Forecasting the Spreading of Technologies in Research Communities

Francesco Osborne
Knowledge Media Institute,
The Open University
MK7 6AA, Milton Keynes, UK
francesco.osborne@open.ac.uk

Andrea Mannocci
Knowledge Media Institute,
The Open University
MK7 6AA, Milton Keynes, UK
andrea.mannocci@open.ac.uk

Enrico Motta
Knowledge Media Institute,
The Open University
MK7 6AA, Milton Keynes, UK
enrico.motta@open.ac.uk

ABSTRACT

Technologies such as algorithms, applications and formats are an important part of the knowledge produced and reused in the research process. Typically, a technology is expected to originate in the context of a research area and then spread and contribute to several other fields. For example, Semantic Web technologies have been successfully adopted by a variety of fields, e.g., Information Retrieval, Human Computer Interaction, Biology, and many others. Unfortunately, the spreading of technologies across research areas may be a slow and inefficient process, since it is easy for researchers to be unaware of potentially relevant solutions produced by other research communities. In this paper, we hypothesise that it is possible to learn typical technology propagation patterns from historical data and to exploit this knowledge i) to anticipate where a technology may be adopted next and ii) to alert relevant stakeholders about emerging and relevant technologies in other fields. To do so, we propose the Technology-Topic Framework, a novel approach which uses a semantically enhanced technology-topic model to forecast the propagation of technologies to research areas. A formal evaluation of the approach on a set of technologies in the Semantic Web and Artificial Intelligence areas has produced excellent results, confirming the validity of our solution.

CCS CONCEPTS

• **Information systems** → Information retrieval; • **Computing methodologies** → Artificial intelligence

KEYWORDS

Scholarly Data, Semantic Web, Technology, Ontology

1 INTRODUCTION

Every new piece of research, no matter how ground-breaking, adopts previous knowledge and reuses tools and methodologies from the past. As emphasised by Isaac Newton, researchers stand "on the shoulders of giants": we constantly reuse ideas, methods and materials. Today, as the number of papers and the available scientific knowledge is growing rapidly, it is becoming increasingly harder to keep track of all the relevant knowledge and methodologies that could facilitate a research initiative, trigger a research idea, or spark a collaboration between experts from different fields. The availability of research material on the Web and the presence of academic search engines (e.g., Google Scholar, Microsoft Academic Search, SciVal Scopus) and systems, which support the exploration of the research environment (e.g., Semantic Scholar[1], aMiner [1], Saffron [2], Rexplore [3]), alleviates only marginally this issue. Indeed, while these systems are effective at processing keyword-based queries on the literature and at producing a variety of analytics about the research landscape, they do not attempt to represent explicitly the knowledge described in scholarly publications.

The vision underlying the work presented here is one in which researchers are assisted by software capable of applying data-driven methodologies to machine-readable descriptions of research knowledge. The aim is to expand the conceptual horizon of researchers and combine human creativity with the data mining ability of computers. The Semantic Web community has already started to work in this direction, by fostering the Semantic Publishing paradigm [4], creating bibliographic repositories in the Linked Data Cloud [5], generating knowledge bases of biological data [6], formalising research workflows [7], implementing systems for managing nano-publications [8, 9] and micro-publications [10], organising relevant workshops (e.g., Linked Science and SemSci at ISWC, Scientometrics and Sepublica at ESWC, SAVE-SD at WWW) and challenges (e.g., the ESWC Semantic Publishing Challenge), and creating a variety of ontologies to describe scholarly data, e.g., SWRC[2], BIBO[3], BiDO[4], FABIO[5]. Recently, Kitano [11] proposed an even more ambitious vision, suggesting the development of an artificial intelligence system able to make major scientific discoveries in biomedical sciences and win a Nobel Prize.

