ARTICLE IN PRESS

Information Processing and Management xxx (2012) xxx–xxx

Contents lists available at SciVerse ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Assessing the quality of textual features in social media

Flavio Figueiredo [a], Henrique Pinto [a], Fabiano Belém [a], Jussara Almeida [a], Marcos Gonçalves [a,*], David Fernandes [b], Edleno Moura [b]

[a] Universidade Federal de Minas Gerais, Department of Computer Science, Belo Horizonte, MG, Brazil
[b] Universidade Federal do Amazonas, Department of Computer Science, Manaus, AM, Brazil

ARTICLE INFO

Article history:
Received 22 March 2011
Received in revised form 12 March 2012
Accepted 14 March 2012
Available online xxxx

Keywords:
Web 2.0
Textual features
Information quality
Social media

ABSTRACT

Social media is increasingly becoming a significant fraction of the content retrieved daily by Web users. However, the potential lack of quality of user generated content poses a challenge to information retrieval services, which rely mostly on textual features generated by users (particularly tags) commonly associated with the multimedia objects. This paper presents what, to the best of our knowledge, is currently the most comprehensive study of the relative quality of textual features in social media. We analyze four different features, namely, TITLE, TAGS, DESCRIPTION and COMMENTS posted by users, in four popular applications, namely, YouTube, Yahoo! Video, LastFM and CiteULike. Our study is based on an extensive characterization of data crawled from the four applications with respect to usage, amount and semantics of content, descriptive and discriminative power as well as content and information diversity across features. It also includes a series of object classification and tag recommendation experiments as case studies of two important information retrieval tasks, aiming at analyzing how these tasks are affected by the quality of the textual features. Classification and recommendation effectiveness is analyzed in light of our characterization results. Our findings provide valuable insights for future research and design of Web 2.0 applications and services.

1. Introduction

The advent and rapid growth of a variety of Web 2.0 applications, enabling and fostering the establishment of online communities and social networks, have contributed to the creation and dissemination of a massive amount of social media content. Social media refers to content created and disseminated via social interactions, and is thus typically associated with *user generated content*. In general, social media is increasingly becoming a significant fraction of the content searched for and retrieved daily by Web users. Take YouTube,[1] the currently most popular social video sharing application, as an example. With reportedly 24 h of videos uploaded per minute and 2 billion video views a day,[2] YouTube often figures among the top four applications in volume of traffic over the Internet.[3]

Social media typically includes a main object, stored in one of various media types (e.g., text, image, audio or video), as well as a variety of other sources of information related to the object, which we refer to as the object's associated *features*. *Content features* are sources of information which can be extracted from the object itself, such as the color histogram of an

* Corresponding author.
E-mail addresses: flaviov@dcc.ufmg.br (F. Figueiredo), hpinto@dcc.ufmg.br (H. Pinto), fmuniz@dcc.ufmg.br (F. Belém), jussara@dcc.ufmg.br (J. Almeida), mgoncalv@dcc.ufmg.br (M. Gonçalves), david@dcc.ufam.edu.br (D. Fernandes), edleno@dcc.ufam.edu.br (E. Moura).

[1] http://www.youtube.com.
[2] http://www.youtube.com/t/fact_sheet.
[3] http://www.alexa.com.

0306-4573/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ipm.2012.03.003

image. *Textual features*, on the other hand, are textual content often created and associated with the object by the users themselves. Examples are the object title, description, tags and comments posted by users. *Social features*, in turn, reflect the social context into which the object is inserted, that is, the user who created it, the users who accessed it, and the user interactions established through it.

Textual features are of particular interest because of their potential as supporting data for a variety of Information Retrieval (IR) services such as search, recommendation and online advertising. In fact, most IR services available in current Web 2.0 applications exploit only the object's *textual features*, in spite of various multimedia IR techniques available in the literature (Rui, Huang, Mehrotra, & Ortega, 1997; Smeulders, Worring, Santini, Gupta, & Jain, 2000).[4] However, being often unsupervised sources of data, created by the end users themselves, textual features as well as social media in general suffer from the potential *lack of quality* as supporting data for effective IR services.

For instance, one particular type of textual feature, namely *tags*, has been the focus of a large body of recent work. Previous studies include the investigation of its use for supporting search, recommendation, classification and clustering (Clements, de Vries, & Reinders, 2010; Guy, Zwerdling, Ronen, Carmel, & Uziel, 2010; Byde, Wan, & Cayzer, 2007; Li, Guo, & Zhao, 2008; Schenkel et al., 2008; Chen, Wright, & Nejdl, 2009; Ramage, Heymann, Manning, & Garcia-Molina, 2009; Guan, Bu, Mei, Chen, & Wang, 2009; Song et al., 2008; Sen, Vig, & Riedl, 2009), as well as various analyses of their usage patterns (Golder & Huberman, 2006; Sigurbjornsson & van Zwol, 2008; Paul Heymann, Andreas Paepcke, & Hector Garcia-Molina, 2010; Santos-Neto, Figueiredo, Mowbray, Gonçalves, & Ripeanu, 2010). However, no consensus has been reached as to the quality of tags for effective IR (Bischoff, Claudiu, Wolfgang, & Raluca, 2008; Marshall, 2009). Moreover, previous research focuses mostly only on tags, neglecting the potential use of other textual features, such as title and description.

In this context, this paper presents what, to the best of our knowledge, is the currently most comprehensive study of the relative quality of textual features in social media. The term *quality* is here used to refer to the feature's potential effectiveness as supporting data for IR services. In that sense, the quality of a feature depends on several aspects, including whether it (1) carries enough content to be useful, (2) provides a good description of the object's content, and (3) is able to effectively discriminate objects into different pre-defined categories (e.g., for tasks such as object classification and directory organization) or into levels of relevance (e.g., for searching). Our goal in this paper is, thus, to *assess the relative quality of different textual features commonly found in various popular Web 2.0 applications*.

Towards our goal, we crawled object samples from four applications which target different media types as primary means for content dissemination. In addition to YouTube, we also consider Yahoo! Video[5] (or simply YahooVideo), another popular social video sharing application, LastFM[6] (or simply LastFM), an online radio and music community website,[7] and CiteULike,[8] a scholarly reference management and discovery service. Our collected samples contain the contents of four textual features, namely, TITLE, TAGS, DESCRIPTION and COMMENTS.

We characterized the collected data in order to extract evidence of quality of each feature with respect to usage, amount and semantics of content, descriptive and discriminative power, as well as evidence of content and information diversity across features associated with the same object. Our main findings are: (1) all four features but TITLE are absent (i.e., with no content) in a non-negligible fraction of the collected objects, particularly in applications that allow for the collaborative creation and edition of the feature's content; (2) collaborative features, if present, do tend to carry larger amounts of content than features whose editorial access is restricted to the object's owner; (3) a significant degree of polysemy as well as a significant amount of non-existing terms affect all four features in all four applications; (4) the typically smaller and more often used TITLE and TAGS features exhibit, in general, higher descriptive and discriminative powers, followed by DESCRIPTION and COMMENTS, although, on both CiteULike and LastFM, DESCRIPTION outperforms TAGS (but not TITLE) in terms of descriptive power if it is estimated based only on the top-5 most descriptive terms; and (5) there is a non-negligible amount of content and information diversity across features associated with the same object, implying that different features contribute with different pieces of information about it.

We also assessed the relative quality of the features with respect to two specific IR tasks, namely, object classification and tag recommendation. These tasks were selected due to their common applicability to a plethora of services (e.g., automatic construction of Web directories, search, browsing, etc.), and because they allow for automatic assessment of feature quality.

Classification experiments were performed varying the source of data for the classifier and the content weighting scheme. In particular, we tested six classification strategies consisting of taking the content of each feature in isolation as well as two combinations of all four features. Our main results, which are supported by our characterization findings, are: (1) weighting schemes that explore discriminative power have better effectiveness either in isolation or in combination with other metrics; (2) TAGS, if present, are in fact the most promising feature in isolation, due to a combination of good discriminative power and large amount of content, two quality-related aspects that have important roles for classification effectiveness; (3)

[4] This is possibly due to the lack of scalability of state-of-the-art multimedia techniques to the size of popular Web 2.0 applications as well as to the detrimental impact that the (often low) quality of user-generated multimedia objects, particularly videos, has on the effectiveness of these techniques (Boll, 2007).

[5] http://video.yahoo.com.

[6] http://last.fm.

[7] Note that, although the musical content available in LastFM is not *generated* by users, its community based organization gives users the ability to describe musical content and interact over social networks.

[8] http://www.citeulike.org.

Please cite this article in press as: Figueiredo, F., et al. Assessing the quality of textual features in social media. *Information Processing and Management* (2012), http://dx.doi.org/10.1016/j.ipm.2012.03.003

combining content from multiple features may improve classification results due to the presence of distinct and somewhat complementary content and information; (4) a simpler feature combination strategy based on bag of words may be as effective or at most slightly worse than concatenating features as different feature spaces; and (5) in spite of its good discriminative and descriptive power and its larger object coverage, TITLE is the feature with lowest quality for object classification, since its effectiveness is very affected by the small amount of content available.

Motivated by our findings regarding the large amount of content and information diversity across textual features of the same object, we also investigated the potential of exploiting the contents of TITLE, DESCRIPTION and COMMENTS for improving the quality of the TAGS feature through the recommendation of candidate terms. Our results show that, unlike observed for classification tasks, TITLE, despite its smaller sizes, is the feature that offers the highest quality content for tag recommendation purposes, since, as observed in our characterization, this feature has the best descriptive power in all four analyzed applications.

This paper's main contribution lies in a more solid understanding about the quality of textual features in social media. In comparison with our previous studies (Figueiredo et al., 2009; Almeida, Gonçalves, Figueiredo, Belém, & Pinto, 2010), we here present a much more comprehensive analysis. In particular, we extend our prior work to (1) characterize the semantic properties of the contents of each feature as well as the diversity of information across features, (2) perform a more extensive analysis of the impact of feature quality for object classification purposes, and (3) investigate the potential of exploiting the contents of different features for tag recommendation. The knowledge produced here may be of use to Web 2.0 application designers, who can address issues such as usefulness and attractiveness of each feature to their users. It may also be of relevance to designers of social media IR services as it uncovers evidence of quality of each feature as a potential source of information. In sum, we here offer a more comprehensive exploration of the problem than previous work, covering different features in different applications and their impact on two different IR services, thus providing more sound conclusions.

The remaining of this paper is organized as follows. Related work is discussed in Section 2. Section 3 introduces the four textual features analyzed, whereas Section 4 summarizes our data collection methodology. The main findings from our textual feature characterization are discussed in Section 5. Section 6 describes our classification experiments and presents their most relevant results, whereas our assessment of the potential benefits of exploiting feature contents for tag recommendation is presented in Section 7. Finally, Section 8 concludes our paper, pointing out possible directions for future work.

2. Related work

Previous efforts towards analyzing and exploiting textual features on the Web 2.0 focused mainly on tagging systems, which have been proposed as a more flexible alternative to describe and organize objects. Indeed, many authors have proposed the use of tags to enhance services such as searching, recommendation, clustering and indexing (Hotho, Jaschke, Schmitz, & Stumme, 2006; Byde et al., 2007; Li et al., 2008; Schenkel et al., 2008; Song et al., 2008; Sigurbjornsson & van Zwol, 2008; Clements et al., 2010; Guy et al., 2010). Others have addressed the characterization of tagging systems (Paul et al., 2010; Golder & Huberman, 2006; Santos-Neto et al., 2010). Nevertheless, there still lie many open questions with respect to the *quality* of tags.

On one hand, Suchanek, Vojnovic, and Gunawardena (2008) analyzed the amount of tags in Delicious[9] that are *known* according to two semantic databases. They found that only approximately half of the used tags have entries in these dictionaries, and that nearly half of these known tags have 10 or more meanings (measured by distinct occurrences in the dictionary). In other words, they concluded that tags contain a significant amount of unknown terms and suffer from polysemy, two issues that might have a detrimental impact on IR effectiveness. In contrast, Bischoff et al. (2008) analyzed the quality of tags in the context of searching in Delicious and LastFM. By checking the overlap between tags and the object's textual content (i.e., bookmarked webpages in Delicious, and song lyrics in LastFM), metadata assigned by experts, and external sources (e.g., search engines), the authors found that tags tend to be accurate with respect to the object's content and related to search queries.

In a recent study, Venetis, Koutrika, and Garcia-Molina (2011) addressed the problem of selecting tags for summarizing a set of results generated by a query. They proposed several metrics to capture the structural properties of tag clouds created for query results, and used these metrics to evaluate their quality. They also evaluated several tag selection algorithms with respect to the quality of their results. In different directions, Paul et al. (2010) analyzed the consistency, quality and completeness of tags posted by users in social cataloging sites, whereas Santos-Neto et al. (2010) proposed a framework to assess the value of user contributions in tagging systems.

In the context of classification and clustering, Ramage et al. (2009) contrasted the usage of tags assigned by users in Delicious with the usage of the full textual content of the bookmarked webpages. The authors showed that combining both webpage contents and tags improves the effectiveness of two different clustering algorithms. In addition, Chen et al. (2009) developed a classification model based on both content features and tagging data of musical tracks from LastFM. The authors showed that the use of both types of features improves classification results when compared to using only content features.

There is also a large body of work on tag and content recommendation methods. Tag recommendation aims at supporting users by suggesting "good" tags for a target object. Sigurbjornsson and van Zwol (2008), for instance, exploited metrics of tag co-occurrence, applying them to an initial set of tags associated with the target object to produce a final ranking of candidate

[9] http://delicious.com.

tags. Tag co-occurrence patterns are also exploited by Menezes et al. (2010), although the authors use a lazy associative tag recommendation method in order to efficiently uncover more sophisticated patterns, which ultimately leads to superior results in comparison with the best method in (Sigurbjornsson & van Zwol, 2008). In Belém et al. (2011) we extended traditional tag co-occurrence based methods to include not only tags that had been previously assigned to the objects but also terms extracted from other textual features, applying several heuristic metrics to capture the relevance of each candidate term as a recommendation for the target object. Following different approaches, Clements et al. (2010) exploited random walks on a social annotation graph combining content, tags and users, whereas Rendle and Schmidt-Thie (2010) proposed a personalized tag recommendation method that exploits the factorization of tag assignment events into matrices modeling the interactions among users, objects and tags. The relationships among people, tags and objects were also investigated by Guy et al. (2010): the authors proposed to recommend objects that are strongly related to people in a user's social network as well as objects related to the user's tags. We note that many existing tag recommendation methods exploit the contents of multiple textual features as sources of candidates (Lipczak, Hu, Kollet, & Milios, 2009; Zhang, Zhang, & Tang, 2009). However, to our knowledge, no previous study has analyzed the relative quality of different features as sources of data for such task. This is one of the contributions of this paper.

In addition to tag analyses, there is also some work on assessing the quality of other textual features and user generated content. Mishne (2007) characterized the use, popularity, temporal distribution of relevant posts, and amount of information in blog comments, showing that the information in comments can be used to increase the amount of documents retrieved by search queries, and that the temporal distribution can be used to re-rank query results according to blog entries' post dates. Other efforts tackled the problem of quality on question/answering (QA) websites. In a QA website, a user posts a question related to a certain topic and receives answers from other users. The relevance of answers to a given question is indicated by the users of the site. Agichtein, Castillo, Donato, Gionis, and Mishne (2008), Bian, Liu, Agichtein, and Zha (2008) and Jeon, Croft, Lee, and Park (2006) made use of textual and social features to identify and/or classify answers with higher quality. Agichtein et al. (2008), for instance, manually classified a data set of questions and answers, indicating whether the answer is of *low, medium or high* quality. They used a series of textual and social characteristics in order to separate high quality content from the rest. Textual quality was analyzed with respect to capitalization, dictionary and grammatical errors, whereas the considered social characteristics were derived from different relationship networks established among users of the system. In (Shah & Pomerantz, 2010), the authors manually evaluated the quality of answers in the Yahoo! Answers website according to various criteria, finding that their evaluation faithfully matched the askers' perceptions. They also extracted features from questions, answers and user profiles to train a number of classifiers to automatically select the best answer. More broadly, information quality is a matter of relevance to any collaboratively created content. Wikipedia,[10] for example, has also been the target of several studies, some of which with very controversial results (Dalip, Gonçalves, Cristo, & Calado, 2009; Giles, 2005; Lih, 2004).

