TIPS: Search and Analytics for Social Science Research

Emanuele Di Buccio[1,2,*], Alberto Cammozzo[3], Federico Neresini[3] and Alberto Zanatta[3]

[1]Department of Information Engineering, University of Padova, Via G. Gradenigo 6/b, 35131, Padova, Italy
[2]Department of Statistical Sciences, University of Padova, Via C. Battisti, 241, 35121, Padova, Italy
[3]Department of Philosophy, Sociology, Education and Applied Psychology, University of Padova, Via M. Cesarotti 10/12, 35123 Padova, Italy

Abstract

The vast amount of digital data available online, which include digitized traditional media, offers new opportunities for social science researchers. This is, for instance, the case of social processes where the temporal dimension is crucial and longitudinal data is necessary. This paper presents a system called TIPS, designed in order to support social science researchers in their investigations. We will present the main modules of the system and how it can support diverse research tasks on longitudinal data such as archives of digitized newspapers.

Keywords

News Search and Analytics, Information Retrieval, Expert Users

1. Introduction

The vast amount of digital data available nowadays provides new opportunities for many disciplines, e.g. Humanities and Social Sciences. Social Science concerns with “any branch of academic study or science that deals with human behavior in its social and cultural aspects”.[1] Social Sciences include several disciplines, e.g. sociology, psychology or political science. A first opportunity for disciplines, such as sociology, is the digitization of traditional media. This is, for instance, the case of newspapers: the analysis of newspapers is traditionally used to carry out research investigations on the public perception of issues along with other methods, such as research surveys. Another opportunity is related to the investigation of social processes where the temporal dimension is crucial and longitudinal data is necessary.

The main contribution of this paper is the description of a system called *Technoscientific Issues in the Public Sphere (TIPS)*. The TIPS system is the result of an interdisciplinary research project which involves researchers in diverse areas, including computer science and engineering, sociology, statistics, psychology and linguistics. The objective of the project is the analysis of the presence of tecnoscience[2] in the mass media, which constitutes a relevant part of the

CIRCLE 2022: Joint Conference of the Information Retrieval Communities in Europe, July 4-7, 2022, Samatan, Gers, France

*Corresponding author.
emanuele.dibuccio@unipd.it (E. Di Buccio); alberto.cammozzo@unipd.it (A. Cammozzo); federico.neresini@unipd.it (F. Neresini); albertozanatta95@hotmail.it (A. Zanatta)
0000-0002-6506-617X (E. Di Buccio); 0000-0003-1551-0022 (A. Cammozzo); 0000-0003-3918-2588 (F. Neresini)

[1] From: https://www.britannica.com/topic/social-science

[2] The term “technoscience” refers to “science and technologies”

public sphere. Support the analysis through automated techniques is crucial because it allows expert users – researchers, journalists, policy makers – to monitor the public opinion relying on very large amount of data. Among the diverse challenges, one is handling heterogeneous data, i.e., diverse source types (blogs, research papers, newspapers, tweet streams). Even if the TIPS system was designed to achieve this goal, in this paper we are going to focus on how support expert users in implementing their methodologies to investigate research activities.

Social Science researchers are expert users that often need many interactions with the information space and at different levels. Multiple searches involving both full-text and metadata may be needed to determine the corpus or the corpora that will be adopted for their research. Indicators based on social science theories might help validating a research hypothesis and a system able to offer ways to build a methodology with "high level" interaction may be helpful also to researchers with limited knowledge in computer science topics or to speedup the methodology implementation.

A first version of the system was documented in [1]. This paper extends that work focusing on how the system can support expert users like sociologists, the extended architecture, and how the methodology proposed and exploited in previous works [2] has been integrated into the system to provide information access beyond the traditional query-response paradigm. The system is available at: https://www.tipsproject.eu/tips

2. Background and System Requirements

TIPS was explicitly designed in collaboration with the members of the Pa.S.T.I.S. Research Unit of the University of Padova in order to support their research methodologies. One of the unit's research topics is Science and Technology Studies (STS), particularly understanding the role of Technoscience in the Media and how Technoscience is represented. Indeed, the representation of technoscientific issues in the Media may greatly differ from that of the specialists; unveiling and monitoring the evolution of that representation could be useful, e.g., to identify critical aspects or communicate more effectively innovations and issues to the public.

