User Interests Identification on Twitter Using a Hierarchical Knowledge Base*

Pavan Kapanipathi[1], Prateek Jain[2], Chitra Venkataramani[2], and Amit Sheth[1]

[1] Kno.e.sis Center, Wright State University
{pavan, amit}@knoesis.org
[2] IBM TJ Watson Research Center
{jainpr, chitrav}@us.ibm.com

Abstract. Twitter, due to its massive growth as a social networking platform, has been in focus for the analysis of its user generated content for personalization and recommendation tasks. A common challenge across these tasks is identifying user interests from tweets. Semantic enrichment of Twitter posts, to determine user interests, has been an active area of research in the recent past. These approaches typically use available public knowledge-bases (such as Wikipedia) to spot entities and create entity-based user profiles. However, exploitation of such knowledge-bases to create richer user profiles is yet to be explored. In this work, we leverage hierarchical relationships present in knowledge-bases to infer user interests expressed as a *Hierarchical Interest Graph*. We argue that the hierarchical semantics of concepts can enhance existing systems to personalize or recommend items based on a varied level of conceptual abstractness. We demonstrate the effectiveness of our approach through a user study which shows an average of approximately eight of the top ten weighted hierarchical interests in the graph being relevant to a user's interests.

Keywords: #eswc2014Kapanipathi; User Profiles; Personalization; Social Web; Semantics; Twitter; Wikipedia; Hierarchical Interest Graph

1 Introduction

A squirrel dying in your front yard may be more relevant to your interests right now than people dying in Africa. - Mark Zuckerberg, Facebooks CEO [3].

* This material is based on the first author's work at IBM Research, complemented in part based upon work supported by the National Science Foundation SoCS program under Grant No.(IIS-1111182, 09/01/2011-08/31/2014). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the employer or funding organization. We would like to thank: (1) Zemanta for their support; (2) Participants of the user study; (3) T K Prasad, Delroy Cameron, Sarasi Lalithasena, Sanjaya Wijeyaratne and Revathy Krishnamurthy for their invaluable feedback.

[3] `http://www.nytimes.com/2011/05/23/opinion/23pariser.html`

Content personalization based on social activities (clicks, posts) is gaining increasing traction with web companies day by day. A variety of services and platforms on the digital web, right from movies on Netflix to navigation routes on GPS (Waze) are personalized based on what you like and what you did. The personalized content for each individual is determined using various metrics such as click behavior, collaborative filtering and cookies. A common element across these techniques is the focus on using current browsing session for providing personalization and therefore a lack of identification of the broader interests.[4]

In this work, we try to address this shortcoming of content personalization by providing a framework for identification of broader user interests based on the content generated by them on Twitter. Specifically, given a tweet *"Now the sensible thing to do would be to conserve the money I have. But I want a new pair of trainers"* [5], our work provides a framework to identify that a person expressing an interest in buying a pair of *"training shoes"* is potentially interested in *"running"*. Once *"running"* is identified as an interest, a recommendation engine can utilize it in conjunction with other metrics to personalize user experience and recommend content. We utilize Twitter due to (1) higher degree of openness, and (2) in [2, 17], tweets have been demonstrated to be a good indicator for determining user interests. For identification of hierarchical categories, we exploit Wikipedia (specifically the category graph) as the knowledge source. The inferred interests are represented in the form of a *Hierarchical Interest Graph (HIG)*. This representation will provide a personalization and recommendation engine with the flexibility to filter content based on abstract interests of users.

The key contributions of our work are as follows: (1) We propose a novel approach that extends the entity-based representation of user interests to a hierarchical representation. (2) We determine the interest scores for the categories in the *Hierarchical Interest Graph* by adapting the spreading activation algorithm [4] for the Wikipedia Category Graph (WCG). (3) We demonstrate a simple but efficient approach to transform the Wikipedia Category Graph into a hierarchy. This hierarchy is used as the base hierarchy for the Interest Graphs. Our evaluation shows an overlap of 87% hierarchical links between mapped categories with a manually created taxonomy - DMoz. (4) We present a user study of 37 participants with a comprehensive evaluation of our approach. The results show that our approach is practically useful with *top-10* ranked interests evaluating a mean average precision of 88%.

Example and Terminology. Consider the following tweets from a user:
– *Great day for Chicago sports as well as Cubs beat the Reds, Sox beat the Mariners with Humber's perfect game, Bulls win and Hawks stay alive*
– *Not sure who the Reds will look too replace Dusty.some very interesting jobs open (Cubs, Mariners, Reds, poss Yanks) Girardi the domino*

Preponderance posting of such tweets, we can determine that the user might be interested in Baseball teams such as *Cincinnati Reds, Chicago Cubs, Boston Red Sox*. We term the entities that can be directly spotted from user's tweets

[4] Netflix and Pandora get explicit input from users to generate broader interests.
[5] `http://bit.ly/sectorRoadMapGigaom`

as ***Primitive Interests***. Further, our approach exploits the knowledge linked to *Primitive Interests* (Baseball teams) in Wikipedia to determine that the user might be also interested in broader categories such as *Category: Major League Baseball, Category: Baseball*. These categories are termed as ***Hierarchical Interests***. Our goal is to determine the most relevant *Hierarchical Interests* by using *Primitive Interests* extracted from tweets.