Technologies, such as algorithms, formats, and applications, are a very important part of the knowledge produced and reused in

K-CAP 2017, December 4–6, 2017, Austin, TX, USA

ACM 978-1-4503-5553-7/17/12…$15.00
https://doi.org/10.1145/3148011.3148030

[1] https://www.semanticscholar.org
[2] http://ontoware.org/swrc
[3] http://bibliontology.com
[4] http://purl.org/spar/bido
[5] http://purl.org/spar/fabio

the research process. Typically, a technology will first appear in a particular research community and will then spread and contribute to a variety of other research areas. For example, Semantic Web technologies (e.g., RDF, OWL) were first created by research communities in the field of Artificial Intelligence, Knowledge Base Systems, Formal Ontology and others; subsequently they contributed to a variety of other research areas, e.g., Information Retrieval, Human Computer Interaction, Biology, and many others. It is however easy to miss an interesting piece of knowledge from a different field. Therefore, the transfer of a technology from a one research area (e.g., Semantic Web) to a different and possibly conceptually distant other area (e.g., Digital Humanities) may take several years, potentially slowing down the research process.

We thus need to develop new methods to foster this process and possibly predict the spreading of technologies across research fields. Unfortunately, standard technology diffusion and forecasting models [19-25] cannot tackle this issue, since they are designed to assess the general potential of technologies associated with a good number of documents. Conversely, we want to predict that a technology that has no or very few initial publications in topic *T* will be adopted by researchers working on *T*.

We hypothesise that technologies which exhibit similar spreading patterns across multiple research topics will tend to be adopted by similar topics. Following this intuition, we introduce the Technology-Topic Framework (TTF), a novel approach which uses a semantically enhanced technology-topic model to predict the technologies that will be adopted by a research field. TTF characterises technologies in terms of a set of topics drawn from a large-scale ontology of research areas over a given time period and applies machine learning on these data to forecast technology spreading. Our goal is to suggest promising technologies to scholars in a field, thus helping to accelerate the knowledge flow and the pace of technology propagation.

The main contributions of this paper are:

- The definition and implementation of the Technology-Topic Framework, a novel approach to characterise and forecast technology propagation;
- A dataset associating technologies to research topics throughout time, which can be used to perform further analysis of technologies in the fields of Semantic Web and Artificial Intelligence;
- An evaluation on 1,118 technologies in the 1990-2013 period, which shows that our methodology can forecast technology spreading with a high precision.

The remainder of this paper is organised as follows. In Section 2, we discuss in details the Technology Topic Framework and the input knowledge bases. In Section 3, we discuss the state of the art of current methods to forecasting technologies. In Section 4 we evaluate our approach by comparing six machine learning algorithms. We conclude in Section 5 by outlining future directions of research.

2 TECHNOLOGY-TOPIC FRAMEWORK

Figure 1 shows the architecture of the Technology-Topic Framework (TTF). It takes as input a dataset of research papers, a list of technologies and a research topic ontology. It then characterises technologies according to their propagation through research topics and uses this representation to forecast the future propagation of novel technologies.

Of course, a technology can propagate to a topic as a result of events that cannot be anticipated by the knowledge of previous spreading patterns. Indeed, the adoption of a technology in a new research topic can be fostered by the creation of multidisciplinary workshops, by a scientific collaboration, by the inclusion of the technology in a commercial application, by the intuition of a researcher, and by many other events. The goal of TTF is therefore to focus on technology propagation events that follow to some extent previously observed patterns and forecast them with high enough precision for reliably suggesting new technologies to researchers. Naturally, the adoption of a technology does not hinge only upon the awareness of its existence, but it depends on a variety of technological, social, and political factors, whose analysis is out of the scope of this paper.

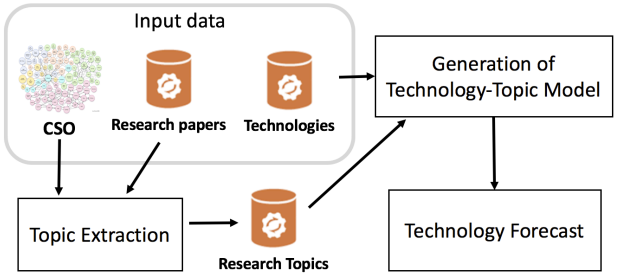

Figure 1. The Technology-Topic Framework architecture.