Research at the intersection of Social Science and Computer Science is not new [3]; Computational Social Science [4], Social Informatics and Digital Social Research [5] are examples of these interdisciplinary research lines. Works relevant to that reported in this paper are systems explicitly designed to support Social Science researchers, e.g., NOAM [6, 7], and to monitor the media such as the European Media Monitor [8]. A number of libraries and tools have been made available and are used by social scientists to carry out their investigations; [5] provides an overview both of methodologies and tools. Even if TIPS shares the objectives and functionalities with many of these systems, we explicitly designed the system to support a complete workflow, from data collection and feature extraction to full-text, metadata and theme-based access; moreover, TIPS is equipped with a number of indicators explicitly designed in collaboration with a team of Social Science researchers.

In order to design the system, we relied on the experience gathered during our prior involvement in interdisciplinary research activities, e.g., [2], we carried out a number of seminars with Social Science researchers, and we exploited feedback from preliminary versions of the system – at the time essentially a full-text and metadata-based search engine – to understand the typical workflow of a user when carrying out activities to investigate a research hypothesis. The main activities performed by the researchers are listed below:

1. defining the specific thematic domain and the corpus of interest for the investigation of the research hypothesis;
2. access both by full-text search and metadata-based search the corpus of interest to explore the thematic space;
3. extract the main (sub)themes in the corpus of interest, access and search the corpus by themes representation and analyze their presence, usually, over time;
4. compute, visualize, and analyze trends of *indicators* which are meant to provide a measure of the degree to which certain properties characterize a set of documents;
5. extract word, sentence, document and document set properties to perform fine-grained analysis and uncover possible relationships among these properties in order to verify a research hypothesis or to formulate a new one;
6. keep track of all the choices made for the diverse research methodology steps in order to foster reproducibility.

The TIPS system aims at automatizing some of the above activities or to support users in performing them through diverse forms of user-system interaction.

3. The TIPS System

3.1. Document Collectors and Repositories

The first versions of the system were focused on the collection, enrichment and content-based search of documents from diverse types of sources, e.g., blogs, online newspapers or tweet streams. Many of the research activities in the TIPS project are carried out on newspapers since they still constitute a large portion of the Media Sphere and allow longitudinal research studies which can involve many decades, e.g., from the 1980's, thanks to digitization of articles.

In TIPS, a document collector is responsible for gathering documents from multiple sources, e.g., RSS of online newspapers, where sources are homogeneous in terms of type and language; for instance, there is a collector for Italian Newspapers, one for Italian Blogs, and one for English Newspapers. A database is used for each collector; currently we are relying on MongoDB.[3] Each document undergoes through a number of steps:

(a) **Harvesting**: articles are collected through RSS feeds; when feeds are not available or not working, HTML website traversal is used, focusing only on the relevant sections.
(b) **Scraping**: the relevant parts of the page are extracted mainly through the library `Newspaper3k`;[4] some additional handcrafted rules resulted from the analysis of some samples are used to remove additional strings, e.g., those peculiar of a specific newspaper.
(c) **De-duplication**: The same article can be published in different news feeds, or updated versions can be published at different times. In the former case, we store only one article but keep track of all the feeds where it was published, especially if it was published on

[3] https://www.mongodb.com/

[4] https://newspaper.readthedocs.io/en/latest/

the homepage. In the latter case, we keep only the newest version of the articles, but we keep track of all publication timestamps. The identification of duplicates relies on MD5 hashing of content and metadata, more specifically, the RSS feed, the URL, and the date it was published. If an article with the same content and metadata hash is already present in the collector database, the article is dropped as a duplicate. If an article is present with the same content hash but a different metadata hash, the original article metadata is enriched.

(d) **Near-duplicates detection**: Previous research activities using early versions of the system revealed a significant number of articles that overlap in most of the content, i.e., near-duplicates. In order to detect them, the current version of the system relies on Locality Sensitive Hashing for Minhash Signatures [9, 10], specifically the implementation made available in the `datasketch` library.[5] Each document is represented using k-shingle with $k = 4$ and considering shingles on tokens, not on characters. We set to the value 0.4 the similarity threshold to determine if two articles should be considered near-duplicates. The threshold was determined considering a week at random for each year between 2010 and 2018, then considering all the articles published in that week and in the same newspaper, and manually inspecting the articles; the most effective threshold for detecting near-duplicates was selected. The procedure for near-duplicates detection is now integrated in TIPS, processes all the incoming articles, and relies on Redis[6] to store and search MinHash signatures; near-duplicates are searched only among articles published in the same newspaper.

