Wiki-MID: a very large Multi-domain Interests Dataset of Twitter users with mappings to Wikipedia

Giorgia Di Tommaso, Stefano Faralli*, Giovanni Stilo, and Paola Velardi

Department of Computer Science, University of Rome,
*Unitelma-Sapienza, Italy
{ditommaso,stilo,velardi}@di.uniroma1.it
stefano.faralli@unitelmasapienza.it

Abstract. This paper presents Wiki-MID, a LOD compliant multi-domain interests dataset to train and test Recommender Systems, and the methodology to create the dataset from Twitter messages in English and Italian. Our English dataset includes an average of 90 multi-domain preferences per user on music, books, movies, celebrities, sport, politics and much more, for about half million users traced during six months in 2017. Preferences are either extracted from messages of users who use Spotify, Goodreads and other similar content sharing platforms, or induced from their "topical" friends, i.e., followees representing an interest rather than a social relation between peers. In addition, preferred items are matched with Wikipedia articles describing them. This unique feature of our dataset provides a mean to categorize preferred items, exploiting available semantic resources linked to Wikipedia such as the Wikipedia Category Graph, DBpedia, BabelNet and others.

Keywords: semantic recommenders, Twitter, Wikipedia, users' interest

Permanent URL: https://doi.org/10.6084/m9.figshare.6231326

1 Introduction

Recommender systems are widely integrated in online services to provide suggestions and personalize the on-line store for each customer. Recommenders identify preferred items for individual users based on their past behaviors or on other similar users. Popular examples are Amazon [1] and Youtube [2]. Other sites that incorporate recommendation engines include Facebook, Netflix, Goodreads, Pandora and many others.

Despite the vast amount of proposed algorithms, the evaluation of recommender systems is very difficult [3]. In particular, if the system is not operational and no real users are available, the quality of recommendations must be evaluated on existing datasets, whose number is limited and what is more, they are focused on specific domains (i.e, music, movies, etc.). Since different algorithms may be better or worse depending on the specific purpose of the recommender,

the availability of multi-domain datasets could be greatly beneficial. Unfortunately, real-life cross-domain datasets are quite scarce, mostly gathered by "big players" such as Amazon and eBay, and they not available to the research community[1].

In this paper we present a methodology for extracting from Twitter a large dataset of user preferences - that we call Wiki-MID - in multiple domains and in two languages, Italian and English. To reliably extract preferences from users' messages, we exploit popular services such as Spotify, Goodreads and others. Furthermore, we infer many other preferences from users' friendship lists, identifying those followees representing an interest rather than a peer friendship relation. In this way we learn, for any user, several interests concerning books, movies, music, actors, politics, sport, etc. The other unique feature of our dataset, in addition to multiple languages and domains, is that preferred items are matched with corresponding Wikipedia pages, thus providing the possibility to generalize users' interests exploiting available semantic resources linked to Wikipedia, such as the Wikipedia Category Graph, BabelNet, DBpedia, and others.

The paper is organized as follows: Section 2 summarizes previous research on creating datasets for recommender systems, Sections 3, 4 and 5 present the methodology to create Wiki-MID, Section 6 is dedicated to dataset statistics and evaluation, and Section 7 describes the released resource, which has been designed on top of the Semantically-Interlinked Online Communities (SIOC) core ontology. Finally, in Section 8 we provide a summary of distinctive features of our resource and some directions for future work.

2 Related work

Recommender systems are based on one of three basic approaches [4]: *collaborative filtering* [5] generates recommendations collecting preferences of many users, *content-based filtering* [6] suggests items similar to those already chosen by the users, and *knowledge-based* recommendation [7] identifies a semantic correlation between user's preferences and existing items. Hybrid approaches are also widely adopted, e.g., [8]. All approaches share the need of sufficiently large datasets to learn preferences and to evaluate the system, a problem that is one of the main obstacles to a wider diffusion of recommenders [9], since only a small number of researchers can access real users data, due to privacy issues.

To overcome the lack of datasets, challenges as RecSys have been launched[2], and dedicated web sites have been created (e.g., SNAP[3] or Kaggle[4]), where researchers can upload their datasets and make them available to the community. However it is still difficult to find appropriate data for novel types of recommenders, as the majority is focused on a single topic, like music [10], [11],) food ([12], [13]), travel ([14], [15]) and more [16]. Furthermore, while a small number

[1] https://recsys.acm.org/wp-content/uploads/2014/10/recsys2014-tutorial-cross_domain.pdf
[2] https://recsys.acm.org/
[3] http://snap.stanford.edu/data
[4] https://www.kaggle.com/datasets/?sortBy=hottest&group=all