Most Wikipedia entities have categories with the same label (*ex: Cincinnati Reds and Category: Cincinnati Reds*). The categories *(Hierarchical Interests)* that syntactically do not match entities (*Primitive Interests*) are termed as ***Implicit Interests***, also because they are not explicitly mentioned in tweets by the user. Formally, *Implicit Interests* $\subseteq$ *Hierarchical Interests*.

Spreading Activation. In this work, the Spreading Activation theory is used to assign appropriate scores for the categories in the Wikipedia hierarchy. Spreading activation theory builds on the assumption that the information in the human memory is represented either through association [10] or via semantic networks [20]. This theory has been utilized for various domains ranging from cognitive, neural sciences to Information Retrieval [5] and Semantic Web. The Spreading Activation theory in its pure form consists of a simple processing technique on a network data structure. A network data structure consists of nodes connected by means of links or edges.

Given a set of initially activated nodes, the processing technique consists of a series of iterations. An iteration can consist of one or more pulses or a termination check. A pulse can consist of three different phases (1) Pre-adjustment phase (2) Spreading (3) Post-adjustment phase. Of the three, pre-adjustment and post-adjustment phases are optional and consist of applying some form of an activation decay to the active nodes. The spreading phase consists of sending activation waves from one node to all the other directly connected nodes. The activation is however, controlled by an application dependent *activation function*. These iterations continue until a stopping condition is reached or the processing is halted by the user. More details on Spreading Activation is presented in [4].

In the next Section (Section 2) we present the approach followed by evaluation in Section 3. Section 4 details the related work whereas the last Section 5 concludes with future work.

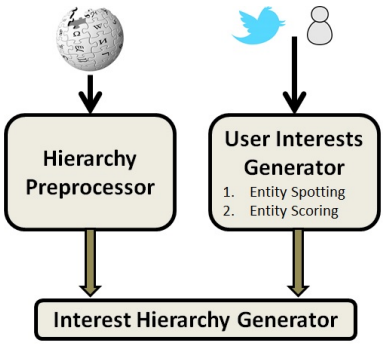


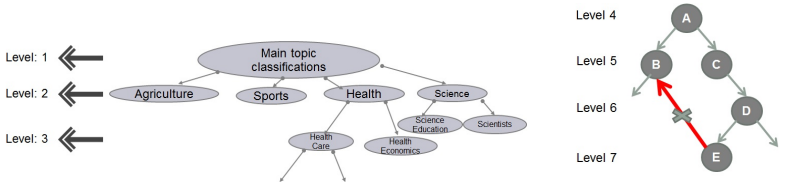
Fig. 1: Architecture

2 Approach

The goal of our approach is to construct a *Hierarchical Interest Graph (HIG)* for a Twitter user. To accomplish this our system as illustrated in Fig. 1, performs the following steps: (1) **Hierarchy Preprocessor** transforms the *Wikipedia Category Graph* (WCG) into a hierarchy that is needed to generate all the *HIG*s. This pre-processing step is necessary because

of the challenges introduced by Wikipedia (detailed in Section 2.1). (2) **User Interests Generator** generates the *Primitive Interests* (defined in Section 1-Terminology) from the tweets of a user. The module (Section 2.2) first spots entities that are Wikipedia articles *(Primitive Interests)* and then scores them to reflect users' interests. (3) **Interest Hierarchy Generator** maps the *Primitive Interests* to *Wikipedia Hierarchy* and uses an adaptation of the spreading activation algorithm to infer a weighted *HIG* for the user (Section 2.3). Step 1 is updated periodically to keep abreast with the changes in Wikipedia whereas Step 2, 3 are performed for each user.

2.1 Hierarchy Preprocessor



(a) Hierarchy Level Assignments (b) Non Hierarchical Links

Fig. 2: Hierarchy Preprocessing

We utilize Wikipedia as the knowledge-base for inferring *Hierarchical Interests*. Although, there are other free ontologies such as OpenCyc[6], and the ODP taxonomy[7], we opted for Wikipedia because of its vast domain coverage. However, a major challenge faced in utilizing Wikipedia as a hierarchy is that, its category graph (*WCG*) comprises of cycles and hence it is neither a taxonomy nor a hierarchy. These cycles make it non trivial to determine the hierarchical relationships between categories. For example, determining that *Category:Baseball* is conceptually more abstract than *Category:Major League Baseball* is difficult if there exists cycles in the graph. Therefore we transform *WCG* to a hierarchy by assigning levels of abstraction for each category.

Firstly, we remove categories that are irrelevant for our work. Specifically, we remove the Wikipedia admin categories[8] that are used only to manage Wikipedia. A sub-string match is employed for the categories with the set of labels used in [18]. Consequently, around 64K categories with 150K links are removed from *WCG* as shown in Table 1.

Level Identification. The root category (node) of *WCG* is *Category: Main Topic Classifications*, which subsumes 98% of the categories (Table 1, 0.80M

[6] `http://www.opencyc.org/`
[7] `http://www.dmoz.org/`
[8] `http://en.wikipedia.org/wiki/Category:Wikipedia_administration`

out of 0.82M categories). Selecting this root node as the most abstract category, we determine the relative hierarchical levels of other categories. We assign the shortest distance to the category from the root as its hierarchical level (level of abstractness) as shown in Fig. 2(a).