(e) **Named Entities extraction**: For many research activities a named entity search is required to identify names of places, people, organizations, and several computer libraries are available. On one hand it is desirable to have a very specific and accurate recognition of entities, on the other, the correct attribution of a name to an entity is often difficult and inaccurate despite being resource consuming: "Galileo" may denote the famous scientist (person), a company or a school with the same name, the name of a place (e.g. "Galileo Galilei square"), and so on, heavily depending on context. Moreover, newspaper articles may often contain new, country-specific entities and misspelled names. We have decided for a layered, concentric approach to named entity recognition. The first step being the recognition of "Named Entity Candidates" (NEC) at download-time, with a simple and swift regular expression pinning capitalized words and uppercase acronyms. A second layer uses the spaCy [7] library, which is trained with many language-specific pre-built models. We store both results, having found that combining the two first layers gives a more accurate results than using just a single approach. While these two first layers are run on the whole corpus, further analysis with more resource-intensive approaches is restricted to sub-corpora according to research needs, providing a third layer. So far we used Stanford NER [11] and several context-specific APIs focused on specific archives of entities (e.g. scientists names are investigated with Elsevier SCOPUS and Orcid APIs). A layered approach allows researchers to spot entities at the corpus level and conveniently select articles where context-specific entities are recognized when inspecting the articles.

[5] http://ekzhu.com/datasketch/index.html

[6] https://redis.io

[7] https://spacy.io

(f) **Extraction of other document properties and indicators**: Besides named entities, a number of document properties are extracted, e.g., document length, number of characters, non-text prevalence. When considering Italian document collectors, distinct nouns and adjectives in the article, and article readability are obtained through *The Italian NLP Tool (Tint)* [12]. Moreover, for each document a number of *indicators* based on controlled vocabularies and the frequency of vocabulary terms. Each indicator is meant to provide a measure of the degree to which certain properties characterize a set of documents. For instance, the *risk indicator* [13] provides a measure of the extent a document or a set of documents may evoke risk, conflict, worry or controversy in the reader. The procedure for computing the risk indicator is described in [14]. Other indicators have been proposed and are available in TIPS, e.g., molecularisation and individualization [2].

(g) **Classification**: Researchers can be involved in multiple research projects. In TIPS each project has at least one classifier to identify documents pertinent to the thematic domain. For instance, the project on “Food Risk Monitoring” has its own classifier; the project on Technoscience has multiple classifiers relying on diverse approaches. One is a Knowledge Engineering approach [1] used since from the early versions of the system. In the new version of the system the default classifier is based on Supervised Machine Learning techniques, more specifically on Stacking [15] of Regularized Logistic Regression using Dual Coordinate Descent [16] and Multinomial Naive Bayes; the implementation relies on the JSAT Library [17]. The model was selected using both Holdout (random split) and 5-fold Cross Validation on a dataset manually labeled by the sociologists and described in [1]; we examined the effectiveness looking both at value and variance of AUC, F1, Precision and Recall. In addition to the labeled set described in [1], we built a new labeled set used as an additional separate test set.

Steps (a)-(f) have been implemented in a library called *hactar* which has been published under the AGPL open source license.[8] Part of step (f) and step (g) have been implemented in a Java module called *tips-data*. Besides classification, indicator computation, and metadata extraction and enrichment, the module is responsible for updating the document repositories with all the “indexable” articles. An article is indexable if it has title, content, date, and is not marked as duplicate or near-duplicate. All the indexable documents are indexed via elasticsearch,[9] a Distributed Search and Analytics Engine. A *document repository* in TIPS corresponds to an elasticsearch index. The index update is performed on a daily basis.

3.2. Repository Search and Exploration

Research activities in the TIPS project require different forms of interaction with document repositories: search is one of them. There are two ways to search a repository. The first way is to use REST Search APIs which are built on top of elasticsearch APIs. Indeed, we designed and implemented the system in a modular way and exposed all the functionalities via REST APIs through the “TIPS Web Server”. REST Search APIs allow users with programming skills to build scripts for interacting with the repositories and retrieving all the documents relevant to her

[8] https://gitlab.com/mmzz/hactar

[9] https://www.elastic.co/elasticsearch/

information need. The second way is through the use a Web User Interface (UI) built on top of the REST APIs.

We equipped the last version of the system with a module to monitor diverse interactions between the users and the system; logs are stored in dedicated index and are anonymized. We analyzed about 3000 log entries gathered since from March 2019 and we observed that most of the queries involved compound queries, where both full-text and metadata-based search are used. The metadata usually adopted in the query include a (filter for a) specific newspaper or a set of newspapers, or a particular set of documents – e.g. only on the documents predicted in the "technoscience" category according to the default classifier. The Search UI allows also the ranking criterion to be specified: possible criteria are by date (most recent first), by classifier score (highest score first), or by query score (BM25 [18] score computed using query terms). The most adopted ranking criterion is that based on the classifier score, followed by date and query score. Most of the interactions aim at determining the corpus later adopted for detailed analysis: queries can be very complex, relying on Boolean or other operators – e.g., prefix match – to retrieve as much documents topically relevant to an issue in the thematic domain as possible — for instance nuclear-related articles in tecnoscience domain. A common task that can be evinced from the query logs is identify useful terms to expand the original, more succinct query. For this reason, we equipped the system with functionalities to get the top terms, named entities, or nouns and adjectives in a document set specified by the user through search parameters; term ranking can be based on document frequency or other measures, e.g., Chi Square.

