Linked-Data Based Suggestion of Relevant Topics

Milan Stankovic[1,2], Werner Breitfuss[1]
[1]Hypios Research
Paris, France
milan.stankovic@hypios.com,
werner.breitfuss@hypios.com

Philippe Laublet
STIH
[2]Université Paris-Sorbonne
Paris, France
philippe.laublet@paris-sorbonne.fr

ABSTRACT
In this paper we propose an alternative method for generating topic suggestions for the needs of expert finding in Open Innovation. An important requirement of Open Innovation scenarios is to be able to identify topics lateral to a given innovation problem, and use them to broaden the broadcast of the problem without compromising on relevancy. We propose an approach based on DBpedia – a Linked Data version of Wikipedia – which enables us to recommend topics facilitating their proximity in the DBpedia concept graph. Relying on this source we can also filter out certain types of concepts irrelevant to industrial problem solving. We evaluate our approach against the adWords keyword suggestion system here we also show the ability of our system to predict lateral topics that appeared in the actual solutions submitted to past problem challenges. Secondly we evaluate user satisfaction with the proposed keywords from both systems, in terms of relevancy and unexpectedness. Finally we show the significant impact of the use of suggested lateral keywords to the raised awareness about the problem in a real Open Innovation problem broadcast.

Categories and Subject Descriptors
I.2.4.6 Semantic networks

General Terms
Algorithm, Experimentation;

Keywords
Semantic Web, DBpedia, topic discovery, keyword recommendation

1. INTRODUCTION
Open Innovation (OI) platforms, such as hypios.com and innocentive.com have emerged to accelerate the innovation process in big companies by offering R&D problems for public solving. On such websites, innovation problems are posted in a form of calls for solutions, that offer rewards for the best, most innovative submissions.

The main promise of such an approach is the diversity in solutions and perspectives to an R&D problem that should emerge as a result of the ability of everyone to post solutions. Problems are thus broadcasted to potential solvers over e-mail and social Web channels.

Solution seekers usually expect serendipity, hoping to have a solver from an unexpected domain who would bring knowledge from his own domain of expertise to the problem in question. Solutions coming from unexpected domains, lateral to the domain of the problem, are shown to be more innovative then those coming from the problem domain [1].

It is thus essential to broadcast the OI problems to the people in diverse but still relevant fields. In such a scenario, discovering lateral topics that are not directly detectable from the problem description, but that are still relevant is crucial for the ability to attract solutions from unexpected domains, and achieve solution transfer from one domain to another.

Although many approaches have been developed for suggesting relevant concepts, most of them target the need for relevancy, and rarely treat the need for unexpectedness and discovery in suggestions. Motivated by recent results showing the potential of Linked Data [2] sources to enhance serendipitous discoveries [3] in the musical domain we have designed an approach for discovering relevant topics using DBpedia.org – a Linked Data version of Wikipedia.

We propose a solution for discovering relevant topics from which an innovative solution can come, based on DBpedia. Our solution has been integrated into the hypios OI process, where the resulting topic suggestions serve as fields for potential solver finding and for problem broadcasting. We provide a threefold evaluation: (1) evaluation on the past innovation problems and solutions; (2) user satisfaction with the proposed keywords, and (3) evaluation of the impact of keyword suggestion on the performance of problem broadcast.

In Section II we present state of the art, and in Section III we give motivation for our approach. Section IV presents our approach for discovering relevant topics, and the Section V presents the different evaluations we have performed. We conclude with perspectives in Section VI.

2. State of the Art
The problem of suggesting relevant concepts has been studied for decades, in various contexts and using different approaches. Suggesting of concepts has been used to help users choose the right tags in collaborative filtering systems [4], to discover alternative search queries [5], to help refine queries [6], to enhance expert finding results [7], and in many other scenarios.

Different techniques and different sources have been used to obtain the topic suggestions. A major branch of approaches relies on co-occurrence of concepts.

I-SEMANTICS 2011, 7th Int. Conf. on Semantic Systems, Sept. 7-9, 2011, Graz, Austria

The approach presented in [8] uses co-occurrence in text of research papers, pondered with a function derived from the tf-idf measure [9] to establish a notion of word proximity.

Approaches like [10] rely on co-occurrence of words in the text of Wikipedia pages, combined with the information about the categories of pages in Wikipedia to deliver a notion of semantic relatedness. Co-occurrence in tags [4] and in search results [11] are also commonly used.