Non Hierarchical Links Removal. Once the hierarchical levels are assigned we remove the edges that do not conform to a hierarchical structure, i.e. all the directed edges from a category of larger hierarchical level (specific) to a smaller hierarchical level (conceptually abstract) are removed. Considering Fig. 2(b), the link such as those from node ***E*** to node ***B*** are removed, since node ***E*** has been determined as a more specific node (Level 7) in the hierarchy than node ***B*** (Level 5). Performing this task reduced *WCG* from 1.9M links to 1.2M links (Table 1), also leading to the removal of cycles in WCG.

The output of this process is a hierarchy with $height = 15$, rooted at the node *Category: Main Topic Classifications*. The nodes in the hierarchy have many to many relationships and hence it is still not a taxonomy. This refined graph with directed edges that conform to a hierarchy is referred to as *Wikipedia Hierarchy* (*WH*).

	Categories	**Links**
Wiki	884,838	2,074,173
Wiki(WA)	820,476	1,922,441
Wiki(WH)	802,194	1,177,558

Table 1: Wikipedia Categories and Links. WA: Without Admin, WH: Hierarchy Preprocessed.

2.2 User Interests Generator

This module identifies *Primitive Interests* from a user's tweets by *Entity Recognition*, and scores them based on their frequency.

Entity Recognition. The first step towards identification of *Primitive Interests* is Entity Recognition[9] in tweets. Entity recognition in tweets is non trivial due to the informal nature and ungrammatical language [23] of tweets. Since the focus of our work is on hierarchical interests identification and not entity recognition, we used an existing solution.

In [6] authors have compared three different state of the art systems namely Dbpedia Spotlight [14], Zemanta[10] and TextRazor[11] for entity recognition in tweets. These results have been summarized in Table 2. We opted to use Zemanta for our work because of the following reasons: (1) Zemanta links the entities spotted in tweets to their corresponding Wikipedia articles (*Primitive Interests*); (2) Zemanta has relatively superior performance to other services as shown in

Extractors	**Pr**	**Re**	**F-M**	**Limit**
Spotlight	20.1	47.5	28.3	N/A
TextRazor	**64.6**	26.9	38.0	500/day
Zemanta	57.7	31.8	**41.0**	**10,000/day**

Table 2: Evaluation of Web Services for Entity Resolution and Linking from [6]. Pr: Precision, Re: Recall, F-M: F-Measure

[9] Details on different techniques for Entity Recognition in tweets is presented in [6]
[10] `http://developer.zemanta.com/`
[11] `http://www.textrazor.com/technology`

Table 2; and (3) Zemanta increased the rate limit of their API[12] to 10,000 per day, on request for research purposes.

Scoring User Interests. Once the *Primitive Interests* are identified, the next task is to score them to find the degree of user's interests across different entities. This is important as the scores of *Primitive Interests* are utilized in scoring the appropriate *Hierarchical Interests* (Section 2.3) for the user. We employ a frequency based scoring mechanism similar to those used in [2, 27]. The score for an entity is determined using the equation: $nf_i = frequency(e_i)/frequency(e_{max})$. The score ranges between 0-1, as in the formula the frequency of mentions of an entity in tweets ($frequency(e_i)$) is normalized by the frequency of the entity that is mentioned the most by the user ($frequency(e_{max})$). To summarize, the results of this module are a set of weighted *Primitive Interests* with weights reflecting the user's degree of interest.

2.3 Interest Hierarchy Generator

For each user, the Interest Hierarchy Generator takes a set of scored *Primitive Interests* and the *WH* as input to generate a weighted *HIG*. The *Primitive Interests* are added as leaf nodes to the *WH* by linking to their appropriate categories. Then, the scores of *Primitive Interests* are propagated up the hierarchy as far as the root using Spreading Activation theory to determine the interest categories and their appropriate weights. The propagation of scores to the categories is performed using an activation function (see Section 1 - Spreading Activation). A basic activation function is shown in Equation 1.

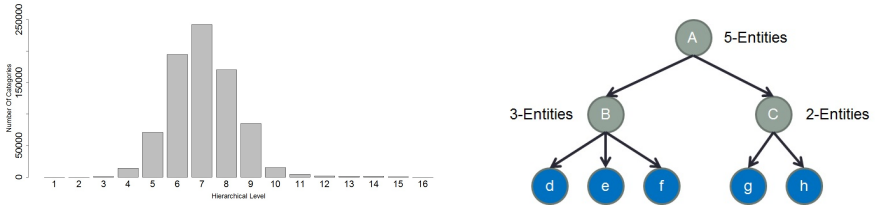
$$A_i = A_i + A_j \times W_{ij} \times D \tag{1}$$

where i is the node to be activated (Parent Category) and A_i is its activation value; j is the activated child node of i (*Primitive Interests*/Child Category); W_{ij} is the weight of the edge connecting node i and j; D is the decay factor. We utilized different variations of *Activation functions* as follows:

1. We experimented with a *no-weight no-decay* option on the basic spreading activation function (Equation 1 with $W_{ij} = 1$, $D = 1$). The resulting *HIG* had higher scores for interest categories that are higher (conceptually abstract) in the hierarchy. This is intuitive, because the activation values were propagated up the hierarchy without any constraints. Further, we experimented with empirically decided, decay factors ($D = 0.4, 0.6, 0.8$) as constraints up the hierarchy. Although there were slight variations, there were no significant improvements with the results. This motivated us to analyze the distribution of nodes in the hierarchy for better normalization.

