Socialising around media. Improving the second screen experience through semantic analysis, context awareness and dynamic communities

Article

Accepted Version

Tomás, D., Gutiérrez, Y., Badii, A., Tiemann, M. and Aisopos, F. (2019) Socialising around media. Improving the second screen experience through semantic analysis, context awareness and dynamic communities. Multimedia Tools and Applications, 78 (18). pp. 25539-25568. ISSN 1573-772 doi: https://doi.org/10.1007/s11042-019-7706-1 Available at https://centaur.reading.ac.uk/85107/

It is advisable to refer to the publisher's version if you intend to cite from the work. See Guidance on citing.

To link to this article DOI: http://dx.doi.org/10.1007/s11042-019-7706-1

Publisher: Springer

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the End User Agreement.

University of Reading

www.reading.ac.uk/centaur

CentAUR

Central Archive at the University of Reading

Reading's research outputs online

David Tomás [1], Yoan Gutiérrez [2], Atta Badii [3], Marco Tiemann [4], Fotis Aisopos [5],
Socialising Around Media: Improving the Second Screen Experience through Semantic Analysis, Context Awareness and Dynamic Communities Journal Multimedia Tools and Applications, 2019. (https://www.springer.com/computer/information+systems+and+applications/journal/11042).

==

Socialising Around Media

Improving the Second Screen Experience through Semantic Analysis, Context Awareness and Dynamic Communities

David Tomás [1], Yoan Gutiérrez [2], Atta Badii [3], Marco Tiemann [4], Fotis Aisopos [5]

1 & 2) University of Alicante, Department of Software and Computing Systems, Carretera, Alicante, Spain
3 & 4) University of Reading, Department of Computer Science, Reading, RG6 6AH, UK
5) National Technical University of Athens, Distributed Knowledge and Media Systems Group, Athens, Greece

Abstract

SAM is a social media platform that enhances the experience of watching video content in a conventional living room setting, with a service that lets the viewer use a second screen (such as a smart phone) to interact with content, context and communities related to the main video content. This article describes three key functionalities used in the SAM platform in order to create an advanced interactive and social second screen experience for users: semantic analysis, context awareness and dynamic communities. Both dataset-based and end user evaluations of system functionalities are reported in order to determine the effectiveness and efficiency of the components directly involved and the platform as a whole.

Keywords: *Social TV Second Screen Semantic Analysis Entity Linking Sentiment Analysis Context Awareness Community Detection Dynamic Communities*

1 Introduction

The introduction of consumer-centric Internet devices, in particular of smart phone devices, has changed the way users interact with media: from having previously been largely passive and unidirectional, the relationship has now be-come proactive and interactive. Users can select what to watch when, and they can comment or rate a TV show and search for related information regarding characters, facts or personalities. The latter type of behaviour is served through second screen use: the use of a computing device (commonly a mobile device, such as a tablet or smartphone) to provide an enhanced viewing experience for content on another device, also called first screen (such as a television). This experience involves providing interactive features during broadcast content, including media postings on social networks and external content discovery related to the assets consumed by the user.

Today, there are no common standards for the second screen environment; no protocols or frameworks exist through which users can discover and ac-cess information related to consumed contents. Most companies provide second screen resources that are proprietary and cannot be shared in a generic way by other applications or platforms, and links or relationships with their con-tent are not possible. Into this scenario emerges SAM1 (Socialising Around Media), an EU-funded research project focused on developing an advanced digital media delivery platform for second screen and content syndication in the domain of social TV. SAM provides open and standardised means of characterising, discovering and syndicating digital media assets (e.g. films, songs, and books), facilitating third party companies to easily build second screen social orientated experiences for their users.

The potential customers of SAM are both business stakeholders (such as media broadcasters, content asset providers, software companies and digital marketing agencies) and private users. For the former, the platform provides a number of benefits, including dynamic social and media content syndication, the ability to manage online reputation, to better understand customers, to track real-time statistics and to monitor media-related social content through second screen. For the latter, SAM offers a complete solution for interactively consuming media and TV programs. The platform integrates context-aware information and complex social functionalities that provide contextual information relevant to the users' current interests. These features provide the end users with an augmented experience in which they can discover new information about assets and share their experience with other users that are also interested in the same topics.

One of the main business objectives of second screen platforms is allowing companies to increase their audience engagement, and one way to achieve this goal is to enhance the end user's experience (Pynta et al, 2014). This paper focuses on describing and evaluating three back-end functionalities of the SAM platform that are responsible for providing an enhanced experience to the end user when interacting with their second screen: semantic analysis (Ng and Zelle, 1997), context awareness (Abowd et al, 1999) and dynamic communities (Tantipathananandh and Berger-Wolf, 2011). Semantic analysis involves a set of features to import, analyse, enrich and exploit content based on Natural Language Processing (NLP) technologies (Cambria and White, 2014). These features include sentiment analysis (Pang and Lee, 2008) and entity linking (Rao et al, 2013): the former provides information about the sentiments ex-pressed by the users' comments sent via the platform, which allows to better cluster similar users based not just in their demographics and consumption habits but also in their opinions; the later provides an ontology-based data integration mechanism to enrich assets before they are sent to the users by creating links to other assets in the platform and to external data sources (such as Wikipedia). The context awareness functionality manages contextual information and aggregates all user interactions within the system, providing personalisation and smart recommendations to them. Finally, the dynamic communities functionality analyses user data and communication, generates candidate user communities and manages user membership in these communities dynamically. As far as we know, there is no other second screen product in the market that offers these three functionalities together (see Section 2.1 for a comparison with other existing approaches). This study includes the results of the intrinsic evaluation carried out for each of the three functionalities mentioned above, together with an extrinsic evaluation performed on the whole platform, involving around sixty participants that were asked to test it and provide their feedback through a set of questionnaires about different aspects of their experience interacting within SAM.

The remainder of this article is organised as follows: the next section presents the related work in the fields of social TV and second screen, entity linking, sentiment analysis, context awareness and dynamic communities; Section 3 provides information about the architecture and use cases of the SAM platform; Section 4 highlights the role in the enhancement process of the three key functionalities described in this paper; Section 5 summarises the intrinsic and extrinsic evaluation carried out on each functionality and on the whole platform; finally, conclusions and future work are presented in Section 6.

2 Related Work

This section presents existing work related to SAM as a social TV and second screen platform, also reviewing works related to the individual functionalities described in this paper: entity linking, sentiment analysis, context awareness and dynamic communities.

2.1 Social Television and Second Screen

The rise of digital technology and social media gave boost to social television, a field where new opportunities for marketers arise, as multi-screen interactivity results in increased levels of viewers' engagement correlated with increased commercial effectiveness (Pynta et al, 2014).

Social interaction is apparent, especially in the case of collocated viewers using second screen applications designed for specific TV programs (Vanattenhoven and Geerts, 2017). Courtois and D'heer (2012) presented a study on the experience of multi-screening, investigating how users incorporate multiple media (e.g. Facebook and Twitter) in their television viewing experience from a connected second screen. The authors analysed the behaviour and responses of an extended group of participants, monitoring their engagement with various second screen applications. Holmes et al (2012) specifically examined visual attention to television programs while interacting with synchronised second screen applications. Second screen garnered considerable visual attention during experimental viewing sessions, especially interactive content and social media feed from Twitter.

Focusing in the second screen market, there are a number of companies that offer similar features as those to be found in the SAM platform. These companies have been researched and analysed for the specific functionalities described in this paper: sentiment analysis, entity linking (generalised as "Con-tent Composition and Enhancement" in the table below), context awareness, and dynamic communities. The result of this analysis is shown in Table 1, providing an easy visualisation of the available features of each system, with - representing lackings and X representing offers. The functionalities mentioned above are highlighted in bold. Additionally, other features considered as novel or relevant to SAM has been included in this table: whether it is a second screen platform, provides a marketplace for buying and selling assets, and offers business intelligence tools for companies to check the pulse of their audiences.

The companies analysed were ContentWise,2 Couchfunk,3 Horizon,4 Leankr,5 Monterosa6 and Tivine.7.

Feature	CW	CF	HZ	LK	MR	TV
Second Screen Platform	-	-	-	-	X	X
Asset Marketplace	-	-	-	-	-	-
Business Intelligence	-	-	-	-	X	X
Content Composition and Enhancement	-	-	-	-	-	X
Sentiment Analysis	-	X	-	X	X	X
Context Awareness	X	X	X	-	-	X
Dynamic Communities	-	X	-	-	X	-

Table 1: Comparison of SAM rivals in the second screen market: ContentWise (CW), Couchfunk (CF), Horizon (HZ), Leankr (LK), Monterosa (MR) and Tivine (TV).

Compared with existing initiatives in the second screen ecosystem, there is no other product that offers all the specific functionalities studied in this article. The strength of SAM is the inclusion of semantic analysis, context awareness, and dynamic communities' technologies to enhance the end user experience and engagement while interacting with their second screen device. The following paragraphs describe each of these individual functionalities in more detail.

2.2 Entity Linking

Entity linking is the task of matching a textual entity mention to a knowledge base, such as a Wikipedia page, that is a canonical entry for that entity (Rao et al, 2013). For instance, given a mention in a text to "Al Pacino", the goal of this task is to determine that it refers to the entity described in this specific entry in Wikipedia: http://es.wikipedia.org/wiki/Al_Pacino. This task is more challenging than traditional named entity recognition (NER), where the goal is to determine the occurrences of names in text and their classification. In the previous example, a NER system would determine that "Al Pacino" is a person or that "Los Angeles" is a location (Nadeau and Sekine, 2007). Entity linking requires a NER system, but this process must be complemented by a following disambiguation phase where this person or location is linked to an unambiguous entity stored in a knowledge base.

