yourHistory – Semantic linking for a personalized timeline of historic events

David Graus
d.p.graus@uva.nl

Maria-Hendrike Peetz
m.h.peetz@uva.nl

Daan Odijk
d.odijk@uva.nl

Ork de Rooij
o.rooij@uva.nl

Maarten de Rijke
derijke@uva.nl

ISLA, University of Amsterdam
The Netherlands

ABSTRACT

In this paper we present yourHistory: a Facebook application that aims to generate a tailor-made, personalized timeline of historic events, by matching a semantically enriched Facebook profile to a pool of candidate historic events extracted from DBPedia. Two aspects are central to our application: (i) semantic linking technologies backed by rich open web knowledge bases for generating semantically enriched user profiles, and (ii) semantic relatedness metrics for ranking historic events to user profiles. This paper describes the development of a Facebook application that aims to be engaging for users, whilst at the same time being a source for data that can be applied to evaluating or improving the application. We describe our Wikipedia-based semantic relatedness metric for event ranking, but also the restrictions and constraints concerning privacy-sensitive and ethical matters, around data storage and user consent. Finally, we reflect on how this type of user data can be applied for evaluating or improving both the semantic linking and event ranking methods in future work.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Entity linking, Facebook, entity ranking, personalization, timeline generation

1. MOTIVATION

Today in history education, students are encouraged to study relations and coherence between events, discern patterns from global history, understand context and 'see the bigger picture'. Looking at historical timelines is interesting in this regards, because it can relate historic events to one another and provide a sense of overview and context. However, with timelines of history students are still looking at the events from a distance, whilst teachers now often teach history starting from what students know and what they are interested in. By connecting history with the lives and interests of students teachers aim to make history more tangible, attractive and accessible.

yourHistory is a Facebook application (available at `http://apps.facebook.com/yourhistory`) that aims to serve a tailor-made, personalized timeline of historic events, by leveraging a Facebook user's interests and profile. The yourHistory timeline displays *historical* events[1] side-by-side to *historic* events that are deemed relevant or interesting to the user. These can be smaller scale events that typically escape history books. By embedding the historic events that match a user's profile in the wider context of the history of the 20th century, yourHistory encourages students to explore, relate events to each other, and put them into context of time periods and their personal interests. Whether in the classroom or at home, exploring historic events and putting them into context of their own life, allows anyone who is interested in history to discover new connections and links between events, time periods and people. To generate this tailor-made timeline, yourHistory leverages rich structured data from online, openly accessible knowledge bases.

The interaction of the user with the application can provide valuable signals on the inner workings of the application. By inviting the user to explore and interact with the yourHistory timeline, and storing these interactions (in the form of clicks), we automatically aggregate data that can be later used for either evaluating or improving the application.

The rest of the paper is structured as follows. In Section 3 we describe our technical approach which involves semantic linking of Facebook user profiles, and retrieving candidate historic events from DBPedia (Section 3.1). Next, we describe our method of ranking candidate events to user profiles in Section 3.2. In Section 4 we discuss some of the technical, privacy-related and ethical challenges we faced during the development of a live Facebook application. Finally, we briefly reflect on the possibilities and uses of logging user interactions with a Facebook application like yourHistory, for evaluation and online learning purposes, in Section 5.

2. RELATED WORK

Central to the yourHistory application are two aspects: semantic linking for enriching user profiles, and semantic relatedness metrics to leverage the enriched profiles for event ranking.

2.1 Semantic linking

Semantic linking is the task of identifying and linking mentions of concepts in raw text, to their referent concepts that are described in a Knowledge Base (KB). As in so-called

[1]Taken from `http://en.wikipedia.org/wiki/Timeline_of_modern_history`

Wikification [5], Wikipedia is the typical KB of choice for semantic linking, due to its wide coverage, rich structure and content. In this case each Wikipedia page is considered to be a distinct and unique concept, and titles and anchor texts of Wikipedia pages are leveraged for lexical-matching based linking.

