uniquely only those entries that have a "generic" or "commonplace" positive, or negative meaning associated to them. This was deemed the only possible solution to the problem of semantic ambiguity, which could only be solved by introducing a phase of Word Sense Disambiguation which was not part of the system. So, we decided to erase all entries that had multiple concepts associated to the same lemma, and had conflicting sentiment values. We also created and added an ad hoc lexicon for the majority of concepts (some 3000) contained in the text we analysed, in order to reduce the problem of ambiguity. This was done again with the same approach, i.e. labelling only those concepts which were uniquely intended as one or the other sentiment, restricting reference to the domain of political discourse.

The system has been lately documented by our participation in the EVALITA (*Evaluation of NLP and Speech Tools for Italian*) challenge[1]. It works in a usual NLP pipeline: the system tokenizes the raw text and then searches for Multiwords. The creation of multiwords is paramount to understanding specific domain related meanings associated to sequences of words. This computation is then extended to NER (*Named Entity Recognition*), which is performed on the basis of a big database of entities, lately released by JRC (*Joint Research Centre*) research centre.[2] Of course we also use our own list of entities and multiwords.

Words that are not recognized by simple matching procedures in the big wordform dictionary (500K entries), are then passed to the morphological analyser. In case also this may fail, the guesser is activated, which will at first strip the word of its affixes. It will start by stripping possible prefixes and then analysing the remaining portion; then it will continue by stripping possible suffixes. If none of these succeeds, the word will be labelled as foreign word if the final character is not a vowel; a noun otherwise. We then perform tagging and chunking. In order to proceed to the semantic level, each nominal expression is classified at first on the basis of the assigned tag: proper nouns are used in the NER task. The remaining nominal expressions are classified using the classes derived from ItalWordNet (*Italian WordNet*)[3]. In addition to that, we have compiled specialized terminology databases for a number of common domains including: medical, political, economic, and military. These lexica are used to add a specific class label to the general ones derived from ItalWordNet. And in case the word or multiword is not present there, to uniquely classify them. The output of this

[1] http://www.evalita.it/
[2] http://irmm.jrc.ec.europa.eu/
[3] http://www.ilc.cnr.it/iwndb/iwndb_php/

semantic classification phase is a vector of features associated to the word and lemma, together with the sentence index and sentence position. These latter indices will then be used to understand semantic relations intervening in the sentence between the main governing verb and the word under analysis. Semantic mapping is then produced by using the output of the shallow parsing and the functional mapping algorithm which produce a simplified labelling of the chunks into constituent structure. These structures are produced in a bottom-up manner and subcategorization information is only used to choose between the assignments of functional labels for argumenthood. In particular, choosing between argument labels like SUBJ, OBJ2, OBL which are used for core arguments, and ADJ which is used for all adjuncts requires some additional information related to the type of governing verb.

The first element for Functional Mapping is the Verbal Complex, which contains all the sequence of linguistic items that may contribute to its semantic interpretation, including all auxiliaries, modals, adverbials, negation, clitics. We then distinguish passive from active diathesis and we use the remaining information available in the feature vector to produce a full-fledged semantic classification at propositional level. The semantic mapping includes, beside diathesis:
- Change in the World; Subjectivity and Point of View; Speech Act; Factitiviy; Polarity.

4 A comparative study

Whereas the aims of syntax and semantics in this system are relatively clear, the tasks of pragmatics are still hard to extract automatically. But, we have to recognize the huge relevance of pragmatics in analyzing political texts.

4.1 The corpus

For the elaboration of preliminary conclusions on the process of the change of the Italian government and president of government, we collected, stored and processed - partially manually, partially automatically -, relevant texts published by three national on-line newspapers having similar profiles[4].

For analytical results to be comparable to those taken so far by second author [20,21], we needed a big corpus, especially considering five rigorous criteria that we list below:

1. Type of message

Selection of newspapers was made taking into account the type of opinions circulated by the Editorial: pro, against Berlusconi and impartial. The following newspapers were thus selected:

a) Corriere della Sera - www.corriere.it (called The People Newspaper).
b) Libero - www.liberoquotidiano.it (pro Berlusconi).
c) La Republica – www.repubblica.it (against Berlusconi).

2. Period of time

[4] www.corriere.it, www.liberoquotidiano.it, www.repubblica.it

The interval time chosen should be large enough to capture the lexical-semantic and syntactic richness found in the Italian press. It was divided into three time periods. We specify them here below with their abbreviations, used during analysis.

A month before the resignation of Berlusconi (12 November 2011), abbreviated to OMBB: October 12 to November 11, 2011

The period between the presentation of Berlusconi's resignation and the appointment of Mario Monti as premier of the Italian Government, abbreviated with PTMB: 12 to 16 November 2011

A month after the resignation of Berlusconi, abbreviated with OMAB: November 17 to December 12, 2011.

Two keywords were commonly used to select items from the Italian press, that is the name of the two protagonists: (Silvio) Berlusconi (and appellations found in newspaper articles: Silvio, Il Cavaliere, Il Caimano) and (Mario) Monti.

We tried to select an archive rich enough for each of the three newspapers (meaning dozens of articles per day), the selected period of time as the one of interest, between average values. Text selection was made taking into account the subcriterion *Ordina per rilevanza* (order articles by relevance) that each web page of the corresponding newspapers made available. We then introduced a new subcriterion of selection: storing articles in the first three positions of each web page for every day of the research period. In particular we collected on average 250 articles per newspaper, that is 750 articles overall. Also number of tokens are on average 150K tokens per newspaper, i.e. 450K tokens overall. Computation time on a tower MacPro equipped with 6 Gb RAM and 1 Xeon quad-core was approximately 2 hours.

4.2 The syntactic and semantic analysis

In Fig. 1 below, we present comparative semantic polarity and subjectivity analyses of the texts extracted from the three Italian newspapers. On the graph we show differences in values for four linguistic variables: they are measured as percent value over the total number of semantic linguistic variables selected from the overall analysis and distributed over three time periods on X axis. To display the data we use a simple difference formula, where Difference value is subtracted from the average of the values of the other two newpapers for that class. Differences may appear over or below the 0 line. In particular, values above the 0x axis mean they assume positive or higher than values below the 0x axis, which have a negative import. The classes chosen are respectively: 1. propositional level polarity with NEGATIVE value; 2. factivity or factuality computed at propositional level, which contains values for non factual descriptions; 3. subjectivity again computed at propositional level; 4. passive diathesis. We can now evaluate different attitudes and styles of the three newspapers with respect to the three historical periods: in particular we can now appreciate whether the articles report facts objectively without the use of additional comments documenting the opinion of the journalist. Or if it is rather the case that the subjective opinion of the journalist is present only in certain time spans and not in others.

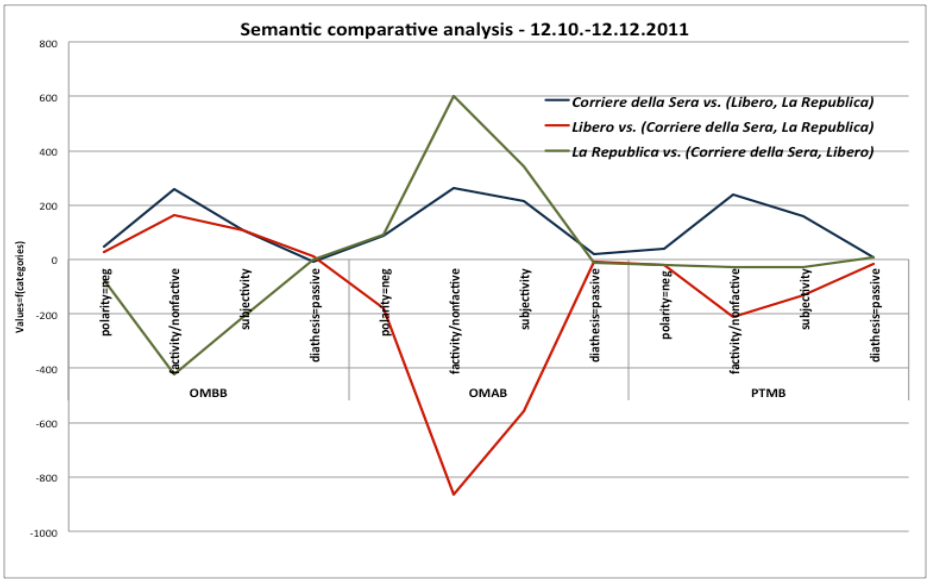


Fig. 1. Comparative semantic polarity analysis of three Italian newspapers.

So for instance, *Corriere*, the blue or darker line, has higher nonfactive values in two time spans, OMBB and PTMB; *Repubblica* values soar in OMAB. In the same period *Libero* has the lowest values; whereas in OMBB, *Libero* and *Corriere* have the highest values when compared with *Repubblica*. PTMB clearly shows up as a real intermediate period of turmoil which introduces a change: here *Repubblica* becomes more factual whereas *Libero* does the opposite. Subjectivity is distributed very much in the same way as factuality, in the three time periods even though with lesser intensity. *Libero* is the most factual newspaper, with the least number of subjective clauses. Similar conclusion can be drawn from the use of passive clauses, where we see again that *Libero* has the lowest number. The reasons for *Libero* having the lowest number of nonfactive clauses in OMAB, needs to be connected with the highest number of NEGATIVE polarity clauses, which is related to the nomination of Monti instead of Berlusconi, and is felt and is communicated to its readers as less reliable, trustable, trustworthy. Uncertainty is clearly shown in the intermediate period, PTMB, where *Corriere* has again the highest number of nonfactual clauses.