The approach presented in [12] goes a step further and combines co-occurrence in Web search results with the extraction of lexico-sytactic patterns from search results snippets to detect patterns that indicate synonymy. Using the notion of similarity based on co-occurrence in search results to refine the search query and increase the relevancy of search results is known as relevance feedback; however, using it in expert finding has not given good results due to heterogeneity of expertise in user profiles [7].

Another significant branch of approaches uses graph structures of concepts to determine semantic proximity of concepts and use this as a base for suggestions in various scenarios. Shortest path is among the most common measures. It is often enhanced by taking into account the informational content of the nodes in the graph [13]. [14] exploits the hypernym graphs of Wordnet[1]; [6] uses Gallois lattice to provide recommendations based on domain ontologies, [15] and [16] use the ODP taxonomy[2]. [17] relies on the graph of Wikipedia categories.

Although most of those approaches give satisfactory results for finding relevant terms to suggest, to our knowledge there has been no effort to show how they perform in supporting the user in discovering unknown relevant terms, and exploring paths unexplored before – a feature of essential importance in our OI scenarios evoked earlier.

3. Motivation

The main motivations behind our DBpedia-based approach are:

- To take advantage of the thesaurus of more then 3.5 million concepts[3], most of them multilingual, connected to a graph the graph of 633,000 categories.
- The potential of Wikipedia categories to provide relevant categories being already demonstrated [17], we whish to explore if the graph of categories performs well in discovering non-obvious, potentially unknown to user, and still relevant topic suggestions.
- Use the semantic information about topic types to distinguish domains of knowledge from other topics that are of no interest to industrial problem solving.
- Explore the potential of DBpedia based recommendations to find potentially interested experts in the domains lateral to the given OI problem.

4. Discovering Relevant Topics

4.1 Knowledge structure in DBpedia

The movement of Linked Data has emerged [18] as an initiative to make the Web content structured by publishing interlinked structured data, for which the structure and meaning is defined by ontology. One of the crucial parts of the Linked Data Web is DBpedia, a Linked Data version of Wikipedia, in which Wikipedia pages are represented as concepts, most of them having explicit types defined in DBpedia ontology[4] and interlinked among each other. In some cases where explicit types are not present, it is possible to infer them by using the concept's properties, e.g., an instance having geographic coordinates is clearly a physical place.

The knowledge of categories of Wikipedia pages is captured using the SKOS ontology[5] and heavily relies on skos:broader[6] property to denote broader category of a given category, and dcterms:subject to denote belonging of a concept to a category. An example graph showing the structure of broader relations can be seen on the Figure 1. Our approach relies on exploring the paths in the graph of categories to find relevant suggestions as well as on using explicit and implicit types of concepts to filter out the unwanted concept types.

4.2 Challenges Faced

The first challenge we have faced is related to the context in which one concept is closely related to another. Although DBpedia provides a nice way to explore connections between concepts through a graph of concept categories, this graph is quite general and agnostic of the user's query. Let us consider a real example coming from our practice.

We had a problem on hypios, related to the concept DBpedia:Kaolinite. By applying simple graph distance measures such as the shortest path, we would find concepts DBpedia:Raw_material and DBpedia:Traditional_medicine at the same distance. Since Kaolinite stands for clay, it is indeed close to both the concept of extraction of clay as raw material and to its use in traditional medicine. For our problem, however, DBpedia:Traditional_medicine was not relevant as the problem dealt with issues in material extraction process. Isolating the context of relevancy is thus a major challenge for our system.

The second challenge deals with the fact that DBpedia contains concepts of different kinds, and that not all of them represent domains of knowledge this can help in industrial problem-solving. For our case, recommendations involving persons, cities and geographic places, buildings, historical events, and books, are considered irrelevant, as they do not represent a problem-solving topic. Establishing a meaningful filter for the recommendations, based on their type, is therefore a necessity.

4.3 Our Approach

Our approach combines the data about categories of concepts in DBpedia, with the information about the types of concepts to deliver meaningful concept suggestions. It consists of (1) searching for relevant concepts in the DBpedia graph, and (2) filtering the found concepts based on their type.