Semantic linking has recently seen a surge in interest; it is a focal point in evaluation campaigns such as the Text Analysis Conference Knowledge Base Population (KBP) track.[2] Consequently, it has seen a wide array of applications, from enriching microblog posts [4], supporting forensic text analysis [10], to feeding second screen applications from subtitles [8]. State of the art linking approaches typically leverage the structure of its underlying knowledge base, by considering, e.g., hyperlinks between pages, category or ontology structure for tasks such as improving disambiguation [7, 5, 3], or measuring "relatedness" between concepts [6].

2.2 Wikipedia-based semantic relatedness

Event retrieval is the task of retrieving (pages describing) events from a KB in response to an explicit query or an implicit one (such as a user's Facebook profile). We consider the event retrieval task as a ranking problem, where our aim is to rank events on descending order of "relatedness" to the user. This is in contrast to, e.g., approaches of collaborative filtering, where the ties in a social network is the main focus for recommending items.

Central to our method of matching user profiles to candidate events is the notion of semantic relatedness between (Wikipedia) concepts, or in the case of yourHistory, the relatedness between candidate historic events and user profiles. The intuition and our underlying assumption is: the more related an event is to a profile entity, the more interesting it is to the user.

To compute this semantic relatedness between events and user profiles, we combine methods that leverage Wikipedia's structure with textual similarity approaches, and aggregate for each event the semantic relatedness scores to all user profile entities. There is a rich history of leveraging Wikipedia to compute semantic similarity; an example is ESA (Explicit Semantic Analysis) [2], where Wikipedia pages are considered "topics" and an approach is employed similar to the topic modeling method of Latent Semantic Indexing (LSI). Other Wikipedia-based similarity and relatedness approaches are based on the Wikipedia graph: a representation of Wikipedia where concepts (pages) are nodes, and an edge is drawn between nodes when the corresponding pages link to or from one another. The topology of the network contains information concerning semantic similarity and relatedness; concepts that are topologically closer, more central or connected, are typically considered more similar, and semantically related. An example approach of explicitly leveraging this property is considering the overlap of the sets of neighbor nodes of two concepts [6], where concepts that share a larger portion of neighbor nodes are considered more related.

3. YOURHISTORY

In order to generate a personalized timeline of historic events we need to consider how we define this personalization, i.e. how to identify events that are interesting or relevant to a particular user given its profile.

We describe our application in four parts, first our preprocessing approach (Section 3.1), next our Wikipedia-based semantic similarity event ranking method (Section 3.2), then we present yourHistory's interface (Section 3.3), and finally, we describe some of the practical implementation details concerning data and infrastructure (Section 3.4).

3.1 Data preprocessing

In this section we describe the preprocessing procedure of: (i) generating *bag-of-concepts* user profiles by applying semantic linking (Section 3.1.1), and (ii) extracting a list of candidate historic events from DBPedia (Section 3.1.2).

3.1.1 Semantic linking user profiles

Once the user has given consent for obtaining data from their Facebook profile (described in Section 4), yourHistory receives the user's profile information in JSON-format through the Facebook API. We extract the values of several fields of the user's likes, movies, music, tv shows, bio information, and work and education history, to yield an initial *bag-of-words* user profile. The resulting *bag-of-words* user profile contains all useful textual data from the users profile and is then linked to the referent Wikipedia concepts using the *semanticizer*.[3] The resulting semantically enriched *bag-of-concepts* user profile forms the basis for yourHistory's event matching process.

To minimize noise (i.e., wrongfully linked concepts) we consider our entity linking framework's confidence score, by setting a threshold on the `SENSEPROB` weight [8]. This weight corresponds to the probability of an n-gram (from the bag-of-words user profile), to refer to a specific Wikipedia concept c. It is derived from two signals:

1. The n-gram's *link probability*: the proportion of the number of times with which n-gram is used as a link, over the total number of times this n-gram occurs in Wikipedia.