4.3 The pragmatic analysis

We show in this section the results outputted by GETARUNS when analysing the streams of textual data belonging to the three sections of the corpus (presented in section 4.1). In Fig. 2 we represent comparative differences between the three newspaper in the use of three linguistic variables for each time period. In particular, we plotted the following classes of pragmatic linguistic objects: 1. references to Berlusconi as entity (Silvio, Silvio_Berlusconi, Berlusconi, Cavaliere, Caimano); 2. references to Monti as entity (Monti, prof_Monti, professore, Mario_Monti,

super_Mario); 3. negative words or overall negative content words. To capture coreference mentions to the same entity we built a specialized coreference algorithm.

One month before Berlusconi's resignation (OMBB), we can highlight the opinions of the three dailies as follows: *Corriere della Sera* and *Libero* are concerned mostly with Berlusconi (see *Berlusconi occurrences*), with a remarkable difference however in terms of positive – *Libero* - vs negative – *Corriere* – comments. After Berlusconi resigned (OMAB) *Libero* is more concerned than the other two newspapers on Monti: negative appreciation is always higher with *Libero* and not with the other two. This can clearly be seen from the sudden dip of positive words. Finally in the intermediate period, both *Libero* and *Corriere* seem to be the most concerned with the new government, with the highest number of negative comments.

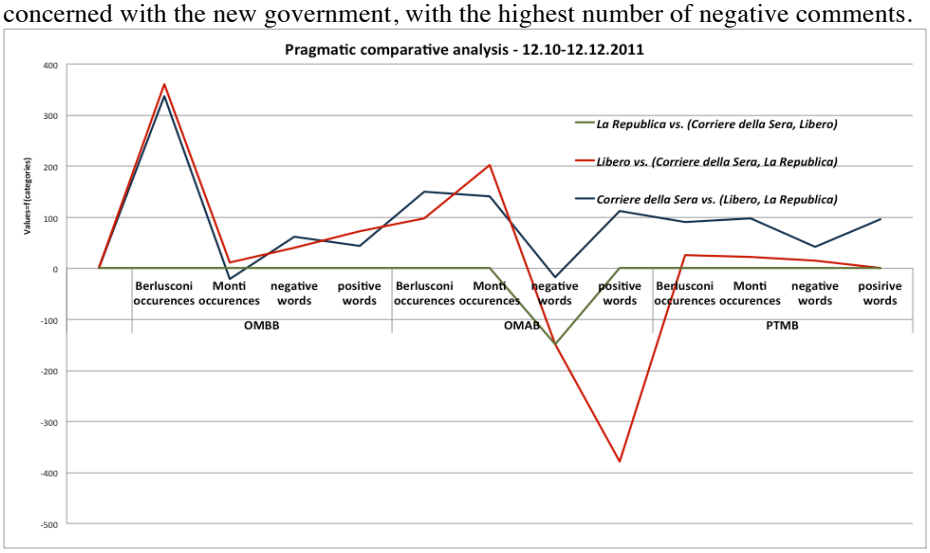


Fig. 2. Comparative pragmatic analysis of three Italian newspapers.

As shown in Fig.2, measuring the overall attitude with positive vs. negative affective content for each newspaper allows a clear cut subdivision in the three time periods. Table 1 below shows the same data in a more perspicuous manner. The percentages from Table 1 are organized as follows. Positive values are computed along time line distribution: for each newspaper, we compute the percentage referred to the each time slot. For instance, in OMBB positive values are distributed with the following subdivision in percent values: 33.88 for *Corriere*, 33.75 for *Libero*, and 32.37 for *Repubblica*. In other words, in OMBB, *Corriere* uses the most number of positive words. In fact, as can be easily noticed, *Corriere* is the newspaper that uses most positive keywords in all the three time periods. On the contrary, *Libero* is the newspaper that uses the least number of positive keywords apart from OMBB. *Repubblica* lies in the middle. The second number included in the same cell is needed to account for differences in number of tokens, and this in turn is due to differences in number of days considered for each time period: 31 for OMBB, 5 for PTBM and 26 for OMAB. Average values for each time period for each newspaper in part confirm percent values but also give a deepest idea of the actual numbers at play.

Newspaper / time period	Corriere della Sera		Libero		La Republica	
	positive	negative	positive	negative	positive	negative
OMBB	33.95% 52.1	35.49% 21.48	33.74% 51.9	32.6% 19.77	32.34% 49.77	31.91% 18.58
PTMB	42.36% 61.2	44.49% 21.8	24.4% 34.2	25.98% 11.4	33.24% 45.8	29.53% 16
OMAB	35.14% 54.88	32.68% 20.42	25.39% 39.58	28.21% 18	39.47% 49.12	39.12% 19.53

Table 1. Sentiment analysis of three Italian newspapers

Negative opinions are computed in the same way. These data can be interpreted as follow:

One month before Berlusconi's resignation (OMBB), both *Libero* and *Corriere della Sera* have more positive contents than *La Repubblica*, which can be interpreted as follows: Berlusconi's Government is considered a good one; in addition, *Libero*, has the lowest percentage of negative opinions about the current economic situation. In the intermediate period between Berlusconi's resignation and nomination of the new Prime Minister, Mario Monti (PTMB) we see that *Corriere* has by far the highest percentage of positive opinions, whereas *Libero* has the lowest. The other period, one month after the nomination of new prime minister, Mario Monti, (OMAB), we assist to a change of opinions. *Corriere della Sera* becomes more positive than other newspapers and also negative opinions are much higher: the new prime minister seems a good chance for the Italian situation; however, the economic situation is very bad. *Libero* – the newspaper owned by Berlusconi - becomes a lot less positive and less negative than the other two. This situation changes in the following time period, where Libero increases in positivity – but remains always the lowest value – and in negativity, but remains below the other two newspaper, on average. This can be regarded as a distinctive stylistic feature of *Libero* newspaper. As a whole, we can see that *Repubblica* is the one that undergoes less changes, if compared to *Libero* and *Corriere* which are the ones that undergo most changes in affective attitude.

We already saw in the Fig. 1 above that Libero is the newspaper with the highest number of nonfactual and subjective clauses in the OMAB time period: if we now add this information to the one derived from the use of positive vs. negative words, we see that the dramatic change in the political situation is no longer shown by the presence of a strong affective vocabulary, but by the modality of presenting important concepts related to the current political and economic situation, which becomes vague and less factual after Berlusconi resigned.

Eventually, we were interested in identifying semantic linguistic common area (identification of common words), also called common lexical fields, and their affective import (positive or negative). From previous tables, it can be easily noticed that all three newspapers use words with strong negative import, but with different frequency. Of course, this may require some specification, seeing the political context analyzed. So we decided to focus on a certain number of specialized concepts and

associated keywords that we extracted from the analysis to convey the overall attitude and feeling of the political situation. We collected in Table 2 below all words related to "Crisis Identification" (CIW for short) and noted down their absolute frequency of occurrence for each time interval.

CIW OMBB	*Corriere*	*Libero*	*Repub.*	***CIW OMAB***	*Corriere*	*Libero*	*Repub.*
1. crisis	124	71	94	1. crisis	50	21	110
sacrifice	4	14	4	sacrifice	9	23	16
rigour	5	4	4	rigour	23	18	10
austerity	0	6	6	austerity	6	2	0
2. battle	6	12	14	2. battle	14	4	8
dissent	2	8	8	dissent	0	4	0
dictator/ship	2	10	18	dictator/ship	2	6	2
3. fail/ure	8	13	9	3. fail/ure	21	8	15
collapse	10	6	12	collapse	8	2	4
drama/tic	12	14	18	drama/tic	4	0	8
dismiss/al	45	39	20	dismiss/al	3	2	15

Table 2. Crisis Identification words in two time periods

If we look at the list as being divided up into three main conceptualizations, we may regard the first one as denouncing the critical situation, the second one as trying to indicate some causes; and the last one as being related to the reaction to the crisis. It is now evident what the bias of each newspaper is, in relation to the incoming crisis:

- *Corriere della Sera* feels the "crisis" a lot deeper before Berlusconi's resignation, than afterwards when Monti arrives; the same applies to *Libero*. *La Repubblica* feels the opposite way. However, whereas "austerity" is never used by *La Repubblica* after B.'s resignation and it was used before it, this is the opposite of what *Corriere della Sera* does, the word appears only after B's resignation, never before. As to the companion word "sacrifice", *Libero* is the one that uses it the most, and as expected its appearance increases a lot after B.'s resignation, together with the companion word "rigour" that has the same behaviour. This word confirms *Corriere's* attitude towards Monti's nomination: it will bring "austerity, rigour and sacrifice".

- in the second half, the other interesting couple of concepts is linked to "battle, dissent, dictator". In particular, "battle" is used in the opposite way by *Corriere della Sera* when compared to the other two newspapers: the word appears more than the double in the second period, giving the impression that the new government will have to fight a lot more than the previous one. As to "dissent", all three newspapers use it in the same manner: it disappears in both *Corriere della Sera* and *La Repubblica*, and it is halved in *Libero*. Eventually the "dictator/ship" usually related to B. or to B.'s government: it is a critical concept for *La Repubblica* in the first period, and it almost disappears in the second one.

- as to the third part of the list, whereas *Libero* felt the situation "dramatic" before B.'s resignation, the dramaticity disappears afterwards. The same applies in smaller percentage to the other two newspapers. Another companion word, "collapse" has the

same behaviour: Monti's arrival is felt positively. However, the fear and the rumours of "failure" is highly felt by *Corriere della Sera* and *La Repubblica*, less so by *Libero*. This is confirmed by the abrupt disappearance of the concept of "dismiss/al" which dips to the lowest with *Libero*.

5 Conclusion

The analysis we proposed in this paper aims at testing if a linguistic perspective anchored in natural language processing techniques (in this case, the scaled version of GETARUNS system) could be of some use in evaluating political discourse in print press. If this proves to be feasible, then a linguistic approach would become a very relevant to an applicative perspective, with important effects in the optimization of the automatic analysis of political discourse.