In order to resolve the challenge of identifying the right context of relevancy, we use several concepts as a starting point for our concept expansion. These are called seed concepts. The core of

[1] http://wordnet.princeton.edu/

[2] http://www.dmoz.org/

[3] As a comparison the current version of Wordnet has 206941 word-sense pairs, and ODP has half a million categories

[4] http://wiki.DBpedia.org/Ontology

[5] `http://www.w3.org/2008/05/skos`

[6] The prefixes namespace prefixes used in this paper can be resolved at http://prefix.cc

our technique is the exploration of the graph of categories of topics, created by skos:broader relations, with two main characteristics:

1. Concepts that are found on a shorter path distance from the starting concepts are considered to be closer and more relevant;
2. Concepts found by exploration of the graph proximity of several initial concepts are more relevant than those appearing in the proximity of just one starting concept.

The proximity of concepts in our sense, called hyProximity is defined by the formula (1).

$$hyP(c,IC) = \sum_{c_i \in IC} \frac{p(c,c_i)}{d(c,c_i)} \quad .1$$

hyProximity of a concept c to the set of initial concepts IC is the sum of pondered inverted distances between the concept c and each concept c_i from the set of initial seed concepts IC. Various distance functions d and pondering functions p can be used. Our formula is in fact a generalization of proximity measures used in shortest path-based approaches. In our implementation, the distance between the concept c and an initial concept c_i , denoted $d(c,c_i)$, is calculated as the shortest path from c_i to the first common ancestor that it shares with c in the graph formed by skos:broader relations. The ponders $p(c,c_i)$ are used to give different importance to different distances, basically a weight function.

Functions that decrease the importance of siblings at higher distances linearly or exponentially are the most logical choice. We have primarily used $e^{-\lambda d(c,ci)}$ as a pondering function with the coefficient λ equal to 0.3 with the objective to exponentially decrease the importance of topics at higher distance. Measures non-dependent on distance can also be used for pondering. For instance, using informational content of the closest common category would be an approach similar to the one used by Resnik [13] for his graph-based similarity measures.

Once the hyProximity values for concepts in DBpedia graph have been calculated, the ranked list of concepts is finally filtered to exclude the topics which are not relevant for the industrial problem solving, such as concepts denoting people, geographical features, companies, lists of topics, and books. The filtering is performed based on the implicit and explicit concept types in DBpedia. For efficiency purposes an index of non-desirable concepts is pre-created to avoid performing SPARQL queries for each concepts each time.

4.3.1 Calculation of the Candidate Suggestions

The calculation of hyProximity is done using the following algorithm:

Algorithm 1

1. get initial topic concepts IC
2. for each seed concept c in IC:
  a. while level++ < maxLevel:
    i. for each c_i, sharing a common ancestor with c at the current level:
      1. value(c_i) = p(c,c_i)/d(c,c_i)
      2. get previousValue(c_i) from Results
      3. put <c_i, previousValue(c_i)+value(c_i)> to Results
3. sort Results
4. filter Results

Taking the graph given on Figure 1 as an example, the given algorithm would count the Topic 4 twice, since this topic shares a common ancestor with Seed 2 at level 1, and a common ancestor with Seed 1 at level 2. Its hyProximity value would then be greater than the value for Topic 1 that only shares one common ancestor with Seed 1 at level 1. The value for Topic 1 would be greater than the value for Topic 3 that only shares one ancestor with Seed 1 at level 2.

Since in our implementation we use the level of the first common ancestor category as a distance function, and the pondering function is directly dependent on distance, it is easy to calculate the values of hyProximity in an iterative process taking first the concepts that share a first level ancestor (a parent category), then second level ancestor (a grand-parent category), etc. Obtaining the concepts that share an ancestor at a particular level is simple using SPARQL queries on DBpedia[7].

The exponentially lower importance of siblings at higher levels, assures that stopping the algorithm at a certain level won't significantly affect the order of topics with higher hyProximity values. We have thus been able to run the algorithm only up to 3rd level without affecting the order of the hundreds of best ranked topics.

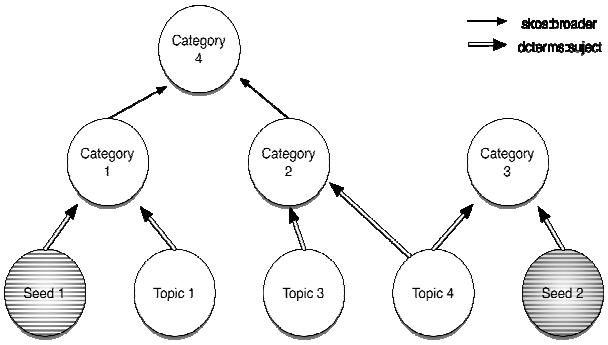

Figure 1 The Structure of Topics and Categories in DBpedia

5. FIGURES/CAPTIONS

We have conducted several experiments in order to evaluate our approach from different angles. We evaluate the ability of the system to allow the discovery of relevant and unknown topic where relevancy and novelty for the user are our main dimensions of evaluation.