2. The distribution of categories across the hierarchy follows a bell curve as shown in Fig. 3(a). This uneven distribution impacts the propagated scores by increasing the scores of categories with more child nodes. Therefore, we normalized the activation value of each of the *Hierarchical Interests* based on the

[12] http://developer.zemanta.com/docs/suggest/



(a) Nodes Distribution (b) Intersect Booster

Fig. 3: Interest Hierarchy Generator

number of sub-categories at its child level. This was experimented with the raw count of the node frequency (Equation 2). As shown in Fig. 3(a), the peak of the bell curve is at level 7 with about 250k nodes. if Equation 2 is used, these large values have a heavier penalty on the interest scores. Therefore, we also experimented with log scale of the raw numbers (Equation 3).

$$F_i = \frac{1}{nodes_{(h_i+1)}} \qquad (2) \qquad FL_i = \frac{1}{\log_{10} nodes_{(h_i+1)}} \qquad (3)$$

where h_i is the hierarchical level of node i; $nodes_h$ is number of nodes at hierarchical level h.

3. *Preferential Path Constraint*: The nodes in *WH* have many categories associated with them. Considering our example in Section 1, (Dated-Jan 9th 2014) *Cincinnati Reds* has categories starting with *Major League Baseball teams, Sports in Cincinnati, Ohio, Sports clubs established in 1882, etc.* One of the problems we noticed is that all these categories were given equal priority and hence equal weights were being propagated. Therefore, we introduced the *preferential path constraint* to prioritize the categories for a node. The motivation is drawn from the Wikipedia category structure where for any article or category, on their Wikipedia page, the parent categories are ordered from left to right in decreasing order of significance. Having the categories of *Cincinnati Reds* in the same order as mentioned above implies that *Category:Major League Baseball teams* is more suitable as a category of *Cincinnati Reds* than the rest. We utilize this heuristic as preferential path constraint in the activation function. This is similar to adding weights to the edges in *WH* and is accomplished using the Equation 4.

$$P_{ij} = \frac{1}{priority_{ji}} \qquad (4)$$

where $priority_{ji}$ is the priority of category i for subcategory j. $priority_{ji}$ increases linearly $(1, 2, ..)$ reflecting the order of categories from left to right.

4. *Intersect Booster*: We utilize this variation to boost the categories (nodes) in the hierarchy that forms the intersecting point of multiple *Primitive Interests* for a given user. For example, consider the hierarchy in Fig. 3(b), where *d, e, f, g, h* are entities and *A, B, C* are categories. If only *d, e, f* are considered as user's *Primitive Interests*, the most appropriate *Hierarchical Interests* would be *Category:B*. On the other hand, if entities *g, f* are also user's interests then

Category:A would be the more appropriate due to the intersection of maximum *Primitive Interests* at *Category:A*. Therefore, to formalize this aspect and boost the score of intersecting nodes, we introduced Equation 5.

$$B_i = \frac{N_{e_i}}{N_{e_{cmax}}} \tag{5}$$

where N_{e_i} is the total number of entities activating node i; $cmax$ is the subcategory of i that has been activated with max number of entities.

In Fig. 3(b), if *d, e, f* are *Primitive Interests* then for *Category A*, $B_A = \frac{3}{3}$. If *g, h* are *Primitive Interests*, then $B_A = \frac{5}{3}$ (increases).

Activation Functions. Using the variations explained above, we created different activation functions which are as follows:

–*Bell*: The Bell function is as shown in Equation 6. This function spreads the activation value up the hierarchy with a raw normalization (Equation 2).

$$A_i = A_i + A_j \times F_i \tag{6}$$

–*Bell Log*: This function (Equation 7) uses the log normalization (Equation 3) to reduce the impact of the raw count.

$$A_i = A_i + A_j \times FL_i \tag{7}$$

–*Priority Intersect*: The final activation function that we experimented with builds on the *Bell Log* function (Equation 3). This function rewards the categories on the left (Equation 4) and boosts the interesting nodes (Equation 5). Formally, the function is represented by the Equation 8.

$$A_i = A_i + A_j \times FL_i \times P_{ij} \times B_i \tag{8}$$

3 Evaluation

The input to our system is a set of tweets for a given user and the *WCG*, whereas the output is a weighted *HIG* for each user. We evaluate the following two aspects of the system: (1) We perform a user study to evaluate the *Hierarchical Interests* generated for each user using the activation functions explained in Section 2.3. (2) Since the Wikipedia Hierarchy plays an important role in generating *HIG*s, we evaluate the hierarchy against a manually constructed taxonomy - DMoz.

3.1 Hierarchical Interests Evaluation

Evaluation of personalization and/or recommendation systems typically involves a user study as performed in various works [1, 17, 19]. The user studies involve a set of users participating in evaluating the results generated by the system.