Entity linking has been used in different domains to enrich content by linking them to existing knowledge bases such as Wikipedia, CYC8 and Freebase9. Odijk et al (2013) presented an approach that automatically generates links to background information (Wikipedia articles) in real-time, in order to be shown in second screen. This paper reflected the work performed in the context of LiMoSINe EU Research Project (LiM, 2013) concerning language-based technology search and employing semantic linking based on subtitles. Specifically, media-related subtitles were used as a textual stream to generate links and also second screen context was efficiently modelled using a graph-based approach.

The NoTube EU project (NoT, 2012) also worked towards TV experience personalisation enhancing online social and semantic data. The project focused on the task of enriching TV metadata and creating a linked data cloud, as well as generating user models and profiles via their social web activities. This resulted into a personalised recommendation system that used existing web services and shared background knowledge to collect, enrich and recommend TV data without an intrusive user profiling process (Aroyo et al, 2011).

Similarly, to these projects, SAM exploits entity linking in order to enrich the assets' content by automatically identifying which assets in the platform are interrelated, connecting them, and also providing external links to related Wikipedia pages building a linked-data ecosystem.

2.3 Sentiment Analysis

Sentiment analysis, also known as opinion mining, is an area of NLP focused on identifying and extracting subjective information from human language (Pang and Lee, 2008). Sentiment analysis systems try to identify the attitude of the author of the analysed source of information with respect to some topic, entity or overall contextual polarity (positive, negative or neutral) of the text. The use of sentiment analysis technologies can benefit companies and users in their decision-making processes, since it makes possible to deter-mine user preferences, opinions and feelings. These technologies can be applied at different levels, depending on the focus of the analysis: global polarity and aspect-based polarity. Global polarity allows inferring the sentiment polarity expressed in the whole text. Aspect-based polarity aims to classify the sentiment with respect to the specific aspects of an entity, since an opinion holder can give different views for different aspects of the same entity (e.g. “The film is not very good, but the main actor is superb. I’m happy to watch him acting again”).

Sentiment analysis has become one of the hottest research areas in computer science. Many companies and research groups are developing sentiment analysis solutions that tries to leverage the huge amount of subjective in-formation available from social media in order to monitor reputation about products and services. Work in Fernández et al (2015) presented a sentiment analysis approach for the social context in a second screen scenario, addressing the Task 1 (sentiment analysis at global level) of the TASS 2015 competition (Villena-Román et al, 2015). In this model, a sentiment lexicon was created using the individual words, n-grams and skip-grams of tweets datasets, with each term being statistically scored according to their appearance within each polarity. This lexicon was exploited using a Support Vector Machine (SVM) algorithm Joachims (1998) to build a classifier. Zhao et al (2011) attempted to extract TV watcher’s sentimental reaction to major events in live broadcast sports games in real-time. They followed a lexicon-based approach to streaming data from Twitter, using WordNet10 as a lexical database and analysing the sentiment polarities evolution over time. The authors also introduced a social TV system that enables the audience to better select interesting programs in real-time and to produce personalised summaries.

Unlike other projects and products in the context of second screen, sentiment analysis is employed in SAM in three different ways: first, it provides information about end user’s opinions on specific assets consumed in the plat-form for business intelligence purposes, providing reports to the content creators on the polarity and intensity of the comments posted by users while consuming their assets; secondly, it is employed to build dynamic groups of users inside the platform based on their similar opinions; finally, sentiments are also considered as part of the context analysis of the user for assets recommendation.

2.4 Context Awareness

Context awareness is the task of capturing and managing users’ contextual information. In a social TV environment, context awareness can be related to the user’s’ public information (e.g. location and gender) or users’ interactions and social behaviour (e.g. friends and comments). Hu et al (2014) presented a multi-screen social TV architecture, using geo-location data and user social features to enrich TV viewing experience. Later work by the same authors (Hu et al, 2015) provides a unified big data platform for social TV analytics, mining social responses associated with TV programs obtained from Sina Weibo11 social platform.

A major result of this analysis targets at the personalisation of mobile digital TV applications, by predicting and recommending to television viewers content that match their interests (Chorianopoulos, 2008). Mitchell et al (2010) used social networks as a mechanism for providing social awareness to users of an Internet Protocol Television (IPTV) system, resulting into a user-user recommendation and rating system.

Work in Cesar et al (2008) investigated the usages of the second screen in an interactive television environment, aiming at controlling, enriching, sharing, and transferring TV content. Geerts et al (2014) analysed a second screen companion application stimulating social interaction to offer more insight into how viewers are experiencing such applications. Finally, Giglietto and Selva (2014) performed a content analysis to a big dataset of second screen tweets throughout a TV season, clarifying the relationship between politics-related shows and social media content.

To the best of our knowledge, no previous work has already taken into account both the first screen (e.g. video full screen, “likes” and comments) and second screen (e.g. show-more and dismiss) interactions in social TV to personalise content delivered. Also, our context management approach includes an innovative graph analysis method that aims at providing a competitive second screen user profiling and recommendation service in terms of efficiency.

2.5 Dynamic Communities

The creation of dynamic user communities through social media is a phenomenon that has become very popular, along with the emerge of social media themselves. Kaplan and Haenlein (2010) describe how several companies are already using social networking sites to support the creation of brand communities. Much work has focused on mining and detecting communities in social media (Papadopoulos et al, 2012) (Greene et al, 2010) (Tang and Liu, 2010), usually employing graph-based algorithms and placing emphasis on their performance for enabling scaling community detection to huge real-world social networks.

The approach taken in SAM is to use a generically applicable scalable community identification algorithm, and to create the graph structure to which the algorithm is applied from data gathered through the SAM platform itself. Specifically, users' social media interactions over the platform as well as their interactions with the platform (e.g. selecting a specific video file or second screen element) at a later stage of development.

3 The SAM Platform

The SAM platform is the main technological outcome of the SAM project. This platform offers a complete end-to-end system both for business users who create, manage and evaluate second screen experiences, and for end users who consume and interact with their second screen through SAM. The platform has been developed as a modular system whose individual components provide the necessary functionalities or enable them through shared services.

Figure 1, below, depicts the overall modular architecture of SAM. The components of the platform are distributed in four different layers:

– Data Management. This layer's components are responsible of importing and storing all the assets and metadata in SAM. This layer contains the Cloud Storage service component and its modules, which realise the storage of different data types. It provides also support to the communication with third party systems and external social components (such as Twitter and Facebook). The components in charge of these functionalities are the

Content Gateways and Social Components, respectively.
– Control. This layer hosts all the components that offer control over the platform, including the core functionalities for enhancing the second screen experience described in this paper. These functionalities are Semantic Ser-vices (provides sentiment analysis and entity linking) and Context Control, which includes Context Manager (responsible of context awareness functionality) and Community Manager (controlling dynamic communities). The other two components that form this layer are Brand and Consumer Protection, addressing brand protection mechanisms and inappropriate content filtering, and Identity and Security Services, focused on the protection of the platform against unauthorised use.

– Communication. This layer includes the Interconnection Bus that coordinates and facilitates the communication between the different SAM components, providing built-in facilities such as message routing, format trans-formation, message queuing, security, and access control.

– Interaction. This layer gathers all the front-end components for business and end users to communicate with the platform. The Marketplace is the area where content providers can publish and access existing assets, making them available to the end users. The Linker allows the composition and aggregation of assets, linking content together with other content or meta-data available in the Marketplace. Analytics is the component in charge of reporting to the different stakeholders using Business Intelligence and advanced Social Mining techniques. Finally, the Dashboard displays video elements on the first screen (usually a smart TV) and additional information on the second screen (a tablet, smartphone, etc.).

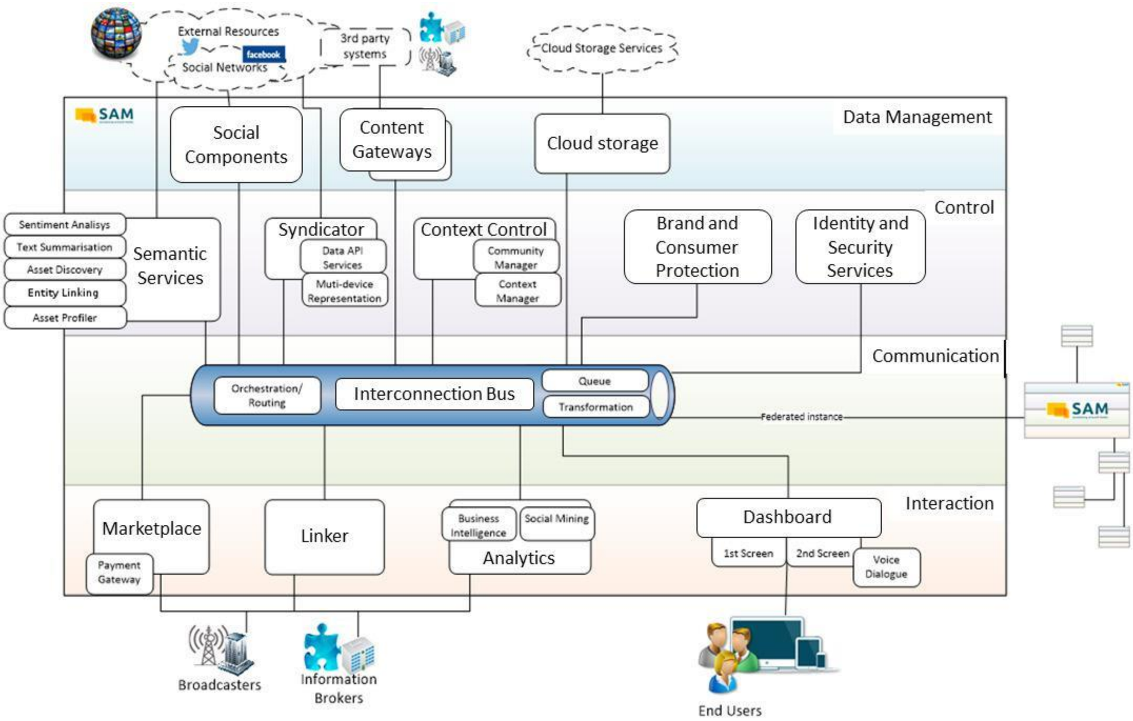


Fig. 1: The SAM Platform architecture.