However, we are aware that this study only sketches a way to go, and a lot more should be studied until a reliable discourse interpreting technology will become a tool in researcher's hands. We should also be aware of the dangers of false interpretation. For instance, if we take as example the three newspapers we used in our experiments, differences at the level of lexicon and syntax, which we have highlighted as differentiating them, should be attributed only partially to their idiosyncratic rhetorical styles, because these differences could also have editorial roots. Theoretically, at least, *Corriere della Sera*, should embody an impartial opinion, *Libero*, pro Berlusconi and *La Repubblica*, against him. But differences are more subtle, and in fact, in some cases, we could likewise classify *Libero* as being impartial, *Corriere della Sera* as being pro current government and *La Repubblica* as the only one being more critical on the current government disregarding its political stance. It remains yet to be decided the impact that the use of certain syntactic structures could have over a wider audience of political discourse. In other words, this study may show that automatic linguistic processing is able to detect tendencies in the manipulation of the interlocutor with the hidden role of detouring the attention of the audience from the actual communicated content in favor of the speaker's intentions.

Different intensities of emotional levels have been clearly highlighted, but we intend to organize a much more fine-grained scale of emotional expressions. It is a well-known fact that the audience can be easily manipulated (e.g., the social and economic class) by a social actor (journalist, political actor) when their themes are treated with excessive emotional tonalities (in our study, common negative words). In the future, we intend to extend the specialized lexicon for political discourse in order to individuate more specific uses of words in context, of those words which are ambiguous between different semantic classes, or between classes in the lexicon and outside the lexicon (in which case they would not have to be counted). We believe that GETARUNS has a range of features that make it attractive as a tool to assist any kind of communication campaign. We wish it to be rapidly adapted to new domains and to new languages (i.e. Romanian), and be endowed with a user-friendly web interface that offers a wide range of functionalities. The system helps to outline distinctive features which bring a new and, sometimes, unexpected vision upon the discursive feature of journalists' writing.

Acknowledgments: In performing this research, the second author was supported by the POSDRU/89/1.5/S/63663 grant.

References

1. Bos, Johan & Delmonte, Rodolfo (eds.): "Semantics in Text Processing (STEP), Research in Computational Semantics", Vol.1, College Publications, London (2008).
2. Esuli, A. and F. Sebastiani. Sentiwordnet: a publicly available lexical resource for opinion mining. In Proceedings of the 5th Conference on Language Resources and Evaluation LREC, 6, 2006.
3. Delmonte, R. (2007). Computational Linguistic Text Processing – Logical Form, Logical Form, Semantic Interpretation, Discourse Relations and Question Answering, Nova Science Publishers, New York.
4. Delmonte, R., Tonelli, S., Tripodi, R.: Semantic Processing for Text Entailment with VENSES, published at http://www.nist.gov/tac/publications/2009/papers.html in TAC 2009 Proceedings Papers (2010).
5. Delmonte, R. (2009). Computational Linguistic Text Processing – Lexicon, Grammar, Parsing and Anaphora Resolution, Nova Science Publishers, New York.
6. Delmonte R. and Vincenzo Pallotta, 2011. Opinion Mining and Sentiment Analysis Need Text Understanding, in "Advances in Distributed Agent-based Retrieval Tools", "Advances in Intelligent and Soft Computing", Springer, 81-96.
7. Gîfu, D. and Cristea, D.: Multi-dimensional analysis of political language, in J. J. (Jong Hyuk) Park, V. Leung, T. Shon, Cho-Li Wang (eds.) In Proc. of 7th FTRA International Conference on Future Information Technology, Application, and Service – FutureTech-2012, Vancouver, vol. 1, Springer (2012).
8. Hobbs, J. R., Stickel, M., Appelt, D., and Martin, P.: "Interpretation as Abduction", SRI International Artificial Intelligence Centre Technical Note 499 (1990).
9. Pennebaker, James W., Booth, Roger J., Francis, Martha E.: "Linguistic Inquiry and Word Count" (LIWC), at http://www.liwc.net/.
10. Kim, S.-M. and E. Hovy. Determining the sentiment of opinions. In Proceedings of the 20th international conference on computational linguistics (COLING 2004), page 1367–1373, August 2004.
11. Pang, B. and L. Lee. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42nd annual meeting of the Association for Computational Linguistics (ACL), page 271–278, 2004.
12. Polanyi, Livia and Zaenen, Annie: "Contextual valence shifters". In Janyce Wiebe, editor, Computing Attitude and Affect in Text: Theory and Applications. Springer, Dordrecht, 1–10 (2006).
13. Pollack, M., Pereira, F.: "Incremental interpretation". In Artificial Intelligence 50, 37-82 (1991).
14. Saurì R., Pustejovsky, J.: "Are You Sure That This Happened? Assessing the Factuality Degree of Events in Text", Computational Linguistics, 38, 2, 261-299 (2012).
15. Taboada, M., Brooke, J., Tofiloski, M., Voll, K. & Stede, M.: "Lexicon-based methods for sentiment analysis". In Computational Linguistics 37(2): 267-307 (2011).
16. Turney, P.D. and M.L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems (TOIS), pages 15–346, 2003.
17. Wiebe, Janyce, Wilson, Theresa, Cardie, Claire: "Annotating expressions of opinions and emotions in language". In Language Resources and Evaluation, 39(2):165–210 (2005).

Distributional Semantics for Answer Re-ranking in Question Answering*

Piero Molino, Pierpaolo Basile, Annalina Caputo,
Pasquale Lops, and Giovanni Semeraro

Dept. of Computer Science - University of Bari Aldo Moro
Via Orabona, 4 - I-70125, Bari (ITALY)
{piero.molino, pierpaolo.basile, annalina.caputo, pasquale.lops,
giovanni.semeraro}@uniba.it

Abstract. This paper investigates the role of Distributional Semantic Models (DSMs) into a Question Answering (QA) system. Our purpose is to exploit DSMs for answer re-ranking in QuestionCube, a framework for building QA systems. DSMs model words as points in a geometric space, also known as *semantic space*. Words are similar if they are close in that space. Our idea is that DSMs approaches can help to compute relatedness between users' questions and candidate answers by exploiting paradigmatic relations between words, thus providing better answer re-ranking. Results of the evaluation, carried out on the CLEF2010 QA dataset, prove the effectiveness of the proposed approach.

1 Introduction

Distributional Semantics Models (DSMs) represent word meanings through linguistic contexts. The meaning of a word can be inferred by the linguistic contexts in which the word occurs. The philosophical insight of distributional models can be ascribed to Wittgenstein's quote *"the meaning of a word is its use in the language"*. The idea behind DSMs can be summarized as follows: if two words share the same linguistic contexts they are somehow similar in the meaning. For example, analyzing the sentences "drink wine" and "drink beer", we can assume that the words "wine" and "beer" have similar meaning. Using that assumption, the meaning of a word can be expressed by the geometrical representation in a *semantic space*. In this space a word is represented by a vector whose dimensions correspond to linguistic contexts surrounding the word. The word vector is built analyzing (e.g. counting) the contexts in which the term occurs across a corpus. Some definitions of contexts may be the set of co-occurring words in a document, in a sentence or in a window of surrounding terms.

* This paper summarizes the main results already published in Molino, P., Basile, P., Caputo, A., Lops, P., Semeraro, G.: Exploiting Distributional Semantic Models in Question Answering. In: Sixth IEEE International Conference on Semantic Computing, ICSC 2012, Palermo, Italy, September 19-21, 2012. IEEE Computer Society 2012, ISBN 978-1-4673-4433-3.

This paper aims at exploiting DSMs for performing a task to which they have never been applied before, i.e. candidate answers re-ranking in Question Answering (QA), exploring how to integrate them inside a pre-existent QA system. Our insight is based on the ability of these spaces to capture paradigmatic relations between words which should result in a list of candidate answers related to the user's question.

In order to test the effectiveness of the DSMs for QA, we rely on a pre-existent QA framework called QuestionCube[1] [2]. QuestionCube is a general framework for building QA systems which exploits NLP algorithms, for both English and Italian, in order to analyze questions and documents with the purpose of allowing candidate answers obtained from the retrieved documents to be re-ranked by a pipeline of scorers. Scores assign a score to a candidate answer taking into account several linguistic and semantic features. Our strategy for exploiting DSMs consists in adding a new scorer to this pipeline, based on vector spaces built using DSMs. In particular, we propose four types of spaces: a classical Term-Term co-occurrence Matrix (TTM) used as baseline, Latent Semantic Analysis (LSA) applied to TTM, Random Indexing (RI) approach to reduce TTM dimension, and finally an approach which combines LSA and RI. The scorer will assign a score based on the similarity between the question and the candidate answers inside the DSMs.

2 Methodology

QuestionCube is a multilingual QA framework built using NLP and IR techniques. Question analysis is carried out by a full-featured NLP pipeline. The passage search step is carried out by Lucene, a standard off-the-shelf retrieval framework that allows TF-IDF and BM25 weighting. The question re-ranking component is designed as a pipeline of different scoring criteria. We derive a global re-ranking function combining the scores with CombSum. More details on the framework and a description of the main scorers is reported in [2]. The only scorers employed in the evaluation are: **Terms Scorer**, **Exact Sequence Scorer** and **Density Scorer**, a scorer that assign a score to a passage based on the distance of the question terms inside it. All the scorers have an enhanced version which adopts the combination of lemmas and PoS tags as features.