User Study. 37 users agreed to participate in our user study by giving us access to their tweets and agreeing to evaluate the results. Our system analyzed their tweets and generated results using the following activation functions: (1) *Bell*, (2) *Bell Log*, (3) *Priority Intersect*.

	Users	Tweets	Entities	Distinct Entities	Tweets with Entities	Categories in HIG
Total	37	31927	29146	13150	16464	111535
Average		864	787	355	445	3014

Table 3: User Study Data

Each activation function when employed generated corresponding weighted *HIGs* for users. The *top-50* scored *Hierarchical Interests* from each *HIG* were selected for user evaluation. The evaluation requested the users to mark the *Hierarchical Interests* as *Yes/No/Maybe* to indicate their interest or lack of. To ensure unbiased results from the users, the *Hierarchical Interests* when presented to the user neither had any order (based on score) nor contained any associated information such as the tweets or the associated activation functions.

The guidelines provided to users included: (1) The interests provided were categories (conceptually abstract) not entities; (2) The interest generation did not involve a temporal aspect. If the user at any point of time was interested in an event or a topic (*Category:Super Bowl*, *Category:United States presidential election, 2012*) then they have to be considered as their interests; (3) It is unlikely that users tweet about everything which is of interest to them. Therefore, the users were asked to mark for relevance based on the topics they tweet or had tweeted in the past. For instance, if a user has never tweeted about *Category:Pets* and the system marks it as an interest then such interests should be marked as irrelevant; (4) The *Maybe* option in evaluation is introduced due to the abstractness of the interests. For example, for users who are only interested in *Baseball, American Football*, an interest such as *Category:Sports* inferred might be too broad/abstract to mark *Yes*.

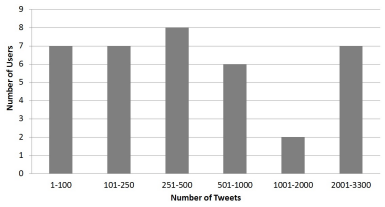
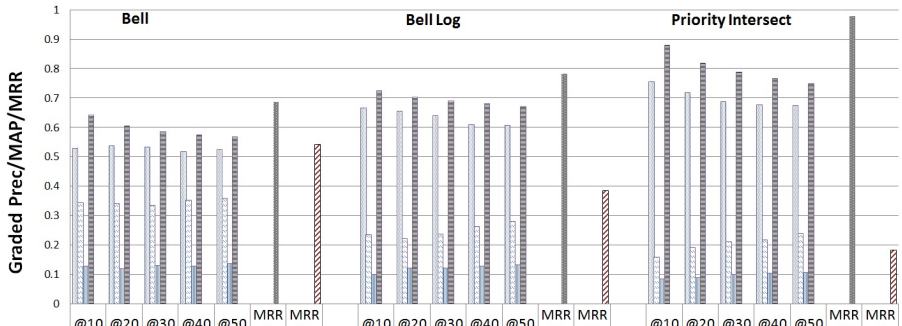


Fig. 4: Users Tweets Distribution

Data. The survey was conducted with 37 Twitter users having varying number of tweets. The "users to tweets" distribution is shown in Fig. 4. Table 3 shows the user statistics, the volume of tweets and number of entities identified in the tweets. Approximately 32K tweets were obtained using the Twitter API[13]. Due to the restriction enforced by Twitter Search API, for seven users who have more tweets, we could retrieve only 3200 per user. From 32k tweets, 29k entities were extracted, out of which approximately 45% are distinct. Further, the users found 58% of the interest categories identified by our system to match with their interests (Yes), limited confidence in 12% (Maybe) and 30% were marked irrelevant (No).

Results. In order to compare the three activation functions and evaluate their results, we answered the below questions and accordingly selected the evaluation metrics. The selected evaluation metrics are standard metrics in Information Retrieval and more details can be found in [13].

[13] `https://dev.twitter.com/docs/api/1.1/get/search/tweets`



	@10	@20	@30	@40	@50	MRR Y	MRR N	@10	@20	@30	@40	@50	MRR Y	MRR N	@10	@20	@30	@40	@50	MRR Y	MRR N
yes	0.53	0.54	0.53	0.52	0.52			0.67	0.66	0.64	0.61	0.61			0.76	0.72	0.69	0.68	0.67		
no	0.34	0.34	0.34	0.35	0.36			0.23	0.22	0.24	0.26	0.28			0.16	0.19	0.21	0.22	0.24		
maybe	0.13	0.12	0.13	0.13	0.14			0.1	0.12	0.12	0.13	0.13			0.08	0.09	0.1	0.11	0.11		
MAP	0.64	0.61	0.59	0.58	0.57			0.72	0.7	0.69	0.68	0.67			0.88	0.82	0.79	0.77	0.75		
MRRY						0.69							0.78							0.98	
MRRN							0.54							0.39							0.18

Fig. 5: Evaluation Results – MAP: Mean Average Precision; MRR-Y/N: Mean Reciprocal Recall-Relevant/Irrelevant Results.