2.1 Input Knowledge Bases

The Technology-Topic Framework (TTF) takes as input three knowledge bases:

1) a dataset of research papers, described by means of their titles, abstracts, and keywords;
2) a list of input technologies, associated to the relevant publications in the research paper dataset;
3) an ontology of research areas, describing topics and their relationships.

In the following, we will discuss the specific knowledge bases adopted for the study of 1,118 technologies in Semantic Web and Artificial Intelligence presented in this paper. We will also suggest some alternative data sources that could be used to implement TTF.

Dataset. We used a dump of the Scopus database in the 1990-2013 period, containing about 16 million papers, mainly in the field of Computer Science. Scopus is a very large and high-quality database of peer-reviewed literature. Each paper is described by title, abstract, and a set of keywords. Similar available datasets which contain titles and abstracts of scholarly publications are Microsoft Academic Graph[6], Core[7], OpenAIRE[8], and CiteSeerX[9].

[6] https://www.microsoft.com/en-us/research/project/microsoft-academic-graph

Technology list. In general it is possible to generate a set of technologies by i) extracting them from research papers by means of automatic methods [12, 13], ii) obtaining them from a manually curated software repository (e.g., the Resource Identification Initiative portal [14]), or iii) getting them from a general knowledge base (e.g., DBpedia [15]). Since the focus of this study is on the analysis of technologies and not on their identification, we created a manually curated set of technologies in the fields of Semantic Web and Artificial Intelligence.

We first selected an initial set of about 2,000 technologies by running TechMiner [12] on a set of 3,000 papers in Semantic Web in order to find technologies that were originated or adopted by this field. We then manually cleaned and enriched the resulting dataset by discarding incorrect results and we also added 500 other technologies sourced from Wikipedia pages listing Artificial Intelligence and Machine Learning algorithms and methods. Specifically, we focused on three categories of technologies: algorithms and approaches (e.g., Support Vector Machines, Particle Swarm Optimisation, Latent Semantic Analysis), formats (e.g., Rule Interchange Format, OWL 2, Systems Modeling Language), and tools and applications (e.g., OntoClean, Taverna, Annotea). We then used an Elasticsearch instance to map each technology to all the papers in the Scopus dump which contained the technology name in the title or in the abstract, as done in previous technology forecasting studies [16]. Finally, we selected a set of 1,118 technologies which appeared in more than 10 publications.

Topic ontology. As a reference topic ontology, we used the Computer Science Ontology (CSO), created to represent topics in the Rexplore system [3], which is currently being trialled by Springer Nature to classify proceedings in the field of Computer Science [17], such as the well-known LNCS series. CSO was created by applying the Klink-2 algorithm [18] to the 16 million publications of our Scopus-derived dataset [3]. The Klink-2 algorithm combines semantic technologies, machine learning and knowledge from external sources (e.g., DBpedia, calls for papers, web pages) to automatically generate a fully populated ontology of research areas, which uses the Klink data model[10].

CSO is an extension of the BIBO ontology[11] which in turn is built on top of SKOS. It includes three semantic relationships: *relatedEquivalent*, which indicates that two topics can be treated as equivalent for the purpose of exploring research data (e.g., Ontology Matching, Ontology Mapping), *skos:broaderGeneric*, which indicates that a topic is a sub-area of another one (e.g., Linked Data, Semantic Web), and *contributesTo*, which indicates that the research output of a topic contributes to another (e.g., Ontology Engineering, Semantic Web).

2.2 Generation of Technology-Topic Matrices

The aim of this phase is to build for each year a matrix that characterises technologies in terms of their number of publications in different research topics. To this end, we first map each paper associated with at least one technology to a set of topics.

The classic way to do so is to adopt keywords as proxy for research topics or to apply a probabilistic topic model. However, as extensively discussed in previous works [3, 17, 18], these solutions ignore the rich network of semantic relationships between research topics and are often unable to distinguish research areas from other terms that may be used to annotate publications. Therefore, we exploit the topic ontology by associating to each paper i) all the concepts in CSO whose label is found either in the title, the abstract or the keyword set, as well as ii) all *skos:broaderGeneric* and iii) all *relatedEquivalent* areas of the initial set of topics extracted from the scholarly dataset. For example, a publication associated with the term SPARQL will be tagged also with higher-level topics such as RDF, Linked Data, Semantic Web, World Wide Web, and Computer Science.