The remainder of this section summarises the services provided by the platform for business users and end users of SAM.

3.1 Business Use Cases: Experience Creation and Management

The goal of the SAM platform for business users and their use cases is to improve the efficiency, quality and integration of the second screen content creation process into work flows and content provider ecosystems, facilitating that second screen experiences can be provided for many programs instead of just for prime-time entertainment content.

Second screen experiences can be created and managed through a dedicated Marketplace component offering a user interface for business users. Here, content managers can import and access media items (referred to as “media assets”) in order to compose a second screen experience. The system developed during the course of the project supports a number of different content types and formats such as Wikipedia entries consisting mostly of textual content, and video clips that complement first screen video content.

Using a Linker component, content editors can search for, receive suggestions for (using the entity linking functionality described in Section 4.1.1) and then compose available assets using a time line-based view of the first screen media asset for which a second screen experience is to be created.

In addition to including media assets to a second screen experience, editors can also configure and control content access restrictions through the Brand and Consumer Protection component (e.g. to protect minors), configure the integration of social media channels into the second screen experience for enabling SAM dynamic community functionalities (described in Section 4.3), or integrate external social media services such as Facebook or Twitter. Once an editor has completed a second screen experience, it can be reviewed and published through the SAM platform so that it is ready for use by end users.

After a second screen experience has been published and used by end users, SAM gives business users access to data analytics tools (Analytics component) that let them analyse and visualise user activities and user interactions, including analyses of their activity, the contents of their interactions on social media and the sentiments expressed (through the sentiment analysis functionality described in Section 4.1.2).

3.2 End User Use Cases: Consumption and Interaction

For end users, the aim of the SAM platform is to provide them with an easy to use, simple and seamless user experience for consuming and interacting with second screen media while audio-visual content is playing on a first screen device.

Second screen experiences created for end users using the SAM platform infrastructure are launched and synchronised automatically with first screen through paired SAM applications that run both on a television set that serves as first screen and on a hand-held device that displays the second screen. Both the content synchronisation and the delivery of second screen content are handled by dedicated content selection and delivery components that are part of the SAM platform (the Syndicator and Dashboard components in Figure 1).

While a first screen video is playing on a SAM-enabled TV, connected SAM applications display related content assets that have been added to the second screen experience for this video by content editors, leveraging the suggestions provided by the entity linking functionality. Users can interact with the content in different ways (e.g. play video clips or expand teaser text for longer text assets). This constitutes the first set of functionalities offered to end users which focuses on augmenting first screen content with additional second screen content. Users of the SAM second screen application can furthermore interact with each other through social media messages while a first screen video is being played. The SAM application includes both external social media services (Twitter and Facebook) and a social media service developed for the SAM plat-form, managed by the Context Control component. User interactions through the SAM social media service resemble interactions over Twitter. Unlike Twitter, the SAM social media service is based on the concept of groups to which messages are posted and only members of a group receive a message sent to that group. Users of SAM can be automatically invited to groups using user profiles that are maintained for each user of the platform.

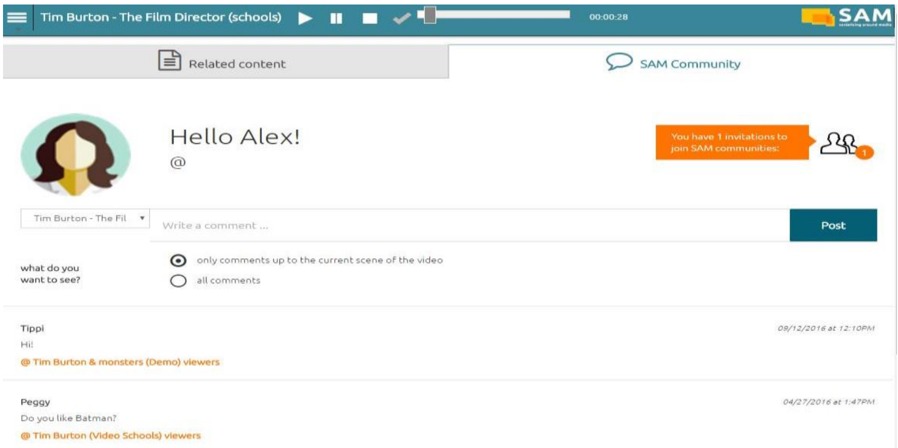


Fig. 2: The SAM Application Community user interface.

Figure 2 illustrates the user interface for SAM dynamic community interactions. In this example, a user is invited to join a Tim Burton's film community. The way these communities are dynamically created is further described in Section 4.3.

4 Improving the Second Screen Experience

This section describes in detail the three functionalities of SAM that are crucial for realising advanced features in the interaction of the end users with the platform through their second screens. Other functionalities provided by the platform, such as the asset marketplace, content syndication and responsive user interfaces, are out of the scope of this paper.

The following paragraphs provide information on the architecture, components and scientific background underlying these functionalities, highlighting their role in achieving the enhancement of the end user experience while interacting with their second screens in SAM.

4.1 Semantic Analysis

The Semantic Services component of SAM is in charge of providing semantic analysis functionalities to the platform. These functionalities include sentiment analysis, entity linking, ontology mapping and exploitation, text summarisation and asset edition. These features are supported by an ontology representing digital media assets and their relationships. Assets in the platform are stored as instances of this ontology (from now on, the SAM ontology), created in order to support semantic representation and querying of the data.[12] The SAM ontology reuses concepts from the Europeana Data Model[13] (EDM) and Schema.org among others, defining new concepts and relationships when no suitable elements were available in the previously mentioned schemas.

Since the sources of media information to be distributed through SAM are heterogeneous, and data imported into the platform follow different formats and schemas, SAM includes a mechanism for ontology mapping that for each concept in the input schema provides a list of suggestions of related concepts in the SAM ontology. In this way, content providers can manually select the best match between their data and the SAM ontology and run a batch importing process.

In order to facilitate the interaction with other SAM components, all the functionalities provided by the Semantic Services are exposed as RESTful interfaces. The following paragraphs highlight the parts of this component employed in the task of enhancing the end user experience while interacting with their second screen: entity linking and sentiment analysis.

4.1.1 Entity Linking

In the context of SAM, entity linking allows to analyse text and identify entity mentions in two different knowledge bases: Wikipedia and the SAM knowledge base, which stores the assets imported and created inside the platform. In this way, the data existing in the platform can be analysed for entities, and these entities can be linked to Wikipedia pages and other assets in SAM (e.g. books, songs, films, actors, etc.), creating a linked data ecosystem.

Depending on the target knowledge base, Wikipedia or SAM, different approaches to the entity linking task have been defined. Although the task at hand is the same, the tools and resources employed are different. The following paragraphs describe these two approaches. For further information refer to Tomás et al (2015).

The approach to entity linking on Wikipedia is based on OpenNLP, 14 DB-pedia15 and DBpedia Lookup.16 OpenNLP is a library that supports different NLP tasks, such as tokenisation, part-of-speech tagging, parsing and named entity extraction. As a first step, this library is employed to identify and ex-tract from text noun phrases and named entities that form the set of candidate terms to be linked to Wikipedia. DBpedia, the structured version of Wikipedia, is used in this approach to facilitate querying and further processing of its contents.

The candidate terms extracted by OpenNLP are passed to DBpedia Lookup. Given a set of keywords, this tool retrieves a set of related DBpedia entries, either because the label of the entry, or an anchor text frequently used in DB-pedia to refer to that entry matches the query. This feature avoids the need for a perfect matching between the query and the resource, overcoming the problem of shortened forms (e.g. Leonardo DiCaprio / Leo DiCaprio), aliases (e.g. Dwayne Johnson / The Rock) and abbreviations (e.g. British Broadcasting Corporation / BBC).

Finally, a disambiguation algorithm is applied to solve the problem of entity ambiguity, since a single mention to an entity can match multiple knowledge base entries (e.g. “Francis Bacon” can refer to both the English philosopher and the Irish artist). This disambiguation algorithm ranks the previously identified DBpedia entries for each candidate entity. This problem is addressed using three different strategies:

– Number of inlinks. In the Web, an inlink to a web page w is a URL of another web page which contains a link pointing to w. This disambiguation method ranks the candidate DBpedia entities taking into account the number of inlinks pointing at those pages from other DBpedia pages (more inlinks implies better ranking).

– Context distance. This method compares the text surrounding the mention to a candidate entity (at sentence level) with the description contained in DBpedia. The similarity between textual contexts is computed applying the Levenshtein distance, defined as the minimum number of single-character edits (insertions, deletions or substitutions) required to change one text into the other (Levenshtein, 1966).

– Hybrid. This approach combines the two previous methods. Number of inlinks and context distance values are computed and normalised between 0 and 1. The resulting value is obtained by computing the average between these two values.

Regarding the SAM knowledge base, the approach to entity linking tries to link entities identified in text to the assets stored as instances of the SAM ontology. This approach uses OpenNLP and Lucene17 as its core tools. Lucene is an information retrieval library, which provides search functionalities by adding content to an index and allowing to perform queries on that index.

In this case, the process of entity linking requires a previous offline process, where the contents of all the instances of the SAM ontology are indexed by Lucene as separate documents. These contents consist of information about media assets (e.g. metadata about a book, including its title, author, publisher, synopsis, number of pages, etc.).

After this offline process, OpenNLP is used to identify candidate entities in text as in the approach to Wikipedia. The candidates obtained are then used to query Lucene (as a string of keywords) to retrieve the most relevant documents (media assets) for them. Lucene was configured to perform a fuzzy search using a similarity measure based on the Damerau-Levenshtein algorithm (Damerau, 1964), reducing in this way the problem of name variations.