Once the query to determine the corpus has been identified, research activities usually require a "compact" representation of a large number of documents in order to have an idea of the thematic structure. A well known class of algorithms to achieve this objective is Topic Modeling [19, 20]. For these analyses, in past works [2] we relied on the Mallet library,[10] in particular on LDA [21] using asymmetric Dirichlet Prior [22]. Also in this case we observed that multiple interactions were needed, for example, in order to "interpret" the topics, build different topic models on refined version of the corpus, or compare the thematic structure of an issue-specific or a general corpus. For this reason we designed and implemented novel system components to perform theme-based analysis, visualize the result and interact with them, implementing some of the visualizations and forms of interaction proposed in previous works [23, 24].

The new components allow the user to submit a query and extract topics from all the documents in the repository satisfying the query; in other the words, the query serves as filter to specify the corpus. Along with the query, the user can specify some of the LDA settings available in Mallet: the number of topics, the number of top words, the number of sampling iterations. Moreover, she can specify a threshold to determine the minimum probability a topic should have in a document to be considered "representative" for that document; note that the threshold is not used in the inference process, but only to access documents where the topic is present, after the inference procedure. The query is then submitted to the TIPS Web Server that forwards the request to a dedicated Topic Modeling and Word Embedding (TM&WE) Web Server that relies on Mallet. The TM&WE Web Server is responsible for:

- gathering all the documents satisfying the query from the document repository;

[10] https://github.com/mimno/Mallet

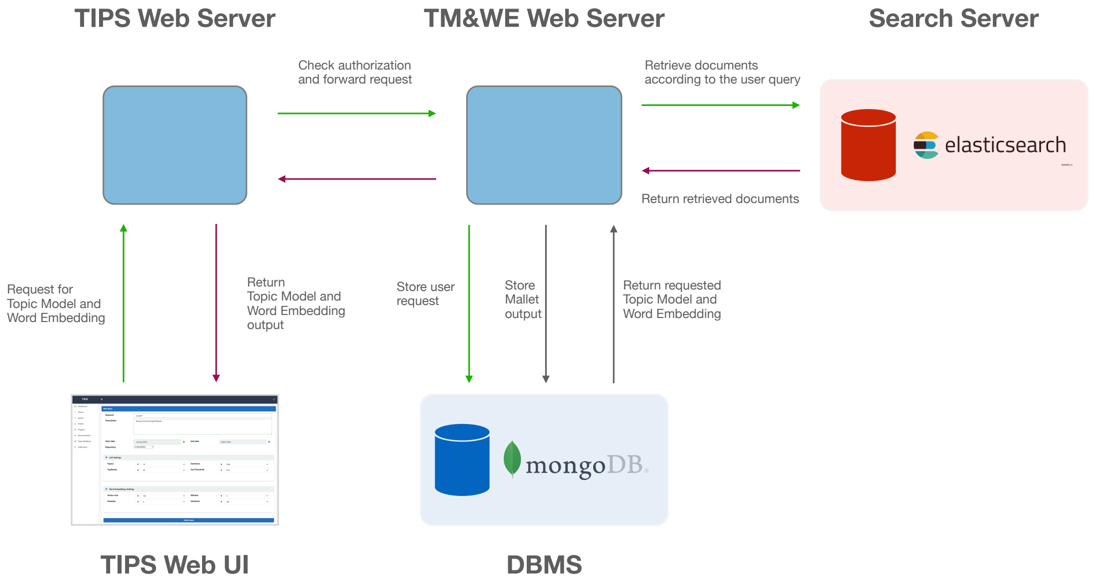

TIPS Web Server
TM&WE Web Server
Search Server
Check authorization and forward request
Retrieve documents according to the user query
elasticsearch
Return retrieved documents
Request for Topic Model and Word Embedding
Return Topic Model and Word Embedding output
Store user request
Store Mallet output
Return requested Topic Model and Word Embedding
mongoDB.
TIPS Web UI
DBMS

Figure 1: Overview of the TIPS Architecture responsible for Topic Modeling and Word Embedding

- build the topic model according to the specified settings;
- store the query and the settings in a dedicated MongoDB database;
- store the model and all the necessary outputs into a MongoDB database.