All of the aforementioned studies provide evidence of quality of textual content on the Web 2.0, but they focus either on a single feature or on a small sample of objects. To the best of our knowledge, few are the attempts at comparing *multiple* textual features on Web 2.0 applications. In particular, we cite an analysis of Flickr,[11] an online photo sharing application, in which the authors *manually* compared the contents of three textual features (TITLE, TAGS and DESCRIPTION) as sources of description of different versions of images with the same content (i.e., a touristic place) (Marshall, 2009). The authors concluded that both TITLE and DESCRIPTION better capture the context within the picture, thus offering better descriptions, whereas TAGS are mostly used for organizational purposes. With a different goal, Lipczak and Milios (2010) analyzed the relations between the titles of the objects and the tags used to describe them, finding that users' tagging decisions are influenced by the title's content. They also found that users are less likely to pay attention to tags posted by other users, but rather tend to maintain the consistency of her personal profile of tags. In comparison with these previous studies, our present analysis is much broader and fully automatic, focusing on much larger object collections from different Web 2.0 applications, and addressing a larger set of quality-related properties of the analyzed features. The present work extends our prior efforts (Figueiredo et al., 2009; Almeida et al., 2010) by providing a much more comprehensive analysis, addressing new quality-related aspects such as the semantic properties of different textual features and the diversity of information across different features of the same object. Moreover, in comparison with our previous work, we here perform a much more extensive analysis of the quality of the features for supporting object classification, including the evaluation of two different widely used classification algorithms, and reach more sound conclusions. We also extend our prior studies by assessing the potential of exploiting different textual features for improving a second type of IR service, namely, tag recommendation.

Information quality has also been studied in other contexts. Language models (Zhou & Croft, 2005) as well as block importance models (Cai, Yu, Wen, & Ma, 2004; de Moura, Fernandes, Ribeiro-Neto, da Silva, & Gonçalves, 2010) have been proposed to analyze quality in traditional (textual) Web documents. In particular, de Moura et al. (2010) proposed several metrics as part of an Information Retrieval Model to assess the importance of different information blocks (e.g., menu, title, etc.) of structured Web documents. We here make use of two of these metrics, adapting them to the specific context of textual features associated with multimedia Web 2.0 objects. Aiming at a completely different goal from the original work, we here apply these metrics as heuristics to assess the descriptive and the discriminative powers of the features (see Section 5.3).

[10] http://wikipedia.org.

[11] http://www.flickr.com.

In the context of organizational databases, Strong, Lee, and Wang (1997) introduced four categories of data quality aspects, namely intrinsic, accessibility, contextual and representational. Intrinsic aspects relate to the accuracy, objectivity and reputation of the data. Accessibility aspects express the concern with easiness of access to the data. Contextual aspects relate to how well the data matches task contexts (e.g., their relevance), whereas representational aspects address the easiness of understanding and interpreting the data as well as the consistency and conciseness of data representation. The authors define a data quality problem as any difficulty encountered along one or more of these dimensions that compromises data usefulness. Although we do not deal with information at the database level, our study explores evidence of quality with respect to three of the four aforementioned aspect categories. We characterize feature quality with respect to usage and amount of content, which, according to Strong et al. (1997), fall into the contextual aspect category. We also analyze the semantic properties of feature contents, which are related, at some level (e.g., syntactic correctness), with representational quality aspects. Finally, we assess descriptive and discriminative capacities, which are intrinsic quality aspects, in the sense that they express the accuracy and credibility of the textual features as related to the associated object. In this paper, we show evidence of data quality problems in each analyzed textual feature, providing valuable knowledge to support the design of future Web 2.0 services.

Complementary to our research agenda, some efforts towards applying multimedia IR techniques on Web 2.0 applications are also worth mentioning. Examples include the use of content and textual features to group similar pictures on Flickr (Rege, Dong, & Hua, 2008) or musical artists (Chen et al., 2009) as well as the combination of advanced feature extraction techniques and high level semantic concept modeling for effective annotation of music documents (Shen, Meng, Yan, Pang, & Hua, 2010). These studies provided evidence that multimedia IR techniques and social media are converging on some level. However, textual features still remain as promising data sources for IR tasks on the Web 2.0.

3. Textual features on the Web 2.0

A Web 2.0 *object* is an instance of a media (such as text, audio, video and image) in a given Web 2.0 application. There are various sources of information related to an object, here referred to as its features. In particular, *textual features*, the focus of this study, comprise the self-contained textual blocks that are associated with an object, usually with a well defined topic or functionality (de Moura et al., 2010). We here select four textual features for analysis, namely, TITLE, DESCRIPTION, TAGS, and COMMENTS, which are found in many Web 2.0 applications, although some of them are referred to by different names in some applications. CiteULike, for instance, a social bookmarking website for scientific publications, refers to the DESCRIPTION feature as ABSTRACT, as it usually contains the publication abstract, and name user COMMENTS as REVIEWS. LastFM refers to COMMENTS as SHOUTS.

Textual features may be categorized according to the level of user collaboration allowed by the application. In particular, the textual features analyzed can be either *collaborative* or *restrictive*. Collaborative features may be altered or appended by any user, whereas restrictive ones may only be altered by one user, typically the one who uploaded the object into the system. This property of a feature is here called its *annotation rights*, a generalization of the previously used *tagging rights* notation (Marlow, Naaman, Boyd, & Davis, 2006).

Table 1 shows the annotation rights of each considered textual feature in each analyzed application. TAGS are a collaborative feature in CiteULike, LastFM and YahooVideo, as any user can add tags to an existing object in these applications. In YouTube, in contrast, only the video owner (i.e., the user who uploaded the video) can add tags to it. Moreover, while restrictive in both YahooVideo and YouTube, DESCRIPTION has a collaborative nature in both LastFM and CiteULike. In the former, users can edit the information on an artist or music in a wiki-like manner. In the latter, users can provide different abstracts to the same publication, although we found that to be very rare in our collected dataset. In all four applications, TITLE is restrictive and COMMENTS is collaborative.

We note that some of the applications may automatically fill some of these features when the object is uploaded. YouTube, for instance, adds the name of the object as its TITLE and as TAGS, if these features are not provided. However, it does allow users to remove automatically added TAGS, if a TITLE is provided. In CiteULike, the user may request for the system to extract TITLE and DESCRIPTION from several digital libraries. Nevertheless, in all but one application, users are allowed to change all features, according to their annotation rights. The exception is LastFM, in which the TITLE (i.e., artist names) is automatically inserted via a media player software based on music files metadata.

4. Data collection methodology and datasets characteristics

To perform our study, we built crawlers to sample objects and associated textual features from each application. For YouTube, YahooVideo and LastFM, our crawlers follow a snowball sampling strategy (Goodman, 1961). Each crawler starts with a number of objects as seeds (at least 800). For each object under consideration, it follows links provided by the applications and retrieves all related objects, storing them for future consideration. In YouTube and YahooVideo, the seeds were randomly selected from the list of all-time most popular videos provided by the applications. In LastFM, they were selected among the artists associated with the most popular tags. That is, our LastFM dataset consists of artist pages including music tracks. For CiteULike, which does not provide explicit related object links but makes daily snapshots[12] of stored objects available for download, we collected a random sample of one snapshot.

[12] http://www.citeulike.org/faq/data.adp.

Table 1
Textual feature annotation rights.

	TITLE	TAGS	DESCRIPTION	COMMENTS
CiteULike	Restrictive	Collaborative	Collaborative	Collaborative
LastFM	Restrictive	Collaborative	Collaborative	Collaborative
YahooVideo	Restrictive	Collaborative	Restrictive	Collaborative
YouTube	Restrictive	Restrictive	Restrictive	Collaborative

Table 2
Object classes.

Application	Classes
LastFM	Blues, Classical, Country, R & B, Electronica, Jazz, Pop/Rock, Rap, World
YahooVideo	Action, Animals, Art & Animation, Commercials, Entertainment & TV, Family, Food, Funny Videos, Games, Health & Beauty, How-to, Movies & Shorts, Mus ic, News & Politics, People & Vlogs, Products & Tech., Science & Environment, Sports, Transportation, Travel
YouTube	Autos & Vehicles, Comedy, Education, Entertainment, Film & Animation, Gaming, Howto & Style, Music, News & Politics, Nonprofits & Activism, People & Blogs, Pets & Animals, Science & Technology, Sports, Travel & Events

The YouTube, YahooVideo and LastFM crawlers ran for approximately 2 weeks in July, September, and October 2008, respectively, whereas the sampled CiteULike snapshot is from September 2008. In total, we collected:

- 678,614 articles from CiteULike;
- 193,457 artist pages from LastFM; and
- 227,562 and 211,081 videos from YahooVideo and YouTube, respectively.

Both YouTube and YahooVideo allow users to assign one of a set of pre-determined categories to their videos. In order to use these categories as object classes in our classification experiments (Section 6), our two crawlers collected the category of each object, along with its associated textual features. For LastFM, even though an official category listing is not provided by the application, external data sources can be used to retrieve the musical genre of artists, which, in turn, can be used as object classes. In particular, as performed by Chen et al. (2009), we sampled musical genres from the AllMusic website,[13] which claims to be the largest musical database available. We were able to find categories for 5536 artists in our dataset, a larger categorized sample than previous work (Chen et al., 2009). Similarly to LastFM, CiteULike does not provide any article category listing. Moreover, we are not aware of any reliable article categorization covering multiple scientific areas (as in our sample). Thus, we do not include CiteULike in our current classification experiments, leaving its analysis for future work.

In total, there are 20 and 15 categories in YahooVideo and YouTube, respectively, whereas 9 musical genres were used in the categorization of artists in LastFM. These categories, referred to as object classes, are listed in Table 2.

5. Textual feature characterization

In this section, we analyze the use of the four selected textual features in the four studied applications. Our characterization, performed over our collected datasets,[14] aims at assessing the relative quality of the features as supporting data for IR tasks. Throughout this section, we refer to the set of terms in a textual feature *F* associated with a given object as an *instance* of *F*.

Our analyses cover five aspects, providing answers and insights to the following questions:

What is the fraction of objects with empty feature instances?

Feature usage is of key importance to IR, as under-explored features, that is, features that have empty content in a non-negligible fraction of the objects, may not be a reliable source of information. We thus characterize feature usage, investigating which features are more explored by users in each application.

What is the amount of content present in non-empty feature instances?

If present (i.e., non-empty), a feature instance should also provide enough content about the object it is associated with to be an effective source of information for IR. We characterize the number of unique terms available in non-empty feature instances. We also correlate the amount of content with object popularity to assess whether textual features of more popular objects tend to carry more content.

What are the relative descriptive and discriminative powers of the features?

Effective sources of data for IR tasks should offer a reasonably accurate description of the object's content and/or discriminate objects into different pre-defined categories (for services such as directory organization, and content recommendation)

[13] http://www.allmusic.com/.

[14] We note that, in this study, we used our complete LastFM datasets, with 193,457 objects.

Table 3
Percentage of empty feature instances (C = collaborative feature, R = restrictive feature).

	TITLE	TAGS	DESCRIPTION	COMMENTS
CiteULike	0.53% (R)	8.26% (C)	51.08% (C)	99.96% (C)
LastFM	0.00% (R)	18.88% (C)	53.52% (C)	54.38% (C)
YahooVideo	0.15% (R)	16.00% (C)	1.17% (R)	96.88% (C)
YouTube	0.00% (R)	0.06% (R)	0.00% (R)	23.36% (C)

and into levels of relevance or pertinence (for searching and advertising). Thus, we here adopt two heuristic metrics to *estimate* the descriptive and discriminative powers of the features.

What are the semantic properties of content in each feature?

The semantic properties of the terms in each feature are also important to IR tasks as some grammatical classes (e.g., nouns) are better content descriptors than others (e.g., adverbs). Moreover, the degree of polysemy and noise (i.e., non-existing terms) affecting each feature is also of interest as both properties may be very detrimental to IR. We here characterize the semantics of terms in each feature according to two widely used semantic databases.

How diverse are the content and the information provided by different features of the same object?

We also quantify the amount of content and information diversity across feature instances associated with the same object. These results provide insights into whether the use of multiple textual features may be more effective for IR than the use of any feature in isolation.

Our characterization is driven by our belief that these five aspects are key to determine the quality of a feature (and of multiple features) as data source for effective IR services, although we are not claiming any sufficiency property. Moreover, we argue that these aspects are not equally important to all services (Almeida et al., 2010), which motivates us to analyze each of them separately. Take automatic object classification as an example. If a feature contains a few highly discriminative terms, the total amount of content available might not be as important for service effectiveness. However, classification algorithms typically combine the discriminative power of multiple terms, using different weights, to make decisions. Thus, more content might indeed be beneficial, as it provides more evidence to support classification decisions. However, if many terms are used indiscriminately across most classes, more content might, in turn, be detrimental. In contrast, for services that exploit content-based filtering (e.g., content recommendation), good descriptive power might be more important. In either case, high descriptive or discriminative powers might be of little use if the feature is empty (i.e., with no content) in most objects.

Our characterization results are presented and discussed in Sections 5.1, 5.2, 5.3, 5.4, 5.5. Our main findings are summarized in Section 5.6.

5.1. Feature usage

Table 3 shows, for each feature and application, the percentage of empty feature instances, i.e., objects with no terms in the given feature. The annotation rights of each feature are shown in parenthesis: *C* for collaborative and *R* for restrictive.

The fraction of empty instances is much larger for collaborative features. In fact, some of these features, such as COMMENTS in all applications but YouTube, and DESCRIPTION in LastFM and CiteULike, are vastly under-explored. Even YouTube, with millions of users, has a large fraction of empty COMMENTS, similarly to previous findings on blog and photo sharing systems (Mishne, 2007; Marshall, 2009). The TAGS feature, focus of most previous efforts to enhance Web 2.0 IR services, is also absent in 16% and almost 19% of the objects collected from YahooVideo and LastFM, respectively. Only TITLE, restrictive in all applications, is present in practically all objects.

These results may be partially explained by restrictions imposed by the application, such as the case of TITLE and TAGS in YouTube, which may be automatically filled by the system. However, none of the applications enforces the usage of DESCRIPTION, for instance. Nevertheless, it has a much larger presence in YahooVideo and YouTube, where it is restrictive, than in the other two applications. In fact, in YahooVideo, it has a much larger presence than TAGS, a collaborative feature in that application. This is surprising, as we expected that users would be more willing to provide a few words as TAGS than a few sentences as DESCRIPTION.

An interesting comparison is that of TAGS in CiteULike and LastFM, where the feature is collaborative. Even though both applications motivate the organization of personal libraries, CiteULike does explicitly ask users to associate TAGS[15] to their articles, whereas no such explicit incentive exists in LastFM. Since TAGS are much more explored in CiteULike than in LastFM, it seems like users may, in fact, need extra incentives to use collaborative features, an issue previously raised for tagging systems (Marlow et al., 2006).

Thus, considering solely object coverage in terms of feature presence, the restrictive TITLE offers the best quality of all features, in all applications. This is consistent with a previous analysis of Flickr, which also reported a much larger presence of TITLE than of TAGS and DESCRIPTION (Marshall, 2009). More broadly, the feature annotation rights seem to play a role on its usage.

[15] We note that “no-tag” entries, automatically added by the system, are considered as empty instances in our analysis.

Please cite this article in press as: Figueiredo, F., et al. Assessing the quality of textual features in social media. *Information Processing and Management* (2012), http://dx.doi.org/10.1016/j.ipm.2012.03.003

Table 4
Vocabulary size of non-empty feature instances.

	TITLE				TAGS				DESCRIPTION				COMMENTS			
	AR	μ	*CV*	*Max*	AR	μ	*CV*	*Max*	AR	μ	*CV*	*Max*	AR	μ	*CV*	*Max*
CiteULike	R	7.5	0.40	73	C	4.0	1.33	194	C	65.2	0.5	3890	C	51.9	1.42	449
LastFM	R	1.8	0.47	23	C	27.4	1.49	269	C	90.1	1.06	3390	C	110.2	3.55	22634
YahooVideo	R	6.3	0.39	16	C	12.8	0.52	52	R	21.6	0.71	141	C	52.2	2.51	4189
YouTube	R	4.6	0.43	36	R	10.0	0.60	101	R	40.4	1.75	2071	C	322.3	1.94	16965

In particular, using collaborative TAGS, DESCRIPTION or COMMENTS as single source of data for IR tasks might not be very effective due to the lack of content for a significant fraction of objects.