Our DSMs are constructed over a co-occurrence matrix. The linguistic context taken into account is a window w of co-occurring terms. Given a reference corpus[2] and its vocabulary V, a $n \times n$ co-occurrence matrix is defined as the matrix $\mathbf{M} = (m_{ij})$ whose coefficients $m_{ij} \in \mathbb{R}$ are the number of co-occurrences of the words t_i and t_j within a predetermined distance w. The $term \times term$ matrix $\mathbf{M}$, based on simple word co-occurrences, represents the simplest semantic space, called Term-Term co-occurrence Matrix (TTM). In literature, several methods to approximate the original matrix by rank reduction have been proposed. The aim of these methods varies from discovering high-order relations between entries to

[1] www.questioncube.com

[2] In our case the collection of documents indexed by the QA system.

improving efficiency by reducing its noise and dimensionality. We exploit three methods for building our semantic spaces: Latent Semantic Analysis (LSA), Random Indexing [1] (RI) and LSA over RI ($LSARI$). $LSARI$ applies the SVD factorization to the reduced approximation of $\mathbf{M}$ obtained through RI. All these methods produce a new matrix $\hat{\mathbf{M}}$, which is a $n \times k$ approximation of the co-occurrence matrix $\mathbf{M}$ with n row vectors corresponding to vocabulary terms, while k is the number of reduced dimensions. We integrate the DSMs into the framework creating a new scorer, the **Distributional Scorer**, that represents both question and passage by applying addition operator to the vector representation of terms they are composed of. Furthermore, it is possible to compute the similarity between question and passage exploiting the cosine similarity between vectors using the different matrices.

3 Evaluation

The goal of the evaluation is twofold: (1) proving the effectiveness of DSMs into our question answering system and (2) providing a comparison between the several DSMs.

The evaluation has been performed on the *ResPubliQA 2010 Dataset* adopted in the *2010 CLEF QA Competition* [3]. The dataset contains about 10,700 documents of the European Union legislation and European Parliament transcriptions, aligned in several languages including English and Italian, with 200 questions. The adopted metric is the accuracy $a@n$ (also called $success@n$), calculated considering only the first n answers. If the correct answer occurs in the top n retrieved answers, the question is marked as correctly answered. In particular, we take into account several values of n =1, 5, 10 and 30. Moreover, we adopt the Mean Reciprocal Rank (MRR) as well, that considers the rank of the correct answer. The framework setup used for the evaluation adopts Lucene as document searcher, and uses a NLP Pipeline made of a stemmer, a lemmatizer, a PoS tagger and a named entity recognizer. The different DSMs and the classic TTM have been used as scorers alone, which means no other scorers are adopted in the scorers pipeline, and combined with the standard scorer pipeline consisting of the Simple Terms (ST), the Enhanced Terms (ET), the Enhanced Density (ED) and the Exact Sequence (E) scores. Moreover, we choosed empirically the parameters for the DSMs: the window w of terms considered for computing the co-occurrence matrix is 4, while the number of reduced dimensions considered in LSA, RI and LSARI is equal to 1,000.

The performance of the standard pipeline, without the distributional scorer, is shown as a baseline. The experiments have been carried out both for English and Italian. Results are shown in Table 1, witch reports the accuracy $a@n$ computed considering a different number of answers, the MRR and the significance of the results with respect to both the baseline (†) and the distributional model based on TTM (‡). The significance is computed using the non-parametric Randomization test. The best results are reported in bold.

Table 1. Evaluation Results for both English and Italian

		English					Italian				
	Run	a@1	a@5	a@10	a@30	MRR	a@1	a@5	a@10	a@30	MRR
alone	TTM	0.060	0.145	0.215	0.345	0.107	0.060	0.140	0.175	0.280	0.097
	RI	0.180	0.370	0.425	0.535	0.267‡	0.175	0.305	0.385	0.465	0.241‡
	LSA	**0.205**	**0.415**	**0.490**	0.600	**0.300**‡	0.155	0.315	0.390	0.480	0.229‡
	LSARI	0.190	0.405	**0.490**	**0.620**	0.295‡	**0.180**	**0.335**	**0.400**	**0.500**	**0.254**‡
combined	*baseline*	*0.445*	*0.635*	*0.690*	*0.780*	*0.549*	*0.445*	*0.635*	*0.690*	*0.780*	*0.549*
	TTM	0.535	0.715	0.775	0.810	0.614	0.405	0.565	0.645	0.740	0.539†
	RI	0.550	0.730	0.785	**0.870**	**0.637**†‡	0.465	**0.645**	**0.720**	**0.785**	0.555†
	LSA	**0.560**	0.725	**0.790**	0.855	**0.637**†	0.470	**0.645**	0.690	**0.785**	0.551†
	LSARI	0.555	**0.730**	**0.790**	**0.870**	0.634†	**0.480**	0.635	0.690	**0.785**	**0.557**†‡

Considering each distributional scorer on its own, the results prove that all the proposed DSMs are better than the TTM, and the improvement is always significant. The best improvement for the MRR in English is obtained by LSA (+180%), while in Italian by LSARI (+161%). Taking into account the distributional scorers combined with the standard scorer pipeline, the results prove that all the combinations are able to overcome the baseline. For English we obtain an improvement in MRR of about 16% with respect to the baseline and the result obtained by the TTM is significant. For Italian, we achieve a even higher improvement in MRR of 26% with respect to the baseline using LSARI. The slight difference in performance between LSA and LSARI proves that LSA applied to the matrix obtained by RI produces the same result of LSA applied to TTM, but requiring less computation time, as the matrix obtained by RI contains less dimensions than the TTM matrix.

Finally, the improvement obtained considering each distributional scorers on its own shows a higher improvement than their combination with the standard scorer pipeline. This suggests that a more complex method to combine scorers should be used in order to strengthen the contribution of each of them. To this purpose, we plan to investigate some learning to rank approaches as future work.

References

1. Kanerva, P.: Sparse Distributed Memory. MIT Press (1988)
2. Molino, P., Basile, P.: QuestionCube: a Framework for Question Answering. In: Amati, G., Carpineto, C., Semeraro, G. (eds.) IIR. CEUR Workshop Proceedings, vol. 835, pp. 167–178. CEUR-WS.org (2012)
3. Penas, A., Forner, P., Rodrigo, A., Sutcliffe, R.F.E., Forascu, C., Mota, C.: Overview of ResPubliQA 2010: Question Answering Evaluation over European Legislation. In: Braschler, M., Harman, D., Pianta, E. (eds.) Working notes of ResPubliQA 2010 Lab at CLEF 2010 (2010)

INSEARCH
A platform for Enterprise Semantic Search

Diego De Cao, Valerio Storch, Danilo Croce, and Roberto Basili

Department of Enterprise Engineering
University of Roma, Tor Vergata
00133 Roma, Italy
{decao,storch,croce,basili}@info.uniroma2.it

Abstract. This paper discusses the system targeted in the INSEARCH EU project. It embodies most of the state-of-the-art techniques for Enterprise Semantic Search: highly accurate lexical semantics, semantic web tools, collaborative knowledge management and personalization. An advanced information retrieval system has been developed integrating robust semantic technologies and industry-standard software architectures for proactive search as well as personalized domain-specific classification and ranking functionalities.

1 Introduction

Innovation is an unstructured process in most of Small and Medium Sized Enterprises (SMEs). The so called "Innovation Management Techniques", considered by the European Commission as an useful driver to improve competitiveness, are still underutilized by SMEs. Such techniques include Knowledge Management, Market Intelligence, Creativity Development, Innovation Project Management and Business Creation. However, within these techniques, the Creativity Development techniques are the less used among SMEs[1]. The only activity performed by almost all SMEs is the search for external information, in different sources such as the web, patent databases, in trade fairs or discussing with clients and partners. The main source of information for SMEs is the Internet search [7], an activity realized by more than 90% of SMEs when dealing with innovation. Knowledge and information are often distributed in heterogeneous and unstructured sources across networked systems and organizations. Search for entities (such as competitors or new products) is not always sufficient as search for knowledge, as the one related to novel processes or brands and marketing analysis (whereas connected to large scale opinion mining), is based upon richer information.

The system targeted in the INSEARCH EU project[2] embodies most of the ideas of the currently *en vogue* Enterprise Semantic Search technologies [4]. In

[1] European Commission, DG Enterprise Innovation management and the knowledge driven economy - January 2004

[2] FP7-SME-2010-1, Research for the benefit of specific groups, GA n. 262491

order to determine the core functionalities in the targeted system, an analysis involving 90 SMEs has been performed during the INSEARCH project to understand the process of searching within the innovation process. Most of the SMEs (92% of 90 interviewed SMEs) declared to make use of market and/or technology information when planning a technological innovation. Such informations are used to collect novel information for innovative ideas, performing prior art investigation, acquiring knowledge for technical planning or just gather inspiration and ideas. This search targets product and processes and it is mainly performed on scientific Web Sites and Competitors web site.

In these scenarios, keyword-based search related to product types and functions of the products are still used to retrieve information related to innovation processes. Search is mostly performed through iterative searches, evaluating search results through the very first lines of documents/web sites. Overall, the most requested knowledge extraction features are related to finding patterns within documents to propose possible innovation or customer requirements. This requirements are in line with the INSEARCH proposed approach of making usage of a TRIZ based methodology [1], to abstract functionalities from the specific innovation case under study and search for information through specific patterns (the TRIZ based Object-Action-Tool patterns) that could propose to SMEs possible technology innovations for the system under study.

In this paper the overall INSEARCH framework and its corresponding distributed system will be described, focusing on the advantage of integrating in a systematic fashion the benefits of analytical natural language processing tools, the adaptivity supported by inductive methods as well as the robustness characterizing advanced document management architectures built over interoperability standards in the Semantic Web (such as the iQser GIN Server). In the rest of the paper, section 2 discusses the different involved paradigms used to support semantic search. The overall architecture is presented in Section 3 that also show some typical user interactions with the system. Finally, section 4 derives the conclusions.