of large datasets are available, such as Movielens [17], Million song dataset [18] and Netflix Prize Dataset [19], many others are quite small and based on very focused experiments.
Concerning the source of data for extracting preferences, social networks are often used (mainly Twitter, Facebook, Google+, LinkedIn or a combination of sources, such as in [20]), since their content is freely available with more or less severe restrictions. The interested reader can refer to [21] for a detailed survey of methods adopted in literature to collect social data for the purpose of inferring and enhancing users' interests profiles. Preferences are induced from users' profiles (e.g, [22]), authoritative (topical) friendship relations [23], followee biographies [24], and messages ([25], [26], [27] and many others).
Data extraction from Twitter messages is a popular strategy, however, it is also computationally expensive and error-prone, since it requires natural language processing techniques to analyze the text. To overcome this difficulty, a number of studies exploited platforms (e.g., Youtube, Spotify) that integrate among their services the ability to post the user's personal content on the most popular social network sites, such as movies that users are watching. Sharing this information is done in a simple and predefined way. Depending on the social network chosen, the content, for example a Youtube video, will be shared with a pre-formatted message formed by the video name, a link, a self-generated text and, if provided, a numerical rating (eg. "How It's Made: Bread" `https://youtu.be/3UjUWfwWAC4` via YouTube). The message can also be enriched and personalized by the user. In [25] these types of messages are extracted from Twitter, to detect music interests. The dataset is based on 100,000,000 tweets with the #nowplaying main tag. Tweets are extracted via Twitter APIs over 3-years and next, MusicBrainz and Spotify are used to add more details. Other studies extract data about music [27] or sport [28] events. However, all the datasets generated in this way concern only one domain of interest.
To the best of our knowledge, the only really multi-domain dataset is presented in [29], where pre-structured tweets about three domains - movies, books and video-clips - are extracted respectively from IMDb (Internet Movies Database), Youtube and Goodreads. With respect to this work, we collect a much wider number of interests, since in addition to pre-formatted messages based on a number of available services, we reliably extract many additional types of interests exploiting users' followees lists. Furthermore, as shown in Section 7, we collected many interest *types* for each user, while the dataset released in [29] includes only 7 users with at least 3 types of interests.

3 Workflow

This section summarizes the data sources and workflow to create the Wiki-MID multi-domain resource. We extract preferences (with unary ratings) from a user's messages and from his/her friendship list, identifying those followees who represent an interest rather than a peer friendship relationship. The process is in three steps:

1. *Extracting interests from users' textual communications.* Using textual features extracted from users' communications, profiles or lists seems a natural way for modeling their interests. However, this information source has several drawbacks when applied to large data streams, such as the set of Twitter users. First, it is computationally very demanding to process millions of daily tweets in real time; secondly, the extraction process is error prone, given the highly ungrammatical nature of micro-blogs. To reliably extract preferences from Twitter users' messages, in line with other works surveyed in Section 2, we use a number of available services, described hereafter, that allow to share activities and preferences in different domains - movies, books etc. - using pre-formatted expressions (e.g, for Spotify: #NowPlaying) followed by the url of a web site, from which we can extract information without errors. The drawback is that a relatively small number of users access these services and in addition, preferences are extracted only in few domains.
2. *Extracting interests from users' friendship lists.* In [30] the authors argue that users' interests can also be implicitly represented by the authoritative (*topical*) friends they are linked to. This information is available in users' profiles and does not require additional textual processing. Furthermore, interests inferred from topical friends are less volatile since, as shown in [31], "common" users tend to be rather stable in their relationships. Topical friends are therefore both relatively stable and readily accessible indicators of a user's interest. Another advantage is that average Twitter users have hundreds of followees, many of which, rather than genuine friends, are indicators of a variety of interests in different domains, such as entertainment, sport, art and culture, politics, etc.
3. *Mapping interests onto Wikipedia pages.* The final step is to associate each interest, either extracted from messages or inferred from friendship relations, with a corresponding Wikipedia page, e.g.:
   @nytimes ⇒ WIKI:EN:The_New_York_Times
   (in this example, @nytimes is a Twitter account extracted from a user's friendship list). Although not all interests can be mapped on Wikipedia, our experiments show that this is possible in a large number of cases, since Wikipedia articles are created almost in real-time in correspondence with virtually any popular entity, either book, or song, actor, event, etc.

We applied this workflow to two Twitter streams in two languages, English and Italian, as explained in the next Sections.

4 Extraction of users' interests

4.1 Extracting interests from messages

Everyday a huge number of people uses on-line platforms (eg. Yelp, Foursquare, Spotify, etc.) that allow to share activities and preferences on different domains on a social network in a standard way. Among the most popular services accessed by Twitter users, we selected those providing pre-formatted messages:

- **Spotify:** Spotify is a music service offering on-demand streaming of music, both desktop and mobile. Users can also create playlists, share and edit them in collaboration with other users. In addition to accessing the Spotify web site, users can retrieve additional information such as the record label, song releases, date of release etc.. Since 2014, Spotify is widely used in America, Europe and Australia. Spotify is among the services allowing to generate self-generated content shares in Twitter. An example of these tweets is: "#NowPlaying The Sound Of Silence by Disturbed `https://t.co/d8Sib5EDVf`". The standard form of these tweets is:
  `#NowPlaying <title> by <artist > <URL>`
  By filtering the tweets stream and using Twitter APIs for hashtag detection, we generated a stream of all the users who listened music using Spotify.
- **Goodreads and aNobii:** Similarly to Spotify, a number of platforms allows to share opinions and reviews on books. In these platforms, users can share both titles and ratings. Similarly to Spotify, generated tweets have a predefined structure and point to an URL. In the book domain, we use Goodreads (10 million users and 300 million books in the database) and for Italian, the more popular aNobii service.
- **IMDb and TVShowTime:** In the domain of movies, currently there are no dominant services. Popular platforms in this area are Flixter, themoviedb.org and iCheckMovies. However, many of these platforms use the IMDb database, owned by Amazon, which handles information about movies, actors, directors, TV shows, and video games. We also use the TvShowTime service for Italian users.