5.2. Amount of content

In this section, we analyze the amount of content available in each textual feature. Since this analysis is language-dependent, we focus on the English language, applying a simple filter that disregards objects containing fewer than three English stop-words in the whole content of all textual features,[16] as it is. We combined feature instances, such as DESCRIPTION and COMMENTS, before counting stop-words in order to maximize the amount of objects selected for analysis, since it is somewhat expected for combined features to have at least three stop-words, if they are in fact in English. We also focus our analysis on non-empty feature instances.

After removing non-English objects and empty feature instances, we were left with a varying number of objects for the analysis of each feature in each application. This number is larger than 150,000 for TITLE, DESCRIPTION and TAGS in all applications but LastFM, on which it exceeds 86,500 objects. Our cleaned datasets also include 152,717, 76,627, 6037 and 76 objects with COMMENTS in YouTube, LastFM, YahooVideo and CiteULike, respectively. We removed terms containing only non-alphanumeric characters as well as stop-words, and applied the Porter stemming algorithm[17] to remove affixes.

We characterize the *number of distinct terms* in each feature instance, here referred to as its *vocabulary size*, which represents the amount of content available for use by IR services. These results are summarized in Table 4, which presents the mean μ, coefficient of variation *CV* (ratio of standard deviation to the mean), and maximum values.[18] The *AR* columns show the features' annotation rights.

In general, TITLE has instances with the smallest vocabulary, followed by TAGS, DESCRIPTION and COMMENTS. This ordering reflects the inherent degree of verbosity expected from each feature: in general, TITLES and TAGS are expected to be shorter than DESCRIPTION, which in turn, is expected to be smaller than COMMENTS. Moreover, *with a few exceptions*, non-empty instances of collaborative features tend to have vocabularies with larger average sizes and high variability (CV). For instance, the table shows that, considering all four features in the same application, COMMENTS instances, if present (i.e., not empty), have the largest vocabularies, on average (except in CiteULike), also exhibiting very large coefficients of variation. Similarly, instances of DESCRIPTION tend to have larger vocabularies in CiteULike and LastFM, where the feature is collaborative, than in the video applications, where it is restrictive. The same holds if we compare TAGS in LastFM (collaborative) and in YouTube (restrictive). Another important factor seems to be the popularity of each feature among users of the application: COMMENTS instances tend to carry much less content in YahooVideo and CiteUlike, where the feature is greatly under-explored (Section 5.1), than in the other two applications. These results are consistent with the ones previously observed for TITLE, TAGS and DESCRIPTION in Flickr (Marshall, 2009).

In CiteULike, specifically, the TAGS feature, despite collaborative, tends to have instances with smaller vocabularies, thus providing less content, than TITLE, unlike in the other applications. This may be explained by articles containing very long titles. It may also be due to the stabilization of tagging vocabulary with time, previously reported by Golder and Huberman (2006) and by Li et al. (2008). Nevertheless, a small fraction of objects have a very large number of different tags (up to 194 terms). Unlike in all other applications, DESCRIPTION has the largest instances, on average, possibly because they typically carry article abstracts, with more content than the poorly used COMMENTS.

In LastFM, TITLE has a much stronger bias towards smaller vocabularies than in the other applications. This is expected since the feature usually contains artist names. In contrast, the TAGS feature tends to carry much more content, with an average of 27 terms per object and reaching as many as 269 terms per object. COMMENTS is the feature with the largest amount of content, on average, although this is biased towards very popular artists (see discussion below). DESCRIPTION instances also have large vocabularies, larger than in the other applications, likely due to its collaborative nature, as users collectively write a text, in a wiki-like manner, describing the artists.

YahooVideo and YouTube show similar trends, although the vocabularies of instances of both DESCRIPTION and COMMENTS tend to be much larger in YouTube, possibly due to its larger audience. In contrast, the instances of TITLE and TAGS have larger vocab-

[16] http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/stop_words.

[17] http://tartarus.org/~martin/PorterStemmer/.

[18] We also analyzed the cumulative distributions of vocabulary size, choosing to omit them as they lead to similar conclusions as the results in Table 4.

Please cite this article in press as: Figueiredo, F., et al. Assessing the quality of textual features in social media. *Information Processing and Management* (2012), http://dx.doi.org/10.1016/j.ipm.2012.03.003

ularies in YahooVideo. Whereas the differences in TAGS may be due to the higher degree of collaboration in YahooVideo, the differences in TITLE may reflect differences in usage patterns.

A relevant issue for IR services is whether the amount of content in each feature tends to be larger in more popular objects. We quantified the correlation, measured by the Pearson coefficient ρ (Jain, 1991), between the size of the vocabulary in each feature instance and the object popularity. Object popularity is estimated by the number of views in YahooVideo and YouTube, by the number of listeners in LastFM, and by the number of posters in CiteULike. In general, we found non-negligible positive correlations only for collaborative features that are largely adopted by users, such as COMMENTS in LastFM ($\rho = 0.5$) and YouTube ($\rho = 0.24$), TAGS in CiteULike ($\rho = 0.23$) and LastFM ($\rho = 0.41$), and DESCRIPTION in LastFM ($\rho = 0.25$). Exceptions are DESCRIPTION in CiteULike ($\rho = 0.006$), and TAGS in YahooVideo ($\rho = 0.003$). The latter may be due to the lack of incentives to YahooVideo users for tagging objects.

In summary, our results show that, if present, collaborative features, in general, tend to carry more content than restrictive ones. This holds even for the same feature (e.g., TAGS) across different applications. Moreover, although in general collaborative features tend to bring more content in more popular objects, the measured correlations, although non-negligible in several cases, as discussed above, are of at most intermediate degree ($\rho \leqslant 0.5$). Thus, other aspects, such as explicit incentives given by the application, how easy and attractive it is for users to edit a feature, which in turn is related to the application's interface, as well as the social networks developed within the application might also play important roles on how features are explored by users. Thus, all such factors should also be considered in the design of future Web 2.0 applications and services.

5.3. Descriptive and discriminative powers

Multiple strategies can be adopted to estimate the descriptive and discriminative powers of a feature. One possibility is to rely on manual assessment by volunteers. However, such user experiments are typically quite costly and have an unavoidable degree of subjectivity that might affect the results. Moreover, in the particular case of discriminative power, the results depend on the aspect under evaluation (e.g., relevance of an object to a query or pertinence to a category or topic), thus requiring assessment in the context of a specific service. An alternative strategy is to use *heuristics* to capture (to some extent) these powers. As an advantage, heuristics can be applied to much larger object samples at lower cost. However, heuristics inevitably have limitations, such as focusing on specific issues and thus ultimately capturing only partially the target aspect, as well as having biases that impact their effectiveness in specific scenarios. In spite of such possible limitations, heuristics can still provide valuable insights into the quality of each feature.

That said, we here choose to apply heuristic metrics to assess both descriptive and discriminative powers of the features. A number of different heuristics, each one with its own characteristics and limitations, are available in the literature. For example, one could use the distribution of term frequencies, information gain, term entropy (Mitchell, 1997), and concentration of terms on object categories (for assessing discriminative power) (Almeida et al., 2010), to name a few options. More sophisticated metrics, such as cohesiveness, coverage and extent, have also been proposed to evaluate the quality of tag clouds (Venetis et al., 2011).

We here choose to make use of two heuristic metrics previously proposed as part of an Information Retrieval model to assert the *importance* of different blocks in traditional structured webpages (i.e., webpages with specific information blocks, such as a menu and a title) (de Moura et al., 2010). We selected these metrics because they can be easily adapted to the present context of textual features of Web 2.0 (multimedia) objects. One can see different textual features as defining different content blocks. For instance, on YouTube, the comments posted by users about a video are placed near to each other, in a common (COMMENTS) block. Thus, we here consider each feature instance associated with an object as an information block. The two metrics, here referred to as *Average Feature Spread* and *Average Inverse Feature Frequency*, are used to estimate the descriptive power and discriminative power, respectively, of each analyzed feature.

5.3.1. Average Feature Spread

Our estimation of the descriptive power of each feature is based on a heuristic metric called *Average Feature Spread*, which is computed as follows. We start by defining the spread of a term in a given object. Let o be an object in the collection O, t a term which appears in at least one feature instance f associated with o, and T_f the set of *distinct* terms that appear in feature instance f. The *term spread*, $TS(t,o)$, measures the number of feature instances associated with o which contain t, that is:

$$TS(t,o) = \sum_{f \in o} i, \quad \text{where } i = \begin{cases} 1 & \text{if } t \in T_f \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

The assumption behind $TS(t,o)$ is that the larger the number of features of o that contain t, the more related to o's content t is. For instance, if the term "Sting" appears in all features of a given video, there is a high chance that the video is related to the famous singer.

Next, we define the *Feature Instance Spread* of a feature instance f associated with an object o, $FIS(f,o)$, as the average term spread across all terms in f. That is, given $|T_f|$ the number of distinct terms in instance f, $FIS(f,o)$ is defined as:

$$FIS(f,o) = \sum_{t \in T_f} \frac{TS(t,o)}{|T_f|}. \tag{2}$$

Table 5
AFS values with 90% confidence intervals. (Best results, at 90% confidence level, shown in bold.)

	CiteULike	LastFM	YahooVideo	YouTube
AFS computed using all available terms				
TITLE	**1.92 ± 0.0011**	**2.65 ± 0.0056**	**2.22 ± 0.0019**	**2.54 ± 0.0031**
TAGS	1.63 ± 0.0020	1.33 ± 0.0015	1.83 ± 0.0019	2.07 ± 0.0027
DESCRIPTION	1.13 ± 0.0003	1.21 ± 0.0012	1.51 ± 0.0015	1.72 ± 0.0025
COMMENTS	–	1.21 ± 0.0013	–	1.12 ± 0.0007
AFS computed using only top-5 terms with highest TS				
TITLE	**2.08 ± 0.0012**	**2.65 ± 0.0056**	**2.41 ± 0.0021**	**2.63 ± 0.0032**
TAGS	1.73 ± 0.0022	2.12 ± 0.0047	2.39 ± 0.0021	2.58 ± 0.0031
DESCRIPTION	2.05 ± 0.0012	2.30 ± 0.0040	2.31 ± 0.0024	2.51 ± 0.0032
COMMENTS	–	2.08 ± 0.0051	–	2.33 ± 0.0036

If feature instance *f* is empty, we define *FIS*(*f*, *o*) = 0. We note that different *filtering criteria*, such as taking only the *k* terms with the largest *TS*, could be applied to compute *FIS*.

The *Feature Instance Spread* heuristic assesses how the terms of a given feature instance *f* are related to the content of instances of other features associated with the same object *o*. It is thus a heuristic to estimate how the feature instance *f* is related to *o*'s content, being here used as an estimate of the average descriptive power of feature instance *f* for the given object *o*.

Following this reasoning, the descriptive power of a feature *F* in the object collection *O* can be captured by averaging the values of *FIS* across all objects in *O*. We refer to this metric as the *Average Feature Spread* of *F*, *AFS*(*F*). Given each object *o* in the collection *O* and the instance of *F* associated with it, *f*, the *AFS*(*F*) is computed as:

$$AFS(F) = \frac{\sum_{o \in O} FIS(f, o)}{|O|} \tag{3}$$

5.3.2. Average Inverse Feature Frequency

The *Average Inverse Feature Frequency (AIFF)* metric, here used to estimate the *discriminative power* of a feature, builds on a small variation of the *IDF* metric (Baeza-Yates & Ribeiro-Neto, 2011), called *Inverse Feature Frequency (IFF)*. The *IFF* considers instances of a feature *F* as a separate "document collection". In other words, given a feature *F* with |*F*| non-empty feature instances,[19] and a term *t* that occurs in at least one instance of *F*, the *IFF*(*t*, *F*) of term *t* in *F* is defined as:

$$IFF(t, F) = \log\left(\frac{|F|}{Frequency(t, F)}\right), \tag{4}$$

where *Frequency*(*t*, *F*) is the number of instances of *F* in which the term *t* appears. Once again, we could apply different filtering criteria, such as disregarding very unpopular terms that might undesirably inflate the IFF value (see discussion below). The *IFF* metric assesses how much information carries *the occurrence of a given term in a given feature*. The assumption is that terms occurring in many instances of the feature are bad content discriminators. For example, whereas the occurrence of "music" in a TITLE of a YouTube object brings little information about its content in relation to other music videos, the occurrence of "Sting" may be more useful to discriminate it from other objects.

We then define the *Average Inverse Feature Frequency* of feature *F*, *AIFF*(*F*), as the average *IFF* over all terms that occur in all instances of *F*. It is thus a heuristic to estimate the discriminative power of the feature *F* in the object collection. Given T_F the complete term vocabulary of feature *F* (i.e., considering all instances of *F* in the object collection), and $|T_F|$ its size, the *AIFF*(*F*) is computed as:

$$AIFF(F) = \frac{\sum_{t \in F} IFF(t, F)}{|T_F|} \tag{5}$$

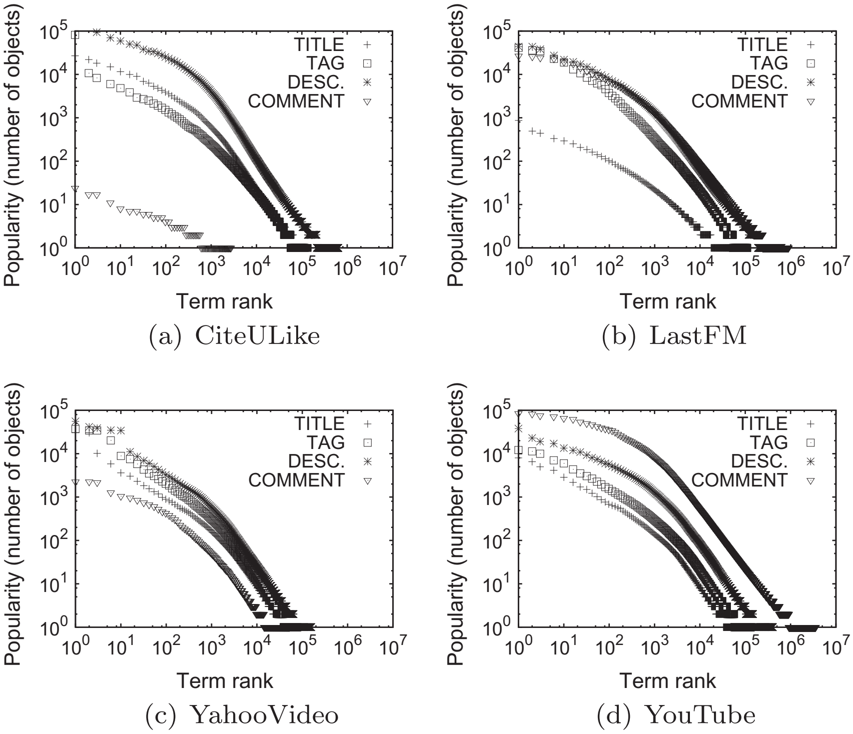
5.3.3. Results

We computed *AIFF* and *AFS* values for each feature in all four applications using our stemmed datasets and considering only objects with non-empty instances of *all* features. Given the negligible fractions of non-empty COMMENTS in YahooVideo and CiteULike, we disregarded this feature in both applications. Thus, *AFS* and *AIFF* are computed only for objects with all four features in LastFM and YouTube, and with TITLE, TAGS and DESCRIPTION in CiteULike and YahooVideo. *AFS* and *AIFF* results, along with corresponding 90% confidence intervals, are shown in Tables 5 and 6, respectively. Each table shows two blocks of results: one computed using all available terms (upper block) and the other computed after filtering some of the terms (lower block). For each block, best results across all analyzed features in each application (including statistical ties measured at the 90% confidence level when using the Mann–Whitney's U and Student's *T* tests (Jain, 1991)) are shown in bold.

[19] We note that |*F*| is not necessarily equal to |*O*|, since some objects in *O* may have empty feature instances of *F*.

Table 6
AIFF Values with 90% confidence intervals. (Best results, at 90% confidence level, shown in bold.)