2. The n-gram's *commonness*: the proportion of the number of times n-gram is used as an anchor to a distinct Wikipedia concept c, over the number total number of times the n-gram is used as an anchor (to any Wikipedia concept c).

By representing a user profile as a *bag-of-concepts* profile, arguably we lose potentially valuable signals that could aid in the event ranking. An example is the temporal dimension; knowing where the user lived or worked at which point in time could prove useful. However, depending on the user's Facebook profile, sparsity issues (i.e. few likes) withheld us from exploring more fine-grained or detailed profiling approaches.

3.1.2 Retrieving candidate historic events from DBPedia

DBPedia is a structured representation of Wikipedia [1], and consists of concepts that are organized in a richly structured ontology.[4] This ontology allows us to smartly query for a subset of Wikipedia concepts that represent historic events from between 1900-01-01 and today. We do so by

[2] http://www.nist.gov/tac/2013/KBP/

[3] http://semanticize.uva.nl

[4] http://mappings.dbpedia.org/server/ontology/classes/

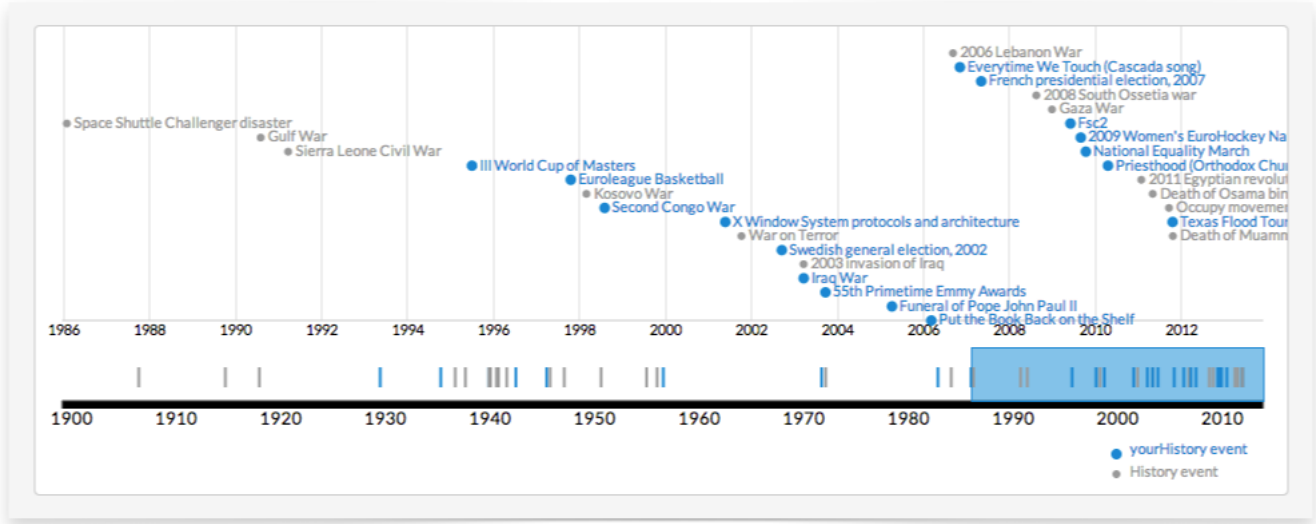

Figure 1: yourHistory's interface.

Table 1: Glossary

Symbol	Description
c	Wikipedia concept (i.e. a Wikipedia page)
U	*bag-of-concepts* user profile
E	pool of candidate events
c_{user}	user profile concept ($c \in U$)
c_{event}	candidate event concept ($c \in E$)

issuing queries to the public DBPedia SPARQL endpoint[5] for concepts that belong to the `dbpedia-owl:Event` class, concepts that have a `startDate` or `xsd:date`-property with a value between 1900-01-01 and today. At the time of research/writing, we ended up with a total of 10,272 candidate events after issuing these queries. To enable us to match events to the *bag-of-concepts* user profiles, we map the retrieved DBPedia events to their Wikipedia equivalents. Finally, we have a pool of candidate events (E) represented by Wikipedia concepts, and a *bag-of-concepts* user profile (U), similarly consisting of Wikipedia concepts. This allows us to easily compare both. See Table 1 for an overview of the terminology used in this paper.