In order to extract users' preferences from these services, we first collect in a Twitter stream TS all messages including a hashtag related to one of the above mentioned services (#NowPlaying, #IMDb ..). Then, we extract from TS the music, movie and book preferences for the set of users U who accessed these services. Unlike [29], we avoid parsing tweets using specific regular expressions, since users are free to insert additional text in the pre-formatted message. Rather, in line with [32], we exploit an element that most pre-formatted tweets have: the URL, e.g., #NowPlaying High by James Blunt https://t.co/7EiepE2Bvz
Accordingly, we collect all tweets containing the selected hashtags and discard those which do not include an URL. The reason for extracting the information from the URL (which is computationally more demanding) rather than from the tweet itself is twofold: i) Tweets can be ambiguous or malformed, and furthermore, users can insert additional text in the pre-formatted message, e.g, "*#NowPlaying Marty*. This guy is amazing.♫`http://t.co/jwxvLiNenW`". Scraping the html page at the URL address ensures that we extract data *without errors*, even for complex items such as book and movie titles; ii) The URL includes additional information (e.g., not only the title of a song, but also the singer and the record label), which provide us a context to reliably match the extracted entity (song, book, movie) with a Wikipedia article, as detailed in Section 5.1. Since the URL in tweets is a short URL, we first extend the original URL so that all URLs belonging to a given platform can be identified (for

example, `https://t.co/oShYDc6DeL` → `http://spoti.fi/2cTPn0U`). Next, we access the web site and scrape its content. For each platform we obtain the following data:
Music: <Title, Author (eg. singer, band)>
Books: <Title, Author>
Movie: <Title, Year of production, Type (eg. movie, tv series)>

4.2 Extracting interests from users' "topical" friends

In addition to preferences extracted from users' messages, we also induce interests from their *topical friends*, a notion that we first introduced in [23]. We denote as topical friends those Twitter accounts in a user's followees list representing popular entities (celebrities, products, locations, events ...). For example, if a user follows @David_Lynch, this means that he/she likes his movies, rather than being a genuine friend of the director. There are several clues to identify topical friends in a friendship list: first, topical relations are mostly *not reciprocated*, second, popular users have a high in-degree. However, these two clues alone do not allow to distinguish e.g., bloggers or very social users from truly popular entities. To learn a classification model to distinguish between topical and peer friends, we first collected a network of Verified Twitter Accounts. Verified accounts [5] are authentic accounts of public interest. We started from a set of seed verified contemporary accounts in 2016, and we then crawled the network following only verified friends, until no more verified accounts could be found. This left us with a network of 107,018 accounts of verified contemporary users (V), representing a "training set" to identify authoritative users' profiles. To learn a model of authoritativeness, we used the set V and a random balanced set of $\neg V$ users. For each account in V and $\neg V$, we extracted three structural features (in degree, out degree and their ratio) and one binary textual feature (presence in the user's account profile of role words such as *singer, artist, musicians, writer..*). Then, we used 80% of these accounts to train a SVM classifier with Laplacian kernel and the remaining 20% for testing with cross-validation, obtaining a total accuracy of 0.88 (true positive rate 0.95 and true negative rate 0.82).

Next, from the set U of users in our Twitter datasets (separately for the English and Italian streams), we collected the set F of Twitter accounts such that, for any $f \in F$ there is at least one $u \in U$ such that u follows f. The previously learned classifier was used to select a subset $F_t \subseteq F$ of *authoritative* users representing "candidate" topical friends.
Finally, an additional filtering step is applied to identify "true" topical friends in F_t, i.e., genuine users' *interests*, which consists in determining which members of the set F_t have a matching Wikipedia page. This step is described in Section 5. The intuition is that, if one such match exists, the entity to which the Twitter account belongs is indeed "topical"[6]. Although this filtering step may affect the recall of the method, it provides high accuracy, as demonstrated in Section 6.

[5] `https://developer.twitter.com/en/docs/api-reference-index`
[6] we do not directly attempt a match of all $f \in F$ with Wikipedia, since it is very computationally demanding and has a reduced precision.

5 Mapping interests to Wikipages

The last step of our methodology consists in mapping the collected users' interests to Wikipages. This step has both the advantage of improving the precision of detected users' interests, and providing a mean to categorize them. We use different mapping methodologies for interests extracted from messages and those induced from users' friendship lists.

5.1 Mapping movies, songs and books

Mapping interests extracted from users' messages to Wikipedia pages is a very reliable process, given the additional contextual information extracted from the URL (see previous Section 4.1). Wikipedia mapping is obtained by a cascade of weighted boolean query on a Lucene Index, as in the example below, used to search the Wikipage of an item:
$< TITLE \in WikiTitle >^{w_1} \wedge < AUTHOR \in WikiGloss >^{w_2} \wedge((< WORDS \in WikiTitle >^{w_3} \vee < AUTHOR \in WikiTitle >^{w_4} \vee \neg(< WORDS \in WikiTitle >^{w_3} \vee < AUTHOR \in WikiTitle >^{w_4} \vee < WORDS \in WikiText >^{w_5}))$
$\vee < WORDS \in WikiText >^{w_5}) < WORDS >$ for music = {"song"} $< WORDS >$ for books = {"books", "novel", "saga" ...} $< WORDS >$ for movie = {"film", "series", "TV series", "episode" ...}
where w_i is a weight assigned to a query. When the page doesn't exist or is not available, we search the page of the item's author, using similar queries.