2 Integrating Ontological and Lexical Knowledge

2.1 Modeling Knowledge for Enterprise Semantic Search

Ontologies correspond to semantic data models that are shared across large user communities. The targeted enterprise or networked enterprises in INSEARCH are a typical expression of such communities where semantics can be produced, reused and validated in a shared (i.e. collaborative) manner. However, while knowledge representation languages are very useful to express machine readable models, the interactive and user-driven nature of most of the task focused by INSEARCH emphasize the role of natural language as the true user-friendly knowledge exchange language. Natural languages naturally support all the expressions used by producers and consumers of information and their own semantics is rich enough to provide strong basis for most of the meaningful inferences needed in INSEARCH. Document classification aiming at recognizing the interests of a

user in accessing a text (e.g. a patent) requires a strongly linguistic basis as texts are mostly free and unstructured, as in [13]. In retrieval, against user queries, document ranking functions are inherently based on lexical preferences models, whose traditional TF-IDF models are just shallow surrogates. Moreover, the rich nature of the patterns targeted by INSEARCH (e.g. Object-Action-Tool triple foreseen by the TRIZ methodology) is strongly linguistic, as the same information is usually expressed in text with a huge freedom, and as for the language variability itself. Consider as an example that if a tool like a *packing machine* is adopted for the manufacturing of coffee boxes, several sentences can make reference to them, e.g. *packing machine applied to coffee*, *coffee is packed through dedicated machines* or *dedicated machines are used to pack small coffee boxes of 10 inch*.

Organizing knowledge through the SKOS concept scheme. Users are able to access, create or refine descriptions of a domain in the form of "tree of topics", or simply topic-trees (modeled as SKOS [18] concept schemes) which will support their contextual search throughout the system. These topics act as collectors for documents which expose all those textual contents that can be naturally associated to their definition. They are under all aspects a controlled hierarchical vocabulary of tags offered to a community of users. Behind every tag a large term vocabulary is used in order to exploit the corresponding topic semantics during search activities. Topic-document associations may be discovered through information push by the mass: users inside a community contribute their bookmarks to the system. On the other hand, it can be achieved by the system itself, by machine learning from the above information, automatically creating topic associations for massive amount of documents which are gathered through the multichannel multimodal document discovery and acquisition component, as discussed in [13]. Examples of SKOS topic for the specific domain of the *Innovation Engineering* domain are reported in Fig. 1. Main SKOS concepts are `Research and Intellectual Properties` (organizing scientific papers or patents) and `Tecnology`. The latter can be specified with the concept `biotecnology` or `material` and so on. Apart from their role of document containers, topics may be described by enriching them with annotations, comments and multiple lexicalizations for the various languages supported by INSEARCH, so that their usage is informally clarified to human users, possibly enforcing their consistent adoption across the community.

User Management. In INSEARCH, standard models and technologies of the RDF [10] family have been adopted to allow each user to view his own SKOS ontology. It requires to model the information associated to user management, domain modeling and user data. The three different aspects have been physically modularized by partitioning the triples content, and each of these partitions is in turn divided into smaller segments to further account for specific data organization requirements such as provenance and access privileges. The partitions are obtained through the use of RDF named graphs, so that, whenever appropriate, the knowledge server may benefit of a single shared data space, or is able conversely to manage each partition (or set of partitions) as a separate dataset. The

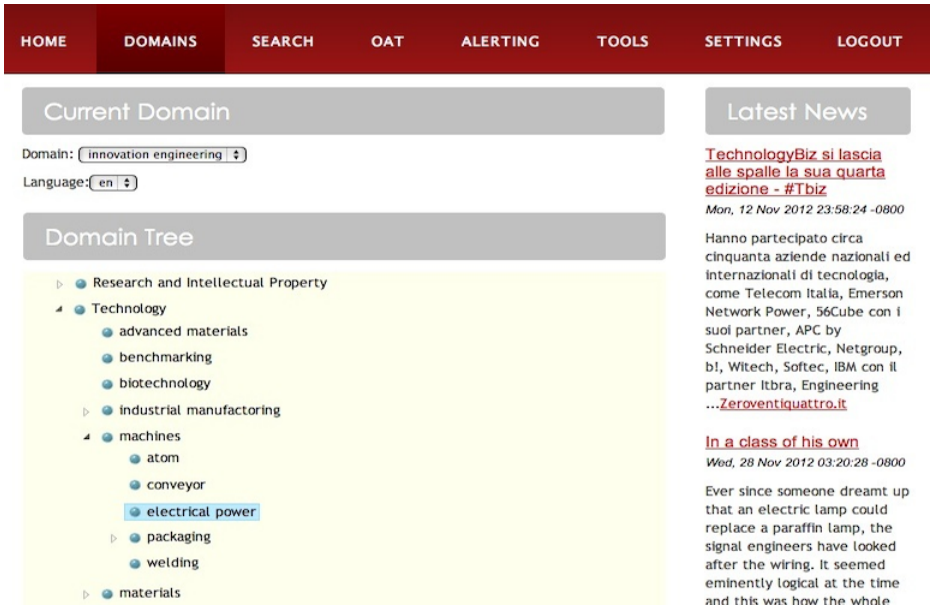


Fig. 1. SKOS topics and bookmarks in the *innovation* domain.

two main categories of users access these partitions in INSEARCH: companies and employees. Companies act like user-groups, collecting standard users (employees) under a common hat and possibly providing shared information spaces (e.g. domain models or reference information) which will be inherited by all of them. Each employee shares with his colleagues common data provided by the company, while at the same time he can be offered a personalized opportunity or a restricted access.

Semantic Bookmarking. In such a scenario, it is crucial to populate the SKOS ontology, thus providing examples for the document categorization process, allowing to link novel documents to existing (or user-defined) SKOS concepts. Semantic Turkey (ST) [14] was born as a tool for semantic bookmarking and annotation, thought for supporting people doing extensive searches on the web, and needing to keep track of: results found, queries performed and so on. Today ST is a fully fledged Semantic Platform for Knowledge Management and Acquisition supporting all of W3C standards for Knowledge Representation (i.e. RDF/RDFS/OWL SKOS and SKOS-XL extension). It is possible to extend it, in order to produce completely new applications based on the underlying knowledge services. The underlying framework allows access to RDF (and all modeling vocabularies already mentioned) through Java API, client/server AJAX communication (proprietary format, no Web service) and client-side Javascript API (hiding TCP/HTTP details). The ST offers among the others functionalities for editing a reference (domain) ontology (i.e. a SKOS-compliant topic taxonomy), bookmarking pages according to the taxonomy as well as organizing query re-

sults according to the hierarchical structure the SKOS taxonomy. Users may surf the web with a standards compliant web browser, associating information found on web documents to concepts from the current knowledge organization systems (KOS). The core framework of ST has been totally reused in INSEARCH without specific customization. However, novel dedicated services have been developed and plugged, flanking the main ones, to meet the specific INSEARCH requirements (see also the discussion in next section on architecture). In particular, the annotation mechanism is merged into the multiuser environment of the INSEARCH platform, so that the system may exploit contributions from different users, whenever the power of mass-contribution is exploitable.

2.2 Robust Modeling of Lexical Information

Computational models of natural language semantics have been traditionally based on symbolic logic representations naturally accounting for the meaning of sentences, through the notion of compositionality (as the Montague's approach in [12] or [3]). While formally well defined, logic-based approaches have limitations in the treatment of ambiguity, vagueness and other cognitive aspects such as uncertainty, intrinsically connected to natural language communication. These problems inspired recently research on **distributional models of lexical semantics** (e.g. Firth [8] or Schütze [15]). In line with Wittgenstein's later philosophy, these latter characterize lexical meanings in terms of their context of use [17]. Distributional models, as recently surveyed in [16], rely on the notion of Word Space, inspired by Information Retrieval, and manage semantic uncertainty through mathematical notion grounded in probability theory and linear algebra. Points in normed vector space represent semantic concepts, such as words or topics, and can be learned from corpora, in such a way that similar, or related, concepts are near to one another in the space. Methods for constructing representations for phrases or sentences through vector composition have recently received a wide attention in literature (e.g. [11]). While, vector-based models typically represent isolated words and ignore grammatical structure [16], the so-called **compositional distributional semantics** (DCS) has been recently introduced and still object of rich on-going research (e.g. [11, 5], [9], [2]). Notice that several applications, such as the one targeted by INSEARCH, are tight to structured concepts, that are more complex than simple words. An example are the TRIZ inspired Object-Action-Tool (OAT) triples that describe *Object*(s) that receive(s) an *Action* from *Tool*(s), such as those written in sentences like "... $[\textit{the coffee}]_{Object}$ *in small quantities* $[\textit{is prepared}]_{Action}$ *by the* $[\textit{packing machine itself}]_{Tool}$ ..." or "... *for* $[\textit{preparing}]_{Action}$ $[\textit{the coffee}]_{Object}$ *by extraction with* $[\textit{hot water}]_{Tool}$, ...".