The last two points are necessary to prevent building multiple times the same topic model and allow a user to access the list of queries previously submitted to the system and their corresponding results; therefore, if the user already performed a request with the same query and the same LDA settings, results are retrieved from the dedicated MongoDB database, where they were stored after the first request. This is crucial also to support reproducibility, since the system automatically keeps track of the history of all the user requests, and the settings used for those requests. Figure 1 depicts the entire process and the relevant components of TIPS. The output generated by the TM&WE component can be accessed through the Web UI and includes:

- the list of topics, whose labels are by default "Topic$n$", where n is the identifier provided by Mallet; the user can edit the label and save the updated version;
- the top words per topic;
- the top documents for each topic, where the probability is above the specific threshold;
- a chart with the topic trends over time, where the time granularity (yearly, monthly, weekly) can be specified by the user and the importance of a topic in a given time interval is computed as in [25].

Actually, the request submitted to the TM&WE server includes a request for training Word Embedding through the Mallet functionalities. The user can specify the settings, which include embedding size, window size, sample size and number of iterations. The learned embedding are

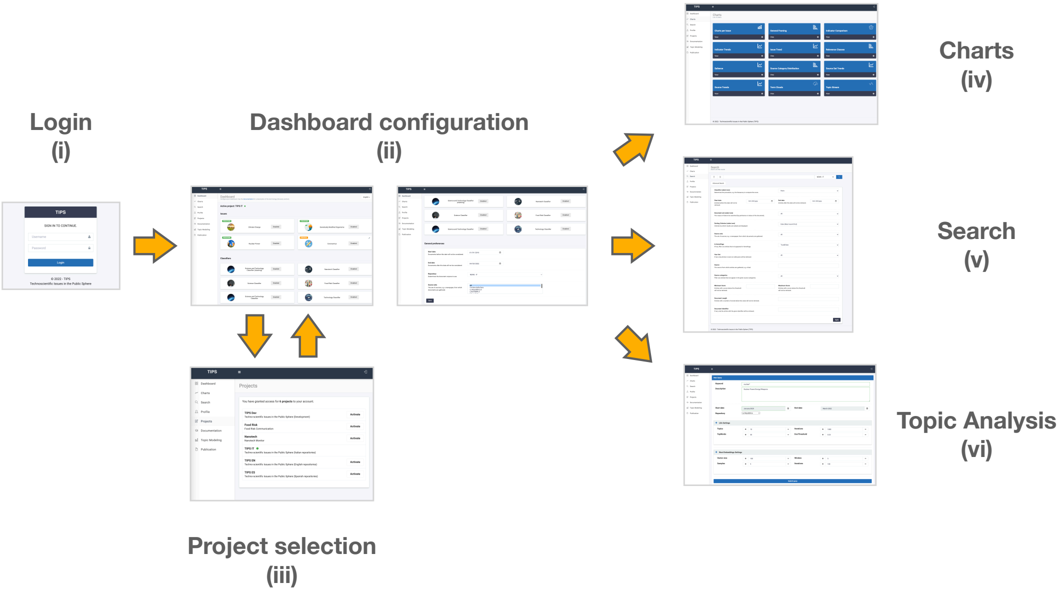


Figure 2: Overview of typical steps performed by a TIPS user and the relevant components

then used to visualize the words “closest” to a user specified keyword in the embedding space — if the query is constituted by a single term, the term is used by default as keyword.

4. A Use Case on Nuclear Issue

In this section we will rely on a use case to illustrate a typical TIPS user workflow; a summary of the steps and the relevant UI components is depicted in Fig. 2.

Let us consider a researcher who wants to perform a longitudinal study on how nuclear-related issues are discussed in a specific newspaper, e.g., *La Repubblica*. The reason for focusing on this newspaper might be that an online archive is available and spans several decades. In TIPS, each user is associated to at least one *project*, which determines repositories, charts, classifiers, and predefined issues (standing queries) she can access. In this case, the researcher should have access to the *TIPS-IT* project, which includes the repository *La Repubblica*.

After the authentication, the user can access the *Dashboard*, which allows the configuration of all the “parameters” of her research activity. A default project is associated to each user and is adopted to populate the dashboard after the login. The user can change the default project via a dedicated page – (iii) in Fig. 2 – where all her projects are listed; this is necessary, for instance, if the research activity is focused on another newspaper archive, e.g., *El País*, *The New York Times*, or *The Guardian*, or the activity consists in a comparative study among different countries as in [2]. In the dashboard, the user can specify the document repository – in our case *La Repubblica* – and the preferred *classifier* – e.g., the “Science and Technology” classifier – which will be used by default to generate some of the charts and for document ranking in the search requests, when sorting is based on the classifier score.