5.2 Mapping topical friends

Matching interests extracted from a user's friendship list with corresponding Wikipedia pages is far more complex, because of homonymy, polysemy and ambiguity. Furthermore, the information included in a user's Twitter profile is very sketchy and in some case misleading, therefore it may not provide sufficient context to detect a similarity with the correspondent Wikipedia article. For example, Bill Gate's description field[7] in his Twitter profile is: "*Sharing things I'm learning through my foundation work and other interests...*" which has little in common with his Wikipedia page: "*William Henry Gates III (born October 28, 1955) is an American business magnate, investor, author, philanthropist, humanitarian and co-founder of the Microsoft Corporation along with Paul Allen.*"
We note that other studies have considered this task. For example, the authors in [33]) use an heuristics based on the overlap coefficient of last 20 topical followees' tweets and the Wikipedia article summary, which is rather data demanding. In [34] the authors use a methodology which is similar to the one we firstly presented in [23], based on a comparison between Twitter description fields and the content of a Wikipage. As previously noted (see the Bill Gates example), this might not be sufficient in many cases. In the present work, to reliably assign a Wikipedia page to a large fragment of users in the set F_t of $U's$ authoritative

[7] as retrieved on January 2018

friends, we use an *ensemble* of methods, with adjudication by majority voting. The methodology is described in what follows.

1. Task Description and data - Given a set $F_t = \{f_1, f_2, ..., f_n\}$ of candidate "topical" Twitter profiles and a set of Wikipages $W = \{w_1, w_2, ..., w_m\}$ we define a mapping function $M : F_t \Rightarrow W \cup \{\lambda\}$ where the value of the function M for a given Twitter profile f_i is a Wikipage w_j, which is the corresponding Wikipage of the entity having the twitter profile f_i or λ, where λ means "no match".

We define an ensemble of three mappers exploiting the information included in Twitter profiles and in DBpedia entities associated to Wikipedia.
Profiles of Twitter users provide, among the others, the following information:

- **profile address**: e.g., `https://twitter.com/katyperry`
- **user_ID**: a numeric value to uniquely identify a user (not visible on the rendered web page);
- **screen_name** a string that can be used to refer to a user when posting a message (e.g. @katyperry);
- **name** the extensive name of the owner of the profile (e.g. "Katy Perry");
- **url**: the link to a profile-relevant homepage (e.g. "katyperry.com"). Only a fragment of profiles have an URL to a homepage.
- **description**: a short description to describe the user and welcome profile visitors.

Furthermore, from each wikipage w_j (e.g., Figure 1, upper right, shows the Wikipedia page of the singer "Katy Perry") it is possible - thanks to DBpedia - to collect additional information, here is a small subset:

- **title**: the title of the page (e.g. "Katy Perry");
- **content**: the textual content of the page;
- **homepage**: a property (collected and included on DBpedia from infoboxes) which (*when present*) links to a web page (homepage) related to the main entity described in w_j (e.g., "katyperry.com")
- **links extracted from the homepage**: are those links included on the source html of the above mentioned homepage, e.g., in the html of the webpage at `katyperry.com` we find: `https://facebook.com/katyperry`,`https://twitter.com/katyperry`, ...

2. Mapping methods - We rely on an ensemble of three different methodologies (M_1, M_2 and M_3) of association between the set F_t of Twitter profiles and Wikipages. The first is based on text mining and structural properties of the social network, the other two are based on finding direct correspondences between the field *url* in a Twitter profile and the property *homepage* in a DBpedia entity.

1. M_1 **- Context Based mapping**: We use the methodology that we first presented in [35], summarized in what follows:
   a) Selection of candidate senses: For any f_i in F_t, find a (possibly empty) list

