

Guess the movie - Linking Facebook pages to IMDb movies

Paolo Fornacciari¹, Barbara Guidi², Monica Mordonini¹, Jacopo Orlandini¹,
Laura Sani¹, and Michele Tomaiuolo¹

¹ Dipartimento di Ingegneria e Architettura - Università di Parma (Italy)

² Dipartimento di Informatica - Università di Pisa (Italy)

Abstract. In Facebook, the set of pages liked by some users represents an important knowledge about their real life tastes. However, the process of classification, which is already hard when dealing with dozens of classes and genres, is made even more difficult by the very coarse information of Facebook pages. Our work originates from a large dataset of pages liked by users of a Facebook app. To overcome the limitations of multilabel automatic classification of free-form user-generated pages, we acquire data also from IMDb, a large public database about movies. We use it to associate with high accuracy a given cinema-related page on Facebook to the corresponding record on IMDb, which includes plenty of metadata in addition to genres. To this aim, we compare different approaches. The obtained results demonstrate that the highest accuracy is obtained by the combined use of different methods and metrics.

Keywords: Online Social Networks, Data Mining, Information Retrieval, Text Similarity

1 Introduction

Nowadays, Online Social Networks (OSNs) are the most popular applications in Internet. They changed the way of how people communicate and interact. Structural properties of OSNs and, in particular, understanding users' behavior when they connect to OSNs is crucial for designing user-centered systems. In particular, analysing every aspect of the online social life of users, we could be able to build a kind of system, as a distributed system [9], to model a current Online Social Network, as Facebook. To analyze aspects and characteristics of users, we need information about the online identity of a user, which involves information about the topology of the social graph, friendships and interactions between users, etc. Given several information regarding an online social network user, one point consists in the definition of multi-dimensional networks [4, 14], in which we can mark users with several attributes (i.e. personal interests about cinema, music, etc.). Multidimensionality allows us to analyze each layer separately and at the same time investigate aggregations of different layers. Multi-dimensional networks can be represented as a multi-graph and due to their complexity, they

are difficult to be analyzed. A recent trend in OSNs is to organize the information related to some specific topics (music genre, actors, movies, ...) in pages which can be linked to users through a like operation. This information represents an important knowledge of the real life of users and permits to categorize users according to specific categorizations.

The study of online behaviour of users is a hard task for at least two main motivations: *(i)* The information is very coarse and not easy to categorize because users are free to create what they want and how they want. An example can be done by considering an information related to music pages. In Facebook we can find several pages related to a singer X and several pages dedicated to songs of the singer X. All of these pages should belong to the same category. *(ii)* The lack of a complete dataset which contains information about users, interaction, interests and temporal behaviour (needed to understand the evolution of the social life).

Our work deals with a specific objective, in the scope of these general motivations. In particular, we have tried to overcome some of the mentioned limitations by gathering a real Facebook dataset in which we have all the required information. Thanks to real data, in this paper we provide a tool which can be used to understand some specific personal data of users, with good precision. Indeed, the tool permits to limit the amount of the coarse information obtained from the OSNs by understanding the topic of the information.

We evaluate our tool on Facebook pages concerning movies. The evaluation shows that the tool is able to find a specific film related to a given title and, as a consequence, we are able to understand the genre of the film.

2 Related Works

Traditional methods of Social Network Analysis (SNA, see [21]), are often supported by a semantic analysis of social media to discovery implicit communities that share common tastes and interests in different domains [2]. Community detection is one of the most popular and most computationally expensive application of SNA; it is useful in recommendation systems, link prediction and suggestion, epidemic spreading and information diffusion. An example of community detection algorithm is presented in [1], which adopts an ego-based approach and exploits the parallelism of modern architectures. The problem of discovering topically meaningful communities is introduced in [20]. The framework allows the discovery of both communities and users' interests, with quite good results. In [11] the authors investigate how social networks can be used to generate recommendations and predict the ratings of clubs. The use of both social and content-based information in recommendation system is described, also, in [3]. It presents an inductive learning approach to recommendation, using both ratings information coming from the individuals of a networks and other forms of information about each artifact (such as, cast lists or movie reviews) in predicting user preferences.