	CiteULike	LastFM	YahooVideo	YouTube
AIFF computed using all available terms				
TITLE	**11.04 ± 0.0096**	**10.09 ± 0.0109**	**10.16 ± 0.0118**	10.29 ± 0.0116
TAGS	10.95 ± 0.0089	9.52 ± 0.0097	9.86 ± 0.0121	10.24 ± 0.0092
DESCRIPTION	10.78 ± 0.0053	9.76 ± 0.0047	10.07 ± 0.0092	10.25 ± 0.0059
COMMENTS	–	9.62 ± 0.0041	–	**10.33 ± 0.0021**
AIFF computed ignoring unpopular terms				
TITLE	7.61 ± 0.024	**6.98 ± 0.046**	**7.05 ± 0.026**	**7.27 ± 0.024**
TAGS	**7.85 ± 0.023**	6.39 ± 0.025	6.86 ± 0.024	7.14 ± 0.019
DESCRIPTION	7.02 ± 0.019	6.12 ± 0.018	6.74 ± 0.022	6.89 ± 0.015
COMMENTS	–	6.09 ± 0.015	–	6.65 ± 0.0091



(a) CiteULike (b) LastFM

(c) YahooVideo (d) YouTube

Fig. 1. Term popularity distributions.

As shown in Table 5 (upper block), *AFS* results provide a consistent ranking of the features in all four applications. According to this heuristic, TITLE is the most descriptive feature, followed by TAGS, DESCRIPTION and, if considered, COMMENTS. One could argue that, as heuristics, *FIS*, and ultimately *AFS*, have biases towards larger values for smaller feature instances. In other words, larger instances might have lower *FIS* values simply because there is a higher chance that most of their terms are not included in the other (smaller) features. In order to reduce the impact of this possible bias on *AFS* results, we recomputed the *FIS* and *AFS* values considering only the *k* terms with largest *TS* values. Table 5 (lower block) shows the results for $k = 5$. Note that, despite the smaller differences among *AFS* values of different features of the same application, TITLE still emerges as the most descriptive feature according to this heuristic. Indeed, the same relative order of the features holds for both video applications: TITLE is followed by TAGS, DESCRIPTION and COMMENTS. On CiteULike and LastFM, on the other hand, DESCRIPTION has a larger *AFS* (computed over the top-5 *TS* terms) than TAGS, implying that, on those two applications, DESCRIPTION contains a few very descriptive terms (according to the *AFS* heuristic). On CiteULike, at least, this result is not completely surprising given that DESCRIPTION instances typically contain the abstracts of the articles, which are expected to carry some terms that are strongly related to the article's content. On LastFM, in turn, the wiki-like manner in which users collaboratively edit the DESCRIPTION of the objects and the nature of its contents (information on artists and their songs) might favor the appearance of some very descriptive terms (e.g., the name of the artist itself or terms related to her musical genre such as *rock*). Considering the results computed over all available terms, whereas the lower *AFS* value for DESCRIPTION might reflect an inherent bias

Table 7
Percentage of terms found in the dictionary.

	TITLE (%)	TAGS (%)	DESCRIPTION (%)	COMMENTS (%)
CiteULike	84.7	61.3	83.1	63.1
LastFM	61.0	74.1	71.9	63.0
YahooVideo	78.2	79.0	80.0	71.0
YouTube	73.3	72.4	72.9	64.5

of the metric, it might also imply that, in spite of the existence of a few very descriptive terms, DESCRIPTION, being typically a larger feature containing full sentences, might also carry many poorly related (or unrelated) terms, which ultimately reduces its overall descriptive power.

In contrast, Table 6 (upper block) shows less distinction in the *AIFF* values across features in any application, with no consistent ranking. In order to understand why the *AIFF* metric is not able to clearly distinguish one feature from the other, we plotted the term popularity distribution of each feature, considering all instances of the feature. In other words, the popularity of term *t* in a feature *F* is assessed by the number of objects with instances of *F* containing *t*. Fig. 1 shows that the distributions are heavy-tailed in all applications, thus containing a large number of terms with very low popularity. These terms have very large *IFF* values, and end up boosting the *AIFF* of all features somewhat similarly.

Thus, we recomputed the *AIFF* values considering only terms that appeared in more than 50 feature instances.[20] The results, also presented in Table 6 (lower block), show a more clear distinction between the features (reported values are statistically different at 90% confidence level). Overall, we observe that, according to the *AIFF* heuristic, TITLE is the most discriminative feature, followed by TAGS, DESCRIPTION and, if considered, COMMENTS, a ranking that is consistent with the *AFS* results. One exception is CiteULike, where the TAGS feature has the best discriminative power (according to the recomputed *AIFF*). The superiority of TAGS over TITLE (in terms of *AIFF*) in CiteULike might be explained by the fact that many words appearing in titles of scientific articles are somewhat general (e.g., "method" or "algorithm" for articles in Computer Science) and tend to be reused in many objects, which lowers their *IFF* values.

It should be noticed that, although statistically different, the relative difference in values between some of the results are small (e.g. *AIFF* values for TITLE and TAGS in CiteULike on Table 6), so that the practical implications for some specific IR services may depend more on other factors such as the amount of content and diversity in content and information (as we shall discuss on Section 5.5.1). Such factors, for instance may have a larger impact in classification results (see Section 6). On the other hand, some differences are in fact very large (e.g. YouTube *AFS* values between TITLE and DESCRIPTION in Table 5), with significant implications on the effectiveness of some services such as recommendation, as we shall see in Section 7.

5.4. Semantic properties

We now turn our attention to the semantic properties of each feature's vocabulary, focusing, as in Section 5.2, on the English language. Golder and Huberman (2006) reported that tagging systems may be very affected by synonymy (multiple words with the same meaning) and polysemy (one word with multiple meanings). These ambiguous terms may have a degenerative impact on the effectiveness of IR tasks. For example, the results produced for a query may not contain a relevant object if it is described by terms that are synonyms of the query terms.

As in (Suchanek et al., 2008), we use the Wordnet (Fellbaum & NetLibrary, 1998) and Yago (Suchanek, Kasneci, & Weikum, 2007) semantic databases to analyze the semantic properties of each feature's vocabulary. Unlike in that study, we here analyze *distinct* terms in their original form, applying only plural stemming. Wordnet is a lexical database of the English language in which words (nouns, verbs, adjectives, etc.) are grouped into sets of synonyms, called *synsets*. These sets are connected based on lexical and semantic relations. Yago is a similar database but describes and relates entities. For example, "Marilyn Monroe", an American Actress, is related to other American Actresses, such as "Jane Fonda". We use Yago in order to determine if a term is a proper name. Collectively, we refer to both databases as *the dictionary*.

Table 7 presents the percentage of terms with at least one entry (i.e., a certain grammatical class for a given term) in the dictionary. The percentage of terms with no entry in the dictionary is here taken as an estimate of the amount of noise in each feature. Table 8 presents the distribution of terms that are present in the dictionary across grammatical classes. Also note that, each line in Table 8 does not add up to 100% as the same term may be categorized into multiple grammatical classes (e.g., "dance", which is counted as both verb and noun).

In consistency with the results for TAGS reported in (Suchanek et al., 2008), we find that a reasonable fraction (15–39%) of the terms in *all* features and applications are not found in the dictionary, and are here considered as *noise*. The fraction is typically larger for COMMENTS, being 29% in YahooVideo and 35–37% in the other applications. It is also worth noting the large fraction of terms not found in the dictionary in LastFM TITLES: indeed, a large fraction of the artist and band names, typical content of this feature, are not found in the dictionary. Moreover, out of the terms with known meaning, most of them (55–80%) are nouns, typically good content descriptors. There is also a non-negligible fraction of (known) proper names,

[20] We experimented with thresholds equal to 10, 50, 100 and 1000, reaching similar *AIFF* ranks for thresholds equal to or greater than 50.

Table 8
Percentage of terms across grammatical classes in the dictionary.

Feature	Noun (%)	Verb (%)	Adjective (%)	Adverb (%)	Proper name (%)
CiteULike					
TITLE	66.7	17.2	27.2	2.3	3.6
TAGS	76.3	16.1	17.6	0.6	5.7
DESCRIPTION	54.7	20.8	33.2	6.1	3.6
COMMENTS	56.6	21.5	28.2	6.4	5.7
LastFM					
TITLE	72.4	23.8	19.4	3.4	15.0
TAGS	65.3	26.8	41.0	8.8	12.7
DESCRIPTION	63.6	23.4	31.5	8.5	8.7
COMMENTS	61.2	28.3	30.5	11.1	10.2
YahooVideo					
TITLE	80.1	28.3	22.7	5.5	7.8
TAGS	76.2	23.1	20.4	5.0	9.4
DESCRIPTION	68.2	30.9	24.2	7.1	6.7
COMMENTS	66.7	32.3	30.8	11.0	8.9
YouTube					
TITLE	76.1	25.4	25.3	4.4	10.9
TAGS	77.8	26.8	20.7	3.5	13.0
DESCRIPTION	67.6	27.4	27.0	7.3	9.8
COMMENTS	61.9	29.0	28.9	9.1	8.8

Table 9
Percentage of terms with at least *k* meanings.

	k = 5				*k* = 10			
	TITLE (%)	TAGS (%)	DESCRIPTION (%)	COMMENTS (%)	TITLE (%)	TAGS (%)	DESCRIPTION (%)	COMMENTS (%)
CiteULike	26.9	15.2	28.4	21.8	9.5	5.2	11.1	9.9
LastFM	17.5	25.6	25.5	23.3	7.3	11.4	12.1	10.9
YahooVideo	28.0	24.7	33.6	31.1	15.7	11.6	16.1	15.9
YouTube	23.6	22.1	27.0	23.6	9.9	8.1	12.7	11.1

which also tend to offer good descriptions, in all applications and particularly in LastFM. This is expected since our LastFM dataset consists of artist pages.

We also measure the percentage of terms in each feature with *k* or more meanings, for values of *k* greater than 1. Table 9 shows the results for *k* equal to 5 and 10. Note that the numbers lie between 5% and 16% for *k* equal to 10, indicating that some degree of polysemy affects *all* features, in all applications, which might ultimately impact the effectiveness of IR tasks. In case these tasks employ any textual context analysis, this impact might be reduced in features which typically contain complete sentences (e.g., DESCRIPTION) as opposed to isolated words (e.g., TAGS).

We note that semantic databases, such as the ones used in this analysis, are not expected to cover the meanings of every single term used in social media applications. For example, Internet slang will most likely not be covered by neither database. Another possible shortcoming is that analyzing terms independently of their context will increase the amount of ambiguous terms, due to polysemy. Moreover, even terms not found in a dictionary may still be useful for different IR tasks. For instance, the co-occurrence of terms with known and unknown/rare meanings may help classification tasks. Such terms may also help users find very specific content in query-based search. Nevertheless, we can still conclude that: (1) a reasonable fraction of terms in all features and applications have no meanings according to the considered databases; (2) most terms with at least one known meaning can be classified as nouns and/or proper names, which are good descriptors of content, and (3) a non-negligible degree of textual ambiguity affects all features and applications, and its impact might be significant when content analysis is performed by taking terms independently.

5.5. Content and information diversity

So far we have characterized different aspects of each feature separately. We now investigate whether different features associated with the same object contribute with different pieces of content (i.e., terms) and with different pieces of information (i.e., semantic meaning) about the object. Towards that goal, we characterize the amount of different terms across features (Section 5.5.1) as well as the semantic similarity between their contents (Section 5.5.2). For both analyses, we focus on the English language, as in Section 5.2. Content diversity is analyzed over the stemmed data sets, whereas information diversity is characterized over the original (non-stemmed) collections.

Table 10
Average content similarities (Jaccard coefficients) between non-empty feature instances with 90% confidence intervals.

	CiteULike	LastFM	Yahoo	YouTube
Average Jaccard coefficients computed using the N = 5 terms with highest TS × IFF				
TITLE × TAGS	0.13 ± 0.0005	0.07 ± 0.0008	0.52 ± 0.0013	0.36 ± 0.0011
TITLE × DESCRIPTION	0.32 ± 0.0007	0.22 ± 0.0008	0.40 ± 0.0012	0.28 ± 0.0010
TAGS × DESCRIPTION	0.14 ± 0.0005	0.14 ± 0.0010	0.44 ± 0.0013	0.33 ± 0.0011
TITLE × COMMENTS		0.13 ± 0.0009		0.14 ± 0.0007
TAGS × COMMENTS		0.10 ± 0.0010		0.18 ± 0.0008
DESCRIPTION × COMMENTS		0.19 ± 0.0015		0.16 ± 0.0008
Average Jaccard coefficients computed using all terms				
TITLE × TAGS	0.09 ± 0.0003	0.01 ± 0.0002	0.33 ± 0.0010	0.26 ± 0.0009
TITLE × DESCRIPTION	0.09 ± 0.0002	0.03 ± 0.0003	0.20 ± 0.0008	0.15 ± 0.0008
TAGS × DESCRIPTION	0.03 ± 0.0001	0.05 ± 0.0002	0.21 ± 0.0007	0.15 ± 0.0007
TITLE × COMMENTS		0.02 ± 0.0004		0.02 ± 0.0001
TAGS × COMMENTS		0.03 ± 0.0002		0.03 ± 0.0001
DESCRIPTION × COMMENTS		0.04 ± 0.0002		0.03 ± 0.0002

5.5.1. Content diversity

We assess the diversity of content across features of the same object by quantifying the *similarity* between their contents. Content similarity between two feature instances is estimated in terms of term co-occurrence using the Jaccard coefficient. Given two sets of items T_1 and T_2, the Jaccard coefficient is computed as:

$$J(T_1, T_2) = \frac{|T_1 \bigcap T_2|}{|T_1 \bigcup T_2|} \tag{6}$$

We compute the content similarity between two features associated with the same object using as input sets the *N* most highly ranked terms from each feature instance based on the product of the *TS* and *IFF* metrics.[21]

Table 10 shows the average similarity, along with corresponding 90% confidence intervals, between all pairs of features in all four applications for $N = 5$ and when all terms of each feature instance are considered.[22] As in the previous section, we disregard COMMENTS in CiteULike and YahooVideo. According to the table, there seems to be more similarity between restrictive features (e.g., TITLE and TAGS in YouTube), as the same user tends to use common words in them. The exception is TAGS in YahooVideo, which, despite collaborative, shares great similarity with the restrictive TITLE and DESCRIPTION features, perhaps an indication that TAGS are often used only by the video owner herself. These results are consistent with those reported in (Lipczak & Milios, 2010) which show a significant overlap between TITLE and TAGS in CiteULike and Delicious objects. Nevertheless, we should note that the average Jaccard coefficients are all under 0.52. Thus, although there is some co-occurrence of highly ranked terms across features, each feature may still bring new (possibly relevant) content about the object.

5.5.2. Information diversity

The Jaccard coefficient used in the previous section does not capture the fact that different words may be used to describe the same underlying meaning, even when using stemmed datasets. This is because it only analyzes co-occurrence of terms, and thus does not consider synonyms. It also does not capture variations in the degree of specificity of the terms: while some users may apply very specific terms (e.g., *poodle*) to describe a concept, others may choose more general terms (e.g., *dog*). We thus extend our analysis to quantify the *semantic similarity* between (non-stemmed) words occurring in different features associated with the same object.

In order to measure semantic similarity between sets of words, we again make use of the Wordnet dictionary, and more specifically of its *hyponym* semantic relation: a word *w* is a hyponym of another word *v* if the semantics of *w* is included in (i.e., is narrower than) the semantics of *v* (*v* is said to be *w*'s hypernym). For instance, *cat* is a hyponym of *animal*, and *chair* is a hyponym of *furniture*. Using the hyponym relation, we can infer a taxonomy, that is, a hierarchical structure describing the relations between word meanings. The inferred taxonomy can then be used to measure semantic similarity. Fig. 2 shows a small example which illustrates a portion of the taxonomy inferred from the Wordnet hyponym relations. The closer two words are in the taxonomy tree, the more similar they are. In the example, *settee* is more similar to *divan* than to *yoke*.

Words can and frequently do have multiple senses, which may or may not be related. The word *chair*, for instance, means either a seat or a person in charge of a meeting, organization or conference. Thus, even though *professorship* is not very related to *wheelchair*, they are both hyponyms of *chair*. Therefore, the taxonomy is built with relations between *word senses*, and not simply between the words themselves. That is, the same word may appear multiple times in the taxonomy, each occurrence representing one of its *senses*. This strategy avoids relations between unrelated *senses*.

[21] It is intuitive to see that the top *TS* × *IFF* terms will be the ones with a higher chance of occurring across features, since: (1) *TS* captures the spread of a term across features; and, (2) *IFF* captures the uniqueness of a term for an object.