3.2 Wikipedia-based semantic relatedness for event ranking

Here we describe how we rank events to user profiles, using our Wikipedia-based semantic relatedness-score S. The relatedness score S is calculated for each c_{user} in U to each c_{event} in E and consists of several signals. When these scores are computed for all possible (c_{user}, c_{event})-pairs, we sum the resulting values and min-max normalize them, to yield for each c_{event} a score between 0 and 1, which represents its "semantic relatedness" to U. The relatedness is based on the following signals:

1. Whether or not the profile concept and event are directly linked: 1 when either one's Wikipedia page contains a link to the other, 0 otherwise.
2. The link overlap between c_{user} and c_{event}: how many linked pages do both concepts share.
3. The textual similarity between the abstracts of the corresponding Wikipedia pages of c_{user} and c_{event}.

For each $c_{event} \in E$ we sum the scores stemming from the different signals, to each $c_{user} \in U$, and yield the final c_{event} relatedness-score. These signals are further detailed below.

3.2.1 Direct link

The first signal is a binary value, representing whether or not the profile concept c_{user} occurs in the set of outlinks of the event c_{event}, or vice versa. The intuition is that events that are directly linked to profile concepts are more (directly) related to the user's profile.

3.2.2 Link overlap

For the second signal we extract the set of outlinks of both c_{user} and c_{event} (i.e., all Wikipedia pages that are (hyper)linked in the concept's Wikipedia page). For performance reasons we create an index of virtual documents, generated by concatenating the IDs of the outlink set for each event. By then considering the (concatenated) set of profile outlinks a query, we quickly retrieve the most similar events.

3.2.3 Textual similarity

For our third and final signal, we measure the cosine similarity between the TF-IDF weighted vectors representing the abstract of the concepts' Wikipedia pages. We use the gensim topic modeling framework for this comparison [9].

The final output is a ranked list of JSON objects representing events, containing the following properties: *event_date*, *event_id*, *event_title*, *event_url*, *score*, *related_entity_id*, *related_entity_title*, and *related_entity_url*.

[5] http://dbpedia.org/sparql

3.3 yourHistory's interface

Given the final ranked list of c_{event}, we draw the events in a timeline using a D3.js timeline visualization. A screenshot of this timeline visualization is shown in Figure 1. We visually distinguish between the two types of events; shown in blue are the personalized *historic* events, and the central *historical* events are shown in gray. Clicking any of the events opens the corresponding Wikipedia page in a new window. The blue bar can be extended or reduced in size, to zoom in or out in time while maintaining a sense of context. Dragging the bar allows the user to move the frame through the timeline.

Currently, the events are shown as-is, but displaying some of the mechanics behind the event ranking (e.g., showing the user that her timeline contains event *Y because she likes band X*) might increase engagement and enable more valuable feedback. Including more information, and relaying it back to the user might increase engagement.

3.4 Data and infrastructure

In this section we describe some of the practical implementation details: the datasets we use and its infrastructure.

3.4.1 Datasets

yourHistory makes use of Wikipedia and its structured counterpart DBPedia. For semantic linking, yourHistory has access to a Wikipedia dump of March 4th, 2013. For candidate event retrieval, yourHistory queries the live DBPedia SPARQL endpoint.[6]

3.4.2 Infrastructure

yourHistory consists of three components:

- A back-end, powered by a Python Flask application
- A data repository: two MongoDB databases, one where we store user profiles, and another where users' interactions are stored
- A front-end, which consists of a web page containing an interactive timeline visualization, powered by D3.js. This web page is shown to the user inside the Facebook application.

The back-end handles communication with the Facebook API, data preprocessing, semantic linking and event scoring and ranking. It outputs a timeline of ranked events in JSON format to the D3.js JavaScript application that runs in the web interface, and draws the interactive timeline. We log the users' clicks by storing for each event clicked its unique identifier, and the time-stamp of the click.