The user can also specify an *issue*, which consists in a query used to determine the corpus for the research activity. Besides a set of predefined issues, the user can define her own issue using a custom query; the history of all the user issues is stored in a MongoDB database to keep track of the research activities. In the considered use case, the user will specify a *custom* issue to determine nuclear related articles, e.g., by the prefix query "nuclear*".

After the dashboard configuration, the user can access documents via full-text or metadata-based search using the basic or advanced search page (v). Multiple search interactions can be adopted, both to gain additional knowledge required for the research activity or, to build the query for the custom issue. For instance, the researcher might be interested only in "nuclear power" and not in "nuclear weapons", and she might modify the issue query accordingly.

The "Charts" page (iv) reports the list of all the available charts. In the considered use case, the researcher might use the *Charts per Issue*, where different charts are dynamically computed on the corpus determined by the *issue*. Examples are the distribution among the newspaper sections of the articles about the issue, the trend of the risk indicator over time, or the readability index. For instance, the researcher can check if the value of the readability index in articles about the issue is lower than in the corpus and comparable with values observed in technoscientific articles. She can study the trend over time of the indicators, e.g., risk, and investigate if the observed peaks are due to specific events like nuclear accidents or if the trend is in line with the results obtained with traditional surveys [13]. Finally, the distribution among sections can help to provide insight into the different "thematic areas" where the nuclear issue is discussed.

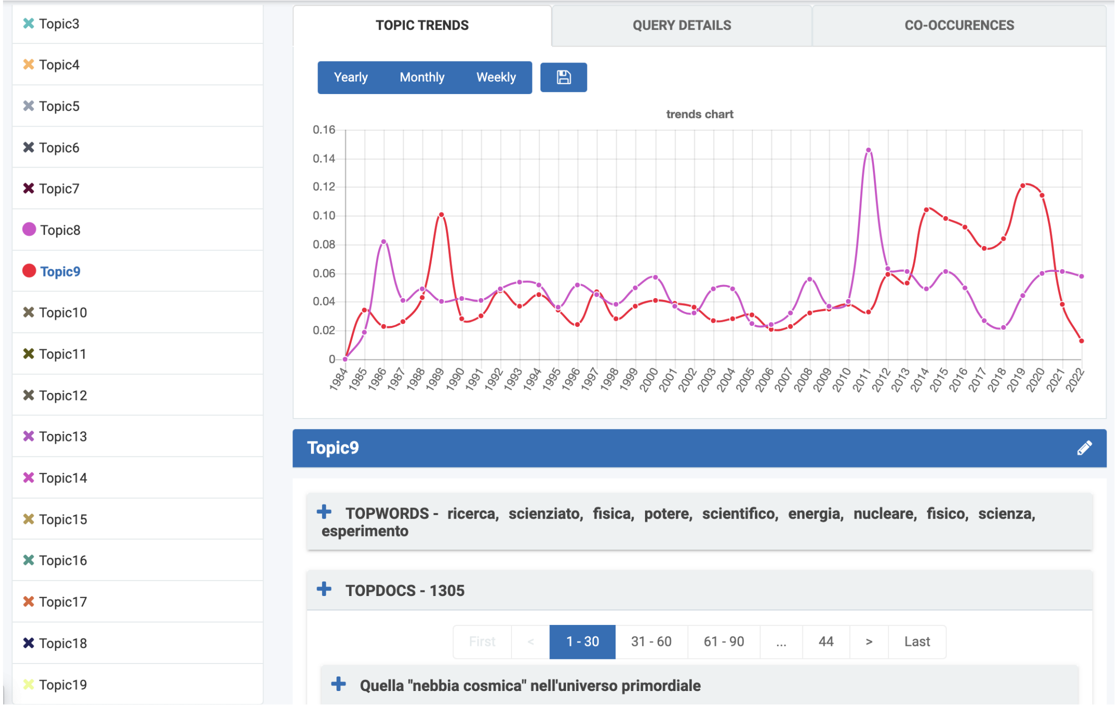
The researcher can then perform fine-grained analysis by retrieving and examining documents by the Search UI (v) or extracting features from sentences, documents, or document sets; the latter is the case of the syntactic features used to study the Communication of Science and Technology in Online Newspapers using Multidimensional Analysis [26].

The user can also explore themes in the corpus using the Topic Modeling functionalities (vi). Using the query built in the previous interactions, the user will submit a request to the TM&WE Web Server; parameters are set by default to 10 topics, 50 top words, 1000 iterations, and 0.25 for the probability threshold. Along with Topic Modeling, also Word Embedding will be trained. Figure 3 reports the page to explore the results. The user can select which topics to visualize; in Figure 3, topics 8 and 9 were selected, and their presence over time is visualized. The top words suggest that topic 9 is about nuclear research. Access to the top documents can help the interpretation of the topic. If only a subset of the topics is pertinent to the research activity, those topics can be later used to determine a subset of the corpus constituted only by documents where those topics are the most prominent — this is the approach used in [2].