Here physical entities (such as *coffee* or *hot water*) play the role of *Object*s or *Tool*s according to the textual contexts they are mentioned in. Compositional models based on distributional analysis provide lexical semantic information that is consistent both with the meaning assignment typical of human subjects to words and to their sentential or phrasal contexts. It should support synonymy and similarity judgments on phrases, rather than only on single words. The

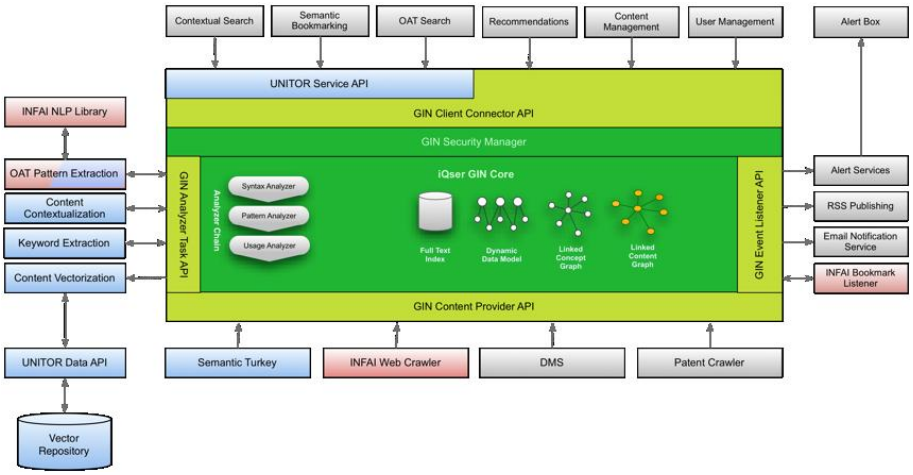


Fig. 2. An high level view of the INSEARCH functionalities and services.

objective should be assigning high values of similarity to expressions, such as "... *buy a car* ..." vs. "... *purchase an automobile* ...", while lower values to overlapping expressions such as "... *buy a car* ..." vs. "... *buying time* ...". Distributional compositional semantics methods provide models to define: (1) ways to represent lexical vectors $\boldsymbol{v}$ and $\boldsymbol{o}$, for words v, o occurring in a phrase (r, v, o) (where r is a syntactic relation, such as verb-direct_object), and (2) metrics for comparing different phrases according to the basic representations, i.e. the vectors $\boldsymbol{v}$, $\boldsymbol{o}$.

While a large literature already exist (e.g. [11]) the user can find more details about the solution adopted in INSEARCH in [2]. Compositional distributional semantic models are used to guide the user modeling of ontological concepts of interest (such as the SKOS topics), feed the document categorization process (that is sensitive to OAT patterns through vector based representation of their composition), concept spotting in text as well as query completion in INSEARCH. The adopted methods are discussed in [2] and [6].

3 The INSEARCH architecture

The INSEARCH overall architecture is designed as a set of interacting services whose overall logic is integrated within the iQser GIN Server for information ecosystems. The comprehensive logical view of the system is depicted in Fig. 2.

The core GIN services are in the main central box. External Analyzers are shown on the left, as they are responsible for text and language processing or, as in the case of the Content vectorization module, for the semantic enrichment of input documents. GIN specific APIs are responsible for interfacing heterogenous content providers and managing other specific data gathering processes (e.g. specific crawlers). Client Connector APIs are made available by GIN for a variety

of user level functionalities, such as User Management, Semantic Bookmarking or Contextual searches that are managed via appropriate GIN interface(s). At the client level in fact, the basic search features from web sources and patents, are extended with:

- Navigation in linked search results and Recommendations for uploaded or pre-defined contents through bookmarks or SKOS topics of interest. Recommendations are strongly driven by the semantically linked content, established by the core analysis features of the GIN server.
- Semantic bookmarking is supported allowing sophisticated content management, including the upload of documents, the triggering of web crawling stages, the definition and lexicalization of interests, topics and concepts described in SKOS. Interesting information items are used for upgrading recommendations, topics and concepts and prepare contextual searches.
- Personalization allows user management functions at the granularity of companies as well as people.

On the backend side, we emphasize that the current server supports the integration with Alfresco[3] as the document and content management system, whereas the defined interests are also managed as Alfresco's content. While the integration of Web sources is already supported by a dedicated crawler, also patents are targeted with an interface to the patent content provider WIPO[4].

Contextual Semantic search is also supported through vector space methods. Vectorization is applied to incoming documents with an expansion of traditional bag-of-word models based on topic models and Latent Semantic Analysis (as discussed in Section 2.2). Moreover, the available vector semantics supports distributional compositional functions that model the representation and inferences regarding TRIZ-like OAT patterns, so that natural language processing and querying based on domain specific patterns are consistently realized. Basic feature extraction services and morphosyntactic analyzers (such as lemmatization and part of speech tagging) are already in place as external GIN analyzers.

The main functionalities currently integrated in INSEARCH are thus:

- **Website monitoring**: Observe changes in given pages/domains, which are added by the user and implemented as bookmarklets
- **Assisted Search**: such as in Query completion, e.g. support the user in the designing proper queries about company's products or markets .
- **Document analysis**: Intelligent Document Analysis is applied to asses their relevance to high-level topics predefined by the user in the SKOS taxonomy. Relevance to individual topics is provided through automatic classification driven by weighted membership scores of results with respect to individual topics.
- **Patent and scientific paper search**: Search for patents and/or scientific papers in existing databases (e.g. European patent office) is supported.
- **OAT-Pattern analysis**: TRIZ-inspired Object-Action-Tool (OAT) triples are searched in documents: these patterns play the role of suggestions for *tools*, which provide a certain function specified by the *object* and the *action*.

[3] `http://www.alfresco.com/`

[4] `http://www.wipo.int/portal/index.html.en`

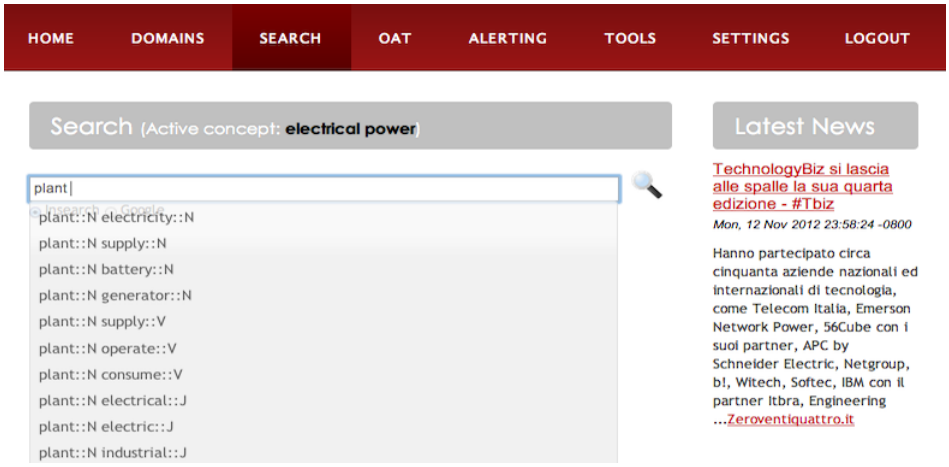


Fig. 3. The INSEARCH front-end and the completion of the Query *plant* when the SKOS concept `electrical power` is selected.

– **Adaptivity**: The system tracks user behaviors and adjusts incrementally its own relevance judgments for the topics and categories of interest.

3.1 Typical user interactions

The system has been recently deployed in its full functional version and provides a unique opportunity to evaluate its application to realistic data sets and industrial processes. The INSEARCH users will be able to quantitatively and qualitatively evaluate the impact of its semantic capabilities, its collaborative features as well as the overall usability of the personalized search environment in a systematic manner.

The front end of the INSEARCH system is shown in an interactive contextual search use-case in Fig. 3 and 4. The main tabs made available here are related to the Domains, Search, Alerting and Tools functionalities. In Domains the user can interact with and refine his own SKOS topics as well as interests and preferences, as shown in Fig. 1. Alerting supports the visualization of the results of Web Monitoring activities: here returned URLs, documents or other texts are conceptually organized around the SKOS concepts thanks to the automatic classification targeted to the ontology categories, made available through the Rocchio Classifiers, as discussed in [13]. In Tools most of the installation and configuration activities can be carried out.

In the Search tab, contextual search and query completion is offered to the user. In Fig. 3 the suggestions related to the ambiguous keyword "*plant*" early provided by the user are shown, where nouns like "*generator*" and "*battery*" (as well verbs like "*generator*" and "*battery*") are the proper continuation of the query, given the underlying domain, i.e `electrical power`. The completion is different when a topic such as `biotecnology` is selected, as shown in Fig. 4.

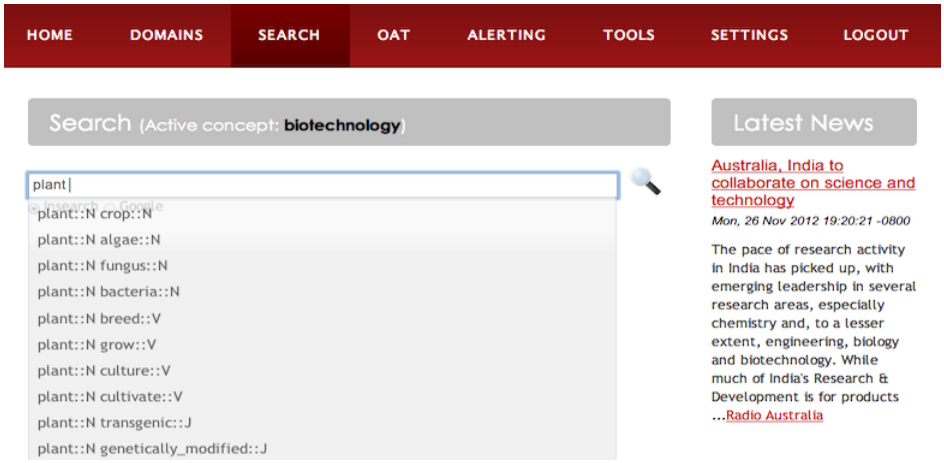


Fig. 4. The INSEARCH front-end and the completion of the Query *plant* when the SKOS concept `biotecnology` is selected.

The different completion is made available by the lexicalization of each concept: these lexical preferences are projected in an underlying Word Space (discussed in Section 2.2) that provides the geometrical representation of all words appearing in the indexed documents. Given the vectors representing all query terms and the lexical preferences of the selected SKOS concepts, the most similar (i.e. nearest) words are selected and proposed for the completion. This adaptivity is achieved also to provide novel information to the final users. In the front-end interface, a list of news is proposed. These are continually downloaded from the web and retrieved using the lexical preferences specified by the user during his own registration as well as the selected SKOS concepts. Notice that news are sensitive to the different SKOS concepts during the session, as in Fig. 3 and 4.