How many relevant/irrelevant Hierarchical Interests are retrieved at top-k ranks?: To assess this question, we adapt the *Precision@k* metric to deal with the graded (*Yes/No/Maybe*) results from our user study. We term the metric as *GradedPrecision* and is as shown in Equation 9

$$GradedPrecision_{res}@k = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{HI_{res}@k}{k} \tag{9}$$

where k is the rank; Q is the set of users in the user study; res is the one of the options evaluated by the user *Yes/No/Maybe*; $HI_{res}@k$ is the total number of *Hierarchical Interests* marked res at rank k.

The equation for graded *Hierarchical Interests* results in the distribution of each grade between the range 0-1. We employed *GradedPrecision* for every rank interval of 10 for *top-50 Hierarchical Interests* for each activation function. Fig. 5 shows that, on an average the *Bell* is able to retrieve 53% relevant *Hierarchical Interests* from the *top-10* interests, whereas the *Priority Intersect* is able to retrieve 76% of relevant results. This is accompanied with lesser retrieval of irrelevant results by *Priority Intersect* compared to *Bell Log* and *Bell*. We need to note that, although the number of *maybe's* are low, they hold a potential of being interesting to the users. Thus to summarize, the *Priority Intersect* retrieves more (23% more than the baseline *Bell* at *top-10*) relevant *Hierarchical Interests* than the other two activation functions.

How well are the retrieved relevant Hierarchical Interests ranked at top-k?: This question is answered by employing the standard ranking evaluation metric *Mean Average Precision (MAP)*. *MAP* is used with binary results and hence, we considered the *Hierarchical Interests* marked *Yes* as relevant and

No/Maybe as irrelevant for further variations of this evaluation. Formally MAP is as follows:

$$MAP(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j} Precision(R_{jk}) \tag{10}$$

where Q is the set of users in the user study; m_j is the total number of relevant *Hierarchical Interests*; $Precision(R_{jk})$ is the $Precision@k$ of user j.

Similar to *GradedPrecision*, we calculated MAP for every interval of 10 ranked *Hierarchical Interests*. Higher the MAP, better are the relevant *Hierarchical Interests* ranked. As shown in Fig. 5, *Priority Intersect* does convincingly better in ranking the *top-10* relevant *Hierarchical Interests* with MAP of 88% compared to 72% of *Bell Log* and 64% of *Bell*. If *Hierarchical Interests* marked *Maybe* by users are considered relevant then MAP at *top-10* increases to 92% for *Priority Intersect*, 82% for *Bell Log* and 71% for *Bell*.[14]

How early in the ranked Hierarchical Interests can we find a relevant result?: The Mean Reciprocal Rank (MRR) metric captures the answer to the above question. Formally, the metric is as shown in Equation 11.

$$MRR_{res} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{11}$$

where Q is the set of users in the user study; $rank_i$ is the rank at which the first yes/no result is found for user i; res is either relevant or irrelevant result (yes/no).

We have employed MRR for both relevant (MRR_Y in Fig. 5) and irrelevant results (MRR_N in Fig. 5). If the system finds a relevant *Hierarchical Interests* sooner in the ranked list for the users then MRR_Y is higher. On the other hand, if an irrelevant interest is ranked higher then MRR_N is higher. Therefore, a system is better if MRR_Y is higher and MRR_N is lower. Fig. 5 shows that, *Priority Intersect* was able to rank a relevant *Hierarchical Interests* at the top for all users but one ($MRR_Y = 0.98$). The *Bell Log* does fairly good with an MRR_Y of 0.78 for relevant result.

How many of the categories, inferred by the system, were not explicitly mentioned by the user in his/her tweets?: By answering this question, we will be able to evaluate the *Hierarchical Interests* that had no syntactic mentions in the tweets of users and hence are inferred by exploiting the knowledge-base. In other words, we evaluate the *Implicit Interests* (see Section 1 for definition and example) detected by our system. Although *Primitive Interests* and *Hierarchical Interests (Implicit Interests)* are semantically different, we intended to signify the value added by the knowledge-bases.

Fig. 6 shows the average percentage of *Implicit Interests* by each activation function to be 81% for *Bell*, 78% for *Bell Log* to 71% for *Priority Intersect* for the *top-50* ranked *Hierarchical Interests*. We then calculated *GradedPrecision* for

[14] Please visit the project page `http://wiki.knoesis.org/index.php/Hierarchical_Interest_Graph`.

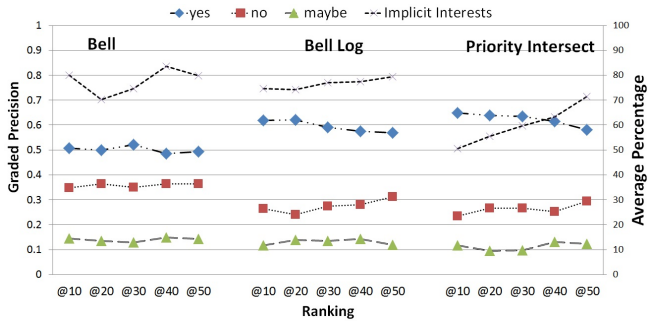


Fig. 6: Evaluation of Implicit Categories. Primary Y-Axis for Graded Precision (yes,no,maybe). Secondary Y-Axis for Average Percentage (*Implicit Interests*)

Implicit Interests (*Yes/No/Maybe*) detected by the activation functions. Fig. 6 shows that *Priority Intersect* achieves the best results (65% of the *top-10 Implicit Interests* were relevant to the users). From this evaluation we can conclude that our approach is able to detect implicit *Hierarchical Interests* that have no explicit mention in users' tweets.