5. Final Remarks and Future Directions

In this paper, we described the TIPS system and how it can support Social Sciences researchers such as sociologists in their research activities. We reported on a specific use case in order to present some of the system functionalities and how they are exploited.

Our research group is currently working on text-mining-based methodologies that will be later included in TIPS. Two are the research directions we are currently pursuing. One concerns the use of heterogeneous sources, e.g., news, tweets, blog and forum posts. Previous works



TOPWORDS - ricerca, scienziato, fisica, potere, scientifico, energia, nucleare, fisico, scienza, esperimento

TOPDOCS - 1305

Quella "nebbia cosmica" nell'universo primordiale

Figure 3: TIPS component to explore Topic Modeling results when considering documents published since form 1984 on an Italian Newspaper and pertinent to the query "nuclear*".

show how "standard" techniques, e.g., LDA, cannot be straightforwardly applied to short texts such as tweets [27] or to heterogeneous collections [28]. The other research direction concerns techniques to support longitudinal studies. We are planning to include existing Dynamic Word Embedding (DWE) approaches in the system. Moreover, we are investigating other time-aware representation of words or of groups of related words, thus complementing methods such as Time-aware Topic Modeling or DWE techniques.

Another line of work is focused on the "optimization" of the system. The current version exploits BM25 for ranking, using default values for the free parameters (b, k_1). As mentioned in [18], optimization can be costly both in terms of human evaluation and computing power; however, specific collections, such as those available in TIPS, are worthy of the cost. Since our collections have multiple fields, we are planning to move from BM25 to BM25F and optimize the parameters using techniques such as those described in [18, 29]. Another aspect is scalability. The indexing and classification module, *tips-data*, relies on (Java) multi-threading to index the diverse repositories in parallel. A possible issue might be the request load. The current number of users is limited and has its peak when TIPS is used for teaching activities, where students are usually divided into groups, and each group has its own TIPS account. However, we are planning to evaluate the robustness of the current system in terms of request load. We opted for MongoDB and elasticsearch because they are scalable and can be distributed on a number of machines. Possible bottlenecks might be the TIPS Web Server or the TM&WE Web Server; a possible strategy might be to run multiple instances of these servers and use a load balancer.

Acknowledgments

The authors would like to thank the Pa.S.T.I.S. Research Unit of Department of Philosophy, Sociology, Education and Applied Psychology (FISPPA), University of Padova, for the fruitful discussions and the feedback in the design of the TIPS System. The contribution by A. Zanatta was provided when he was working at the Department FISPPA of the Univeristy of Padova.

References

[1] A. Cammozzo, E. Di Buccio, F. Neresini, Monitoring technoscientific issues in the news, in: ECML PKDD 2020 Workshops - Workshops of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2020), Ghent, Belgium, Sept. 14-18, 2020, Proceedings, volume 1323 of *Communications in Computer and Information Science*, Springer, 2020, pp. 536–553. doi:10.1007/978-3-030-65965-3_37.

[2] F. Neresini, S. Crabu, E. Di Buccio, Tracking biomedicalization in the media: Public discourses on health and medicine in the uk and italy, 1984–2017, Social Science & Medicine 243 (2019) 112621. doi:https://doi.org/10.1016/j.socscimed.2019.112621.

[3] G. Sadowsky, Future developments in social science computing, in: Proceedings of Spring Joint Computer Conference, 1972, pp. 875–883.

[4] R. M. Alvarez (Ed.), Computational social science: Discovery and Prediction, Analytical methods for social research, New York: Cambridge University Press, 2016.

[5] G. A. Veltri, Digital social research, John Wiley & Sons, 2019.

[6] I. Flaounas, O. Ali, M. Turchi, T. Snowsill, F. Nicart, T. De Bie, N. Cristianini, Noam: News outlets analysis and monitoring system, in: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, SIGMOD '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 1275–1278. doi:10.1145/1989323.1989474.

[7] I. Flaounas, T. Lansdall-Welfare, P. Antonakaki, N. Cristianini, The Anatomy of a Modular System for Media Content Analysis, Arxiv - Social Media Intelligence (2014). arXiv:1402.6208.

[8] R. Steinberger, B. Pouliquen, E. van der Goot, An introduction to the Europe Media Monitor family of applications, in: Proceedings of the SIGIR 2009 Workshop on Information Access in a Multilingual World, volume 43, 2009. arXiv:1309.5290.