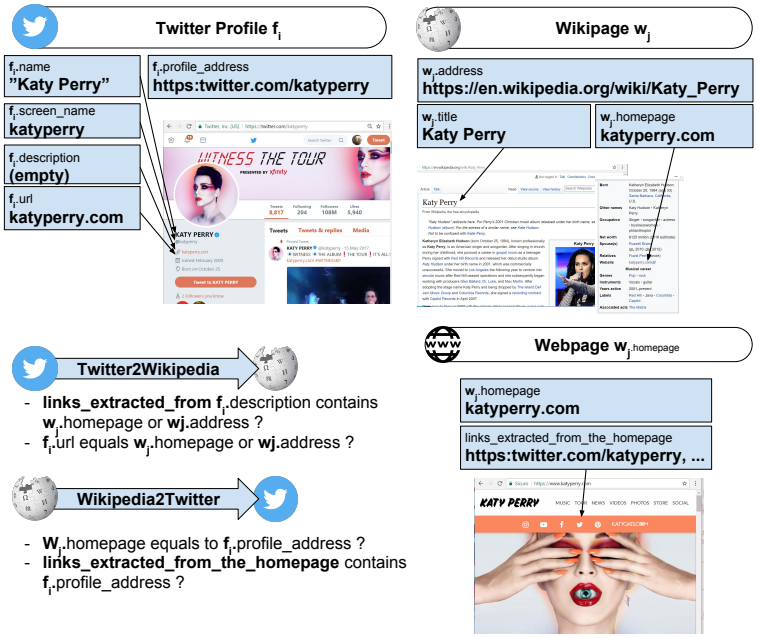

Fig. 1. Example of Twitter2Wikipedia and Wikipedia2Twitter mapping

of *candidate wikipages*, using BabelNet [36] synonym sets (in BabelNet, each "BabelSynset" points to a unique Wikipedia entry). For example, *@kathyperry* has candidates *Katy_Perry* and *Katy_Perry_discography*, but there are cases with dozens of candidates (e.g., `https://en.wikipedia.org/wiki/John_Williams_(disambiguation)`);
b) BoW Disambiguation: Compute the bag-of-words (BoW) similarity between the user description in f_i's Twitter account and each candidate wikipage. The BoW representation for each wikipage is obtained from its associated BabelNet relations (relations are described in [37]);
c) Structural Similarity: If no Wikipages can be found with a sufficient level of similarity (as for the previous example of Bill Gates description field), select from f_i's friendship list those friends already mapped to a wikipage -if any- and compute the similarity between those wikipages and candidate wikipages. For example, to correctly map the Twitter account of Bill Gates to Wikipedia, profile information of the following Twitter users in his friendship list are used: Paul Allen, Melinda Gates, TechCrunch, Microsoft Foundation, and more. Note that Paul Allen is explicitly mentioned in the first sentence of Bill Gate's wikipage.

2. M_2 **- Twitter2Wikipedia**: as sketched in Figure 1, we first collect a set of URL from a given profile f_i including: the link (if any) in the field *url* and all the links extracted from the profile *description* filed (In the example of Figure 1, since the profile description is empty, we collect only

the link `katyperry.com`). Second, we search a Wikipage w_j (if any) for which one of the links collected for f_i (in our example we collected the link `katyperry.com`) matches with the link provided in the *homepage* property, (in our example the Wikipage with title "Katy Perry" has a property *homepage* whose value matches exactly the link `katyperrry.com`), or directly with the address of the page itself (e.g. `https://en.wikipedia.org/wiki/Katy_Perry`). Note that this mapping method is error prone: for example, from the Twitter profile of Paul Gilmour we extract the following url: `skysports.com` matching with the homepage property of the following wikipage: `https://en.wikipedia.org/wiki/Sky_Sports`. Although related, this is not Paul Gilmour's page `https://en.wikipedia.org/wiki/Paul_Gilmour`.

3. M_3 **- Wikipedia2Twitter**: M_3 is symmetric to M_2. As shown in Figure 1, we map a given f_i to a Wikipage w_j if the *homepage* property, or one of the links extracted from the source html of the homepage in w_j, matches the Twitter *profile address*. Like for Twitter2Wikipedia, this mapping method is error prone.

For each of the above three approaches we add three additional mapping functions ESM_1, ESM_2, ESM_3 where each mapping function is defined as :

$$ESM_k(t_i) = \begin{cases} w_j, & \text{if } M_k(t_i) = w_j \text{ and } t_i.name = w_j.title \\ \lambda, & \text{otherwise} \end{cases}$$

In other words, ESM_k "reinforces" the result of M_k if the *name* field in the Twitter profile perfectly matches with the *title* of the Wikipedia page. Note that this is often not the case, as for *@realDonaldTrump*.

3. Ensemble Voting - For a given Twitter profile f_i the ensemble voting mechanism selects the Wikipage w_j for which there is maximum agreement among the 6 mapping functions (M_1,M_2,M_3,ESM_1,ESM_2,ESM_3), and there are at least 2 M_j, M_k in agreement ($j \neq k$). The threshold 2 has been empirically selected to obtain the best compromise between number of mapped interests and precision, as detailed in Section 6.

6 Wiki-MID statistics and evaluation

The outlined process has been applied to two streams of Twitter data, in English and Italian, extracted during 6 months (April-September 2017) using Twitter APIs. We collected the maximum allowed Twitter traffic of English users mentioning service-related hashtags (e.g., #NowPlaying for Spotify), and the full stream of messages in Italian, since they do not exceed the maximum. As a final result, we obtained for a large number of users a variety of interests along with their corresponding Wikipedia pages. An excerpt of a Twitter user's interests is shown in Table 1. In the example, we selected two interests from each of the four sources from which they have been induced: IMDb (movies), Goodreads (books),

USER ID:787930***		
Source	**Interest**	**Wikipage**
IMDb	Eyes Wide Open - 2009 - movie	WIKI:EN:Eyes_Wide_Open_(2009_film)
	Okja - 2017 - movie	WIKI:EN:Okja
Goodreads	The Beautifull Cassandra - Jane Austen	WIKI:EN:Jane_Austen
	The Beach - Alex Garland	WIKI:EN:The_Beach_(novel)
Spotify	I Don't Know What I Can Save You From - Kings of Convenience	WIKI:EN:Kings_of_Convenience!
	Nothing Matters When We're Dancing - **The Magnetic Fields**	WIKI:EN:The_Magnetic _Fields
Topical friends	**@IMDb**	WIKI:EN:IMDb
	@UNICEF_uk	WIKI:EN:UNICEF_UK
	@TheMagFields	WIKI:EN:The_Magnetic _Fields
	@BarackObama	WIKI:EN:Barack_Obama
	@Spotify	WIKI:EN:Spotify

Table 1. Excerpt of a Twitter user's interests

Spotify (music) and the user's topical friends. Although a detailed analytics of interest categories is deferred to further studies, the example shows the common trend that a user's interests, either extracted from his/her messages or from topical friends, are strongly related, and in same case identical. For example, the user in Table 1 frequently accesses the IMDb and Spotify services, and he/she is also a follower of the IMDb and Spotify Twitter accounts. Furthermore, his/her interest in the band The Magnetic Field emerges from both source types.