Facebook is one of the most popular online social networks and its structure is widely investigated: for example, in [23] the authors describe the social structure obtained by the complete Facebook network of 100 American colleges and universities from a single-day snapshot. Many companies and brands use Facebook to increase their social reputation, or for so-called word-of-mouth marketing. In a recent research [22] the authors investigate the impact of social influence of a Facebook fan page on movie box offices. In fact, almost every movie has its own Facebook fan page nowadays. The research shows that it is possible to establish a relationship between the information diffusion among users on Facebook and the engagements of a fan page without explicit knowledge of the social network, and to measure the social influence of leading users in a fan page. HappyMovie is a Facebook application for recommending movies to groups [18]. The final recommendation is influenced by the personality of each member of the group and the way in which they are connected. This way, the application can offer a product to a group of people that fits the individual needs of every member, trying to achieve high satisfaction. Another interesting Facebook application for a movie recommendation system is presented in [17]. It basically exploits the information coming from Linked Data datasets in the Linked Data cloud, mainly DBpedia and LinkedMDB (the semantic-enabled version of the Internet Movie Database IMDb), to compute a semantic similarity between movies and to provide a recommendation to the user. In [12], authors use both videos and subtitles of movie trailers to classify the genre and MPAA rating.

In the fields of Information Retrieval and document clustering, the possibility of discriminating and selecting documents using text semantic similarity is a quite relevant challenge. In [10], the authors discuss the existing works on text similarity through partitioning them into three approaches: String-based, Corpus-based and Knowledge-based similarities. String-based similarity measures operate on string sequences and character composition. In [5], a variant of the Damerau–Levenshtein Distance is used for spelling corrections. In [13], Cosine Similarity is used for document clustering. Another string matching algorithm is the Gestalt Pattern Matching, or Ratcliff/Obershelp pattern recognition, that is based on the longest common subsequence [19]. An overview of Corpus-based and Knowledge-based similarity algorithms is described in [16]. Corpus-based measures of word semantic similarity try to identify the degree of similarity between words using information exclusively derived from large corpora, while knowledge-based measures try to quantify the degree to which two words are semantically related, using information drawn from semantic networks.

3 Datasets

3.1 The SocialCircles! Dataset

We use a real dataset gathered by a Facebook application, called SocialCircles!³, which exploited the Facebook API to retrieve social information about registered

³ <http://social.di.unipi.it/>

users. The application is now unavailable because Facebook no longer support the applications exploiting this API since the 1st May 2015. The goal of the application was to retrieve a good number of Ego Networks, which are explained in detail in [8]. Instead our specific scenario is explained in [7] and is associated with the following sets of data:

- Topology and profile information. We gathered complete profile information about 337 registered users and the friendship relations existing between them.
- Interaction information. We collected information about interactions between users registered to the application and their friends, such as posts, comments, likes, tags and photos.
- Online presence data. We monitored the chat presence status and obtained information about the time spent online by registered users and their friends, for a total of 95.578 users.

The dataset obtained from the SocialCircles! application contains 337 complete Ego Networks, for a total of 144.481 users (ego and their alters). The Ego Networks we retrieved have the advantage of representing a very heterogeneous population: 213 males and 115 females, with age range of 15-79 with different education, background and provenience. We believe this is an advantage compared to other works where the dataset comes from an analysis of groups of people with the same background and education [15]. In [6] a complete analysis of the dataset is shown.

3.2 Multi-layer ego network: users, interactions, and movies

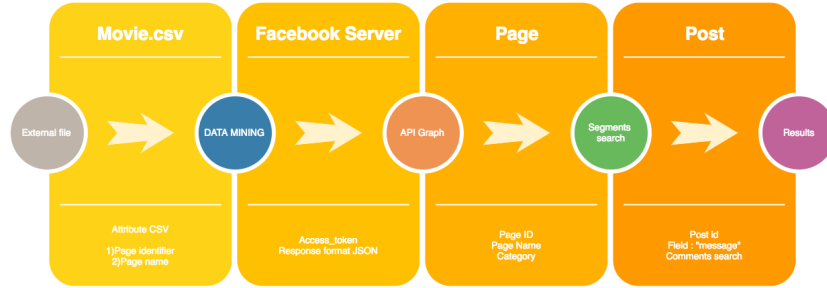


Fig. 1. Workflow of data acquisition from Facebook, for a given public page.