5.1 Evaluation based on existing problems and solutions

Our first evaluation is based on previous problems and solutions on hypios. Since the purpose of our system is to provide topic concepts that would be used to attract relevant innovative solutions to our problems, it is reasonable to look back at previous problems and try to see if the suggested keywords appear in the solutions that were actually submitted.

Finding a system to compare with is a special difficulty as most of the state of the art systems are tailored to provide recommendations in order to refine and enrich user's input, and not necessarily help the user discover new and unknown relevant concepts. We have however found one system that is used to

[7] http://DBpedia.org/sparql

provide keyword suggestions for adWords campaigns[8]. This system, offered by Google, takes an initial set of keywords and provides suggestions, based on search patterns on google.com.

For the purpose of this evaluation, we have taken the texts of all the problems posted on hypios.com since the launch of the website in September 2009, which gave a total of 26 problems. A total of 142 solutions were received for those problems. We used a concept extraction tool, Zemanta[9], to extract the seed concepts from problem descriptions. Those concepts served as seed for finding suggestions using (1) our hyProximity and (2) adWords keyword suggestion tool. Zemanta returns concepts in the form of Wikipedia pages (consumed by hyProximity), but they are easily transformable to textual topics (consumed by adWords). Finally we have counted the number of suggested topics that appeared in the actual problem solutions.

Since we are only interested in measuring the discovery aspect we have not taken into account the topics that appear in both problems and solutions, but only solution-specific topics. Since hyProximity works with DBpedia concepts and adWords with textual keywords we have evaluated their success differently. We have admitted that hyProximity has found a topic if its suggestion appears in the topics that Zemanta has extracted from solutions to the problem. For adWords, we have admitted that the system has found a topic if a textual keyword it suggested appears in the text of any solution submitted to the given problem.

In our tests we have varied the number of best ranked suggestions, from both hyProximity and adWords, that we took into account in comparing with the topics in the actual solutions. This allowed us to study the impact of the number of suggested relevant keywords on the potential that they would be found in solutions. The results are presented on the Figure 2 – the x-axis shows the number of best-ranked suggestions taken into account[10], and the y-axis shows the corresponding number of topics found in solutions.

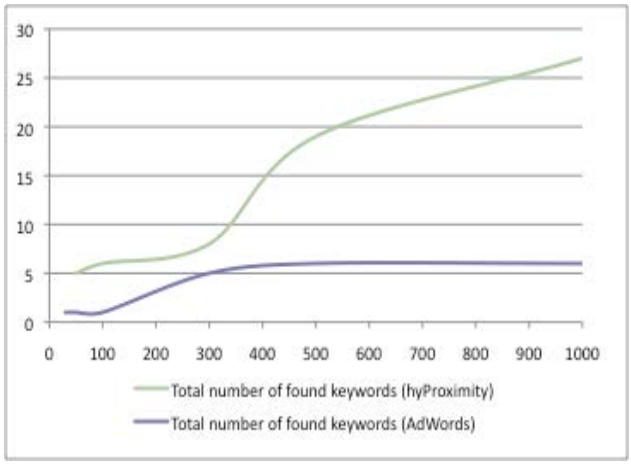

Figure 2 Number of lateral topics found in actual problem solutions and suggested by (1) hyProximity and (2) adWords on vertical axis. The horizontal axis represents the number of best ranked topic suggestions taken into account.

To give an example of correctly found topics, we consider a hypios problem about Ice Thickness Measurement, whose initial set of topics was: *Ice, Measurement, Electromagnetic Induction, Geology, Mass Balance*. The solution topics correctly "predicted" by AdWords were: *thickness measurement,* and *light measurement*; while hyProximity has found: *ice sheet, temperature, electrical resistance, ice age, radioglaciology, Young's modulus, speed*.

5.2 Evaluation based on user satisfaction

The existing gold standard data sets, such as Wordsim353[11], are mostly helpful in evaluating the relevancy of suggestions, and not their unexpectedness and non-obviousness; that are in the core of our interest, so we could not use them for this evaluation of this aspect.