Overall, we followed 444,744 English-speaking and 25,135 italian-speaking users (the set U) who accessed at least one of the services mentioned in Section 4.1. Tables 2 and 3 show general statistics of interests extracted from users' messages respectively, for English and Italian speaking user. In the English dataset we crawled more than 20M tweets from these users, of which, about 2.7M could be associated to the URL of a corresponding book, movie or song. On average, we collected 6 interests per user. What is more, several users have interests in at least two of the three domains. Figure 2 compares the Venn diagram of interest types in our dataset (left) with that reported in [29] (right), to demonstrate the superior coverage of our dataset, even when considering only preferences extracted from users' messages. The last line of Tables 2 and 3 (precision) shows that the methodology to extract and map preferences from messages is very reliable. We evaluated the precision (two judges with adjudication) on a randomly selected balanced sample of 1200 songs, books, and movies in English, obtaining a precision of 96% with a k-Fleiss Inter Annotator Agreement (IAA) of 1[8]. For the Italian dataset, we evaluated 750 songs, books, and movies, obtaining a precision of 98%, and a k-Fleiss of 0.97.

The number and variety of extracted preferences is mostly determined by the interests induced from users' topical friends, as shown in Table 4 (Table 5 for the Italian dataset). The average number of interests induced for each user is as high as 82, and the distribution is shown in Figure 3, left (English stream), and right (Italian stream). Figure 3 (left) shows, e.g., that there are 100,000 users in U with ≥ 100 interests induced from their topical friends. As far as the topical interests mapping performance is concerned, in [35] we estimated that inducing interests from topical friends and subsequent mapping to Wikipedia with mapping method $M1$ has an accuracy of 84%. Since our aim in this work

[8] The evaluation is rather straightforward, as readers may verify inspecting the released dataset and mappings.

message-based interests ($\|U\|$=444,744 English speaking users)				
	Music	**Books**	**Movie**	**Total**
Platform:	*Spotify*	*Goodreads*	*IMDb*	*All*
#crawled tweets (tweets with selected hashtags)	19,941,046	693,975	97,772	20,732,793
#cleaned tweets (tweets fro which an URL was extracted)	2,519,166	139,882	88,355	2,747,403
# of unique interests with a mapping to a Wikipage	253,311	20,710	8,282	282,303
average #interests per user	6	8	6	6
average #users per interest	7	3	7	6
precision of Wikipedia mapping (on 3 samples of 400 items each)	94%	96%	97%	**96%**

Table 2. 6-months (April-September 2017) statistics on **message-based** interests extracted from English-speaking users

message-based interests ($\|U\|$= 25,135 Italian speaking users)				
	Music	**Books**	**Movie**	**Total**
platform	*Spotify*	*ANobii*	*IMDb* *TVShowTime*	*All*
#crawled tweets (tweets with selected hashtags)	273,256	12,198	2,229	287,683
#cleaned tweets (tweets for which an URL was extracted)	70,330	12,193	2,119	84,642
# of unique interests with a mapping to a Wikipage	9,926	4,690	279	14,895
average #interests per user	3	9	7	6
average #users per interest	5	2	5	4
precision of Wikipedia mapping (on 3 samples of 250 items each)	96%	98%	100%	**98%**

Table 3. 6-months (April-September 2017) statistics on **message-based** interests extracted from Italian-speaking users

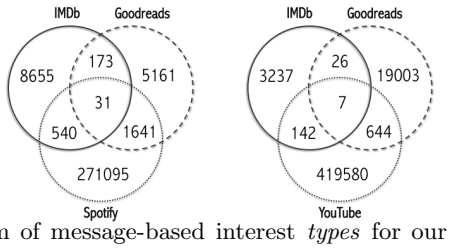

Fig. 2. Venn Diagram of message-based interest *types* for our English dataset (left) and the dataset in Dooms et al. (right)

is to generate a highly accurate dataset, first, we used an ensemble of methods, as detailed in Section 5.2, and furthermore, we considered only the subset F'_t in F_t with indegree (with respect to our population U) higher than 40. In fact, we noted that less popular topical friends may still include bloggers or Twitter

Interests induced from topical friends ($\|U\|$=444,744 English speaking users)	
# of topical friends F'_t with indegree $\geq$ **40** in U	**409,743**
# of unique interests with a mapping to a Wikipage	58,789
average #interests per user	82
precision of Wikipedia mapping (tested on a sample of 1,250 items in F'_t)	90%

Table 4. 6-months (April-September 2017) statistics on interests induced from **topical friends** of English-speaking users

Interests induced from topical friends ($\|U\|$=25,135 Italian speaking users)	
# of topical friends F'_t with indegree $\geq$ **42** in U	**29,075**
# of unique interests with a mapping to a Wikipage	4,580
average #interests per user	41.96
precision of Wikipedia mapping (tested on a sample of 1,250 items in F'_t)	90%

Table 5. 6-months (April-September 2017) statistics on interests induced from **topical friends** of Italian-speaking users

users for which, despite some popularity, a Wikipage does not exist. In these cases, our methodology may suggest false positives. When applying the indegree filter, the precision - manually evaluated with adjudication on 1250 accounts randomly chosen in this restricted population F'_t - is as high as 90%, as shown in the last line of Tables 4 and 5. The k-Fleiss IAA are 0.95 and 0.92, respectively. We remark that we are not concerned here with measuring the recall, since the objective is to release a dataset with *high precision and high coverage*, in terms of number of interests per user, over the considered populations. To this end, the indegree threshold 40 was selected upon repeated experiments to obtain the best trade-off between the distribution of interests in the population U and precision of Wikipedia mapping.