As explained before, we are able to build a multi-layer network which contains different layers, one to describe the ego network of a generic ego, one layer to describe the interaction between users in an ego network and, finally, one layer to describe an interest of users, in our case we focus on movie genres.

This multi-dimensional ego networks could be extended in the future with other information contained in our dataset, such as music, books, etc., which can be used to categorize users, find homophily and, to recommend information based on user interest. As shown in Figure 1, from the SocialCircle dataset we extract information about 69519 Facebook movie pages and the relations between users and the movie pages (who likes what). Several of them are pages without a significant title and the most important problem when we need to tag a user with a genre is to understand the genre of a page from the title of the page, that is the only information available. Thus, for each page we also collect all available posts and comments.

3.3 The IMDb Dataset

The Internet Movie Database (IMDb)⁴ is an online relational database of information related to movies, television programs and video games, including cast, production crew, fictional characters, biographies, plot summaries and reviews, handled by IMDb.com, Inc., a subsidiary of Amazon. IMDb contains the Top Rated Movie list, ranked by a formula which includes the number of ratings each movie received from users, and the value of ratings received from regular users. To be included on the list, a movie must receive ratings from at least 25000 users. Using the same threshold for popularity, the dataset used in this work is made up of all the 4636 unique films that received ratings from at least 25000 users on IMDb. In addition to the title, the dataset contains synopsis, plot and genre. Each movie can be associated to one (or more) of the 27 genres. The textual elements analyzed in our work are titles, plots and synopsis.

4 Methodology

Algorithm 1 Pseudo-code of the main function.

```

1: function FINDMOVIE( $p, \mathcal{M}$ )
2:    $\mathcal{L} \leftarrow \text{SelectMovies}(p, \mathcal{M})$ 
3:    $s_{best} \leftarrow 0$ 
4:    $m_{best} \leftarrow \text{None}$ 
5:   for all  $ts, m \in \mathcal{L}$  do
6:      $s \leftarrow \text{PageSimilarity}(p, m, ts)$ 
7:     if  $s > s_{best}$  then
8:        $s_{best} \leftarrow s$ 
9:        $m_{best} \leftarrow m$ 
10:    end if
11:  end for
12:  return  $m_{best}$ 
13: end function

```

⁴ <http://www.imdb.com/>

In our system, the main algorithm confronts a given Facebook page with a list of IMDb movies, to find the best matching movie. Algorithm 1 is essentially a function to select the movie with the maximum similarity, with respect to the given page. The parameters represent, respectively, the given page (p) and the set of movies ($\mathcal{M}$, in our case, the whole dataset acquired from IMDb). Both the *SelectMovies* function and the *PageSimilarity* function are customizable in different ways. In fact, we experimented different mechanisms and policies, to finally find the most suitable combination.

Algorithm 2 Pseudo-code of the selection of movies.

```

1: function SELECTMOVIES( $p, \mathcal{M}$ )
2:    $\mathcal{L} \leftarrow \emptyset$ 
3:   for all  $m \in \mathcal{M}$  do
4:      $ts \leftarrow TitleSimilarity(p, m)$ 
5:      $\mathcal{L} \leftarrow \mathcal{L} \cup \{(ts, m)\}$ 
6:   end for
7:    $\mathcal{L} \leftarrow Sort(\mathcal{L})$ 
8:    $n \leftarrow SliceSize(\mathcal{L})$ 
9:    $\mathcal{L} \leftarrow Take(n, \mathcal{L})$ 
10:  return  $\mathcal{L}$ 
11: end function

```

In all our experiments, but the very first, the *SelectMovies* function is an implementation of Algorithm 2. Each IMDb movie is associated with its title similarity, with respect to the given Facebook page. Thus, a list of pairs (*similarity, movie*) is obtained. This list is then sorted and its first n elements are considered for further analysis. Here, n is the result of an implementation of the *SliceSize* function, which we will discuss later.