Finally, a disambiguation approach based on context distance (the same described before for Wikipedia) is applied to re-rank the list of candidate instances retrieved by Lucene, providing a final set of ranked assets for each entity initially identified by OpenNLP.

Regarding the benefits for end users in their second screens, entity linking in SAM provides an augmented experience in which users can discover new information about an asset. For instance, a user watching the film “Casino Royale” in the SAM platform, taking advantage of the entity linking functionality, would get additional information related to actors “Daniel Craig” and “Mads Mikkelsen” from Wikipedia, and also to other related media assets in the platform based on the linking to the SAM knowledge base, such as books created by “Ian Fleming”, the writer of the series of spy novels, or references to its original soundtrack.

4.1.2 Sentiment Analysis

In SAM, the application of sentiment analysis to user comments allows extracting valuable information about the consumption of products and services. The benefit that sentiment analysis provides to the end users is the discovery of user preferences, feelings and attitudes while using the platform, which contribute as an input to improve content awareness (see Section 4.2) and dynamic communities (see Section 4.3). Moreover, this information is used by the Analytics component to provide content creators with advance reports for business intelligent purposes, although this application of sentiment analysis in the context of SAM is out of the scope of this work.

The approach to sentiment analysis starts with a normalisation of user comments, translating the usual informal style of these texts to a more formal register. Each user comment is normalised following these steps:
Lower-case conversion. Characters in the text are converted to lower case.
Character repetition removal. When a character appears more than two times in a row, further repetitions are removed. For example, the words “gooood” and “gooooood” would be normalised to “good”, but the word “good” would remain unchanged.

Users’ names and hashtags substitution. Every user’s name and hashtag found is generalised by replacing them with USERNAME and HASHTAG labels, respectively.

The next step is to tokenise the comments to extract all their terms, which are combined to create lexical patterns of skip-grams Guthrie et al (2006).18 In this step, a group of adjacent characters of the same type (letters, numbers, or punctuation symbols) is considered a term. For example, the text “want2go!!” would be tokenised as four different terms: “want”, “2”, “go”, and “!!”.

Each skip-gram is weighted based on the number of terms that it contains and the number of gaps between these terms. This weighting schema is based on the concepts of density and relevance of a skip-gram:

$$density(s,t) = \frac{terms(s)}{terms(s)+skip(s,t)}, \qquad (1)$$

where $density(s,t)$ represents the density of the skip-gram s in the text t, $terms(s)$ is the number of terms in the skip-gram s, and $skip(s,t)$ is the number of skips of the skip-gram s in the text t; and

$$\mathrm{relevance}(\mathrm{s},\mathrm{t}) = \frac{terms(s)}{terms(t)}, \qquad (2)$$

where $relevance(s,t)$ represents the relevance of the skip-gram s in the text t. The final weight of a skip-gram can be defined as the product between its density and its relevance:

$$weight(s,t) \; = \; density(s,t) \; \times relevance(s,t) \qquad (3)$$

Finally, skip-gram patterns are employed as features of an SVM algorithm. By extracting these features from a corpus with user comments labelled as positive, negative or neutral, SVM is trained to classify new user comments in these three categories. The algorithm used was the SVM implementation provided by LibSVM (Chang and Lin, 2011), using a linear kernel with parameters $C \; = \; 1$ and $\in = 0.1$. For a complete description of the approach to sentiment analysis followed in SAM refer to Fernández et al (2015).

4.2 Context Awareness

Context Awareness is a crucial characteristic of SAM. To personalise the user experience, the platform needs to collect and analyse all context-related data, such as user interests and interactions, and provide user recommendations for items of priority (high relevance). Resulting recommendations can personalise the SAM environment in the two screens by: (i) prioritising the video carousel items presented to users in the first screen based on their scores, highlighting the most relevant recommendations; and (ii) suggesting content in second screen by only showing in the top side of the Dashboard widgets with most relevant assets for the user.

The context awareness functionalities have been realised in SAM by means of the Context Manager component, which communicates with the rest of the platform through a RESTful web services API. In both first and second screen applications, specific action listeners have been implemented, capturing and sending user interactions with the various elements and buttons to the Context Manager component in order to aggregate all contextual data for further analysis.

As noted in relevant literature (Jaiswal and Agrawal, 2013), NoSQL graph databases such as Neo4j19 can be applied efficiently to user-entity relation-ship models such as the one defined in SAM. Moreover, the flexibility of such databases is another characteristic that makes databases like Neo4j suitable for context representation (Batra and Tyagi, 2012). Taking these studies into ac-count, a Neo4j graph database was used as the Context Manager back-end, saving contextual data as triples of subject-object interactions (e.g. “user-LIKES-asset”). Thus, in the context of this graph, registered SAM users, existing videos, second screen assets or even plain keywords are represented as nodes, whereas interactions are modelled as graph edges connecting these nodes.

In what follows, we provide an extension and concrete validation of the work presented in Aisopos et al (2016) regarding context awareness in SAM.

4.2.1 Contextual Information Graph

The list of interactions of interest that are collected from the SAM first and second screen listeners (for videos or other assets, respectively) can be seen in Table 2.

Root assets	Second screen assets
consume comment full screen	like dislike dismiss show more

Table 2: List of collected user interactions.

A visual example of the context graph illustrating some button-pressed actions collected in Neo4j database is shown in Figure 3. In this figure, various SAM end-users can be observed (blue nodes) interacting (e.g. consuming, commenting, pressing "Like" button, etc.) with SAM assets (green nodes).

4.2.2 Context Analysis

The exploitation of the aforementioned contextual information aggregation resides within the analysis of the relationship of every user and the various assets. To this end, the system provides a relevance score REL_{ij} of each user u_i with every SAM asset a_j in order to personalise the user second screen experience by providing on screen recommendations of assets of interest.

The specific problem that is targeted by context analysis is asset recommendation: given a set of users U and a set of assets A, the goal is to prioritise all $a_j \in A$ for every user $u_i \in U$, based on their relevance REL_{ij}.

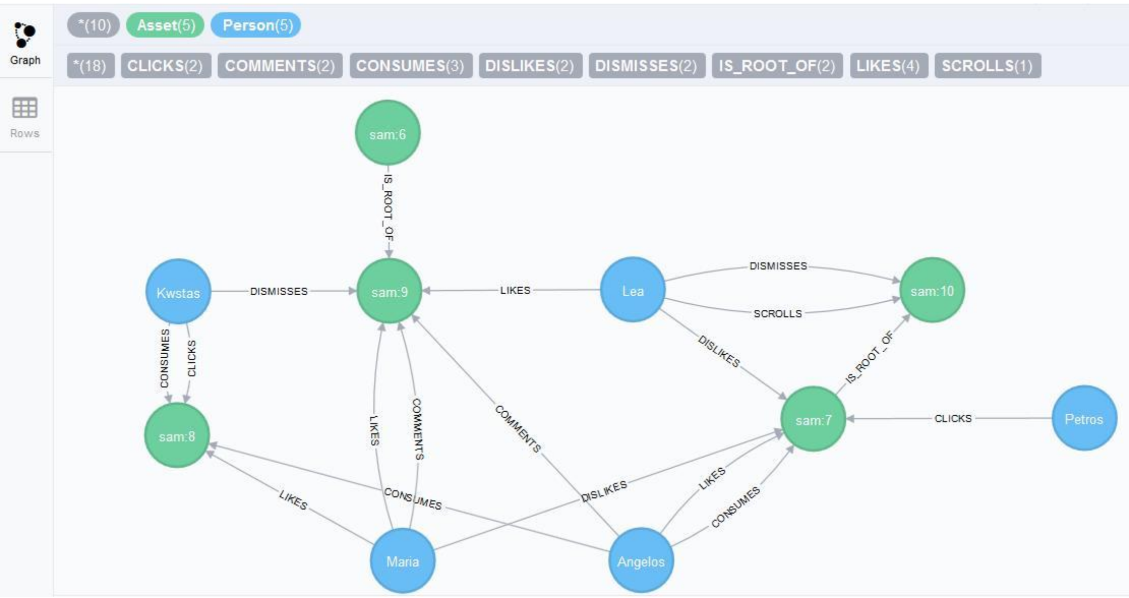


Fig. 3: Screenshot of Neo4j browser showing an example of users/relationships graph.

The calculation of a specific asset relevance a_j for a user u_i, depends on all the interactions collected, as well as the user contextual connection with asset-related keywords (e.g. via a relation with a keyword's parent node). For this purpose, asset keywords are also imported into the Neo4j graph as separate nodes, with the sentiment analysis functionality of SAM providing user sentiment scores on each keyword by analysing their comment history (see Section 4.1.2). These scores are reduced in the $[-1, 1]$ interval, with -1 representing the negative edge and 1 the positive edge.

Regarding the user interactions, each one is also weighted between $[-1, 1]$, which is the chosen weighting climax in order to be included in the analysis. Interactions that explicitly declare relevance/irrelevance ("like" / "dislike") are weighted with the absolute scores -1 or 1 respectively:

$$W_e = \begin{cases} 1 & for\ positive\ interaction \\ -1 & for\ negative\ interaction \end{cases} \quad (4)$$

For the rest of the interactions ("non-explicit"), the approach dictates that the sum of their weights will not overshadow an "explicit" interaction (so their sum in the worst case will be less than 1 or greater than -1). Thus, the relevance weight for each non-explicit interaction can be calculated from the following equation:

$$W_{ne} = \frac{1}{|NE|+1}, \quad (5)$$

where $|N\ E|$ is the maximum number of non-explicit interactions that can be collected for an asset type. The relevance weights applied to all user inter-actions collected in the current case are shown in Table 3.

Interactions	Root asset	Second screen asset	Keyword
Explicit weights (w_e)			
Comment	$[-1,1]$		$[-1,1]$
Like	1	1	
Dislike	-1	-1	
Non-explicit weights (w_{ne})			
Full screen	1/3		
Consume	1/3		
Dismiss		-1/3	
Show More		1/3	

Table 3: Relevance weights of the various user interactions with root or second screen assets.