Concerning coverage, when merging the two sources of information, our English dataset includes an average of 90 interests per user for about 450k users, and a total of $282,303 + 58,789 = 341,092$ unique interests in a large variety of domains. As a comparison, even when considering single domains, the largest available datasets[9], like MovieLens and Bookcrossing, do not exceed 150,000 users and 250,000 items, with a much lower density in terms of interests per user -although these resources provide ranked preferences rather than unary, as in WikiMED. Even the popular Million Songs Dataset Challenge [18] consists of a larger set of users (1.2 million users) but a comparable number of unique interests in a single domain (380,000 songs). To the best of our knowledge, *this is the largest freely available multi-domain interest dataset reported in literature, and furthermore, we provide the unique feature of a reliable mapping to Wikipedia.*

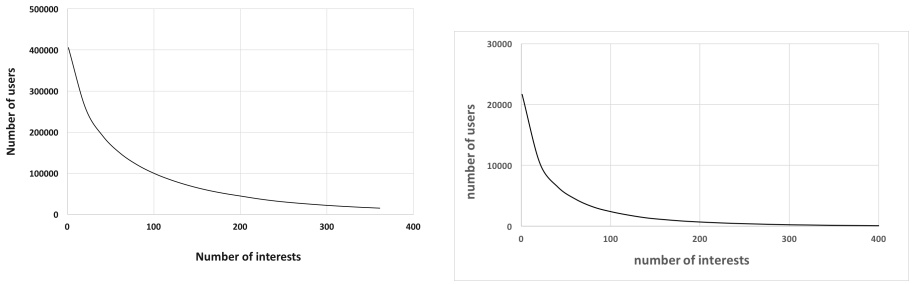

Fig. 3. Distribution of interests induced from users' topical friends: English dataset (left) and Italian dataset (right).

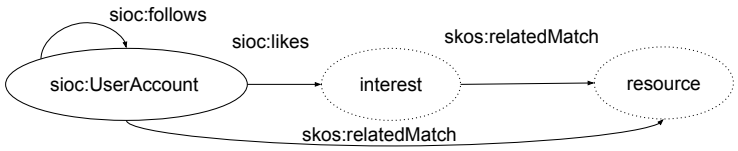

Fig. 4. The data model adopted for the design of our resource.

7 The Wiki-MID resource

Our resource is designed on top of the Semantically-Interlinked Online Communities (SIOC) core ontology.[10] The SIOC ontology favors the inclusion of data mined from social networks communities into the Linked Open Data (LOD) cloud. As shown in Figure 4 we represent Twitter users as instances of the SIOC *UserAccount* class. Topical users and message based user interests are then associated, through the usage of the Simple Knowledge Organization System Namespace Document (SKOS)[11] predicate *relatedMatch*, to a corresponding Wikipedia page as a result of our automated mapping methodology. We release at `http://wikimid.tweets.di.uniroma1.it/wikimid/` both the dataset and the related software under Creative Commons Attribution-Non Commercial-Share Alike 4.0 License.

8 Concluding Remarks

In this paper we presented Wiki-MID, a LOD-compliant resource that captures Twitter users' interests in multiple domains. With respect to other available datasets for Recommender Systems, our resource has several unique features:
1) users' interests are induced from their messages and authoritative ("topical")

[9] `https://www.kdnuggets.com/2016/02/nine-datasets-investigating-recommender-systems.html`
[10] `http://rdfs.org/sioc/spec/sioc.html`
[11] `http://www.w3.org/2004/02/skos/core.html`

friends, and associated with corresponding Wikipedia articles, thus providing a mean to derive a semantic categorization of interests through the exploitation of available resources linked to Wikipedia, such as the Wikipedia Category Graph, DBPedia, BabelNet, and others;
2) for every user, we are hence able to extract in two languages (English and Italian) a variety of interests in multiple categories, such as art, science, entertainment, politics, sport and more;
3) the dimension of the dataset is comparable with the largest single-domain interest datasets in literature, and the average number of multi-domain interests per user is far more large than other multi-domain datasets.

Further note, as shown in Section 6, that extracting interests from messages and topical friends, and subsequent mapping to Wikipedia, is a very reliable process (4% error rate for message-induced interests and 10% for friendship-induced). In addition, the availability of semantic resources linked to Wikipedia offers the possibility to identify for each user the "dominant" interest *categories*, on which recommenders could rely when suggesting new items. We leave to future research the exploitation of these features.

Acknowledgments. This work has been supported by the IBM Faculty Award #2305895190 and by the MIUR under grant "Dipartimenti di eccellenza 2018-2022" of the Department of Computer Science of Sapienza University.