4.1 Approach #1: Content Similarity

Since titles are not accurate, the first approach we have experimented is simply to discard them completely. This analysis is based on the content of Facebook pages and IMDb movies, without considering their titles. Given a certain page p , we confront it with each movie m in our database, based on their textual content. For the Facebook page, the text is collected from all posts and comments on the page. For the IMDb movie, the text is obtained by concatenating its synopsis (if it is available) with all its plots.

Referring to Algorithm 1, the *SelectMovies* function is essentially replaced by an all-pass function. All available movies are accepted and assigned the same title distance. The *PageSimilarity* function is evaluated as the *Cosine Similarity* between the page p and the movie m . As we will describe later, this approach is very computational intensive, as it compares a given Facebook page with every available movie. Moreover, its accuracy is not satisfactory.

Cosine Similarity has been used as it is one of the best known and most precise algorithms for evaluating the similarity between two long texts. It uses the classical BagOfWords transformation, for representing a text in a multidimensional space. Features are then weighted according to the TfIdf model, which highlights the most peculiar words of a given text, instead of the most frequent words. Texts in our datasets are all prefiltered, for removing stop words, and stemmed.

4.2 Approach #2: Variable Slice

The second approach uses the similarity between titles, in order to obtain a list of the most promising movies. This has the aim of both reducing calculations and also guiding the process towards the most interesting set of movies.

For confronting titles, we have initially considered the Levenshtein distance and the Gestalt Pattern Matching (GPM) function. However, from the initial results, the Levenshtein distance has demonstrated to be unfit for this kind of analysis. Thus, for implementing the *TitleSimilarity* function, we have used the Gestalt approach for pattern matching.

For deciding the size of the slice of movies to consider, we have studied the data in our annotated dataset. In particular, we have plotted a set of points for these data, in a Cartesian space, where the x-axis represents the obtained best value of similarity (not necessarily for the correct movie) and the y-axis represent the position of the correct movie in the sorted list. A linear regression gives better clue about the most useful region of this space. In fact, when the best obtained similarity is low, the position of the correct movie tends to be higher. Thus, in our system, the number of movies considered for full textual analysis is a variable slice. The implementation of the *SliceSize* function closely approximates this regression line, i.e., the slice size is inversely proportional to the best obtained similarity (which may not correspond to the correct movie).

4.3 Approach #3: Gestalt Title Similarity

As an alternative evaluation, instead of measuring the Cosine Distance between textual content, we use only the title similarity (based on Gestalt Pattern Matching). In Algorithm 2, this corresponds to a constant slice size of a single element: the movie with the best matching title.

4.4 Approach #4: Levenshtein Title Similarity

For direct comparison, we repeat the previous approach, using a measure of similarity based on the the Levenshtein Distance, instead of the Gestalt Pattern Matching. The similarity measure is essentially obtained from the inverse of the distance. In the following formula, x is the Levenshtein Distance and y the similarity (k is a parameter to optimize).

$$y = \frac{1}{1 + k \cdot x}$$

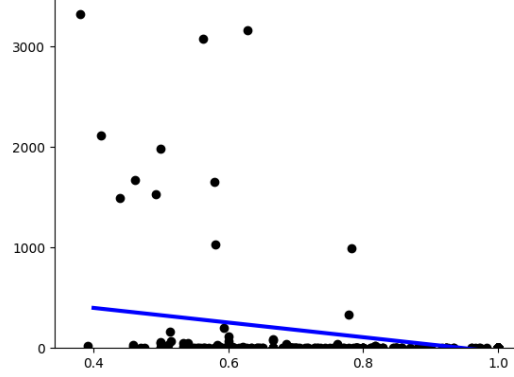


Fig. 2. Linear regression applied to sample data. Each dot represents an instance of the sample dataset, as a pair $(similarity_{best}, position_{correct})$. I.e., we have matched each obtained best value of similarity (which may not correspond to the correct movie), with the position of the correct movie in the sorted list. The line is the linear regression.