Given the aforementioned ratings, a first user-asset relevance score can be considered as the sum of "explicit" and "non-explicit" weights:

$$REL_{ij1} = \sum W_e + \sum W_{ne} = \sum W_e + \sum \frac{1}{|NE|+1} \quad (6)$$

However, apart from a direct relation to an asset, a user interaction with neighbouring assets is also an indication of relevance/irrelevance to it. For example, in Figure 4, case (a) shows a user disliking an asset of a video. Every such relationship over this video's related assets implies potential irrelevance of the video itself to this user. Case (b) shows a user consuming in full screen mode and also commenting on a video (supposedly in a positive way). All these strong relevance weights to this video can also be considered as indications of relevance with the video's related assets (those assets are connected to it in the graph via the IS_ROOT_OF relationship).

Similarly, the relevance value of those indirect relationships with neighbouring assets $a_j' \in A$ must not overshadow the direct or explicit interactions. The total relevance score resulting from those is set as the sum of their weights (now set as W_e' and W_{ne}') divided by a distance factor d (number of hops in the graph between the user and asset, following the path of those edges/relations):

$$REL_{ij2} = \frac{REL_{ij'1}}{d} = \frac{\sum W_e' + \sum W_{ne}'}{2}, \quad (7)$$

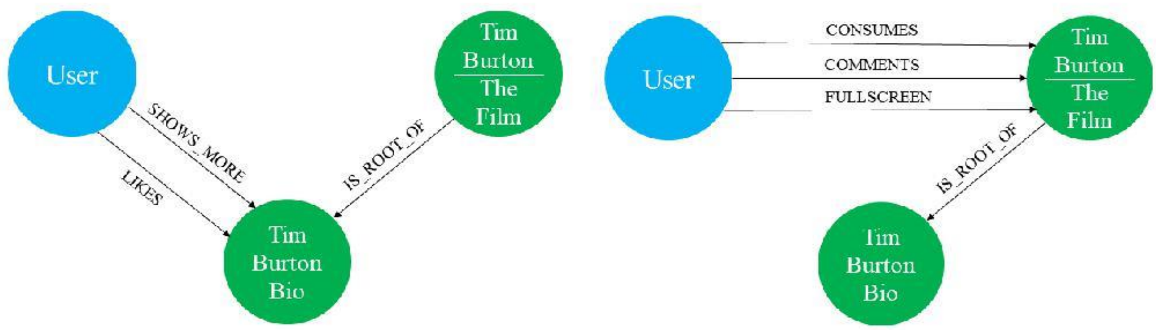
where $d = 2$ in the case of direct neighbours. The total relevance score is defined as a heuristic function, equal to the sum of those two side scores:

$$REL_{ij} = REL_{ij1} + REL_{ij2} = \sum W_e + \sum \frac{1}{|NE|+1} + \frac{\sum W_e' + \sum W_{ne}'}{2} \qquad (8)$$

4.3 Dynamic Communities

The Community Manager component of SAM manages the social media functionalities provided by the SAM platform, including the exchange of messages and the identification and recommendation of communities to end users. Hence it both operates the technical infrastructure that is required in order to transmit messages between users of a community and handles mechanisms for managing user membership in such groups. As a core feature, it implements algorithms that identify which communities might be interesting to an end user given their user profile data, inviting the user to join a relevant community.

The dynamic community's backend functionalities have been implemented using a MySQL database system, a RESTful web services API, and a draft version of the W3C ActivityStreams social messaging format.20 Processing is triggered by changes to the user profile data stored for a user (including new social media activity).



(a)User interacts with an asset linked to a video.

(b) User interacts with the root video of an asset.

Fig. 4: Indirect relation to an asset.

The Context Manager triggers a process for running algorithms in order to compute community invitations for a user when it has processed a profile up-date for a user. Such a profile update can occur, for instance, when a user has started playing back a new video, or when a user has made a comment on social media that has been analysed using sentiment analysis. For the technical implementation of the dynamic community suggestion algorithms, a modular infrastructure has been implemented so that different types of algorithm implementations can be swapped out easily.

The dynamic community's system supports both a deterministic and a machine learning approach for community detection. This is intended to sup-port both editorial-level community creation and detection and inclusion of naturally emerging communities.

4.3.1 Deterministic Community Creation

The goal of deterministic community creation is to give administrators running SAM the opportunity to define when an end user should receive an invitation to a particular community. A rule-based approach has been selected in order to achieve this goal. Based on the Drools Rule Engine library, 21 an event-condition-action rule processing module allows the explicit specification of rules and the execution of community management actions given the outcome of rule processing activities. In addition to basic Rete rule processing (Forgy, 1982), Drools also supports data aggregation and temporal restrictions on data to consider (e.g. sliding windows over time-stamped data.

Listing 1, below, depicts a typical rule that may be defined in order to target a specific user demographics (in this instance, young male adults who like a particular content asset).

4.3.2 Clustering for Community Detection

Using a clustering method in order to identify communities within groups of users does not require manual effort for dynamic community generation, and may identify communities that would not have explicitly been defined if the explicit community definitions described in the previous sections had been used. Especially, since social networks and social media channels have established themselves as important communication channels for large user communities, the automatic detection of user communities in a large group of users has attracted research interest (Paliouras et al, 2015).

Many different clustering algorithms can be applied to the problem of com-munity discovery in large user communities. For the SAM platform, an initial selection of hierarchical divisive clustering, k-means clustering and standard graph-based clustering techniques was initially examined. Graph-based clustering approaches were found to be the most flexible in implementation and also considered as both well-suited to community discovery problems and as an interesting and active research field of interest within the project.

```
rule "Young Male Adults "
when
        m : ContextNotification ( $consumedAsset :
        consumedAsset,
              person.getGender( ) == "male", $user :
        person )
        eval( m.getPerson( ).getAge( ) < 21 )
        eval( m.getPerson( ).getAge( ) > 15 )
        eval( !$user.isMemberOf ( "Young Adults who like "
               + $consumedAsset.getTitle( )) )
        eval( m.getCreatedGroupName( ) == null )
then
        m.setCreateGroup ( t r u e ) ;
        m.setCreatedGroupName ( "Young Adults who like "
            + $consumedAsset.getTitle( ) ) ;
end
```

Listing 1: Example rule "Young Male Adults"

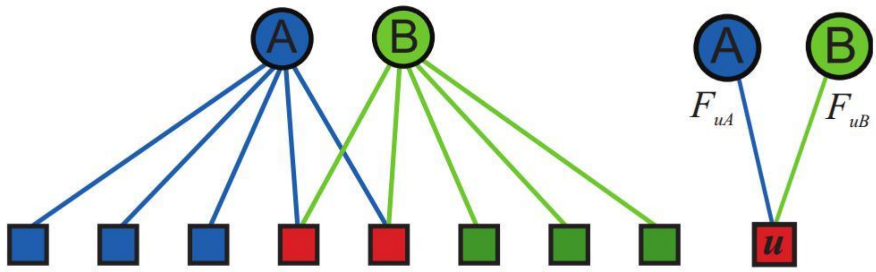


Fig. 5: Illustration of affiliation graph structure (Yang and Leskovec, 2013).

After initial investigations of the "classic" Girvan-Newman algorithm for community discovery (Girvan and Newman, 2002), the BigCLAM algorithm was selected as a simple, fast and scalable algorithm for graph-based community detection.

The BigCLAM algorithm, proposed by Yang and Leskovec (2013), has been designed for operating in big social or similar networks. The algorithm searches for the most likely affiliation factor matrix that maps a number of communities to an undirected and unlabelled network. The algorithm can detect overlapping and non-overlapping clusters of users. The number of communities to fit can be defined as a parameter, or it can be estimated from the network under consideration. Figure 5 shows an example where circles represent communities, squares represent users and the edges between them represent node affiliation (e.g. membership of a community). Edges are weighted (F_{uA} and F_{uB}) to represent the degree of affiliation.

The underlying optimisation problem of finding an affiliation factor matrix is considered as a variation of Non-negative Matrix Factorisation (NMF) for finding an approximation matrix. In the implementation used for SAM, Noesis framework for network data mining, a block gradient ascent algorithm, is used to solve this optimisation problem (Martínez et al, 2015).

In the investigations for the SAM platform, the focus was placed on the identification of concepts and user attitudes towards those concepts. In this model, concepts are used in order to create user profiles of bags of words with concept representations identified using sentiment analysis of social media messages, which are used in order to create the basic user graph that is used by the BigCLAM algorithm (Leskovec et al, 2014).

In the created graph, users are represented by feature vectors of concepts (such as "loves-football") and weighted edges between user nodes are created by determining the similarity of user feature vectors using squared Euclidean distance (edges are omitted when no features are shared between feature vectors). This approach to user representation exclusively uses data extracted from social media contributions via sentiment analysis. It can be replaced or augmented by incorporating additional factors. In particular, information on which first/second screen content has been viewed by users represented in the graph if sufficient data for this is gathered, although implementing this point was outside of the scope of the SAM project. Irrespective of how the described graph has been created, the BigCLAM algorithm is then applied in order to identify communities given the created graph representation.

5 Evaluation

This section presents two types of evaluation of the system described in this article. First, it describes the data-driven evaluation of the individual functionalities and algorithms described in the article in order to provide information about their performance on standard datasets. Second, it reports on a large-scale end user evaluation carried out as part of the SAM project, in which the overall approach of the SAM platform and its appeal to end users has been evaluated.

.1 Core Functionalities

This section reports the evaluation results for the three technical topics covered in Section 4, providing references to relevant publications with additional details on the individual evaluations.

5.1.1 Semantic Analysis

The intrinsic evaluation carried out for this functionality involved the entity linking (in both SAM assets and Wikipedia pages) and sentiment analysis tasks. Results for both tasks are summarised in Table 4, below.