[22] Results for $N = 15$, 30 and 50 are in the same range.

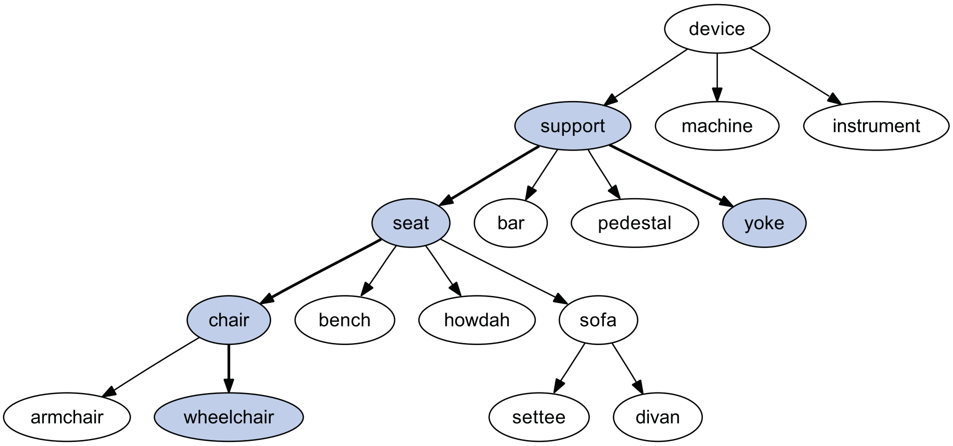


Fig. 2. Example of a portion of the taxonomy inferred from the wordnet hyponym relations.

Table 11
Average information similarities between non-empty feature instances with 90% confidence intervals.

	CiteULike	LastFM	Yahoo	YouTube
Average information similarities between feature instances in our real collections				
TITLE × TAGS	1.50 ± 0.0011		1.44 ± 0.0013	1.39 ± 0.0024
TITLE × DESCRIPTION	1.55 ± 0.0007		1.50 ± 0.0012	1.42 ± 0.002
TAGS × DESCRIPTION	1.44 ± 0.0010	1.41 ± 0.0007	1.47 ± 0.0008	1.40 ± 0.0015
TITLE × COMMENTS				1.39 ± 0.0016
TAGS × COMMENTS		1.41 ± 0.0008		1.38 ± 0.0012
DESCRIPTION × COMMENTS		1.42 ± 0.0008		1.41 ± 0.0010
Average information similarities between feature instances in randomized collections				
TITLE × TAGS	1.14 ± 0.00305		1.22 ± 0.0170	1.23 ± 0.0143
TITLE × DESCRIPTION	1.13 ± 0.0156		1.21 ± 0.0157	1.23 ± 0.0147
TAGS × DESCRIPTION	1.12 ± 0.0219	1.19 ± 0.0128	1.19 ± 0.0115	1.21 ± 0.0119
TITLE × COMMENTS				1.20 ± 0.0147
TAGS × COMMENTS		1.20 ± 0.0173		1.20 ± 0.0114
DESCRIPTION × COMMENTS		1.16 ± 0.0154		1.18 ± 0.0111

Given the taxonomy, we make use of the Leacock–Chodorow (*LC*) metric (Fellbaum & NetLibrary, 1998) in order to measure semantic similarity. Given two *word senses* s_1 and s_2, *LC* estimates the semantic similarity between them by taking the logarithm of the length of the shortest path connecting them in the taxonomy, normalized by the maximum depth of that taxonomy, that is:

$$LC(s_1, s_2) = -\log\left(\frac{taxonomy\_dist(s_1, s_2)}{2 \times maxdepth}\right) \tag{7}$$

where *taxonomy_dist* is the length (in nodes) of the shortest path between s_1 and s_2, and *maxdepth* is the maximum depth of the taxonomy. Since path length is measured in nodes (as opposed to edges), the metric is defined even when the comparison is between a word sense and itself: the path between a node and itself includes one node. Note that, for a given taxonomy, the closer two word senses, the larger their *LC* values and thus the stronger their semantic similarity (as captured by the *LC* heuristic).

To illustrate the computation of the *LC* metric, let's use the taxonomy shown in Fig. 2 to estimate the semantic similarity between word senses *wheelchair* and *yoke*.[23] The maximum depth of the taxonomy is 5, whereas the shortest path connecting the two word senses also contains five nodes (*wheelchair, chair, seat, support* and *yoke*). Thus, the semantic similarity is given by:

$$LC(wheelchair, yoke) = -\log\left(\frac{5}{2 \cdot 5}\right) = -\log\frac{1}{2}.$$

We can then estimate the semantic similarity between two *words* w and v, $sim(w, v)$, by computing the Leacock–Chodorow similarity between each sense s_1 of w and every sense s_2 of v, and taking the maximum. In other words:

$$sim(w, v) = \max_{s_1 \in S_w,\ s_2 \in S_v} LC(s_1, s_2), \tag{8}$$

[23] In this example, we ignore the portion of the taxonomy omitted from the figure.

where S_u is the set of senses of word u.

Given this definition, we can compare the sets of words that appear in different features associated with the same object by measuring the average similarity of all pairs of words in those features. Thus, given two feature instances f_1 and f_2 and their corresponding sets of *distinct* words, W_{f_1} and W_{f_2}, we define the *information similarity* between f_1 and f_2, $IS(f_1,f_2)$, as follows:

$$IS(f_1,f_2) = \frac{\sum_{w \in W_{f_1}, v \in W_{f_2}, w \neq v} sim(w,v)}{|W_{f_1}| \times |W_{f_2}| - |W_{f_1} \cap W_{f_2}|} \quad (9)$$

Note that words that appear in both feature instances are excluded from the *IS* computation as their similarities are captured by the Jaccard Coefficient. Such words would, otherwise, artificially inflate the similarity between feature instances. The greater the value of *IS* between feature instances f_1 and f_2, the higher the semantic similarity between their contents.

Average values of information similarity between different feature instances, along with corresponding 90% confidence intervals, are shown in Table 11. These numbers are computed considering only *nouns* that have at least one word sense since the taxonomy only contains nouns.[24] Once again, we disregard COMMENTS in CiteULike and YahooVideo. We also disregard TITLES in LastFM as they contain mostly names of artists or bands (i.e., proper names) which are not captured in the Wordnet hyponym relations.

In general, the semantic similarity between words in different features of the same object can be considered low, with average *IS* values under 1.6. In order to better understand these results, we further investigated the values of the Leacock–Chodorow coefficient measured in our data sets. We found that, on average, pairs of words appearing in different features of the same object have a *taxonomy_dist* (given their different senses) of approximately 9, which can be considered large. This is also the mode of the distribution of measured *taxonomy_dist* values. Some examples of pairs of words that are related in the taxonomy at such a distance are "video" and "hypocrisy", "thing" and "prosecutor", and "February" and "video", which clearly have very weak (if any) semantic similarity. We also found pairs of words, such as "thing" and "law", which, according to Wordnet, are at a distance equal to 3 in the taxonomy, but whose semantic relation seems very weak to us. In fact, we observed that most pairs of words are connected to each other via other words of very general nature, such as "thing", "entity", and "person".

More broadly, we also interpret the *IS* values reported in Table 11, taking as baseline, for each application, a uniform random model consisting of 1000 objects. These randomized collections are built as follows: considering each feature separately, we assign words to each instance of the feature, selecting them with equal probability from the feature's word vocabulary (extracted from the corresponding original data set), while keeping the distribution of the number of words per feature instance the same as in the original collection. Note that, by independently selecting words to different features, we are indeed breaking any semantic dependency there might exist among feature instances of an object. Average *IS* values and 90% confidence intervals for the randomized collections are also shown in Table 11. In general, they are only slightly smaller than corresponding measures for the original data sets. In other words, the semantic similarity between words across features of the same object is indeed low, only slightly higher than the similarity that arises, by chance, in the randomized collections.[25] These results provide evidence that textual features associated with the same object do carry not only different content but also different information.

5.6. Summary of our findings

Our characterization results may be summarized into five main findings. First, all four features but TITLE, have a non-negligible fraction of empty instances in at least two of the analyzed applications, and thus might not be effective as single source of data for IR services. More broadly, restrictive features tend to be more often explored by users than collaborative ones, even within the same feature category. Second, in contrast, considering only non-empty feature instances, the amount of content tends to be larger in collaborative features. Third, the typically smaller and more often used TITLE and TAGS features exhibit, in general, higher descriptive and discriminative powers, followed by DESCRIPTION and COMMENTS. Nevertheless, on both CiteULike and LastFM, DESCRIPTION has a higher descriptive power than TAGS (but not than TITLE) if it is estimated based solely on the top-5 most descriptive terms, implying that DESCRIPTION instances do carry a few very descriptive terms on those two applications. Fourth, all textual features have a large amount of nouns and proper names, according to our dictionary, which can be considered good content descriptors. However all of them also suffer from semantic problems such as presence of noise and polysemy. Finally, through the use of two different metrics, namely Jaccard and Information-Similarity coefficients, we found evidence that there is a significant amount of content and information diversity across features associated with the same object.

[24] Unlike in the previous section, we here choose not to filter words based on the $TS \times IFF$ product. This is because, in the current analysis, these metrics would have to be applied to words, as opposed to terms, and thus would produce results that cannot be compared with those reported in Table 10. Rather, we choose to report the information similarity considering the complete set of words of each feature instance.

[25] We also considered an alternative random model which was built by first randomly selecting 1000 objects from each data set, and then randomly exchanging the whole contents (i.e., all words) of pairs of feature instances, taking each feature independently. The Information Similarity coefficients computed over the collections built according to the second random model are similar to those measured for the first one, and thus are omitted.

In the following sections, we assess the relative quality of the textual features when applied to two specific IR tasks, namely, object classification and tag recommendation. Results for both tasks are discussed in light of our characterization findings, since each of the analyzed quality aspects may impact the effectiveness of each task differently. We also experiment with strategies based on the combination of multiple features, motivated by the content and information diversities observed across object features in our datasets.

6. Object classification

This section presents our object classification experiments. These experiments use the collected categories, introduced in Section 4, as object labels. As discussed in that section, we focus our experiments on LastFM, YouTube and YahooVideo. In Section 6.1, we present the model adopted for object representation, and define the various classification strategies considered. Our experimental setup is described in Section 6.2, whereas the main results are discussed in Section 6.3.

6.1. Object representation model and classification strategies

In order to perform object classification, we first need to define a model to represent the objects. We adopt the vector space model (VSM), representing each object as a vector in a real-valued space whose dimensionality $|V|$ is the size of the vocabulary of the object feature(s) being considered as data source. Each term in the feature(s) is represented by a real-valued weight in the vector that represents the degree of relationship between the term and the object it is assigned to.

There are two important issues here: (1) how to determine the weight of each term in order to capture the semantics of the object, and (2) how to model the objects in the VSM using their distinct textual features. Regarding the former, we consider the following term weighting schemes:

TS : The weight of a term t in an object o is equal to the spread of t in o, $TS(t,o)$, which heuristically captures the descriptive power of the term (see Section 5.3.1).

IFF : The weight of a term t in a feature instance $f \in F$ is equal to the IFF of t in F, $IFF(t,F)$, which heuristically captures the discriminative power of the term (see Section 5.3.2). F represents the feature (or feature combination) used to represent the object (see below).

TS × IFF : The weight of t is equal to the product $TS(t,o) \times IFF(t,F)$, capturing, thus, both descriptive and discriminative powers of t.

TF : The weight of t is given by the frequency of t in the feature instance (or feature instance combination) f used to represent the object (see below), that is, the weight is given by $TF(t,f)$.

TF × IFF : The weight is given by the product $TF(t,f) \times IFF(t,F)$.

The last two schemes allow us to compare TS and the more commonly used Term Frequency (TF) metrics.

We also examine the following six strategies to model an object o as a vector V, which is normalized so that $\|V\| = 1$:

Title : Only terms present in the TITLE of o constitute the vector V, which is then referred to as V_{title}.

Tags : V is composed of only terms in the TAGS of o, i.e., $V = V_{tags}$.

Description (Desc.) : V is composed of only terms in the DESCRIPTION of o, i.e., $V = V_{desc}$.

Comments (Comm.) : V is composed of only terms in the COMMENTS of o, i.e., $V = V_{comm}$.

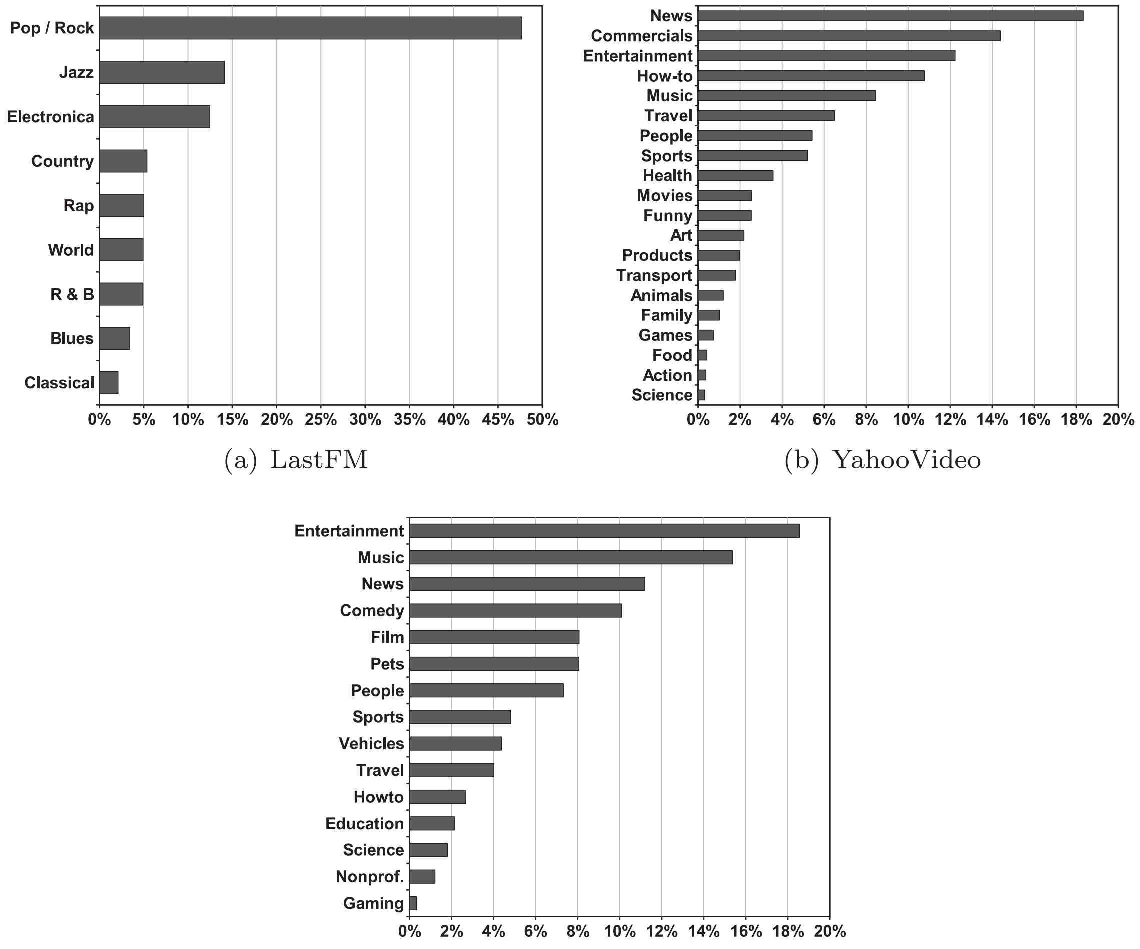
Bag-of-Words (Bagow.) : All four features are taken as a single document, as done by traditional IR algorithms, which do not consider the block structure in a Web page. Therefore, the same terms in different features are represented by the same element in the vector, which is referred to as V_{bagow}.

Concatenation (Conc.) : All textual features are concatenated in a single vector, so that the same term in different features are considered different elements in the vector, i.e., vector $V = V_{conc}$ is given by $\langle V_{comm}, V_{desc}, V_{tag}, V_{title} \rangle$.