Once the query is submitted, documents are retrieved, automatically classified and clustered with respect to the existing SKOS concepts, as in Fig. 5. This clustering phase allows users to browse documents exploring their relatedness to specific SKOS concepts, such as `electrical power` or `research`. The user interface also allows to implement a *relevance feedback* strategy to improve the quality and adaptivity of text classifiers by simply clicking over the "thumbs up" or "thumbs down" icons. They allow to accept or reject each concept/document association, that reflects the underlying text classification. When the user accepts a classification, the Rocchio classifier associated with the corresponding concept is incrementally fed with the document, that becomes a positive example. On the contrary, the selected document is provided as a negative example, by clicking on the "thumbs down" icon.

Finally, the Object-Action-Tool (OAT) pattern-based search is shown in Fig. 6. The user is allowed to retrieve documents specifying specific actions (*pack*), objects (*coffee boxes*) or tools (*dedicated machine*). During the data-gathering

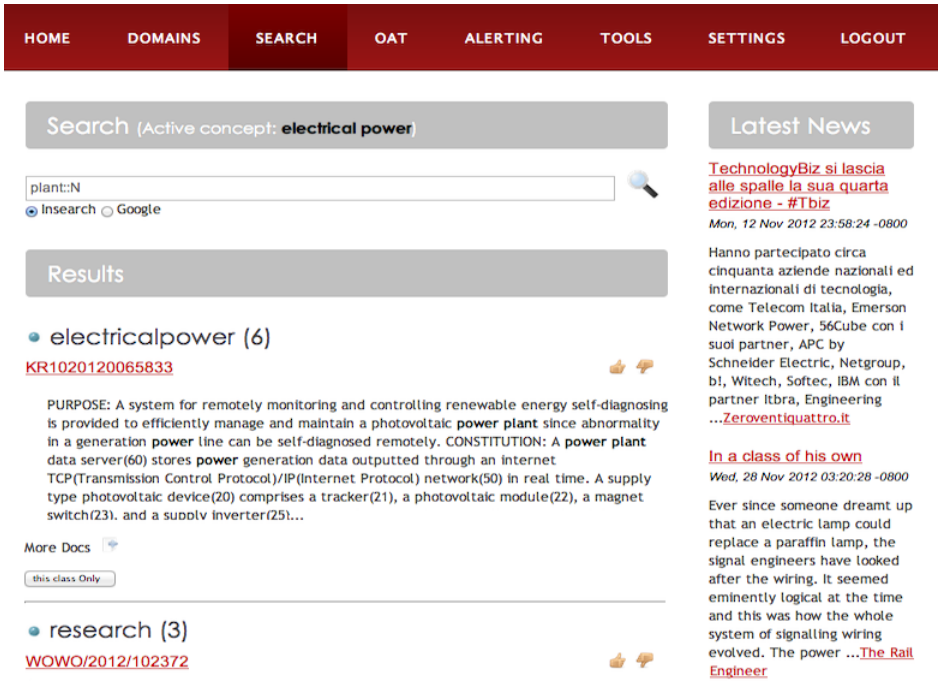


Fig. 5. The INSEARCH front-end the presentation schema of retrieved documents for the query *plant* when the SKOS concept `electrical power` is selected. Here 6 and 3 documents are related to the `electrical power` and `research` concept, respectively. The "Thumbs up"/"Thumbs down" icons allow to implement a relevance feedback strategy.

phase, the *OAT pattern extraction* module (see Fig. 2) extracts all patterns from the documents, by exploiting a set of pre-defined morphosyntactic patterns, such as Subject-Verb-Object. The extracted OAT patterns are used during the indexing phase, thus enabling semi-structured queries through (possible incomplete) OAT patterns. Fig. 6 summarizes a session where the user is interested in documents related to the action *control* and object *nuclear fission*. Initially the system suggests a set of possible tools, such as *method*, *system* or *product*. The user can select one or more tools to browse the related documents.

4 Conclusions

In the innovation process, the search of external information represents a crucial activity for the most of Small and Medium Sized Enterprises. In this paper the system targeted in the INSEARCH EU project is discussed. It embodies most of the state-of-the-art techniques for Enterprise Semantic Search: highly accurate lexical semantics, semantic web tools, collaborative knowledge management and personalization. The outcome is an advanced integration of analytical natural

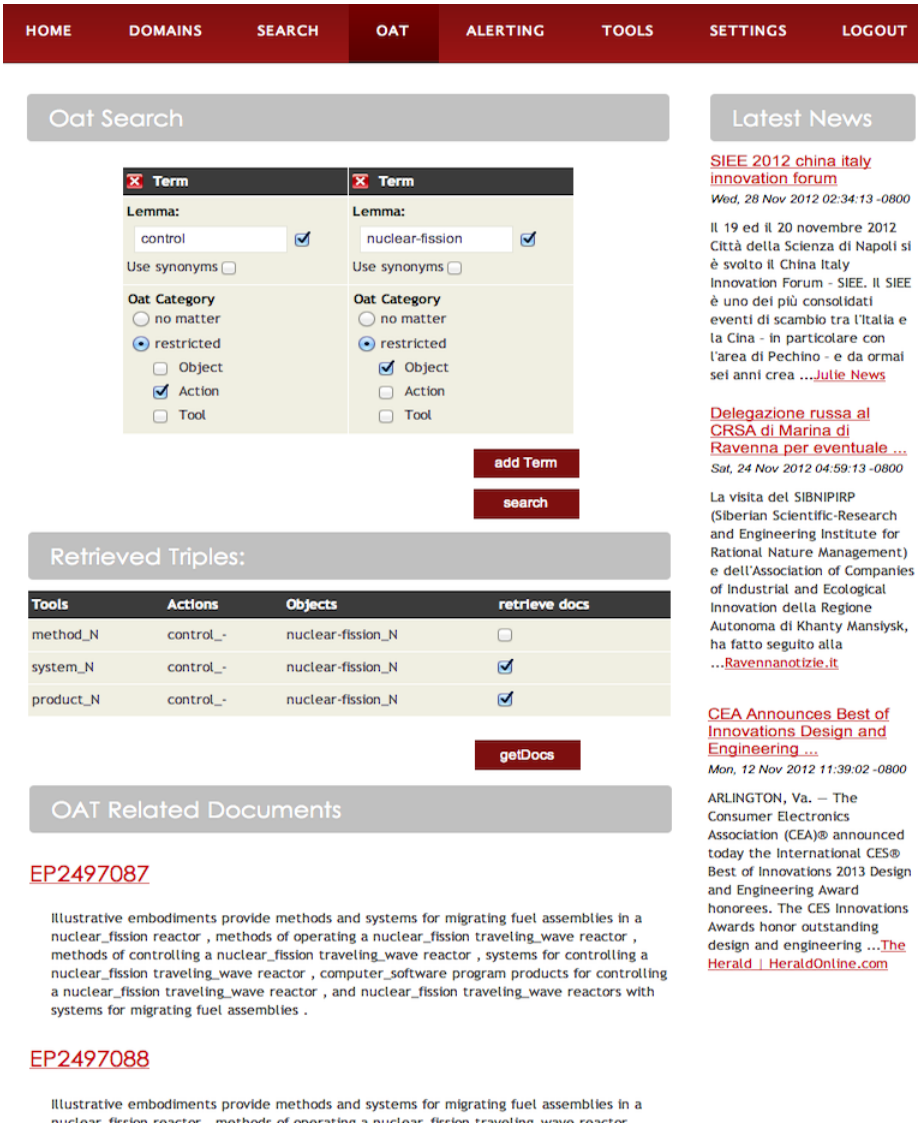


Fig. 6. The INSEARCH front-end for the Object-Action-Tool (OAT) triple-based search schema

language analysis tools, robust adaptive methods and semantic document management systems relying over the Semantic Web standards. The knowledge bases personalization as well as the semantic nature of the recommending functionalities (e.g. query completion, contextual search and Object-Action-Tool triple-based search) will be evaluated in systematic benchmarking activities, carried at the enterprise premises, within realistic and representative scenarios.

Acknowledgment The authors would like to thank all the partners of the IN-SEARCH consortium as they made this research possible. In particular, we thank Armando Stellato and Daniele Previtali from UNITOR, Jorg Wurzer from iQSer, Paolo Salvatore from CiaoTech, Sebastian Dunninger, Stefan Huber from Kusftein, Antje Schlaf from INFAI, Mirko Clavaresi from Innovation Engineering, Cesare Rapparini from ICA and Hank Koops from Compano.