[9] A. Z. Broder, Identifying and filtering near-duplicate documents, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 1848, 2000, pp. 1–10. doi:10.1007/3-540-45123-4_1.

[10] M. Henzinger, Finding near-duplicate web pages, in: Proceedings of the Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, volume 2006, 2006, p. 284. doi:10.1145/1148170.1148222.

[11] J. R. Finkel, T. Grenager, C. Manning, Incorporating non-local information into information extraction systems by Gibbs sampling, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), Association for Computational Linguistics, Ann Arbor, Michigan, 2005, pp. 363–370. doi:10.3115/1219840.1219885.

[12] A. Palmero Aprosio, G. Moretti, Tint 2.0: an all-inclusive suite for nlp in italian (2018).

[13] F. Neresini, A. Lorenzet, Can media monitoring be a proxy for public opinion about technoscientific controversies? The case of the Italian public debate on nuclear power., Public understanding of science (Bristol, England) (2014).

[14] E. Di Buccio, A. Lorenzet, M. Melucci, F. Neresini, Unveiling Latent States Behind Social Indicators, in: R. Gavaldà, I. Zliobaite, J. Gama (Eds.), Proceedings of the SoGood@ECML-PKDD 2016, Riva del Garda, Italy, September 19, 2016., volume 1831 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016.

[15] D. H. Wolpert, Stacked generalization, Neural Networks 5 (1992) 241–259. doi:10.1016/S0893-6080(05)80023-1.

[16] H.-F. Yu, F.-L. Huang, C.-J. Lin, Dual coordinate descent methods for logistic regression and maximum entropy models, Machine Learning 85 (2011) 41–75. doi:10.1007/s10994-010-5221-8.

[17] E. Raff, Jsat: Java statistical analysis tool, a library for machine learning, Journal of Machine Learning Research 18 (2017) 1–5. URL: http://jmlr.org/papers/v18/16-131.html.

[18] S. Robertson, The Probabilistic Relevance Framework: BM25 and Beyond, Foundations and Trends® in Information Retrieval 3 (2009) 333–389. doi:10.1561/1500000019.

[19] D. M. Blei, J. D. Lafferty, Topic Models, in: A. Srivastava, M. Sahami (Eds.), Text Mining: Classification, Clustering, and Applications, New York, New York, USA, 2009.

[20] J. Boyd-Graber, Y. Hu, D. Mimno, Applications of Topic Models, Foundations and Trends® in Information Retrieval 11 (2017) 143–296. doi:10.1561/1500000030.

[21] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent dirichlet allocation, The Journal of Machine Learning Research 3 (2003) 993–1022.

[22] H. M. Wallach, D. Mimno, A. McCallum, Rethinking lda: Why priors matter, in: Proceedings of the 22Nd International Conference on Neural Information Processing Systems, NIPS'09, Curran Associates Inc., USA, 2009, pp. 1973–1981.

[23] J. Chuang, C. D. Manning, J. Heer, Termite: Visualization techniques for assessing textual topic models, in: Proceedings of the Workshop on Advanced Visual Interfaces AVI, ACM Press, New York, New York, USA, 2012, pp. 74–77. doi:10.1145/2254556.2254572.

[24] S. Liu, M. X. Zhou, S. Pan, Y. Song, W. Qian, W. Cai, X. Lian, TIARA: Interactive, topic-based visual text summarization and analysis, in: ACM Transactions on Intelligent Systems and Technology, volume 3, 2012, pp. 1–28. doi:10.1145/2089094.2089101.

[25] D. Mimno, Computational historiography, Journal on Computing and Cultural Heritage 5 (2012) 1–19.

[26] V. Zorzi, The Communication of Science and Technology in Online Newspapers: a Multidimensional Perspective, Ph.D. thesis, University of Padova, Italy, 2018.

[27] J. Qiang, Z. Qian, Y. Li, Y. Yuan, X. Wu, Short Text Topic Modeling Techniques, Applications, and Performance: A Survey, IEEE Transactions on Knowledge and Data Engineering 34 (2022) 1427–1445. doi:10.1109/TKDE.2020.2992485. arXiv:1904.07695.

[28] J. Qiang, P. Chen, W. Ding, T. Wang, F. Xie, X. Wu, Heterogeneous-Length Text Topic Modeling for Reader-Aware Multi-Document Summarization, ACM Transactions on Knowledge Discovery from Data 13 (2019) 1–21. doi:10.1145/3333030.

[29] A. Costa, E. Di Buccio, M. Melucci, G. Nannicini, Efficient parameter estimation for information retrieval using black-box optimization, IEEE Transactions on Knowledge and Data Engineering 30 (2018) 1240–1253. doi:10.1109/TKDE.2017.2761749.