Another reason for not using them is the fact that our notion of relatedness is dependent on the innovation problem and does not operate on a word-to-word basis. We have therefore conducted a user study to evaluate the relevancy and unexpectedness of the suggestions given by our system.

Ten users familiar with open innovation scenarios have participated in the study, generating a total of 22 evaluated problems, and evaluations of 2152 suggested keywords. Users have been asked to think of an innovation problem close to their knowledge or experience, write it down and provide 5-7 initial keywords. The initial keywords were provided in the form of Wikipedia pages representing the concepts embodied by the keywords, for disambiguation purposes.

In the next step we proposed the lists of 50 first ranked suggestions provided by adWords and by hyProximity, and asked the user to rate their relevancy and unexpectedness on a scale of 1 to 5. On the scale of relevancy 1 meant totally irrelevant, and 5 meant relevant. In our case relevant meant that one could expect a potential solver to be found by using the proposed topic in expert search. In the case of unexpectedness, the scale went from 1 - for the topics that were directly fundable in the problem text, to 5 – for the topics that the user did not even know about (and had to look up their meaning in a dictionary or encyclopedia).

The users were either employees of hypios with knowledge about problem formulation or PhD students having experience with OI problems in the industrial setting. The choice of evaluators was important for both of our measures. Their ability to judge the relevancy in this particular sense came out of their experience with OI problems, and at the same time they were not domain experts, but had rather general knowledge so the topics that they would judge as unexpected would most likely be also unexpected for an average innovation seeker from a client company.

[8] https://adwords.google.com/select/KeywordToolExternal

[9] http://www.zemanta.com/api/ . We do not especially focus on evaluating Zemanta against competing systems as it is out of the scope of our paper and the fact that the same system was used to obtain seed concepts for both adWords and our system ensures a level of comparability of their respective results.

[10] Google adWords provides a maximum of 800 topic suggestions

[11] http://alfonseca.org/eng/research/wordsim353.html

Table 1 Average note ± standard deviation obtained in the study

measure	adWords	hyProximity
relevancy	3.682±0.16	3.362±0.17
unexpectedness	2.580±0.16	3.461±0.18
unexpectedness (relevancy >=4)	2.322±0.24	3.169±0.30
unexpectedness (relevancy = 5)	1.939±0.25	2.502±0.38

As shown in the Table 1, the adWords keyword suggestion system performed better in terms of general relevancy, but was beaten by hyProximity in terms of general unexpectedness (by 34%). We tested the differences in unexpectedness for the two systems using the paired T-test over subjects individual means, and the unexpectedness was significantly different (p <0.0001). The difference in relevancy was not significant (p > 0.05).

5.3 Experience with Open Innovation Challenges

To test the system in a real life scenario we selected an open innovation problem hosted on hypios.com. The problem concerns finding a solution for a bake stable, non-sweet, intermediate water activity filling that is able to maintain a smooth creamy texture for a period of at least six months, sought by an international food company.

We used Zemanta to obtain the topics that appear directly in the problem description, and then generated suggestions of lateral topics using adWords keyword suggestion tool, and using hyProximity. Direct topics were used as seed for both keyword suggestion systems.

In the next step we selected experts from an existing expert database that were associated to those topics (both directly obtained from the problem using Zemanta, and lateral from adWords and hyProximity). The expert database was created at hypios in a 6 month long crawl of public data revealing some kind of expertise (university homepages, publications, etc.). In the third step we addressed all those experts with an automatically generated message and invited them to participate in the solution finding process for this specific problem, which is a standard way to perform problem broadcasting on hypios.

We were interested to see if the lateral topics would generate the same interest in the problem as the topics extracted directly from the problem text. In this experiment we measured the total of experts related to the suggested keywords and their response rate to the message.

The experts were asked to access a specific website, where they could find information concerning the problem and requirements for solving it through a link in the message that allowed us to track how many of the addressed subjects would actually follow the link.

The first column of Table 2 shows the total number of experts that were identified and subsequently addressed using the direct topics, and the lateral topics suggested by the two keyword generation systems. The second column (Response) shows the amount of people that used the link in the message to access the problem description website. The third column shows the percentage of the active response over the amount of people addressed.

We also addressed a random group of one thousand experts from the database to compare those results with the random selection, however the results for these were insignificant, with only one person out of the one thousand clicking the link in the message.