References

1. Altshuller, G.: 40 principles, TRIZ keys to technical innovation. No. 1 in Triz tools, Technical Innovation Center, Worcester, Mass., 1. ed edn. (1998)
2. Annesi, P., Storch, V., Basili, R.: Space projections as distributional models for semantic composition. In: Gelbukh, A.F. (ed.) CICLing (1). LNCS, vol. 7181, pp. 323–335. Springer (2012)
3. B. Coecke, M.S., Clark, S.: Mathematical foundations for a compositional distributed model of meaning. Lambek Festschirft, Linguistic Analysis 36 (2010)
4. Baeza-Yates, R., Ciaramita, M., Mika, P., Zaragoza, H.: Towards semantic search. Natural Language and Information Systems pp. 4–11 (2008)
5. Baroni, M., Zamparelli, R.: Nouns are vectors, adjectives are matrices: representing adjective-noun constructions in semantic space. In: Proceedings of EMNLP 2010. pp. 1183–1193. Stroudsburg, PA, USA (2010)
6. Basili, R., Giannone, C., De Cao, D.: Learning domain-specific framenets from texts. In: Proceedings of the ECAI Workshop on Ontology Learning and Population. ECAI, ECAI, Patras, Greece (July 2008)
7. Cocchi, L., Bohm, K.: Deliverable 2.2: Analysis of functional and market information. TECH-IT-EASY (2009)
8. Firth, J.: A synopsis of linguistic theory 1930-1955. In: Studies in Linguistic Analysis. Philological Society, Oxford (1957), reprinted in Palmer, F. (ed. 1968) Selected Papers of J. R. Firth, Longman, Harlow.
9. Grefenstette, E., Sadrzadeh, M.: Experimental support for a categorical compositional distributional model of meaning. CoRR abs/1106.4058 (2011)
10. Klyne, G., Carroll, J.J.: Resource Description Framework (RDF): Concepts and Abstract Syntax (2004)
11. Mitchell, J., Lapata, M.: Vector-based models of semantic composition. In: In Proceedings of ACL-08: HLT. pp. 236–244 (2008)
12. Montague, R.: Formal Philosophy: Selected Papers of Richard Montague. Yale University Press (1974)
13. Moschitti, A., Basili, R.: Complex linguistic features for text classification: a comprehensive study. In: Proc. of the ECIR. pp. 181–196. Springer Verlag (2004)
14. Pazienza, M.T., Scarpato, N., Stellato, A., Turbati, A.: Semantic turkey: A browser-integrated environment for knowledge acquisition and management. Semantic Web journal 3(2) (2012)
15. Schütze, H.: Automatic Word Sense Discrimination. Computational Linguistics 24, 97–124 (1998)
16. Turney, P.D., Pantel, P.: From frequency to meaning: Vector space models of semantics. Journal of artificial intelligence research 37, 141 (2010)
17. Wittgenstein, L.: Philosophical Investigations. Blackwells, Oxford (1953)
18. World Wide Web Consortium: SKOS Simple Knowledge Organization System Reference (Aug 2009)

Wikipedia based Unsupervised Query Classification

Milen Kouylekov, Luca Dini, Alessio Bosca, and Marco Trevisan

CELI S.R.L., Torino, Italy
{kouylekov,dini,bosca,trevisan}@celi.it

Abstract. In this paper we present an unsupervised approach to Query Classification. The approach exploits the Wikipedia encyclopedia as a corpus and the statistical distribution of terms, from both the category labels and the query, in order to select an appropriate category. We have created a classifier that works with 55 categories extracted from the search section of the Bridgeman Art Library website. We have also evaluated our approach using the labeled data of the KDD-Cup 2005 Knowledge Discovery and Data Mining competition (800,000 real user queries into 67 target categories) and obtained promising results.

Keywords: Query Classification, Wikipedia, Vector Models

1 Introduction

In Information Science logs analysis and more specifically Query Classification (QC) has been used to help detecting users web search intent. Query classification studies have shown the difficulty of achieving accurate classification due to the inevitably short queries. A common practice is to enrich query with external information, and to use an intermediate taxonomy to bridge the enriched query and the target categories. A good summary of such approaches is made by the KDDCUP 2005 organizers [2005]). Associating external information to queries is costly as it involves crowing the web. The goal of our approach is to create an lightweight unsupervised language independent approach to query classification using the rich content provided by the Wikipedia, a resource easily accessible in many languages.

In Section 2 we present this approach. In Section 3 we provide a evaluation of its capabilities.

2 Unsupervised Query Classification Approach

Our approach is based on the vector space model (VSM). The core of the VSM is representing text documents as vectors of identifiers, such as index terms. Each element of the vectors corresponds to a separate term found in the document. If a term occurs in the document, its value in the vector is non-zero. The dimensionality of the vector is the number of words in the vocabulary (the number

of distinct words occurring in the corpus). In VSM, weights associated with the terms are calculated based on a heuristic functions. Some of the more popular approaches are term frequency and inverse document frequency.

Using the VSM a similarity function between the vectors of two documents and a query can be defined. The standard function used is the cosine similarity coefficient, which measures the angle between two vectors.

We have adapted the vector space model to the query classification task using the following approach: First, we associate a document that describes the category with each category C_k. We name this document category document CD. For example a CD for the category Basketball must contain information about: the rules of the game, *National Basketball Association*, *FIBA* and famous players etc. A CD for the Arts must contain information about i) paining; ii) sculptures; 3) art museums etc. In the second stage of the approach we associate to each query a set of relevant documents found in a document collection. We define the category score of query q for category C_k as the maximum value of the cosine similarity between term vectors of the CD of the category and a document relevant to the query ${}_Qi$. For example we expect to find a lot of common terms between a document relevant to the query *Michael Jordan* and the CD for category Basketball and few common terms with the CD of the category Art. Finally as output the approach returns the categories which CD has the highest cosine similarity with a relevant document of the query.

We use relevant documents for a query and not the query because we compare the query terms with CD documents that are not relevant to the query itself. For example the CD of the category Arts does not contain a mentioning of Pablo Picasso but the intersection between it and documents relevant to Pablo Picasso contain a lot of common terms like painting, art, surrealism etc.

In order to make our approach feasible we need a document collection that contains sufficient number of documents in order to: 1) Find a big enough CD document for each category. 2) Find documents relevant to the classified queries.

The advantage of the proposed approach is that it does not require training data and it is language dependent. The approach will befit greatly from a short category description as this will allow a more correct selection of a CD.

3 Experiments

In our experiments we used the Wikipedia, a free, web-based, collaborative, multilingual encyclopedia[1]. We assign as CD for the category with the name X the Wikipedia page with the same title. For example the Wikipedia Page with title *Basketball* can be used as a CD for the category *Basketball*. The page describes almost all the important aspect of the game and has a lot of terms in common with documents relevant to queries like: *Michael Jordan*, *NBA Playoffs* and *Chicago Bulls*. Respectively the page with the title *Hardware* contains a lot of terms in common with documents relevant for queries like: *intel processors*, *computer screens* and *nvida vs intel*.

[1] http://wwww.wikipedia.org

For some categories the CD assigned by the approach contains short texts that are not sufficient for a complete overview of the category. We expand the CD for these categories by concatenating the texts of the pages that contain the name of the category as sub part of the page title. For example the CD for the category *Arts* can be expanded by concatenating the text of the pages: *Liberal arts*, *Visual arts*, *Arts College*, *Art Education*, *Islamic Arts* etc. If the approach does not find a page with the same title as the category it assigns as CD for the category the concatenation of pages with the name of the category as sub part of the page title or pages that contain the category name in the first sentence of the page text.

3.1 Bridgeman Art Library

Our first evaluation is done using the taxonomy and query dataset created for the ART domain in the Galateas Project [2011]. The domain is defined by the contents of the Bridgeman Art Library(BAL) website[2]. To understand their use and meaning the categories have been grouped by BAL domain experts into three groups: Topics (Land and Sea, Places, Religion and Belief, Ancient and World Cultures etc. 23), Materials (Metalwork, Silver, Gold & Silver Gilt, Lacquer & Japanning, Enamels etc. 10), and Objects (Crafts and Design, Manuscripts, Maps, Ephemera, Posters, Magazines, Choir Books etc. 22) (total: 55 top-categories).

Our approach was evaluated on the 100 queries annotated by the three annotators into upto 3 categories. The queries were in four languages English , French, German, Dutch and Italian. We have created a classification instance for each language by manually translating the name of the categories in each language. For each of these queries we automatically assign the top 3 categories returned by the classifier. Example:

Query: navajo turquoise
Category1: Semi-precious Stones Score: 0.228
Category2: Silver, Gold & Silver Gilt: 0.1554
Category3: Botanical Score: 0.1554

In this example the category assigned to the query is Category1 *Semi-precious Stones*. The results of the evaluation are summarized in Table 1.

	Precison	F-Measure
Bridgeman Art Library	16.1	14.5
KDD Cup Results	29.0	32.2

Table 1. Results

[2] http://www.bridgemanart.com/

3.2 KDD Cup 2005

To evaluate our approach we have experimented also with the KDD-Cup 2005 data [2005]. The data is from query classification task selected by the organizers as interesting to participants from both academia and industry. The task of the competition consists in classifying Internet user search queries. The participants had to categorize 800,000 queries into 67 predefined categories. The meaning and intention of search queries is subjective. A search query *Saturn* might mean *Saturn car* to some people and *Saturn the planet* to others. The participants had to tag each query with up to 5 categories. The systems participating in the competition were ranked by the organizers on the obtained average **F-Measure**.

We have evaluated our approach against a gold standard provided by the organizers (800 queries). Our approach obtained an average F-Measure of 32.2 (Table 1). The state of the art system [2006] in the competition achieves F-Measure of 46.1.

4 Discussion

The results obtained are encouraging having in prospective the unsupervised nature of the approach. One of the main difficulties were the generic names of categories like *Icons* and *The Arts and Entertainment*. The documents for these categories contained terms that were not relevant to the art domain. Also a significant problem for the system posed queries that are named entities. These queries were classified based on their descriptions into categories relevant to their peculiarities and not in *People and Society*, *Personalities* and *Places* categories. A possible solution to this problem will be to map DBPedia[3] hierarchy to the domain categories and use it as a additional source of knowledge.

The results we obtained is encouraging particularly because we did not associated additional information to each queries apart of the documents obtained using Wikipedia. Many of the queries did not produce relevant Wikipedia documents, which is one of the main limitations of our approach. Additional domain corpora will decrease its effect.

References

[2005] Ying Li, Zijian Zheng, and Honghua (Kathy) Dai. Kdd cup-2005 report: facing a great challenge. ACM SIGKDD Explorations Newsletter Homepage archive, 7(2), 2005.

[2006] Dou Shen, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Building bridges for web query classification. In SIGIR06, 2006.

[2011] Eduard Barbu, Raphaella Bernardi, T.D. Le, Milen Kouylekov, V. Petras, Massimo Poesio, Juliane. Stiller, E. Vald, D7.1 First Evaluation Report of Topic Computation and TLIKE , Galateas Project http://www.galateas.eu

[3] http://dbpedia.org/