References

1. Linden, G., Smith, B., York, J.: Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet computing **7**(1) (2003) 76–80
2. Davidson, J., Liebald, B., Liu, J., et al.: The youtube video recommendation system. In: Proc. of the 4th RecSys, ACM (2010) 293–296
3. Fouss, F., Saerens, M.: Evaluating performance of recommender systems: An experimental comparison. In: WI-IAT 2008. Int. Conf. on. Volume 1., IEEE 735–738
4. Felfernig, A., Jeran, M., Ninaus, G., Reinfrank, F., Reiterer, S., Stettinger, M.: Basic approaches in recommendation systems. In: RSSE. Springer (2014) 15–37
5. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: The adaptive web. Springer (2007) 291–324
6. Pazzani, M.J., Billsus, D.: Content-based recommendation systems. In: The adaptive web. Springer (2007) 325–341
7. Trewin, S.: Knowledge-based recommender systems. Encyclopedia of library and information science **69**(Supplement 32) (2000) 180
8. Burke, R.: Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction **12**(4) (2002) 331–370
9. Gunawardana, A., Shani, G.: A survey of accuracy evaluation metrics of recommendation tasks. JMLR **10**(Dec) (2009) 2935–2962
10. Dror, G., Koenigstein, N., Koren, Y., Weimer, M.: The yahoo! music dataset and kdd-cup'11. In: Proc. of KDD Cup 2011. (2012) 3–18
11. Shepitsen, A., Gemmell, J., Mobasher, B., Burke, R.: Personalized recommendation in social tagging systems using hierarchical clustering. In: RecSys, 2008, ACM

12. Kamishima, T., Akaho, S.: Nantonac collaborative filtering: A model-based approach. In: Proc. of the 4th RecSys, ACM (2010) 273–276
13. Sawant, S., Pai, G.: Yelp food recommendation system (2013)
14. Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis on review text data: a rating regression approach. In: Proc. of the 16th ACM SIGKDD. (2010) 783–792
15. Mavalankar, A.A., et al.: Hotel recommendation system. Internal Report (2017)
16. Çano, E., Morisio, M.: Characterization of public datasets for recommender systems. In: RTSI, IEEE 1st International Forum on, IEEE (2015) 249–257
17. Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. TiiS16
18. McFee, B., Bertin-Mahieux, T., Ellis, D.P., Lanckriet, G.R.: The million song dataset challenge. In: Proc. of the 21st WWW, ACM (2012) 909–916
19. Bennett, J., Lanning, S., et al.: The netflix prize. In: Proc. of KDD, NY (2007)
20. Yan, M., Sang, J., Xu, C.: Mining cross-network association for youtube video promotion. In: Proc. of the 22nd ACMMM, ACM (2014) 557–566
21. Piao, G., Breslin, J.G.: Inferring user interests in microblogging social networks: A survey. In: arXiv:1712.07691v3. (2017)
22. Chaabane, A., Acs, G., Kaafar, M.A., et al.: You are what you like! information leakage through users? interests. In: Proc. of the 19th NDSS Symposium. (2012)
23. Faralli, S., Stilo, G., Velardi, P.: Large scale homophily analysis in twitter using a twixonomy. In: Proc. of 24th IJCAI, 2015, Buenos Aires,Jul. 25-31, 2015. (2015) 2334–2340
24. Piao, G., Breslin, J.G.: Inferring user interests for passive users on twitter by leveraging followee biographies. In: Proc. of ECIR. (2017)
25. Pichl, M., Zangerle, E., Specht, G.: # nowplaying on# spotify: Leveraging spotify information on twitter for artist recommendations. In: ICWE. (2015) 163–174
26. P., K., P., J., C., V., A., S.: User interests identification on twitter using a hierarchical knowledge base. LNCS: The Semantic Web: Trends and Challenges (2014)
27. Schinas, E., Papadopoulos, S., Diplaris, S., Kompatsiaris, Y., Mass, Y., Herzig, J., Boudakidis, L.: Eventsense: Capturing the pulse of large-scale events by mining social media streams. In: Proc. of the 17th PCI, ACM (2013) 17–24
28. Nichols, J., Mahmud, J., Drews, C.: Summarizing sporting events using twitter. In: Proc. of the 2012 Int. Conf. on Intelligent User Interfaces, ACM (2012) 189–198
29. Dooms, S., De Pessemier, T., Martens, L.: Mining cross-domain rating datasets from structured data on twitter. In: Proc. of the 23rd WWW, ACM (2014) 621–624
30. Barbieri, N., Bonchi, F., Manco, G.: Who to follow and why: link prediction with explanations. In: Proc. of the 20th ACM SIGKDD, ACM (2014) 1266–1275
31. Myers, S.A., Leskovec, J.: The bursty dynamics of the twitter information network. In: Proc. of the 23rd WWW, ACM (2014) 913–924
32. Pichl, M., Zangerle, E., Specht, G.: Combining spotify and twitter data for generating a recent and public dataset for music recommendation. In: Grundlagen von Datenbanken. (2014) 35–40
33. Besel, C., Schlötterer, J., Granitzer, M.: Inferring semantic interest profiles from twitter followees: Does twitter know better than your friends? SAC '16 (2016)
34. Nechaev, Y., Corcoglioniti, F., Giuliano, C.: Sociallink: Linking dbpedia entities to corresponding twitter accounts. In: The Semantic Web – ISWC 2017. (2017)
35. Faralli, S., Stilo, G., Velardi, P.: Automatic acquisition of a taxonomy of microblogs users' interests. Journal of Web Semantics (2017)
36. Navigli, R., Ponzetto, S.P.: BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. AI (2012) 217–250
37. Delli Bovi, L., Telesca, L., Navigli, R.: Large-scale information extraction from textual definitions through deep syntactic and semantic analysis. TACL **3** (2015)