Table 2 Results of Open Innovation Experiment

System	Identified	Response	Percentage
Direct topics (by Zemanta)	1802	33	1.83
adWords	2061	41	1.98
hyProximity	2849	64	2.24

The response rate obtained through expert finding by using the lateral keywords was not only comparable but even better than in the case where only direct keywords were used. The obtained average response rate in the experiment is also higher than the average response rate of 1.5 % based on the history of hypios problem broadcasts.

This result shows that it is very promising to use the lateral keywords in problem broadcasting. An issue not addressed yet is the overlap of experts from different keyword groups. As there are some similarities in the terms selected, several experts were suggested by at least two sets of topics.

The relations between those selected experts and the topics is illustrated in the Venn diagram in Figure 3. The response rates where only effected by a rather small amount (9 people out of the total overlaps) where 6 of them are both in the adWords selected group and the Direct topics group plus 3 of them out of both adWords and hyproximity keyword group.

Since the problem challenge is still open we are not yet in position to measure the impact on the number of solution submissions that came from experts identified through lateral keywords. However the amount of interest generated within experts – potential solvers, is a significant indicator of the usefulness of the lateral keywords, especially those generated by hyProximity, for the purposes of OI problem broadcast.

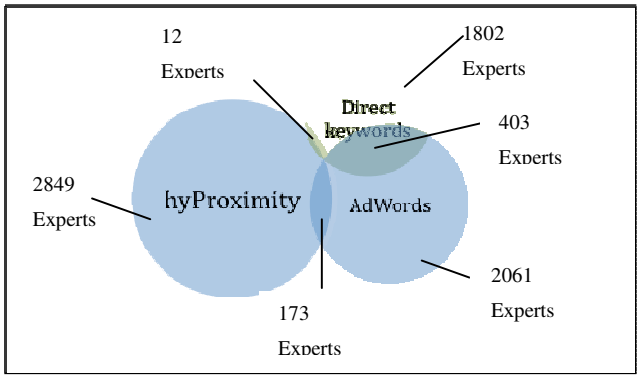

Figure 3 Venn diagram of overlap of experts in different groups

We repeated this setup with three more problems using the same amount of keywords and topics for two problems based on energy consumption and problem concerning a new approach for embedding a mixture of ion-exchange resins (IER) within a polymer matrix.

The three additional experiments showed similar results even thought the overall participation was lower as in the study

discussed in detail earlier. Figure 3 depicts a summery of the results of those three follow up studies.

Table 3 Average Results of Open Innovation Experiment for the last 3 hypios problems

System	Identified total	Response	Percentage
Direct topics (by Zemanta)	4562	67	1.47
adWords	5120	83	1.62
hyProximity	6997	141	2.01

The lower response rate might be a consequence of the fact that the competition for the solution of these problems was still ongoing while we wrote the paper and the submission deadlines where still far off.

6. Conclusions and Future Work

In this paper we have considered the problem of recommending relevant topics for expert search in OI scenarios. The essential difference of such a task from other topic recommendation tasks, such as tag recommendation and search query refinement, is the need to enable the discovery of lateral topics – unexpected and potentially unknown to the user, but still relevant.

We have explored the ability of DBpedia – a Linked Data version of Wikipedia, to serve as a ground for suggesting lateral keywords. Based on similar approaches for exploiting graph structures of background knowledge, we proposed a notion of proximity of topics in the DBpedia graph, called hyProximity, as well as a pragmatic approach for its calculation and filtering of the topics, which are of no interest for the industrial problem solving.

Our different evaluations demonstrate the usefulness of the suggested lateral keywords in the OI problem broadcasting, as well as the good performance of our approach against commercial state of the art systems for keyword suggestion. When performed on past OI problem definitions from hypios.com, hyProximity returns a much larger number of lateral topics that appeared in the actual solutions to those OI problems, then what we find with Google adWords' keyword suggestion system. In a user evaluation Google's system tends to give suggestions of slightly higher relevancy, but hyProximity gives results with much higher unexpectedness. The level of unexpectedness remains stronger with hyProximity even if only highly relevant suggestions are taken into account.

Finally we have demonstrated the usefulness of lateral topics suggested by our system in a real OI problem broadcast, where lateral keywords have generated a higher response rate to the problem notifications, then the topics that appeared directly in the problem.

We intend to continue the investigation of the impact that the lateral topics suggested by our system might have on the diversity of obtained solutions, by studying the domains that the solutions come from and comparing their diversity with the cases where the lateral topics were not used for problem broadcast.