Overall, Fig. 5 and Fig. 6 illustrates all our evaluations on the quality of *Hierarchical Interests* generated by our approach. We can hence conclude that our approach with *Priority Intersect* activation function performs the best in determining *Hierarchical Interests* of a user that is represented as *HIG*.

Comparative Evaluation. The closest work to our approach that has been published in its initial stages is a system called Twopics [15]. Twopics generates a ranked list of Wikipedia categories as user interests from tweets. Twopics extracts entities from user's tweets and then for each of these entities infer the categories upto five levels from WCG. The scoring of these categories is based on the frequency of it being inferred for a user. The paper has a very initial evaluation and does not provide any gold standard dataset to compare against. Therefore, we implemented their approach. Although their approach did not result in a hierarchical representation of interests, we found that the ranked interest categories were similar to our *no-weight no-decay* activation function where the more abstract categories were ranked higher. We compared Twopics to our baseline –*Bell*, using a small scale user study with 6 users. The evaluation of *top-50* results showed that *Bell* activation function with 52% relevant results performed better than Twopics with 38% relevant results. Confirming our intuition of similarity with *no-weight no-decay* results, the analysis of the small scale evaluation ranked abstract categories higher in the interest list.

3.2 Wikipedia Hierarchy Evaluation

In order to evaluate the quality of the automatically generated hierarchy by our approach, we followed a similar methodology used by [18]. In [18] the authors have constructed a taxonomy from *WCG* and have evaluated it by comparing

it with Research Cyc. We would have preferred to use their implementation, however we did not receive any response for our request.

We evaluated the *WH* with the category system of manually constructed taxonomy DMoz. The information on DMoz category hierarchy is available on DMoz download page.[15] The methodology is as follows:(1) We mapped categories from the *WH* to DMoz categorization. We performed a simple string match between the category labels of Wikipedia and DMoz. 141,506 categories matched. (2) Next, we traversed through the *WH* to find category-subcategory relationships of all distances (transitively related sub-categories) between the mapped categories. We found 46,226 category-subcategory relationships. Our Next step was to check the quality of these links by its presence in DMoz (Gold Standard). (3) In order to verify the Wikipedia category-subcategory relationships from Step 2, we traversed through DMoz category hierarchy to check the existence of directed paths between the same categories and subcategories. **87.62%** of the *WH* relationships were found in DMoz. Therefore, we concluded that our automatic hierarchy generation approach has high overlap of category-subcategory relationships for the mapped categories with manually created DMoz. This is a good indication about the quality of links in the *WH*, which in-turn evaluates the quality of links present in the *HIG*s generated by our approach.

NOTE: More analysis, evaluations and datasets is available on *project page*[16].

4 Related Work

Personalization on the web started by analyzing web documents that users visit in order to generate user's interests [8, 22]. Recently, the increasing adoption of social networks such as Twitter, has shifted the personalization systems to analyze user activities on these platforms. Each of these work either uses Bag of Words [16], Topic Models [9, 21] or Bag of Concepts [1, 2, 17] approach. In our work we started with the Bag of Concepts approach due to the availability of knowledge-bases linked to the concepts that are leveraged to infer *Hierarchical Interests*. On the other hand, Bag of Words and Topic Models (shallow inferencing) lack this advantage of utilizing explicit semantics. Furthermore, it has been argued that these techniques may not perform so well on tweets as the tweet content is short and informal [26].

In the area of web personalization and recommendation, generating hierarchical interests for a user involves analyzing web documents. In [8, 29], the authors have realized top-down techniques to hierarchically cluster web documents the user is interested in. Both the techniques are built upon Bag Of Words approach and the hierarchical clusters of terms form the user profiles. On the other hand, work in [22, 25] analyze web documents and leverage ontologies to create contextual user profiles. The former [22] use Bag Of Words approach to map web documents to Wikipedia concepts. Sieg et al. [25] used DMoz with an adaptation of spreading activation to map web documents to DMoz articles.

[15] `http://www.dmoz.org/rdf.html`

[16] `http://wiki.knoesis.org/index.php/Hierarchical_Interest_Graph`

User interests extracted from social messages have been represented as Bag Of Concepts in various works [2, 12, 17, 27]. One of the main aspects of these works is the weighting schemes used to reflect user's interests towards the concepts. Abel et al. in their work [2] compare hashtag-based, entity-based and topic-based user models generated from tweets, for news recommendation. The approach scores the concepts/interests based on simple term frequency technique. The same technique is employed by TUMS system developed by Tao et al. [27] to generate semantic user profiles from tweets. However, the focus of TUMS is on the semantic representation of the user profiles. The weighting scheme used by Orlandi et al. [17] to generate semantic user profiles, provides an aggregated score for concepts from multiple social networks (Facebook and Twitter) with a temporal decay. Other techniques such as tf-idf, temporal scoring [3, 17] have also been used to score interests. Although, it will be interesting to evaluate the impact of these scoring mechanisms (specifically the temporal factor) on the weights of interest categories in *HIG* (see future work in Section 5), in this work we have focused on including the most relevant categories in the *HIG*.