Experiment	Precision	Recall	F1 score
Entity linking			
Wikipedia	0.87	0.89	0.88
SAM knowledge base	0.90	0.89	0.88
Sentiment analysis			
Global polarity	0.75	0.68	0.71
Aspect-based polarity	0.61	0.55	0.58

Table 4: Performance of the semantic analysis functionalities.

In the case of entity linking, the goal of the experiments was to evaluate the performance of the system in identifying mentions to different named entities (person, fictional character, book, video game, organisation, album, and song) in plain text (a set of paragraphs) and linking them to their corresponding Wikipedia page or SAM asset. In absence of a suitable dataset in the digital media domain, a corpus was developed in order to evaluate this task. To this end, a list of IMDB22 500 top rated films was retrieved. A crawler processed this list to retrieve the description for each film from Wikipedia, i.e., the set of paragraphs occurring before the table of contents. The resulting corpus contained more than 4,500 entities, with person as the most common type of entity and video game as the least one. The descriptions obtained for each film contained on average 27.21 words and 2.22 entities.

In the case of entity linking on Wikipedia, for each possible entity an average of 1.91 candidates were found. As shown in Table 4, the system obtained 87% precision, 89% recall and 88.47% F1 score in the corpus described above. As a reference to value the performance of this approach, state-of-the-art system in the entity linking task organised at SemEval-201523 obtained 88.9% F1 score, very close the performance obtained by the approach developed in SAM. In this competition, the corpus consisted of 138 documents and 1,175 entities tagged, approximately three times smaller than the corpus developed in SAM for this evaluation.

For entity liking on the SAM knowledge base, taking into account that in the current stage the SAM knowledge base is not massively populated, the Wikipedia corpus described above was also employed in this experiment. To this end, every film description was imported into the platform as an individual asset to populate the SAM knowledge base. The system obtained 90% precision, 89% recall and 89.49% F1 score. Again, this result is comparable to that achieved by the state-of-the-art approach presented at SemEval-2015.

With regard to the sentiment analysis functionality, it was evaluated for both global polarity and aspect-based polarity detection. The global polarity algorithm was evaluated with SemEval-2013 Task 2 dataset24 (10,709 user comments) and Metacritic dataset25 (5,888 users' movie reviews). In the case of aspect-based polarity, the evaluation dataset was SemEval-2014 Task 4 - Restaurants26 (3,009 user comments).

As shown in Table 4, the performance achieved by global polarity was 75% precision, 68% recall and 71.32% F1-score. This result overcomes the 69.02% F1 score obtained by the best system reported in (Theresa Wilson and Ritter, 2013). The performance achieved by aspect-based polarity was 61% precision, 55% recall and 57.84% F1 score, which is comparable to the 58.16% obtained by the best performing system in the SemEval competition.

In order to perform a real testing in the SAM context, an evaluation campaign was carried out with real users interacting with the SAM platform (see Section 5.2 for further details on the experimental setup). The result of this experiment provided 143 interactions (messages sent to the platform) by the users of SAM while consuming different assets. This dataset was manually annotated by three reviewers. Each message was classified as positive, negative or neutral. The inter-annotator agreement was computed using Fleiss' Kappa metric (Fleiss, 1971). The overall Kappa for the three annotators was 0.79, which is considered as “substantial” according to the scale proposed by Landis and Koch (1977). The results obtained for global and aspect-based polarity in this experiment was 75% precision, recall and F1 score. In this experiment, all the user comments included only one aspect per sentence.

5.1.2 Context Awareness

The purpose of this evaluation was to validate that users get accurate recommendations from the Context Manager component, as well as to evaluate its overall efficiency.

In order to properly test the recommendation algorithm, a large dataset from a Carnegie Mellon study (Zhang, 2005) was imported into the platform to work with a meaningful set of user interactions. These dataset records a considerable number of user interactions of various types with online articles, including number of clicks, scrolls, likes, time spent on each article metrics (and preliminary dismisses), as well as explicit relevance ratings as a gold standard.

In the SAM platform, the second screen application contains a recommendation tab in order to suggest content (assets) of interest to users. The aforementioned corpus was selected as it illustrates a case very similar to the interaction with assets in the second screen. Some basic characteristics of the evaluation dataset are: 24 users, 5,921 documents, 10,010 relevance scores, 9,238 scrolls and 8,736 clicks.

Using this corpus, the Weka toolkit27 was employed to train various base-line machine learning algorithms and compare their accuracy with the SAM relevance predictions. More specifically, the best performing baseline algorithms from a representative set of families (linear classifiers, decision tress and neural networks) were selected and configured to the default parameters provided by Weka: Multi-Layer Perceptron (Rosenblatt, 1958), Linear Regression using the Akaike information criterion (Akaike, 1974), Random Forest (Breiman, 2001), Neural Network with 100 neurons (Haykin, 1994), linear regression using epsilon-SVR and nu-SVR (Chang and Lin, 2001), decision trees using m5rules (Holmes et al, 1999) and REPTree (Snousy et al, 2011), and DecisionTable (Kohavi, 1995). The splitting ratios between the training and testing data was set to 70% and 30% respectively. The aggregated results of these experiments are shown in Table 5.

As shown in this table, the SAM algorithm effectiveness (0.2307 MAE and 0.3856 RMSE) is very close to the best classification accuracy, achieved by the DecisionTable classifier (0.2198 MAE and 0.3246 RMSE). The biggest advantage that the SAM approach presents with respect to the other algorithms is that it does not require a training dataset, as mentioned in Section 4.2.

Besides the precision of the recommendation algorithm, another key feature of this type of real-time multimedia applications is the response time, since delays may impoverish user experience. Based on the graph analysis described in Section 4.2.2, the SAM algorithm localises its heuristic calculations only in the neighbourhood of the user and the asset in the Neo4j graph, implying extremely low resource consumption and optimising the recommendation efficiency. To evaluate this point, it is not possible to make a direct comparison of the SAM approach with machine learning classifiers that require a training step. Thus, in order to evaluate the efficiency of our technique, two baseline Collaborative Filtering recommendation algorithms we employed, k-NN clustering (kNN) and Pearson Correlation (CF), using JMeter28 to generate 500 HTTP calls comparing their performance as a web service.

As shown in Figure 6, the SAM web service was by far more efficient than the two aforementioned approaches, with an average response time of 147 ms, verifying the optimised performance expectations.

Taking into account the results shown above, the conclusion is that the SAM approach to context analysis provides a good trade-off between effective-ness and efficiency.

5.1.3 Dynamic Communities

In order to validate whether the BigCLAM algorithm should be a well-suited candidate for use in the SAM platform implementation, an evaluation of the graph clustering performance was carried out. Given the absence of a suitable dataset for the specific target application in SAM, the evaluation reported in Paliouras et al (2015) was repeated with a number of additional candidate algorithms in order to compare results with those reported by the aforementioned authors.

Three datasets generated using social media data by third-party researchers were used in order to carry out the evaluation, allowing for comparison with related work that also employed these data for evaluation purposes. The data was generated by McAuley and Leskovec (2014) with data from Twitter, Face-book and Google+ social networks.

The procedure reported in McAuley and Leskovec (2014) was reproduced as described in their work for three candidate algorithms representing the state-of-the-art in applicable graph clustering techniques: BigCLAM, METIS and MLR-MCL. METIS is a fast graph partitioning algorithm (Karypis and Kumar, 1998) and MLR-MCL is a multi-level flow-based clustering algorithm (Satuluri and Parthasarathy, 2009).
As shown in Figure 6, the SAM web service was by far more efficient than the two aforementioned approaches, with an average response time of 147 ms, verifying the optimised performance expectations.

Taking into account the results shown above, the conclusion is that the SAM approach to context analysis provides a good trade-off between effective-ness and efficiency.

5.1.3 Dynamic Communities

In order to validate whether the BigCLAM algorithm should be a well-suited candidate for use in the SAM platform implementation, an evaluation of the graph clustering performance was carried out. Given the absence of a suitable dataset for the specific target application in SAM, the evaluation reported in Paliouras et al (2015) was repeated with a number of additional candidate algorithms in order to compare results with those reported by the aforementioned authors.

Three datasets generated using social media data by third-party researchers were used in order to carry out the evaluation, allowing for comparison with related work that also employed these data for evaluation purposes. The data was generated by McAuley and Leskovec (2014) with data from Twitter, Face-book and Google+ social networks.

The procedure reported in McAuley and Leskovec (2014) was reproduced as described in their work for three candidate algorithms representing the state-of-the-art in applicable graph clustering techniques: BigCLAM, METIS and MLR-MCL. METIS is a fast graph partitioning algorithm (Karypis and Kumar, 1998) and MLR-MCL is a multi-level flow-based clustering algorithm (Satuluri and Parthasarathy, 2009).

Results for the Graclus and Louvain algorithms that were obtained in McAuley and Leskovec (2014) using identical methods and datasets are also re-ported on to allow for comparison with those state-of-the-art algorithms. This study also provides results for baseline algorithms using the same datasets and evaluation procedures.

Table 6 provides the results for the three datasets and the F1 scores for the different algorithms. Refer to McAuley and Leskovec (2014) for a discussion of the precision and recall achieved for the different datasets. The table also reports the number of vertices (|V|) and edges (|E|) in the datasets. As can be seen, BigCLAM obtained the best performance in Twitter (0.2761) and Google+ (0.1799) dataset, performing the second best in Facebook (0.3505), only surpassed by Louvian (0.3868).

Classifier	Mean Absolute Error	Root Mean Square Error
MLP	0.3311	0.4456
Linear Regression	0.2423	0.3398
Random Forest	0.241	0.3588
Neural Network 100 neurons	0.2663	0.3472
epsilon-SVR	0.2285	0.3339
nu-SVR	0.2441	0.3373
m5rules	0.2241	0.3261
REPTree	0.2273	0.3355
DecisionTable	0.2198	0.3246
SAM	0.2307	0.3886

Table 5: Performance of the SAM algorithm compared to different machine learning baselines.