In each strategy, vector V is defined as $\langle w_{f1}, w_{f2}, \ldots, w_{fn} \rangle$ where w_{fi} is the weight of term t_i in the considered feature instance f. Notice that for vector V_{bagow}, the IFF metric is equivalent to the more traditional IDF metric. These strategies do not cover all possible combinations of features, but are useful to compare their quality for object classification. In particular, the last two strategies are motivated by the results in Section 5.5, and allow us to investigate how textual features that are effective when used in isolation compare with the combination of multiple features.

6.2. Experimental setup

Our classification experiments are performed using two well-known and widely used classification algorithms: a Support Vector Machine (SVM) algorithm with linear kernel implemented in the Liblinear tool (Fan, Chang, Hsieh, Wang, & Lin, 2008), and a k-Nearest Neighbors Classifier (kNN) (Baeza-Yates & Ribeiro-Neto, 2011) implemented by ourselves, which uses a cosine measure to estimate distances between vectors. The first algorithm was selected because it is an effective state-of-art



(a) LastFM

(b) YahooVideo

(c) YouTube

Fig. 3. Distribution of objects across classes.

classification algorithm for large samples of textual data, and because linear kernels work particularly well for text classification tasks (Joachims, Nedellec, & Rouveirol, 1998). SVMs, however, are very complex to understand, making it very hard to interpret their results. Moreover, this algorithm was originally proposed for binary classification problems. Thus, it might have potential scalability problems when applied to tasks with more than two classes, as is the case of our Web 2.0 applications, due to the strategies used to adapt it to this scenario (e.g., one-against-one or one-against-the-rest strategies (Godbole, Sarawagi, & Chakrabarti, 2002)). We here adopt the one-against-the-rest strategy, as it is the standard approach implemented in the Liblinear tool.

The kNN algorithm was chosen as an alternative method which has been shown to have a competitive effectiveness (Yang & Liu, 1999) and potentially superior performance to SVM in problems with many classes, besides allowing for more interpretable results. We note, however, that kNN might suffer from scalability issues on very high dimensional vector spaces, as it needs to compute vector distances. We also note that we chose to use the cosine metric to compute such distances as it is directly impacted by the adopted term weighting scheme, thus allowing us to have a better assessment of the relative impact of using each of the four proposed schemes on classification effectiveness.

As before, we use our stemmed datasets, considering only labeled objects (i.e., objects with an associated class) with non-empty instances of all features, except in YahooVideo, for which the COMMENTS feature was disregarded. Moreover, as Fig. 3 shows, some object classes are highly underpopulated in our datasets.[26] Thus, we filter out all classes with fewer than 2.2% of the objects, removing 8 and 4 classes from YahooVideo and YouTube, respectively.

Our experiments consist of 10 runs, each with a distinct sample of 5000 objects from each application, using 10-fold cross-validation within each sample. Best SVM parameters (e.g., kernel parameter and cost C) were searched for within each training sample, using cross-validation, and the default values (linear kernel and $C = 1$) were found to be the best ones, which

[26] The class names in Fig. 3 are reduced for the sake of clarity. We refer to Table 2 for their complete names.

Table 12
kNN Classification: Macro F1 results along with 90% confidence intervals. (Best results for each object representation model in each application, including statistical ties at 90% confidence, are shown in bold.)

System	Weighting scheme	TITLE	TAGS	DESCRIPTION	COMMENTS	BAGOW	CONCAT
LastFM	IFF	**0.197 ± 0.016**	**0.790 ± 0.015**	**0.638 ± 0.022**	0.397 ± 0.019	**0.782 ± 0.013**	0.711 ± 0.022
	TS	**0.202 ± 0.019**	0.764 ± 0.020	0.503 ± 0.019	0.281 ± 0.011	0.644 ± 0.024	0.664 ± 0.024
	TS × IFF	**0.197 ± 0.017**	**0.794 ± 0.014**	0.630 ± 0.026	**0.429 ± 0.020**	**0.782 ± 0.012**	0.702 ± 0.020
	TF	**0.204 ± 0.019**	**0.795 ± 0.013**	0.542 ± 0.020	0.217 ± 0.016	0.751 ± 0.011	**0.767 ± 0.009**
	TF × IFF	**0.197 ± 0.016**	**0.794 ± 0.012**	**0.653 ± 0.017**	0.394 ± 0.012	0.766 ± 0.017	**0.769 ± 0.013**
YouTube	IFF	**0.401 ± 0.015**	**0.543 ± 0.014**	**0.403 ± 0.020**	0.396 ± 0.012	**0.544 ± 0.016**	0.536 ± 0.016
	TS	0.374 ± 0.017	0.516 ± 0.017	0.339 ± 0.015	0.352 ± 0.014	0.462 ± 0.014	0.504 ± 0.015
	TS × IFF	**0.397 ± 0.015**	0.537 ± 0.016	**0.404 ± 0.018**	**0.419 ± 0.011**	**0.544 ± 0.018**	0.522 ± 0.014
	TF	0.366 ± 0.016	0.532 ± 0.016	0.330 ± 0.014	0.370 ± 0.016	0.458 ± 0.017	0.519 ± 0.017
	TF × IFF	**0.400 ± 0.016**	**0.545 ± 0.014**	**0.401 ± 0.020**	**0.427 ± 0.016**	0.522 ± 0.019	**0.549 ± 0.015**
YahooVideo	IFF	**0.447 ± 0.015**	**0.534 ± 0.016**	**0.481 ± 0.010**	–	**0.610 ± 0.012**	**0.578 ± 0.021**
	TS	0.413 ± 0.017	0.519 ± 0.021	0.434 ± 0.011	–	0.569 ± 0.017	0.545 ± 0.024
	TS × IFF	**0.448 ± 0.014**	0.513 ± 0.015	**0.476 ± 0.009**	–	0.595 ± 0.013	0.533 ± 0.022
	TF	0.408 ± 0.015	0.488 ± 0.017	0.429 ± 0.009	–	0.516 ± 0.015	0.542 ± 0.021
	TF × IFF	**0.444 ± 0.018**	**0.526 ± 0.015**	**0.476 ± 0.008**	–	0.579 ± 0.011	0.570 ± 0.024

is consistent with the literature for textual classification (Joachims et al., 1998). The number of neighbors used by the kNN classifier to make a decision (i.e., parameter k) was set to 30 in all experiments, as this value produced the best results in various preliminary experiments using the training set (also with cross-validation).

We assess classification effectiveness using a commonly used metric, namely F1, which captures both precision and recall of the classification (Yang & Liu, 1999), two complementary measures. Precision captures the fraction of correctly classified objects out of the ones which are assigned to a given class, whereas recall captures the fraction of correctly classified objects out of those that should have been classified to the class. Both are commonly combined into another metric, called F1, which consists of their harmonic mean, i.e.:

$$F1(c) = \frac{2 \cdot P(c) \cdot R(c)}{P(c) + R(c)} \tag{10}$$

Precision, recall and F1 are applicable to each class. A general result for all classes can be obtained by considering either **Macro F1**, defined as the average F1 across all classes, or **Micro F1**, defined by first computing Micro precision and recall considering all classes and then computing the harmonic mean of these two measures. Micro F1 tends to be dominated by the classifier's performance on more popular categories while Macro F1 gives equal importance to all categories. As our collections are very unbalanced, Macro F1 results are of more importance. Nevertheless, we point out that Micro F1 results led to similar overall conclusions. Thus, for the sake of improving the clarity of result presentation and discussion, we chose to omit Micro F1 results, focusing rather on discussing the results in terms of Macro F1.

6.3. Most relevant results

We perform classification experiments combining each term weighting scheme with each textual feature (and feature combination) object model. Tables 12 and 13 show the results for each configuration considering the three applications, five weighting schemes, six object representation models, and two classification algorithms. The tables show average results, computed over 100 samples (i.e., 10 runs, with 10-fold cross-validation within each run), along with corresponding 90% confidence intervals. Note that, according to such intervals, the results deviate from the reported means by less than 10%. For each object representation model, in each application, the best results across all weighting schemes (including statistical ties at the 90% confidence level) are indicated in bold.

Before discussing the results, recall that, as argued in Section 5 and in (Almeida et al., 2010), different quality-related aspects (e.g., amount of content, discriminative power, descriptive power) are not equally important to all IR tasks. Indeed, even their relative impact on the effectiveness of a given task (e.g., classification) may also vary depending on the inherent biases and on the robustness of the specific algorithm adopted to perform it. As we shall see, in the particular case of the algorithms analyzed here, discriminative power and amount of content play important roles on classification effectiveness. Other aspects, such as how the classes are defined, the level of semantic overlap among multiple classes, and biases in class distribution, might also impact the results.

We start our discussion by analyzing the impact of the different term weighting schemes on the effectiveness of each classifier. When comparing all weighting schemes against each other, we can see that the *IFF* weighting scheme, which exploits a metric associated with discriminative power, is consistently one of the best performers for both classifiers in all applications. Indeed, *IFF* is the best weighting scheme, or very close to the best one, for many object representation models. This is perhaps more clearly seen across the results produced by the kNN classifier (Table 12), where the differences among the weighting schemes are somewhat more distinct, although some differences also arise with SVM (check, for instance, the results for

Table 13
SVM Classification: Macro F1 results along with 90% confidence intervals. (Best results for each object representation model in each application, including statistical ties at 90% confidence, are shown in bold.)

System	Weighting scheme	TITLE	TAGS	DESCRIPTION	COMMENTS	BAGOW	CONCAT
LastFM	IFF	**0.204 ± 0.019**	0.801 ± 0.013	0.704 ± 0.021	0.470 ± 0.022	0.790 ± 0.013	0.804 ± 0.013
	TS	0.191 ± 0.017	0.805 ± 0.016	0.701 ± 0.020	0.435 ± 0.024	0.784 ± 0.013	0.807 ± 0.016
	TS × IFF	**0.205 ± 0.017**	0.802 ± 0.013	0.705 ± 0.018	0.456 ± 0.027	0.789 ± 0.012	0.794 ± 0.014
	TF	0.190 ± 0.021	**0.828 ± 0.018**	**0.722 ± 0.012**	0.478 ± 0.017	**0.824 ± 0.013**	**0.828 ± 0.015**
	TF × IFF	**0.204 ± 0.019**	0.819 ± 0.019	**0.721 ± 0.014**	**0.493 ± 0.024**	0.803 ± 0.014	**0.830 ± 0.017**
YouTube	IFF	**0.407 ± 0.019**	**0.560 ± 0.014**	**0.431 ± 0.017**	0.450 ± 0.011	**0.604 ± 0.014**	0.604 ± 0.021
	TS	**0.402 ± 0.021**	**0.561 ± 0.014**	0.411 ± 0.018	0.431 ± 0.013	0.575 ± 0.012	0.599 ± 0.020
	TS × IFF	**0.403 ± 0.021**	0.553 ± 0.017	0.423 ± 0.017	**0.453 ± 0.009**	**0.602 ± 0.012**	0.599 ± 0.024
	TF	**0.407 ± 0.019**	**0.559 ± 0.012**	0.417 ± 0.019	**0.461 ± 0.009**	0.576 ± 0.014	0.606 ± 0.018
	TF × IFF	**0.407 ± 0.019**	**0.558 ± 0.015**	**0.426 ± 0.019**	**0.462 ± 0.007**	0.594 ± 0.014	**0.616 ± 0.021**
YahooVideo	IFF	**0.527 ± 0.015**	**0.643 ± 0.016**	**0.566 ± 0.015**	–	**0.667 ± 0.014**	**0.670 ± 0.012**
	TS	**0.529 ± 0.017**	**0.637 ± 0.016**	0.540 ± 0.015	–	0.652 ± 0.017	0.663 ± 0.013
	TS × IFF	0.522 ± 0.015	0.621 ± 0.011	**0.570 ± 0.012**	–	**0.660 ± 0.013**	0.631 ± 0.014
	TF	**0.529 ± 0.017**	0.635 ± 0.017	0.553 ± 0.013	–	0.653 ± 0.015	0.663 ± 0.013
	TF × IFF	**0.524 ± 0.015**	**0.645 ± 0.017**	**0.568 ± 0.015**	–	0.652 ± 0.012	**0.668 ± 0.016**

YouTube and BAG-OF-WORDS in Table 13).[27] Moreover, we can also see that the results produced with the *TF* and *TS* weighting schemes are, in many cases, improved (to some extent) when these schemes are combined with *IFF*. Once again, such improvements are more significant with kNN, reaching as much as 80% (see COMMENTS for LastFM in Table 12). In the few cases when the combination with *IFF* yields worse classification results in comparison with the results produced using only *TS* or *TF*, the losses are very marginal (mostly under 1%). The following discussion considers the best results produced across all weighting schemes.

Turning our attention to the relative effectiveness of both classifiers, our results reveal that, in LastFM, the simpler kNN is very competitive with SVM, with a relative difference falling under 5%.[28] For YouTube and YahooVideo, the differences between the two classifiers are greater, with a clear advantage for SVM.

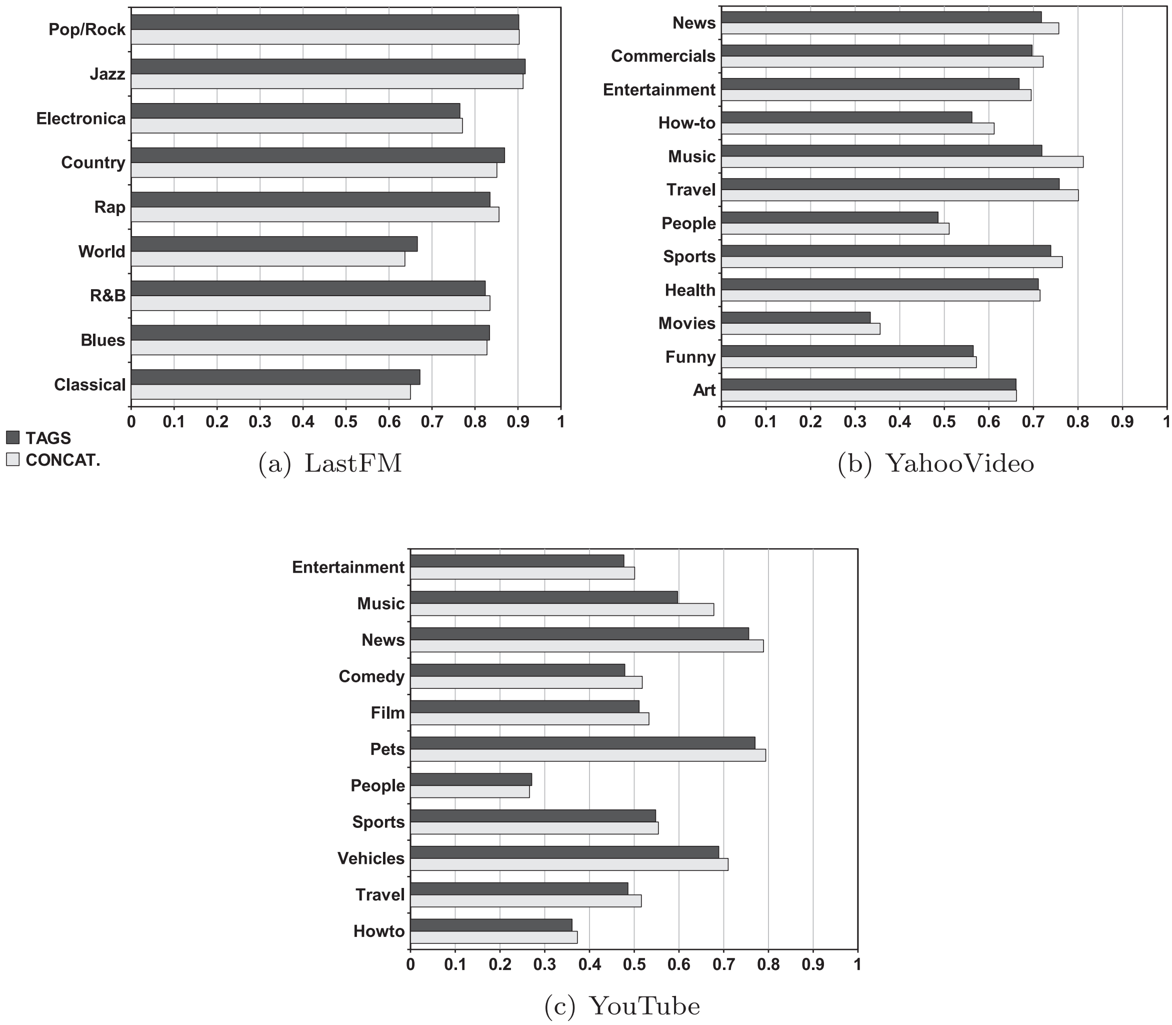
We now discuss the classification effectiveness as a function of the various object representation models proposed. Considering the results obtained when each feature is used in isolation as object representation, TAGS are, undoubtedly, the best single feature in all applications for both classifiers. This is consistent with our characterization results which show that TAGS have: (1) good descriptive and discriminative powers, according to our heuristics, with *AFS* and *AIFF* values close to those of TITLE, and (2) at least twice more terms than TITLE, on average, in the three applications. Thus, the larger amount of content clearly favors TAGS as source of data for the classifier, particularly for SVM, which is known to work better in the presence of larger content (term) spaces (Joachims et al., 1998). This issue of amount of content may also explain the poor performance of TITLE, the worst feature in all applications, for both classifiers, in spite of its *AFS* and *AIFF* values being the largest ones. For example, instances of TITLE in LastFM typically contain artist names, which are usually very short. These may be very good for searching purposes, for example, but are very restrictive for classification as the terms in the artist names of training objects may be too specific and may not occur in the test, therefore not generalizing.