4. SETTING UP THE APPLICATION

When setting up a live Facebook application for running user studies, we are faced with specific constraints and challenges, both technical ones and ethical, privacy-related ones. The need for real-time processing is an example of a technical constraint, while there are more privacy-related questions and issues related to data storage. These and other constraints are described in Section 4. This section consists of two parts: preliminary challenges faced concerning user's privacy and ethics when using Facebook for running online user studies (Section 4.1), as well as the technical challenges and constraints faced when working on a live, real-time Facebook application (Section 4.2).

4.1 Privacy and ethics

In the following we provide a brief guideline on how to fulfill the privacy requirements of user experiments.

4.1.1 Facebook's minimum age restriction

Ethically, collecting data of minors is questionable. Facebook handles a minimum age restriction of 13. In the case of yourHistory, the ethical review board of the University of Amsterdam required the application to be restricted to users over 18 years of age. However, providing a different birthday is an easy way to circumvent this restriction, so we can not be guaranteed to rely on the birthday information from Facebook alone. We rely on the following heuristics as a further check: if the birthday extracted from the Facebook profile identifies the user as below 18, the yourHistory application cannot be accessed. Next, the user is asked to declare she is over 18 years old in the yourHistory welcome screen, if the user here does not declare this, yourHistory will likewise not launch.

4.1.2 Using the Facebook API

Facebook provides examples on how to use the Facebook API.[7] Important aspects to note, privacy-wise, are the *scope* of the access of user data, as well as the transfer of this user data. The scope of access is set by the developer, and determines the type of user data the application can access, after the user grants the application permission to access their data. It includes such types as biography information, such as the user's birthday (*user_birthday*), relationship status (*relationship_status*), information regarding the user's religious and/or political beliefs (*user_religion_politics*), but also the user's list of friends (*user_friends*), the posts on his wall (which include messages, photos and links shared), and the user's *likes*. Finally, Facebook enforces the data transfer over a secure connection (through an SSL certified server), assuring a safe transfer of data.

4.1.3 User consent

There are two stages in requesting user consent. The first stage of user consent is initiated by Facebook when the user accesses the yourHistory application. In this stage, users are asked by Facebook if they agree with the scope of data access. We aim for a more informed consent where the user is more elaborately informed about the storage of the data as well as using the data for improving the application – currently, this is restricted to the second stage of requesting user consent.

Following guidelines provided by the ethical review board of the University of Amsterdam, we designed the next step of asking user consent. This next stage is initiated by the application itself, where we welcome the user with the dialog window shown in Figure 2. After a brief explanation of the app and its authors, the user is given a choice concerning data usage. The user may choose to have data accessed only by the authors of the application, by the authors and other researchers at the University of Amsterdam, or by nobody but the application. Concerning data storage (if the user

[6] http://dbpedia.org/sparql

[7] https://developers.facebook.com/docs/games/friend-smash/

Figure 2: yourHistory's welcome screen, and second stage of requesting user consent.

has complied), we offer the user a choice between allowing the aggregated profile to be stored for an indefinite amount of time, for 3 months at most, or for a single day. In any case, the data will never leave the University of Amsterdam. Finally, once the user declares she is over 18 years old, she is directed to yourHistory's main interface.

4.2 Technical Constraints

Due to the live and real-time nature of this application, an important constraint is that the application has to run fast and be responsive. The event ranking part in particular proved challenging in this regard: the numerous pair-wise comparisons between user profile concepts and candidate events are demanding. We addressed this primarily by optimizing the link overlap calculation procedure, the most compute-intensive operation, as described in Section 3.2.

However, numerous additional improvements towards speeding up the application could be considered, e.g., by downsizing the search space in which to make pair-wise comparisons, by clustering events and user profile concepts before computing relatedness. The clustering could be focused on the content of the concepts (e.g., cluster categorically similar concepts), or a temporal dimension or range.