Wikipedia Graph has been leveraged as the base for generating *HIG* in our approach. Other approaches have utilized it for tasks such as ontology alignment [11], and clustering [28], classification of tweets [7]. Further, Spreading Activation theory used in our approach to assign interest scores has also been adapted to tasks such as document categorization [24] and search results personalization [25].

5 Conclusion and Future Work

In this paper, we have presented an approach that generates *Hierarchical Interest Graph* for Twitter users by leveraging *Wikipedia Category Graph*. We showed that the approach is practically useful in determining *Hierarchical Interests* with an extensive user study involving Twitter users with mean average precision close to approximately 90% for the *top-10 Hierarchical Interests*. We also showed the advantage of utilizing background knowledge (automatically created Wikipedia Hierarchy) for user interest identification. In future, we intend to utilize the *Hierarchical Interest Graphs* for personalizing and recommending Tweets/News articles. Further, we want to include temporal aspect to score interests where recently mentioned interests are scored higher.

References

1. Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Analyzing User Modeling on Twitter for Personalized News Recommendations. UMAP '11.
2. Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Semantic Enrichment of Twitter Posts for User Profile Construction on the Social Web. ESWC '11.
3. M-Dyaa Albakour, Craig Macdonald, and Iadh Ounis. On Sparsity and Drift for Effective Real-time Filtering in Microblogs. CIKM '13.
4. Allan M. Collins and Elizabeth F. Loftus. A spreading-activation theory of semantic processing. *Psychological Review*, 82(6):407–428, November 1975.

5. F. Crestani. Application of Spreading Activation Techniques in InformationRetrieval. *Artificial Intellence Review*.
6. Leon Derczynski, Diana Maynard, Niraj Aswani, and Kalina Bontcheva. Microblog-genre Noise and Impact on Semantic Annotation Accuracy. HT '13.
7. Yegin Genc, Yasuaki Sakamoto, and Jeffrey V. Nickerson. Discovering Context: Classifying Tweets Through a Semantic Transform Based on Wikipedia. FAC'11.
8. Daniela Godoy and Analía Amandi. Modeling User Interests by Conceptual Clustering. *Inf. Syst.*, 2006.
9. Morgan Harvey, Fabio Crestani, and Mark J. Carman. Building User Profiles from Topic Models for Personalised Search. CIKM '13.
10. Geoffrey E. Hinton. *Parallel Models of Associative Memory*. 1989.
11. Prateek Jain, Pascal Hitzler, Amit P. Sheth, Kunal Verma, and Peter Z. Yeh. Ontology Alignment for Linked Open Data. ISWC'10.
12. Pavan Kapanipathi, Fabrizio Orlandi, Amit P Sheth, and Alexandre Passant. Personalized Filtering of the Twitter Stream. In *SPIM Workshop at ISWC 2011*.
13. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. 2008.
14. Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia Spotlight: Shedding Light on the Web of Documents. I-Semantics '11.
15. Matthew Michelson and Sofus A. Macskassy. Discovering Users' Topics of Interest on Twitter: A First Look. AND '10.
16. Alan Mislove, Bimal Viswanath, Krishna P. Gummadi, and Peter Druschel. You Are Who You Know: Inferring User Profiles in Online Social Networks. WSDM '10.
17. Fabrizio Orlandi, John Breslin, and Alexandre Passant. Aggregated, Interoperable and Multi-domain User Profiles for the Social Web. I-SEMANTICS '12.
18. Simone Paolo Ponzetto and Michael Strube. Deriving a Large Scale Taxonomy from Wikipedia. AAAI '07.
19. Feng Qiu and Junghoo Cho. Automatic Identification of User Interest for Personalized Search. WWW '06.
20. M. R. Quilian. *Semantic Memory. In: M. Minski (ed.). Semantic Information Processing*. MIT Press, Cambridge, MA, 1968.
21. Daniel Ramage, Susan T. Dumais, and Daniel J. Liebling. Characterizing Microblogs with Topic Models. ICWSM '10.
22. Krishnan Ramanathan and Komal Kapoor. Creating User Profiles Using Wikipedia. In *Conceptual Modeling - ER 2009*.
23. Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. Named entity recognition in tweets: An experimental study. EMNLP '11.
24. Peter Schonhofen. Identifying Document Topics Using the Wikipedia Category Network. WI '06.
25. Ahu Sieg, Bamshad Mobasher, and Robin Burke. Web Search Personalization with Ontological User Profiles. CIKM '07.
26. Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short Text Classification in Twitter to Improve Information Filtering. SIGIR '10.
27. Ke Tao, Fabian Abel, Qi Gao, and Geert-Jan Houben. TUMS: Twitter-Based User Modeling Service. In *The Semantic Web: ESWC 2011 Workshops*.
28. Tan Xu and Douglas W Oard. Wikipedia-based Topic Clustering for Microblogs. *Proceedings of the American Society for Information Science and Technology*, 2011.
29. Yabo Xu, Ke Wang, Benyu Zhang, and Zheng Chen. Privacy-enhancing Personalized Web Search. WWW '07.