4.5 Approach #5: Title + Popularity

Additionally, for selecting the most promising movies, their popularity may also be useful. Thus, we evaluate a popularity index, on the basis of votes collected on IMDb. We consider the total number of votes, disrespecting their particular value. Either voted positively or negatively, a movie has to be considered popular or known, if it collects a large quantity of feedback. For evaluating the popularity index, we use the following function, which penalizes the less popular movies (h is a parameter to optimize). It is worth to remember here that we have collected only IMDb movies with at least 25000 votes. The title similarity is then combined linearly with this popularity index. Popularity is weighted with a small coefficient, as it is useful to discriminate only among very similar titles.

$$y = \left(1 - \frac{25000}{x}\right)^h$$

4.6 Approach #6: Title + Popularity + Content

Finally, according to this last approach, both the textual content and the title provide useful information for finding the right movie. Thus, in this case the *PageSimilarity* returns a linear combination of title similarity (Gestalt Pattern Matching) and content similarity (Cosine Similarity).

Cosine Similarity is inferior, alone. But it provides a good contribution, if joined with title similarity.

5 Results

Table 1. Results of the different approaches, for both quality and computing time.

Approach	Accuracy	Precision	Recall	F-measure	Time
#1	14.98%	0.1887	0.1503	0.1596	22611.52 s
#2	68.50%	0.8021	0.6840	0.7095	267.18 s
#3	83.79%	0.8824	0.8374	0.8439	70.09 s
#4	81.65%	0.8901	0.8160	0.8279	7.46 s
#5	84.10%	0.8870	0.8405	0.8445	75.92 s
#6	86.24% (93.88%)	0.8944	0.8620	0.8666	314.26 s

We have tested all the approaches described in Section 4 on a subset of Facebook pages, from the global dataset. These randomly selected pages have been manually annotated and accurately matched with their corresponding IMDb movies⁵. This testset includes 327 pages, which are of very different kinds, including some titles with misspelled words, cryptic citations, names of actors and characters instead of the movie title, non-English language, etc.

In particular, for the content similarity, we always prefilter the textual content for removing English classical stop words and stemming the remaining words; for title similarity, we remove only the words “the” and “movie”, which are often used in names of Facebook pages and largely disturb the analysis, and then remove all spaces. In all approaches, we use the following optimizations: the slice size decreases from 720 at $r_{best} = 0$, to 1 at $r_{best} \geq 0.9$; for the Levenshtein similarity, k is 0.05; for the popularity index, h is 2. In all our tests, the Levenshtein similarity has always provided lower accuracy than the Gestalt similarity, though executing in much less time.

We have found that in many cases the correct movie can be determined simply through the comparison of titles (approach #5). However, for obtaining the correct result when the page title is misleading, the comparison of page content and movie description is needed (approach #2). The final result (approach #6) is obtained by combining title and content similarity, weighted at 0.40 and 0.60, respectively; weights are optimized, meaning that content similarity gives an important contribution. Results are shown in Table 1 and Figure 3. We list here few anecdotal examples (when a FB page is dedicated to a whole saga, in the test set it is linked to the first episode in IMDb).

- FB page: ... *Twilight.*, *New Monn.*, *Eclipse and Breaking dawn...* [typo in the title], correctly classified as: *The Twilight Saga: New Moon*;
- FB page: *~TwILiGhT~neW MoOn~eCLipse~bReakiNg DAwN~*, correctly classified as: *The Twilight Saga: New Moon*;

⁵ The main datasets of this work are available at <http://sowide.unipr.it/datasets>