5. DISCUSSION

Since evaluation of the applications' performance is still work in progress, in this section we briefly reflect on the potential of using stored user interactions for this purpose. By storing the user's interactions with the timeline (in the form of clicks on events), we have access to a valuable signal of (implicit) feedback. This signal could be used for two goals: evaluating the application, and improving the application through (online) learning. We elaborate on both applications in the sections that follow.

5.1 Evaluating yourHistory

An example application of analyzing user interactions is the evaluation of our semantic relatedness scoring function. In this case, we consider clicks on events as positive feedback. The intuition is that in yourHistory's goal of serving a personalized-timeline, inviting users to explore and learn, clicks represent the user's interest in an event, or can be considered an instantiation of the user's intent to read more: a measurable signal of user engagement.

By feeding this signal back to the scoring function, we can analyze whether the ranking correlates with clicks (i.e. do higher ranked events generate more interactions?), or whether the individual scoring functions' rankings might be more indicative (i.e. does scoring function #1 rank the more frequently clicked events higher than scoring function #2?). In the setting where we consider clicks positive feedback, and we aim to optimize the application for clicks to increase user engagement, we can additionally get insights into in what way combining various scoring functions is most effective.

5.2 Online learning

The current event ranking method is in a way "static": we rank event entities based on the individual relatedness to profile concepts. Next to hand-tuning the algorithms based on clicks as described in the previous section, the user feedback might too be applied in an online learning setting, to automatically improve the application. Here the scoring functions will be no longer used to compute a definitive score, but rather as features for a machine learning model. The positive feedback (clicks) can then be applied for training the model, enabling the application to learn to rank event entities given a user profile. Additionally, the current feedback signal can be extended towards being explicit. A possible use-case for explicit user feedback is improvement of the semantic linking component. This component now too is static: we apply a threshold on the entity linker's confidence score. A possible extension would be to ask the user to judge their generated *bag-of-concepts* user profile by removing or adding concepts that she does or does not feel associated with, and correcting wrongfully linked concepts. This explicit feedback can be used to improve the semantic linking, in an online learning setting similar to the one described above.

Acknowledgments

This research was partially supported by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr 288024 (LiMoSINe project), the Netherlands Organisation for Scientific Research (NWO) under project nrs 640.004.802, 727.011.005, 612.001.116, HOR-11-10, the Center for Creation, Content and Technology (CCCT), the QuaMerdes project funded by the CLARIN-nl program, the TROVe project funded by the CLARIAH program, the Dutch national program COMMIT, the ESF Research Network Program ELIAS, the Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW), the Netherlands eScience Center under project number 027.012.105 and the Yahoo! Faculty Research and Engagement Program.

6. REFERENCES

[1] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. Dbpedia - a

crystallization point for the web of data. *Web Semant.*, 7(3):154–165, Sept. 2009.

[2] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1606–1611, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[3] D. Graus, T. Kenter, M. Bron, E. Meij, and M. de Rijke. Context-based entity linking–university of amsterdam at tac 2012. *TAC 2012*, 2012.

[4] E. Meij, W. Weerkamp, and M. de Rijke. Adding semantics to microblog posts. In *WSDM '12*, pages 563–572, 2012.

[5] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *CIKM '07*, pages 233–242, 2007.

[6] D. Milne and I. H. Witten. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, pages 25–30. AAAI Press, July 2008.

[7] D. Milne and I. H. Witten. Learning to link with wikipedia. In *CIKM '08*, pages 509–518, 2008.

[8] D. Odijk, E. Meij, and M. de Rijke. Feeding the second screen: Semantic linking based on subtitles. In *OAIR '13*, 2013.

[9] R. Řehůřek and P. Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA.

[10] Z. Ren, D. van Dijk, D. Graus, N. van der Knaap, H. Henseler, and M. de Rijke. Semantic linking and contextualization for social forensic text analysis. In *European Intelligence and Security Informatics Conference (EISIC 2013)*, pages 96–99, August 2013.