Regarding the other two features, COMMENTS is usually a bit better than DESCRIPTION in YouTube, considering both classifiers. In spite of the smaller *AFS* (descriptive power) and a somewhat similar *AIFF* (discriminative power), COMMENTS instances have, on average, more than eight times more terms than DESCRIPTION instances, which turns out to be a more dominant factor for classification effectiveness (see further discussion below). On the other hand, we find that DESCRIPTION outperforms COMMENTS in LastFM, in spite of the somewhat larger amount of content of the latter and comparable *AFS* and *AIFF* values. Despite not being completely supported by our characterization, this result may reflect the wiki-like collaborative nature of the DESCRIPTION feature in LastFM, which brings a tendency for a higher quality semantic content, a phenomenon also observed in Wikipedia (Hu, Sun, Lauw, & Vuong, 2007). This is an aspect involving social behavior that is not captured by our current metrics, being subject of future work.

Recall that the Jaccard and Leacock–Chodorow Coefficients, analyzed in Section 5.5, suggest that there exist distinct content and information in each feature, which can be leveraged for classification purposes. This is confirmed by the results of both feature combination strategies, which, except in LastFM, do bring some improvements over using features in isolation, particularly considering SVM. These results, along with the natural expansion in the amount of content, provide evidence supporting the notion that content and information diversity across features can improve the efficacy of IR tasks. We note, however, that such improvements are not very large. Indeed, the largest improvement across all analyzed configurations is obtained with the kNN classifier in YahooVideo. In such case, using BAG-OF-WORDS, instead of the best single feature (i.e., TAGS), yields a 14% higher

[27] We note that the smaller differences among the results produced by SVM with different weighting schemes are consistent with the literature. Indeed, previous work has shown that SVM is more robust to the selected weighting scheme (Lan, Tan, Low, & Sung, 2005), and that it produces similar results with several of them. On the other hand, the more significant differences obtained with kNN is also expected, since, as previously mentioned, this classifier directly explores the weights to compute the distances between neighbors (using the cosine measure), and use them to make the classification decisions.

[28] We note that, in terms of Micro F1, results (omitted) are even closer, with a maximum difference of 2.3%.



(a) LastFM

(b) YahooVideo

(c) YouTube

Fig. 4. Per class average F1 results tags and concatenation object models.

Macro F1. Considering the results produced by SVM, the largest improvement of using bag-of-words instead of tags is 7% (on YouTube). These differences in classification effectiveness can be considered small given that the sizes of the vocabularies of bag-of-words and concatenation are 12–47 times larger than that of tags in the three applications. In other words, in this case, the much larger amount of content does not lead to a corresponding increase in performance, as other aspects (e.g., discriminative power of the available content) also impact classification results. Moreover, we should note that such larger vocabularies might have strong implications for the time complexity of the classification process, particularly during training.

Comparing the results obtained using both feature combination strategies, we find that, in most cases, the results for concatenation are only slightly better than those obtained with the simpler bag-of-words representation, which is favored by a reduced dimensionality of the vector space. This finding is not in consonance with recent results by Ramage et al. (2009), which demonstrate a clear advantage for the concatenation approach over the one based on bag-of-words. However, in that work, the authors concatenated tags with the whole content of the Delicious webpage, a very different approach from the one adopted here.

Comparing the classification effectiveness in the three applications, our results show that, for any given feature (or feature combination), the results in LastFM are much higher than in the other applications (except for title, for reasons already discussed). These results are further illustrated in Fig. 4, which shows per class average F1 values for both tags and concatenation object models. Clearly, the classifier achieves higher F1 values in LastFM, being greater than 0.65 for all classes. Moreover, the results are more evenly distributed across classes. In contrast, in YouTube and YahooVideo, average F1 values can be as low as 0.27 and 0.33, respectively, being also more unevenly distributed.

The higher average F1 values in LastFM may be explained by the fact that LastFM object classes are defined by experts from the music industry, guaranteeing a higher level of consistency in the class assignments when compared to the other applications, in which individual users are responsible for the manual classification. This problem related to the manual classification is exacerbated if we consider that YouTube and YahooVideo have a larger number of classes, and that some of them may have some degree of semantic overlap, making it even harder for users to determine to which class an object belongs. For example, a comedy video may be rightfully inserted into either the "Entertainment & TV" or the "Funny Videos" category, on YahooVideo.

In sum, considering the automatic classification of objects in the analyzed applications, we conclude that: (1) weighting schemes that explore discriminative power have good effectiveness either in isolation or in combination with other metrics; (2) kNN may be competitive with SVM in some applications (notably, LastFM); (3) TAGS, when present, is the best feature in isolation for classification purposes due to a combination of good discriminative power and large amount of content (besides good descriptive power), leading to results that are often very close to the two analyzed feature combination strategies, while producing a much smaller feature space; (4) feature combination may bring some benefits due to the presence of distinct and somewhat complementary content; (5) a simpler feature combination strategy based on bag of words may be as effective or at most only slightly worse than concatenating features as different feature spaces, which is a much more expensive approach; (6) a combination of fewer classes, with possibly less ambiguity, and more qualified human labelers, made automatic classification more effective in LastFM than in the two video applications.

7. Tag recommendation

We now turn our attention to the relative quality of the textual features for supporting a different IR task, namely, tag recommendation. The goal of this task is to support users during the process of tagging Web 2.0 objects, by *recommending* "good" (i.e., high quality) tags for describing an object (Guan et al., 2009; Song et al., 2008; Sigurbjornsson & van Zwol, 2008; Byde et al., 2007; Lipczak et al., 2009; Rendle & Schmidt-Thie, 2010; Belém et al., 2011; Menezes et al., 2010; Clements et al., 2010). We here focus on recommending tags because: (1) it is the most studied textual feature on the Web 2.0, being of more interest to the community; (2) our characterization and classification results indicate that this is a feature of quality (with respect to several aspects) when used in isolation; (3) recommending new terms to other features does not make much sense as they are typically composed of full sentences with semantic meanings, while the TAGS feature contains set of terms; and (4) the recommended tags can be, at least partially, automatically evaluated, as we shall see.

We should note that our goal here *is not* to propose new tag recommendation mechanisms. Rather, our primary goals are to *assess the potential benefits of using* TITLE, DESCRIPTION *and* COMMENTS *as data sources for effective tag recommendation, and to evaluate the potential of different metrics, which capture different quality-related aspects, to support such task*. To make our assessment viable, we here adopt a simple approach to recommend tags, which consists of extracting new and possibly "relevant" (according to given criteria) terms from the other textual features and adding them as new TAGS. In other words, the idea is to expand the contents of the TAGS feature by introducing new recommended terms extracted from the other three features, and then evaluate the quality of such recommendations. We note that, despite the great variety of different tag recommendation methods and tag analyses available in the literature (Menezes et al., 2010; Sigurbjornsson & van Zwol, 2008; Belém et al., 2011; Rendle & Schmidt-Thie, 2010; Lipczak et al., 2009; Zhang et al., 2009), we are not aware of any previous effort in quantifying the quality of different sources of candidate terms for recommendation purposes. This is our objective here, which distinguishes the present effort from previous studies.

Three problems must be tackled in order to effectively recommend new tags: (1) maximize the amount of relevant terms being recommended; and (2) recommend more relevant terms before irrelevant ones, thus minimizing noise in initial suggestions; while, (3) minimizing the amount of irrelevant terms. In other words, the problem of recommending tags is here treated as a problem of *ranking the best candidate tags to recommend*. This rank will be produced by applying a quality metric (or combinations of metrics) to candidate terms extracted from other features of the same object and ranking those terms according to the value of the metric (s).

Note that determining whether a term is relevant to be recommended as a new tag could require laborious manual inspection, which might be affected by the subjectivity of human judgments. Thus, following the approach adopted in (Rendle & Schmidt-Thie, 2010; Sigurbjornsson & van Zwol, 2008; Lipczak et al., 2009), we here consider as "relevant" recommendations only terms that already appear in the TAGS feature of the target object, i.e., the current content of the TAGS feature is considered as the "gold standard". Even though this is a somewhat limited evaluation, in the sense that we cannot say whether a recommended tag that is not currently in the object is not relevant, it allows for an automatic evaluation and serves our purpose of comparing the quality of each textual feature when applied to such a task. In other words, our results can be seen as a lower bound for the effectiveness of a tag recommendation approach that exploits contents from the other features. We also note that this scenario covers common situations such as: (1) when a user is uploading a new object and has filled the contents of (some of) the other features (e.g., TITLE and DESCRIPTION); or (2) when a new tagging application is being built for content already existent in the system.

Next, we first present the metrics used to rank the candidate terms as well as the metrics used to evaluate the recommendations (Section 7.1), and then discuss the main results (Section 7.2).

7.1. Metrics

In order to rank terms extracted from TITLE, DESCRIPTION and COMMENTS as candidates for recommendation, we make use of a similar vector space model as the one described in Section 6.1 to represent the contents of each such feature. Moreover, we also adopt a term weighting scheme, here applied as an estimate of the quality of a term as a candidate for recommendation. In other words, given an instance *f* of a feature *F* and a term weighting scheme *ws*, we produce a set of tag recommendations by *ranking* all terms in *f* according to *ws*.

As in Section 6, we consider five different weighting schemes (i.e., quality metrics), namely, *TS*, *TS* × *IFF*, *TF*, *TF* × *IFF* and *IFF*. We note that, for computing such metrics, we consider the content of the TAGS feature associated with the target object as "empty" so that such content would not influence the results of the metrics. For instance, when computing the *TS* of a candidate term *t*, we only consider its spread in the other three features as an estimate of its quality.

The ordered set of terms of *f* (ranked according to *ws*) is here taken as an ordered set of tag recommendations (or simply a *recommendation*) for the associated object, and is referred to as *Rec*(*f*,*ws*). In other words, terms appearing in higher positions of *Rec*(*f*,*ws*) are considered as more promising ones for recommendation, and thus should be suggested first. We here evaluate the quality of each textual feature *F*, for each considered weighting scheme *ws*, by assessing the effectiveness of the recommendations produced from each feature instance, i.e., by quantifying the effectiveness of *Rec*(*f*,*ws*), for each instance *f* of *F*.

Note that, despite related to some of the analyses performed in Section 5.3, our goal here is different. We are evaluating *ranked lists of terms*, therefore we here take special look at the *order* in which the candidate terms are ranked. This order is not directly captured by any of the considered weighting schemes. Thus, we believe that evaluating how they perform in this task is an interesting exercise. Moreover, since our main interest is in the produced ranking, in a task similar to the retrieval of documents, we evaluate the effectiveness of each recommendation using metrics more pertinent to such task, as discussed next.

A good tag recommender would ideally recommend relevant terms before any noise is suggested. To measure this capability, we compute the Average-Precision (*AP*) of the produced ranked list of recommendations, i.e., of *Rec*(*f*,*ws*). *AP* is a common IR evaluation metric which not only considers if the elements from an answer set are relevant, but also their positions in the ranking. Thus, higher *AP* values are obtained for recommendations that include terms of the TAGS feature (our "gold standard") in the top positions of the ranking.

In order to compute *AP* for a recommendation *Rec*(*f*,*ws*), we must initially compute Precision-at-k (*P@k*), which is the fraction of the *k* most highly ranked terms in *Rec*(*f*,*ws*) that are relevant. In other words, it measures the fraction of the top-*k* ranked terms from the recommendation that appear in the TAGS feature of the object:

$$P@k(Rec(f,ws),k) = \frac{|f_{tags} \bigcap Rank(Rec(f,ws),k)|}{k} \tag{11}$$

where *Rank*(*Rec*(*f*,*ws*),*k*) is a set containing the first *k* ranked terms in the recommendation, that is, *Rank*(*Rec*(*f*,*ws*),*k*) = {t_i,t_i ∈ *Rec*(*f*,*ws*) and $0 < i \leqslant k$}.

AP is then defined as the average *P@k* for each possible value of *k*, that is:

$$AP(Rec(f,ws)) = \frac{1}{|f_{tags}|}\sum_{k=1}^{|f_{tags}|} P@k(Rec(f,ws),k) \cdot rel(k) \tag{12}$$

where *rel*(*k*) is a binary function indicating if the term at position *k* is relevant or not, that is, if it belongs to f_{tags} or not.

We can also compute the mean of *AP* values, i.e., *MAP*, for all recommendations. Given |*F*| the number of instances *f* of feature *F*, the *MAP* of all recommendations produced using *F* is computed as:

$$MAP(F) = \frac{1}{|F|}\sum_{f \in F} AP(Rec(f,ws)). \tag{13}$$

Table 14
Tag Recommendations: *MAP* results along with 90% confidence intervals. (Best results for each object representation model, including statistical ties at 90% confidence level, are shown in bold.)

Weighting scheme	TITLE	DESCRIPTION	COMMENTS	TITLE	DESCRIPTION	COMMENTS
	LastFM			YouTube		
TF	**0.241** ± **0.003**	**0.220** ± **0.002**	0.165 ± 0.002	0.661 ± 0.003	0.383 ± 0.002	0.207 ± 0.002
TS	**0.242** ± **0.003**	0.215 ± 0.002	**0.204** ± **0.002**	**0.719** ± **0.003**	**0.518** ± **0.003**	**0.416** ± **0.003**
IFF	**0.239** ± **0.003**	0.102 ± 0.001	0.095 ± 0.001	0.659 ± 0.003	0.342 ± 0.002	0.102 ± 0.001
TF × *IFF*	**0.239** ± **0.003**	0.163 ± 0.001	0.144 ± 0.002	0.662 ± 0.003	0.406 ± 0.002	0.276 ± 0.002
TS × *IFF*	**0.240** ± **0.003**	0.135 ± 0.001	0.132 ± 0.002	0.708 ± 0.003	0.461 ± 0.003	0.228 ± 0.002
	CiteULike			YahooVideo		
TF	0.238 ± 0.002	0.233 ± 0.003	–	0.735 ± 0.002	0.491 ± 0.002	–
TS	0.251 ± 0.003	0.240 ± 0.003	–	**0.781** ± **0.002**	**0.638** ± **0.002**	–
IFF	0.246 ± 0.003	0.099 ± 0.001	–	0.700 ± 0.002	0.430 ± 0.002	–
TF × *IFF*	0.252 ± 0.003	**0.252** ± **0.003**	–	0.712 ± 0.002	0.517 ± 0.002	–
TS × *IFF*	**0.262** ± 0.003	0.179 ± 0.002	–	0.751 ± 0.002	0.527 ± 0.002	–

7.2. Most relevant results

We perform experiments using stemmed data sets from the four applications, namely, CiteULike, LastFM, YahooVideo and YouTube. Results are computed over randomly selected samples of 50,000 objects, one from each application. Table 14 shows *MAP* results, along with corresponding 90% confidence intervals, for all three textual features and five term weighting schemes across the four applications. Note that, for both CiteULike and YahooVideo, we disregard COMMENTS (as source of candidate terms and from the computation of the weighting schemes) as this feature is vastly unused in these applications. For each considered feature, best results, including statistical ties at the 90% confidence level, are shown in bold.