Fig. 6: Comparison of the efficiency of SAM recommendation web service with k-NN clustering (kNN) and Pearson Correlation (CF).

5.2 Prototype System User Evaluation

An extensive end user evaluation has been carried out as part of the SAM project. The purpose of the evaluation was to determine the acceptance and the enjoyability of the end user experience that is provided through second screen experiences created with and delivered through the SAM platform. The evaluation included the system functionalities described in this article as part of the overall system setup.

5.2.1 Evaluation Setup

The end user evaluation was carried out within the premises of two schools affiliated with the SAM project. Within each school, pupils aged 13 to 17 participated in the user evaluations. The selected age group represents a digitally native user population that can be assumed to be accustomed to using smart phone devices as part of their regular daily activities, and it represents a tar-get audience demographic of high interest to broadcasters and for the SAM project. It should be noted that findings may not be generalisable to other demographic groups and that the participant group is not a representative sample of the population. The evaluation setup was created in a classroom environment, where a setup considered typical of television classroom use was employed. A single television (first screen) was used for the evaluation, and participants were asked to install the SAM application on their mobile phones (or use a web-based alternative if their device was not compatible with the Android application created in the project).

Dataset	$\|V\|$ $\|E\|$	BigCLAM	METIS	MLR-MCL	Graclus	Louvain
Twitter	81306 1342303	**0.2761**	0.1625	0.1146	0.2147	0.1086
Facebook	4039 88234	0.3505	0.2356	0.2701	0.3026	**0.3868**
Google+	107614 12238285	**0.1799**	0.1663	0.0100	0.1789	0.0549

Table 6: Graph clustering performance $F1$ scores.

Participants were asked to complete four evaluation rounds. Each evaluation round followed an identical procedure:

- Brief verbal introduction to the evaluation round
- Written introduction to the evaluation round
- Presentation of a short form video with the respective second screen content
- Free interaction after completion of playback
- Completion of a paper questionnaire

In each of the four evaluation rounds, different properties of the SAM application and the created second screen experiences were evaluated: (1) a basic augmented content version, (2) using basic SAM community features, using advanced SAM community features and (4) all functionalities of the overall system. Each of the evaluation rounds took between 20 and 30 minutes to complete.

For each of the rounds, a questionnaire based on the Technology Acceptance Model (TAM) questionnaire was developed (Davis, 1986). In the questionnaire, participants were asked to evaluate a number of statements on a 5-point Likert scale. Furthermore, they were encouraged to provide written responses to open questions at the end of each of the evaluation questionnaires. Two evaluations using this approach were carried out as part of the project, one for piloting the evaluation setup and for gathering formative feedback at midterm, and one for gathering summative evaluation feedback for the overall project at the end. In the remainder of this subsection, we focus on the results of the summative user evaluation.

5.2.2 Participants

A total of 90 participants completed trial questionnaires as part of the summative SAM end user evaluation. The vast majority of participants were in the 15-16-year age bracket (81.1%), 15.6% were in the 13-14-year age bracket and 3.3% were in the 17-18-year age bracket. Gender balance was quite even with 47.8% female and 48.9% male participants (3.3% of participants preferred not to answer this question).

5.2.3 Results and Discussion

For brevity, only a small subset of individual responses can be presented in this article. Overall, participants in the end user evaluations responded positively to the overall concept of the SAM platform. Since augmenting first screen content with additional second screen content is an important goal of the SAM project, it is also important to determine whether the content is considered useful. Figure 7 shows a subset of responses concerning this topic. A majority of participants found that the provided SAM application makes it easy to find out more about topics in a video and that using the SAM application improves viewing a video on the TV.

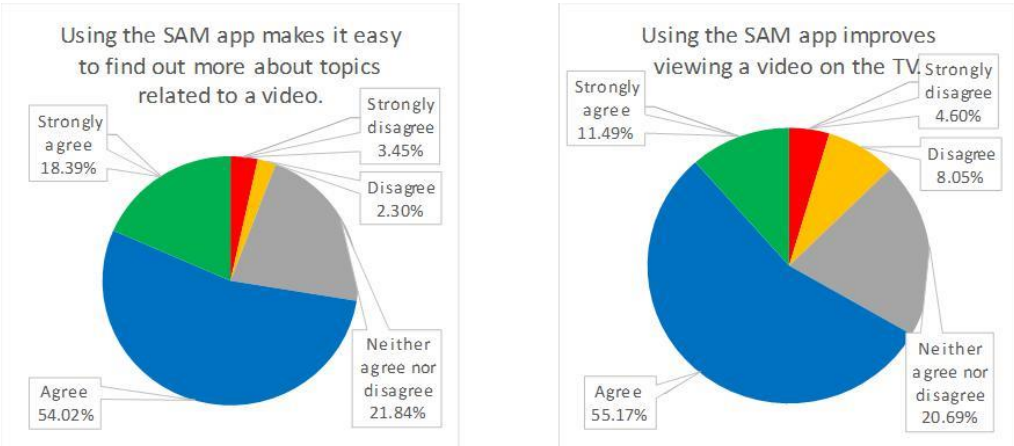


Fig. 7: Participant responses concerning content augmentation.

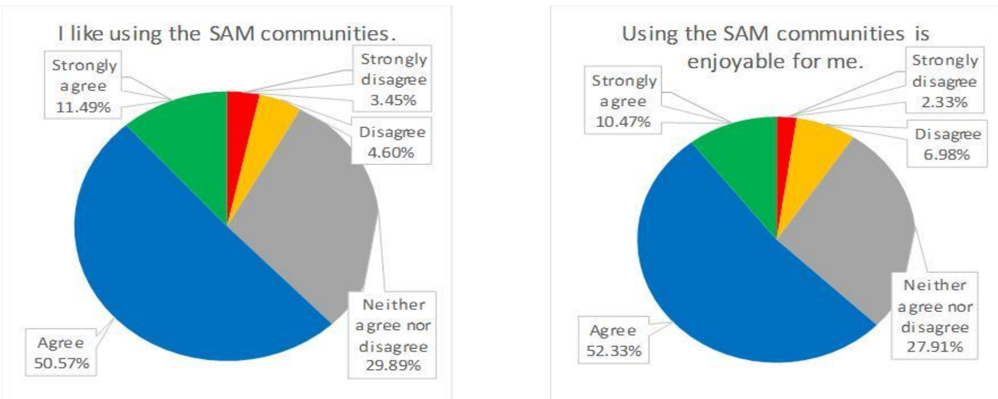


Fig. 8: Participant responses concerning dynamic communities.

Part of the evaluation focused on dynamic content delivery and dynamic community features. The majority of participants similarly responded that using the SAM dynamic communities was enjoyable or very enjoyable (Figure 8). It should be noted here that a specially configured version of the dynamic community functionality was used in user trials for evaluation purposes; a more appropriate longitudinal evaluation was not carried out due to resource and copyright restrictions concerning providing large amounts of video content to participants over a longer period of time.

Generally, participants also responded that they would like to use the SAM application in school environments (65.91% agree or strongly agree) as well as at home (70.46% agree or strongly agree) to complement classroom materials. Given that the SAM platform and related application are still prototype systems at the end of the project, this could be interpreted as pointing to-wards significant interest in such a system from the end user population that participated in these trials.

6 Conclusion

This article has introduced the SAM platform as a system for easily and quickly creating second screen experiences around video content. As a practically oriented project, SAM has focused on real-world implementation and evaluation activities, which have been presented in this and related publications concerning the SAM platform. While developed as a research prototype during the project, commercial use of the overall platform has been a major concern throughout the project, resulting in a practically usable and readily commercially exploitable system.

The article has described three key processes of the workflow that are involved with enabling an enhanced user experience through the use of semantic analysis, user and context modelling and dynamic community discovery. This workflow illustrates how a real-world pipeline for implementing such functionalities can be provided, and the user evaluations carried out indicate user interest in such functionalities as part of multiscreen viewing experiences.

Future research in this area should focus on the creation of publicly avail-able datasets that incorporate multiple types of data sources in order to enable investigating such complex workflows better in experimental environments. To further investigate the appeal and potential acceptance of systems such as the SAM platform, longitudinal studies and evaluations with more heterogeneous first and second screen video content could be carried out to identify the most promising areas for further development and research.

Acknowledgements: This work has been partially funded by the European Commission under the Seventh (FP7 2007-2013) Framework Programme for Research and Technological Development through the SAM (FP7-611312) project, by the Spanish Government under project REDES (TIN2015-65136-C2-2-R) and by the Generalitat Valenciana under project (PROMETEU/2018/089).

End Notes

1. http://wiki.socialisingaroundmedia.com.
2. https://www.contentwise.tv/.
3. https://www.couchfunk.de/.
4. https://www.horizon.tv/.
5. https://www.leankr.com/.
6. https://www.monterosa.co/.
7. https://www.tivine.com/.
8. 8 http://www.cyc.com/.
9. 9 https://developers.google.com/freebase/.
10. 10 https://wordnet.princeton.edu/.
11. https://www.weibo.com/.
12. The ontology definition and additional information, including a use case example, is available at https://github.com/perma-id/w3id.org/tree/master/media/dma.
13. http://labs.europeana.eu/api/linked-open-data-data-structure.
14. https://opennlp.apache.org/.
15. http://wiki.dbpedia.org/.
16. https://github.com/dbpedia/lookup/
17. https://lucene.apache.org/.
18. In addition to allowing adjacent sequences of words (n-grams), skip-grams allow tokens to be skipped. https://neo4j.com/.
19. https://www.w3.org/TR/2016/CR-activitystreams-vocabulary-20160906/.
20. https://www.drools.org/.
21. http://www.imdb.com/.
22. http://alt.qcri.org/semeval2015/task13/index.php
23. https://www.cs.york.ac.uk/semeval-2013/task2/.
24. http://www.metacritic.com/.
25. http://alt.qcri.org/semeval2014/task4/.
26. http://www.cs.waikato.ac.nz/ml/weka/.
27. http://jmeter.apache.org/.