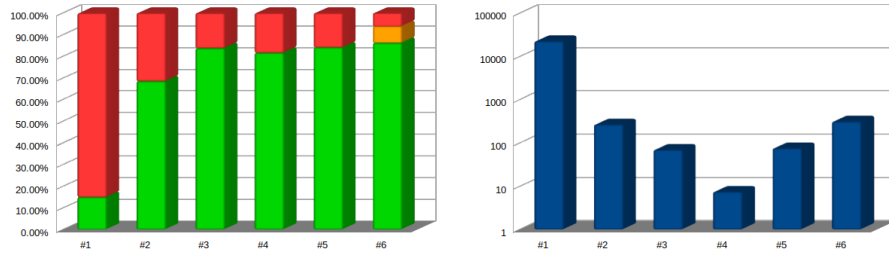


Fig. 3. Accuracy (left) and execution time (right) of the six different approaches described in Section 4. For the accuracy, the last bar shows also the quota of wrong movies, but still in the correct series/saga. For the execution time, the results are in seconds and the scale is logarithmic.

- FB page: *noi...twilighthers forever...*, correctly classified as: *The Twilight Saga: New Moon*;
- FB page: *Toothless the unique NightFury dragon*, correctly classified as: *How to Train Your Dragon*;
- FB page: *“So that’s who Finnick loves. A poor, mad girl back home.”* [quotes in the title], correctly classified as: *The Hunger Games*;
- FB page: *New Zealand Hobbit Fans*, correctly classified as: *The Hobbit: An Unexpected Journey*;
- FB page: *Meet the Parents: Little Fockers*, wrongly classified as: *Meet the Fockers*, instead of: *Little Fockers*;
- FB page: *Matrix Trilogy*, wrongly classified as: *The Matrix Reloaded*, instead of: *The Matrix*;
- FB page: *“boom boom!...fire powerr!”* (*Night At the Museum 2*) [quotes in the title], wrongly classified as: *Night at the Museum*, instead of: *Night at the Museum: Battle of the Smithsonian*.

Finally, by a closer look at the errors, it can be easily observed that many wrong movies are still selected from the correct series/saga. In many cases this might still be a useful result. Counting also these instances as acceptable, the final accuracy would be at 93.88%. In our opinion, the final result is pretty good also for real world application. It is mainly due to the inclusion of external data, IMDb in this case, in the process. These data add valuable knowledge, which is exploited by the prediction system. As a downside, this kind of approach cannot deal with pages related to less known or too recent movies, which are not present in the IMDb list of popular movies, by number of votes.

In the end, it is worth to consider that a bayesian multi-label classifier (27 labels) has shown an accuracy of 68.38%, in predicting *any one* of the genres associated with a movie. This classifier has been trained and tested, with 10-folds cross validation, on the synopsis and plots from IMDB, which usually are more coherent than Facebook pages. In [12], using movie subtitles, the genre is classified with an accuracy of 43.66%, using a Support Vector Machine classifier, with 7 possible output labels.

6 Conclusions

The motivation for this research work was to study the possibility of inferring some Facebook users' tastes about cinema genres, by analysing the pages they like. Multi-class automatic classification with high accuracy, as required in the case of the dozens of movies genres, is always challenging. It is even more so in case of coarse user-generated pages. Our approach is based on the use of IMDb data, to associate a Facebook page with an IMDb record, instead of direct genre classification. This way, on the basis of title and content similarity, we are able to obtain a high level of accuracy in finding the exact movie which the Facebook page is dedicated to. Thus, we gain not only genre information, but also all the metadata in IMDb.