We start by noting that, in general, rankings produced based on the *IFF* metric alone tend to have the lowest *MAP* values among all considered weighting schemes, likely because this metric, as an estimate of discriminative power, considers the importance of a term across the whole collection of objects, and not to the specific object that is target of the recommendation. In contrast, heuristics that capture the descriptive power of each candidate term, such as *TS* and *TF*, achieve much higher *MAP* values, for most features and applications. Moreover, the use of such heuristics along with *IFF*, in the *TS* × *IFF* and *TF* × *IFF* schemes, rarely produces significant gains over using only the heuristics (one exception is TITLE in CiteULike), but rather may be very detrimental to the recommendation (e.g., see results for *TS* on YouTube). Thus, based on our heuristic metrics, descriptive power does seem to be more important for tag recommendation.

Comparing the descriptive heuristics, we note that, in many cases, such as in YahooVideo and YouTube (for all features), *TS* brings great improvements over *TF* and over *TF* × *IFF*. The superiority of *TS* over *TF* as well as of *TS* × *IFF* over *TF* × *IFF* in most cases (with a few notable exceptions) indicates that simply taking very frequent terms in the feature instance, as done by *TF*, regardless of whether they occur in other features of the object, may not be a good strategy for tag recommendation. In contrast, taking terms that are spread across multiple features, tends to lead to better suggestions in general. Indeed, in the very few cases in which *TS* does not lead to the highest *MAP* values, the differences from the best result are under 5%. One worth mentioning case is TITLE in LastFM: the results for all weighting schemes are statistically tied at 90% confidence level. This is probably due to most terms of TITLE instances having the same weight, for any of the considered weighting schemes. This happens because most terms (i.e., artist names) occur only once in a TITLE instance (thus having *TF* = 1), tend to occur in all other textual features (maximum *TS*), while being very specific to the object and thus not occurring in other TITLE instances (maximum *IFF*). Ultimately, all five weighting schemes lead to the same rank of candidate terms.

Moreover, considering the results produced by any given weighting scheme, TITLE is the feature that leads to the best *MAP* values, followed by DESCRIPTION and, if applicable, COMMENTS in all four applications. On YouTube and YahooVideo, in particular, TITLE outperforms DESCRIPTION in 39% and 22%, respectively. On the other two applications, the gains are more modest (up to 10%) but still significant. This is consistent with our characterization results that showed that TITLE has excellent descriptive power. We note however that, since *MAP* is based on the fraction of returned items that are relevant and their relative positions, the smaller sizes of TITLE does not severely impact our reported results, not even for LastFM. Nevertheless, this factor might be of relevance in practice, as the smaller sizes ultimately limit the number of candidate terms that can be extracted from that feature (see further discussion below).

Comparing the best results for each application, we note that the *MAP* values obtained for the two video applications are much higher than the results produced for LastFM and CiteULike. This is possibly due to the smaller overlaps between the contents of multiple features of the same object in LastFM and CiteULike (see Table 10 in Section 5.5). Such smaller overlaps ultimately lead to a greater concentration of *TS* around smaller values, making it difficult to distinguish "good" from "bad" terms. Moreover, in the case of CiteULike specifically, Table 5 (Section 5.3) shows that any given feature in CiteULike has lower *AFS* value, a trend also observed for *TS* values, than the same feature on the other applications, thus implying in less descriptive features (according to our heuristics), which ultimately impact recommendation effectiveness on that application.

In sum, TITLE seems to be the most promising feature for providing term candidates for tag recommendation. Moreover descriptive capacity which is strongly present in TITLE instances, appears to play a very important role for the effectiveness of that task. We should note, however, that the smaller sizes of most TITLE instances might restrict the number of available term suggestions. In contrast, DESCRIPTION and COMMENTS, usually carrying many more terms, might provide more suggestions. Nevertheless, in practice, presenting a large amount of suggestions to the user may likely overwhelm her. We argue that, instead, only the most promising terms, that is, those more highly ranked, should be recommended. In that case, extracting candidates from the TITLE, provided that it carries enough content, is more likely to produce better suggestions. As a final note, we emphasize, once again, that our analyses can only measure whether the produced rankings include terms that we know to be relevant, that is, terms that are already present in the TAGS instances. Analyzing the relevance of other suggested terms, which requires manual inspection, is left for future work.

8. Conclusions and future work

The advent of the Web 2.0 drastically changed the way users explore Web applications. Currently, users may act not only as consumers but also as creators of content, generating what is known as *social media*. Social media includes not only the main object of interest in an application (e.g., videos on YouTube) but also several features associated with that content. However, by offering no guarantee of quality, social media poses challenges to the IR community. Indeed, common IR

Please cite this article in press as: Figueiredo, F., et al. Assessing the quality of textual features in social media. *Information Processing and Management* (2012), http://dx.doi.org/10.1016/j.ipm.2012.03.003

services, such as content recommendation, searching and directory organization, still rely mostly on the use of textual features (e.g., tags) associated with the multimedia objects. Nevertheless, the question of which textual features provide potentially better data sources for supporting effective IR services remains open.

In this work, we investigated the relative quality of different textual features as supporting data for IR services in different Web 2.0 applications. To provide insights on the matter, we sampled data from TITLE, DESCRIPTION, TAGS and COMMENTS associated with objects in CiteULike, LastFM, YahooVideo and YouTube. Our characterization of these features, using different heuristic metrics, revealed that collaborative features, including TAGS, are absent in a non-negligible fraction of the objects crawled from all analyzed applications. However, if present, they tend to provide more content than restrictive features, such as TITLE. We also found that, in general, the smaller TITLE and TAGS features have higher descriptive and discriminative powers, according to our heuristics. We should note, however, that, on both CiteULike and LastFM, DESCRIPTION outperforms TAGS (but not TITLE) in terms of descriptive power if only the top-5 most descriptive terms are considered, implying that, on those two applications, DESCRIPTION instances do carry a few very descriptive terms. Finally, we showed that every feature suffers from semantic problems, such as presence of noise and polysemy, and that there exists a significant amount of content and information diversity across features associated with the same object.

As case studies, we also assessed the quality of the four textual features when applied to two relevant IR tasks, namely object classification and tag recommendation. Our classification results showed that weighting schemes that explore discriminative power have better effectiveness either in isolation or combined with other metrics. They also revealed that TAGS, if present, are the most promising feature used in isolation, and that combining content from all features can lead to some classification improvements. In particular, a simple combination strategy based on *bag of words* may be as effective or at most only slightly worse than concatenating features as different feature spaces. Moreover, our results showed that, although widely present and highly ranked according to both descriptive and discriminative power heuristics, TITLE yields the worst classification results, being severely impacted by the typically small amount of content.

Our tag recommendation results showed that, in contrast, TITLE appears as the most promising feature, producing the best *MAP* results. We believe that this is likely due to TITLE, like TAGS, being commonly used to summarize the contentn of an object with a few keywords. Thus, terms extracted from the TITLE tend to be more relevant suggestions than terms from other features. We note however that such conclusions are constrained by our evaluation methodology, which considers only previously assigned tags as relevant. A more thorough evaluation, based on manual inspection by a group of volunteers, is left for future work.

These results illustrate that different quality aspects may impact differently the effectiveness of different services. In other words, the most promising textual feature may vary depending on the target service. We here analyzed two specific IR tasks, while also producing characterization results that uncover knowledge and raise insights that may be quite valuable for future designs and developments. As future work, we intend to explore the metrics and knowledge developed in this study to improve actual IR services and to design different strategies for feature quality enhancement. Another interesting direction for future investigation is to correlate our results, targeting the quality of different sources of content, with individual and social behavior aspects (Santos-Neto et al., 2010).

Acknowledgements

This work is partially supported by the INCT-Web (MCT/CNPq Grant 57.3871/2008-6), and by the authors individual grants and scholarships from CNPq, FAPEMIG, and CAPES. Flavio Figueiredo and Jussara Almeida were also sponsored by Universo Online (UOL), through its UOL Bolsa Pesquisa program, process number 20090215103600.

References

Agichtein, E., Castillo, C., Donato, D., Gionis, A., & Mishne, G. (2008). Finding high-quality content in social media. In *Proc. WSDM*.

Almeida, J. M., Gonçalves, M. A., Figueiredo, F., Belém, F., & Pinto, H. (2010). On the quality of information for Web 2.0 services. *IEEE Internet Computing*.

Baeza-Yates, R., & Ribeiro-Neto, B. (2011). *Modern information retrieval* (second ed.). Addison-Wesley Professional.

Belém, F., Martins, E., Pontes, T., Almeida, J.M., Gonçalves, M.A., Pappa, G. (2011). Associative tag recommendation exploiting multiple textual features. In *Proc. ACM SIGIR*.

Bian, J., Liu, Y., Agichtein, E., & Zha, H. (2008). A few bad votes too many? Towards robust ranking in social media. In *Proc. AIRWeb*.

Bischoff, K., Claudiu, S., Wolfgang, N., & Raluca, P. (2008). Can all tags be used for search? In *Proc. CIKM*.

Boll, S. (2007). MultiTube – where Web 2.0 and multimedia could meet. *IEEE Multimedia*, 14 (1).

Byde, A., Wan, H., & Cayzer, S. (2007). Personalized tag recommendations via tagging and content-based similarity metrics. In *Proc. ICWSM*.

Cai, D., Yu, S., Wen, J. -R., & Ma, W. -Y. (2004). Block-based Web Search. In *Proc. SIGIR*.

Chen, L., Wright, P., & Nejdl, W. (2009). Improving music genre classification using collaborative tagging data. In *Proc. WSDM*.

Clements, M., de Vries, A. P., & Reinders, M. (2010). The task dependent effect of tags and ratings on social media access. *ACM Transactions on Information Systems, 28*(4).

Dalip, D. H., Gonçalves, M. A., Cristo, M., & Calado, P. (2009). Automatic quality assessment of content created collaboratively by Web communities: A case study of Wikipedia. In *Proc. ACM JCDL*.

de Moura, E. S., Fernandes, D., Ribeiro-Neto, B. A., da Silva, A. S., & Gonçalves, M. A. (2010). Using structural information to improve search in Web collections. *JASIST, 61*.

Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research, 9*.

Fellbaum, C., & NetLibrary, I. (1998). *WordNet: An electronic lexical database*. USA: MIT Press.

Figueiredo, F., Belém, F., Pinto, H., Almeida, J. M., Gonçalves, M. A., Fernandes, D., et al. (2009). Evidence of quality of textual features on the Web 2.0. In *Proc. CIKM*.

Giles, J. (2005). Special report: Internet encyclopedias go head to head. *Nature*, 438(15).

Godbole, S., Sarawagi, S., & Chakrabarti, S. (2002) Scaling multi-class support vector machines using inter-class confusion. In *Proc. ACM SIGKDD*.

Golder, S. A., & Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science, 32*(2).

Goodman, L. A. (1961). Snowball sampling. *Annals of Mathematics and Statistics, 32*(1).

Guan, Z., Bu, J., Mei, Q., Chen, C., & Wang, C. (2009). Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. In *Proc. ACM SIGIR*.

Guy, I., Zwerdling, N., Ronen, I., Carmel, D., & Uziel, E. (2010). Social media recommendation based on people and tags. In *Proc SIGIR*.

Paul Heymann, Andreas Paepcke, & Hector Garcia-Molina (2010). Tagging human knowledge. In *Proc. ACM WSDM*.

Hotho, A., Jaschke, R., Schmitz, C., & Stumme, G. (2006). Information retrieval in Folksonomies: Search and ranking. *Lecture Notes in Computer Science, 4011*, 411.

Hu, M., Lim E., Sun, A., Lauw, H., & Vuong, B. (2007). Measuring article quality in Wikipedia: Models and evaluation. In *Proc. CIKM*.

Jain, R. (1991). *The art of computer systems performance analysis: Techniques for experimental design, measurement, simulation, and modeling*. Wiley-Interscience.

Jeon, J., Croft, W. B., Lee, J. H., & Park, S. (2006). A framework to predict the quality of answers with non-textual features. In *Proc. SIGIR*.

Joachims, T., Nedellec, C., & Rouveirol, C. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proc. European conference on machine learning*.

Lan, M., Tan, C. L., Low, H. B., & Sung, S. Y. (2005). A comprehensive comparative study on term weighting schemes for text categorization with support vector machines. In *Special interest tracks and posters of the 14th international conference on World Wide Web*.

Li, X., Guo, L., & Zhao, Y. E. (2008). Tag-based social interest discovery. In *Proc. WWW*.

Lih, A. (2004). Wikipedia as participatory journalism: reliable sources? Metrics for evaluating collaborative media as a news resource. In *Proc. of 5th international symposium on online journalism*.

Lipczak, M., Hu, Y., Kollet, Y., & Milios, E. (2009). Tag sources for recommendation in collaborative tagging systems. In *Proc. ECML PKDD discovery challenge workshop*.

Lipczak, M., & Milios, E. (2010). The impact of resource title on tags in collaborative tagging systems. In *Proc. 21st ACM conference on hypertext and hypermedia*.

Marlow, C., Naaman, M., Boyd, D., & Davis, M. (2006). Position paper, tagging, taxonomy, Flickr, Article, ToRead. In *Proc. Collaborative Web tagging workshop*.

Marshall, C. (2009). No Bull, No Spin: A comparison of tags with other forms of user metadata. In *Proc. JCDL*.

Menezes, G., Almeida, J., Belém, F., Gonçalves, M., Lacerda, A., Moura, G. et al. (2010). Demand-driven tag recommendation. In *PKDD*.

Mishne, G. (2007). Using blog properties to improve retrieval. In *Proc. ICWSM*.

Mitchell, T. (1997). *Machine learning*. McGraw-Hill Companies.

Ramage, D., Heymann, P., Manning, C. D., & Garcia-Molina, H. (2009). Clustering the tagged Web. In *Proc. WSDM*.

Rege, M., Dong, M., & Hua, J. (2008). Graph theoretical framework for simultaneously integrating visual and textual features for efficient Web image clustering. In *Proc. WWW*.

Rendle, S., & Schmidt-Thie, L. (2010). Pairwise interaction tensor factorization for personalized tag recommendation. In *Proc. WSDM*.

Rui, Y., Huang, T. S., Mehrotra, S., & Ortega, M. (1997). A relevance feedback architecture for content-based multimedia information retrieval systems. In *Proc. CBAIVL*.

Santos-Neto, E., Figueiredo, F., Mowbray, M., Gonçalves, M. A., & Ripeanu, M. (2010). Assessing the value of contributions in tagging systems. In *Proc. IEEE 2nd international conference on social computing*.

Schenkel, R., Crecelius, T., Kacimi, M., Michel, S., Neumann, T., & Parreira, J. X. (2008). Efficient top-k querying over social-tagging networks. In *Proc. SIGIR*.

Sen, S., Vig, J., & Riedl, J. (2009). Tagommenders: Connecting users to items through tags. In *Proc. WWW*.

Shah, C., & Pomerantz, J. (2010). Evaluating and predicting answer quality in community QA. In *Proc. ACM SIGIR*.

Shen, J., Meng, W., Yan, S., Pang, H., & Hua, X. (2010). Effective music tagging through advanced statistical modeling. In *Proc. ACM SIGIR*.

Sigurbjornsson, B., & van Zwol, R. (2008). Flickr tag recommendation based on collective knowledge. In *Proc. WWW*.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. IEEE TPAMI.

Song, Y., Zhuang, Z., Li, H., Zhao, Q., Li, J., & Lee, W.C. (2008). Real-time automatic tag recommendation. In *Proc. SIGIR*.

Strong, D. M., Lee, Y. W., & Wang, R. Y. (1997). Data quality in context. *Communications of ACM, 40*(5).

Suchanek, F. M., Kasneci, G., Weikum, G. (2007). Yago: A core of semantic knowledge. In *Proc. WWW*.

Suchanek, F. M., Vojnovic, M., Gunawardena, D. (2008). Social tags: Meaning and suggestions. In *Proc. CIKM*.

Venetis, P., Koutrika, G., & Garcia-Molina, H. (2011). On the selection of tags for tag clouds. In *Proc. WSDM*.

Yang Y., & Liu, X. (1999). A re-examination of text categorization methods. In *Proc. ACM SIGIR*.

Zhang, N., Zhang, Y., & Tang, J. (2009). A tag recommendation system based on contents. In *Proc. PKDD*.

Zhou, Y., & Croft, W. B. (2005). Document quality models for web ad hoc retrieval. In *Proc. ACM SIGIR*.

Please cite this article in press as: Figueiredo, F., et al. Assessing the quality of textual features in social media. *Information Processing and Management* (2012), http://dx.doi.org/10.1016/j.ipm.2012.03.003