References

(2012) Notube: bringing web and tv closer together. URL http://www.notube.tv/

(2013) Limosine project integrates the studies of leading researchers over di-verse topics with a view to enable new kinds of language-based technology search. URL http://limosine-project.eu/

Abowd GD, Dey AK, Brown PJ, Davies N, Smith M, Steggles P (1999) Towards a better understanding of context and context-awareness. In: Gellersen HW (ed) Handheld and Ubiquitous Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 304–307

Aisopos F, Valsamis A, Psychas A, Menychtas A, Varvarigou T (2016) Efficient context management and personalized user recommendations in a smart social tv environment. In: GECON - Conference on the Economics of Grids, Clouds, Systems, and Services (GECON)

Akaike H (1974) A new look at the statistical model identification. IEEE Transactions on Automatic Control 19(6):716–723, DOI 10.1109/TAC.1974. 1100705

Aroyo L, Nixon L, Miller L (2011) Notube: the television experience enhanced by online social and semantic data. In: Consumer Electronics-Berlin (ICCE-Berlin), 2011 IEEE International Conference on, IEEE, pp 269–273

Batra S, Tyagi C (2012) Comparative analysis of relational and graph databases. International Journal of Soft Computing and Engineering (IJSCE) 2(2):509–512

Breiman L (2001) Random forests. Machine Learning 45(1):5–32
Cambria E, White B (2014) Jumping nlp curves: A review of natural language processing research. Comp Intell Mag 9(2):48–57

Cesar P, Bulterman DC, Jansen A (2008) Usages of the secondary screen in an interactive television environment: Control, enrich, share, and transfer tele-vision content. In: European Conference on Interactive Television, Springer, pp 168–177

Chang CC, Lin CJ (2001) Libsvm - a library for support vector machines. URL http://www.csie.ntu.edu.tw/\~cjlin/libsvm/

Chang CC, Lin CJ (2011) Libsvm: a library for support vector machines. ACM transactions on intelligent systems and technology (TIST) 2(3):27

Chorianopoulos K (2008) Personalized and mobile digital tv applications. Mul-timedia Tools and Applications 36(1-2):1–10

Courtois C, D'heer E (2012) Second screen applications and tablet users: con-stellation, awareness, experience, and interest. In: Proceedings of the 10th European conference on Interactive tv and video, ACM, pp 153–156

Damerau FJ (1964) A technique for computer detection and correction of spelling errors. Communications of the ACM 7(3):171–176

Davis F (1986) A technology acceptance model for empirically testing new end-user information systems: Theory and results. Thesis, URL http://

dspace.mit.edu/handle/1721.1/15192

Fernández J, Gutiérrez Y, Tomás D, Gómez JM, Martínez-Barco P (2015) Evaluating a sentiment analysis approach from a business point of view. In: TASS at SEPLN, pp 93–98

Fleiss JL (1971) Measuring nominal scale agreement among many raters. Psy-chological bulletin 76(5):378

Forgy CL (1982) Rete: A fast algorithm for the many pattern/many object pattern match problem. Artificial intelligence 19(1):17–37

Geerts D, Leenheer R, De Grooff D, Negenman J, Heijstraten S (2014) In front of and behind the second screen: viewer and producer perspectives on a companion app. In: Proceedings of the 2014 ACM international conference on Interactive experiences for TV and online video, ACM, pp 95–102

Giglietto F, Selva D (2014) Second screen and participation: A content analysis on a full season dataset of tweets. Journal of Communication 64(2):260–277 Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proceedings of the National Academy of Sciences 99(12):7821–7826

Greene D, Doyle D, Cunningham P (2010) Tracking the evolution of communi-ties in dynamic social networks. In: Advances in social networks analysis and mining (ASONAM), 2010 international conference on, IEEE, pp 176–183

Guthrie D, Allison B, Liu W, Guthrie L, Wilks Y (2006) A closer look at skip-gram modelling. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06), European Language Resources Association (ELRA)

Haykin S (1994) Neural networks: a comprehensive foundation. Prentice Hall PTR

Holmes G, Hall M, Frank E (1999) Generating rule sets from model trees. In: Twelfth Australian Joint Conference on Artificial Intelligence, Springer, pp 1–12

Holmes ME, Josephson S, Carney RE (2012) Visual attention to television pro-grams with a second-screen application. In: Proceedings of the symposium on eye tracking research and applications, ACM, pp 397–400

Hu H, Wen Y, Luan H, Chua TS, Li X (2014) Toward multiscreen social tv with geolocation-aware social sense. IEEE MultiMedia 21(3):10–19

Hu H, Wen Y, Gao Y, Chua TS, Li X (2015) Toward an sdn-enabled big data platform for social tv analytics. IEEE Network 29(5):43–49

Jaiswal G, Agrawal AP (2013) Comparative analysis of relational and graph databases. IOSR Journal of Engineering (IOSRJEN)

Joachims T (1998) Text categorization with support vector machines: Learning with many relevant features. In: European conference on machine learning, Springer, pp 137–142

Kaplan AM, Haenlein M (2010) Users of the world, unite! the challenges and opportunities of social media. Business horizons 53(1):59–68

Karypis G, Kumar V (1998) Multilevel k-way partitioning scheme for irregular graphs. Journal of Parallel and Distributed Computing 48(1):96 – 129

Kohavi R (1995) The power of decision tables. In: 8th European Conference on Machine Learning, Springer, pp 174–189

Landis JR, Koch GG (1977) The measurement of observer agreement for cat-egorical data. biometrics pp 159–174

Leskovec J, Rajaraman A, Ullman J (2014) Mining of Massive Datasets. 2nd Edition. Cambridge University Press

Levenshtein VI (1966) Binary codes capable of correcting deletions, insertions and reversals. Soviet Physics Doklady 10(8):707–710

Martínez V, Berzal F, Cubero JC (2015) The noesis open source framework for network data mining. In: 2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), vol 01, pp 316–321

McAuley J, Leskovec J (2014) Discovering social circles in ego networks. ACM Transaction on Knowledge Discovery from Data 8(1)

Mitchell K, Jones A, Ishmael J, Race NJ (2010) Social tv: toward content navigation using social awareness. In: Proceedings of the 8th international interactive conference on Interactive TV&Video, ACM, pp 283–292

Nadeau D, Sekine S (2007) A survey of named entity recognition and classifi-cation. Linguisticae Investigationes 30(1):3–26

Ng HT, Zelle J (1997) Corpus-based approaches to semantic interpretation in nlp. AI magazine 18(4):45

Odijk D, Meij E, De Rijke M (2013) Feeding the second screen: Semantic linking based on subtitles. In: Proceedings of the 10th Conference on Open Research Areas in Information Retrieval, Le Centre De Hautes Etudes In-ternationales d'Informatique Documentaire, pp 9–16

Paliouras G, Papadopoulos S, Vogiatzis D, Kompatsiaris Y (2015) User Com-munity Discovery. Springer
Pang B, Lee L (2008) Opinion mining and sentiment analysis. Found Trends Inf Retr 2(1-2):1–135

Papadopoulos S, Kompatsiaris Y, Vakali A, Spyridonos P (2012) Community detection in social media. Data Mining and Knowledge Discovery 24(3):515– 554

Pynta P, Seixas SA, Nield GE, Hier J, Millward E, Silberstein RB (2014) The power of social television: Can social media build viewer engagement? Journal of Advertising Research 54(1):71–80

Rao D, McNamee P, Dredze M (2013) Entity Linking: Finding Extracted En-tities in a Knowledge Base, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 93–115

Rosenblatt F (1958) The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review 65(6):386–408

Satuluri V, Parthasarathy S (2009) Scalable graph clustering using stochastic flows: Applications to community discovery. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, KDD '09, pp 737–746, DOI 10.1145/ 1557019.1557101, URL http://doi.acm.org/10.1145/1557019.1557101

Snousy MBA, El-Deeb HM, Badran K, Khlil IAA (2011) Suite of decision tree-based classification algorithms on cancer gene expression data. Egyptian Informatics Journal 12(2):73–82, DOI https://doi.org/10.1016/j.eij.2011.04.003

Tang L, Liu H (2010) Community detection and mining in social media. Syn-thesis Lectures on Data Mining and Knowledge Discovery 2(1):1–137

Tantipathananandh C, Berger-Wolf TY (2011) Finding communities in dy-namic social networks. In: IEEE 11th International Conference on Data Mining, pp 1236–1241, DOI 10.1109/ICDM.2011.67

Theresa Wilson PNSRVS Zornitsa Kozareva, Ritter A (2013) Semeval-2013 task 2: Sentiment analysis in twitter. In Proceedings of the International Workshop on Semantic Evaluation, SemEval âĂŹ13

Tomás D, Gutiérrez Y, Agulló F (2015) Entity linking in media content and user comments: Connecting data to wikipedia and other knowledge bases. In: eChallenges e-2015 Conference, pp 1–10

Vanattenhoven J, Geerts D (2017) Social experiences within the home us-ing second screen tv applications. Multimedia Tools and Applications 76(4):5661–5689

Villena-Román J, García-Morera J, Cumbreras MÁG, Martínez-Cámara E, Martín-Valdivia MT, López LAU (2015) Overview of tass 2015. In: TASS at SEPLN, pp 13–21

Yang J, Leskovec J (2013) Overlapping community detection at scale: a non-negative matrix factorization approach. In: ACM International Conference of Web Search and Data Mining

Zhang Y (2005) Bayesian graphical models for adaptive information fil-tering. Thesis, URL https://users.soe.ucsc.edu/~yiz/papers/data/ YOWStudy

Zhao S, Zhong L, Wickramasuriya J, Vasudevan V (2011) Analyzing twitter for social tv: Sentiment extraction for sports. In: Proceedings of the 2nd International Workshop on Future of Television, vol 2, pp 11–18

****