References

1. Amoretti, M., Ferrari, A., Fornacciari, P., Mordonini, M., Rosi, F., Tomaiuolo, M.: Local-first algorithms for community detection. In: Proceedings of the 2nd International Workshop on Knowledge Discovery on the WEB, KDWeb 2016, Cagliari, Italy, September 8-10, (2016), <http://ceur-ws.org/Vol-1748/paper-07.pdf>
2. Angiani, G., Fornacciari, P., Iotti, E., Mordonini, M., Tomaiuolo, M.: Models of participation in social networks. *Social Media Performance Evaluation and Success Measurements* p. 196 (2016)
3. Basu, C., Hirsh, H., Cohen, W.: Recommendation as classification: Using social and content-based information in recommendation. In: Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence. pp. 714–720. AAAI '98/IAAI '98, American Association for Artificial Intelligence, Menlo Park, CA, USA (1998)
4. Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C.I., Gómez-Gardenes, J., Romance, M., Sendina-Nadal, I., Wang, Z., Zanin, M.: The structure and dynamics of multilayer networks. *Physics Reports* 544(1), 1–122 (2014)
5. Cucerzan, S., Brill, E.: Spelling correction as an iterative process that exploits the collective knowledge of web users. In: EMNLP. vol. 4, pp. 293–300 (2004)
6. De Salve, A., Dondio, M., Guidi, B., Ricci, L.: The impact of user's availability on on-line ego networks. *Comput. Commun.* 73(PB), 211–218 (Jan 2016)
7. De Salve, A., Guidi, B., Mori, P., Ricci, L.: Distributed coverage of ego networks in f2f online social networks. In: Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCCom/IoP/SmartWorld), 2016 Intl IEEE Conferences. pp. 423–431. IEEE (2016)
8. Everett, M., Borgatti, S.P.: Ego network betweenness. *Social Networks* 27(1), 31 – 38 (2005)
9. Franchi, E., Poggi, A., Tomaiuolo, M.: Blogracy: A peer-to-peer social network. *International Journal of Distributed Systems and Technologies (IJ DST)* 7(2), 37–56 (2016)
10. Gomaa, W.H., Fahmy, A.A.: A survey of text similarity approaches. *International Journal of Computer Applications* 68(13) (2013)

11. Groh, G., Ehming, C.: Recommendations in taste related domains: collaborative filtering vs. social filtering. In: Proceedings of the 2007 international ACM conference on Supporting group work. pp. 127–136. ACM (2007)
12. Helmer, E., Ji, Q.: Film classification by trailer features (2012)
13. Huang, A.: Similarity measures for text document clustering. In: Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. pp. 49–56 (2008)
14. Kazienko, P., Musial, K., Kukla, E., Kajdanowicz, T., Bródka, P.: Multidimensional social network: Model and analysis. In: Jedrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI (1). Lecture Notes in Computer Science, vol. 6922, pp. 378–387. Springer (2011)
15. La Gala, M., Arnaboldi, V., Conti, M., Passarella, A.: Ego-net digger: a new way to study ego networks in online social networks. In: First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research (ACM HotSocial 2012) (2012)
16. Mihalcea, R., Corley, C., Strapparava, C., et al.: Corpus-based and knowledge-based measures of text semantic similarity. In: AAAI. vol. 6, pp. 775–780 (2006)
17. Mirizzi, R., Di Noia, T., Ragone, A., Ostuni, V.C., Di Sciascio, E.: Movie recommendation with dbpedia. In: IIR. pp. 101–112 (2012)
18. Quijano-Sanchez, L., Recio-Garcia, J.A., Diaz-Agudo, B.: Happymovie: A facebook application for recommending movies to groups. In: Tools with Artificial Intelligence (ICTAI), 2011 23rd IEEE International Conference on. pp. 239–244. IEEE (2011)
19. Ratcliff, J.W., Metzener, D.E.: Pattern-matching - the gestalt approach. *Dr Dobbs Journal* 13(7), 46 (1988)
20. Sachan, M., Contractor, D., Faruque, T.A., Subramaniam, L.V.: Using content and interactions for discovering communities in social networks. In: Proceedings of the 21st international conference on World Wide Web. pp. 331–340. ACM (2012)
21. Scott, J.: Social network analysis. Sage (2017)
22. Tang, W.H., Yeh, M.Y., Lee, A.J.: Information diffusion among users on facebook fan pages over time: Its impact on movie box office. In: Data Science and Advanced Analytics (DSAA), 2014 International Conference on. pp. 340–346. IEEE (2014)
23. Traud, A.L., Mucha, P.J., Porter, M.A.: Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications* 391(16), 4165–4180 (2012)