We were aware that the concept space of the Web might be bigger than DBpedias however since our studies concentrated on preselected domains the difference is neglect- able. We also intend to consider improvements to the ranking of keyword suggestions that would help increase the user satisfaction with the results. This could be done by trying different pondering functions in hyProximity calculation. Another important direction of the future work is to experiment with our approach in other scenarios such as online advertising.

7. ACKNOWLEDGMENTS

The work of Milan Stankovic has been partially funded by ANRT (French National Research Agency) under the grant number CIFRE N 789/2009. We are also grateful to our user evaluators.

8. REFERENCES

[1] Lakhani, K. R., Jeppesen, L. B., Lohse, P. A., Panetta, J. A., & Research, H. B. S. D. (2007). The Value of Openness in Scientific Problem Solving. *Biotech Business*.

[2] Berners-Lee, T. (2006). Linked Data Desing Principles. Retrieved from http://www.w3.org/DesignIssues/LinkedData.html.

[3] Passant, A. (2010). dbrec — Music Recommendations Using DBpedia. In P. F. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Z. Pan, et al. (Eds.), *Proceedings of the 9th International Semantic Web Conferene - ISWC 2010* (Vol. 1380, pp. 1-16). Shanghai, China: Springer Verlag.

[4] Sigurbjörnsson, B., & Zwol, R. van. (2008). Flickr tag recommendation based on collective knowledge. *Proceedings of the 17th international conference on World Wide Web - WWW '08*, 327. New York, New York, USA: ACM Press.

[5] Mei, Q., Zhou, D., & Church, K. (2008). Query suggestion using hitting time. *Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08*, 469. New York, New York, USA: ACM Press. doi: 10.1145/1458082.1458145.

[6] Safar, B., & Kefi, H. (2004). OntoRefiner, a user query refinement interface usable for Semantic Web Portals. *Proceedings of Application of Semantic Web technologies to Web Communities, Workshop ECAI'04* (pp. 65-79).

[7] Macdonald, C., & Ounis, I. (2007). Expertise drift and query expansion in expert search. *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management - CIKM '07*, 341. New York, New York, USA: ACM

[8] Matos, S., Arrais, J. P., Maia-Rodrigues, J., & Oliveira, J. L. (2010). Concept-based query expansion for retrieving gene related publications from MEDLINE. *BMC bioinformatics*, *11*, 212.

[9] Salton, G. and McGill, M. Introduction to Modern Information Retrieval. New York: McGraw-Hill, 1983

[10] Waltinger, U., & Mehler, A. (2009). Social semantics and its evaluation by means of semantic relatedness and open topic models. *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 01* (p. 42–49). IEEE Computer Society.

[11] Cilibrasi, R. L., & Vitanyi, P. M. B. (2007). The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering*, *19*(3), 370-383. doi: 10.1109/TKDE.2007.48.

[12] Bollegala, D. (2007). Measuring semantic similarity between words using web search engines. *Proceedings of the 16th international conference on World Wide Web - WWW '07*, *V*(December), 757. New York, USA: ACM Press. doi: 10.1145/1242572.1242675

[13] Resnik, P. (1995). Using Information Content to Evaluate Semantic Similarity in a Taxonomy. Arxiv preprint cmp-lg/9511007, 1.

[14] Burton-Jones, A., Storey, V., Sugumaran, V., & Purao, S. (2003). A heuristic-based methodology for semantic augmentation of user queries on the web. *Conceptual Modeling-ER 2003*, 476–489. Springer.

[15] Ziegler, C.-N., Simon, K., & Lausen, G. (2006). Automatic Computation of Semantic Proximity Using Taxonomic Knowledge Categories and Subject Descriptors. *CIKM '06 Proceedings of the 15th ACM international conference on Information and knowledge management* (pp. 465-474). Arlington, Virginia, USA:

[16] Maguitman, A. G., Menczer, F., Roinestad, H., & Vespignani, A. (2005). Algorithmic detection of semantic similarity. *Proceedings of the 14th international conference on World Wide Web* (p. 107–116).

[17] Strube, M., & Ponzetto, S. P. (2006). WikiRelate! Computing semantic relatedness using Wikipedia. *Proceedings of the National Conference on Artificial Intelligence* (Vol. 21, p. 1419). Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press;

[18] Bizer, C., Heath, T., & Berners-lee, T. (2009). Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, (Special Issue on Linked Data