Finally, for each technology, we count the number of papers for each topic in each year. The result is a sequence of matrices, one matrix for each year, in which rows represent technologies, columns represent topics, and cells contain the number of publications of a technology (e.g., Annotea) in a topic (e.g., Semantic Web) yielded in a given year.

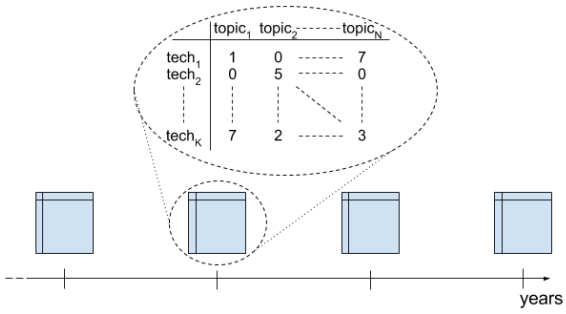

Figure 2. The technology-topic matrices.

Naturally, this representation rests on the assumption that a research topic will not change radically in time. We recognise that this is not necessarily true, being possible for a research area to shift, as new paradigms and ideas emerges. It is however a very common assumption that is adopted by most system for exploring research [1, 2, 3] and tends to work well in most cases.

In our implementation of TTF, the sequence of matrices is materialised as a *json* file that is fed to the forecasting module, which will use it for predicting which technology will emerge in which topic in the years to follow[12].

When analysing the spreading of technologies through the research landscape, it is useful to know the most frequent patterns.

[7] https://core.ac.uk
[8] https://www.openaire.eu
[9] http://citeseerx.ist.psu.edu
[10] http://technologies.kmi.open.ac.uk/rexplore/ontologies/BiboExtension.owl
[11] http://purl.org/ontology/bibo

[12] The technology-topic matrices and the TTF code are available at http://rexplore.kmi.open.ac.uk/TTF/downloads/code_and_data.zip.

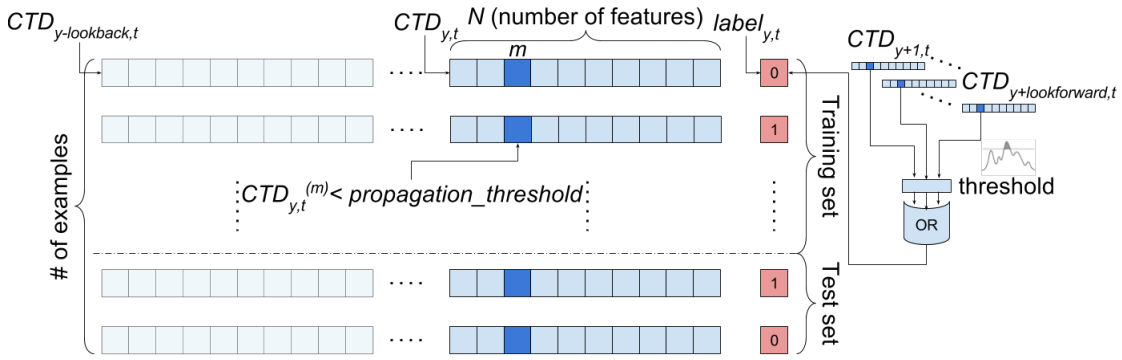

Figure 3. Construction of examples for topic *m*.

To do so, we implemented a python script that extracts frequent technology spreading patterns from the matrices. Since the focus of this paper is on forecasting, we will only describe it briefly.

We define a technology spreading pattern across a set of topics as an ordered tuple of *n* topics associated with the number of technologies that traversed the topics during a time interval. We say that topic T1 is subsequent to topic T2 in a pattern for a certain technology only if the technology that traversed T1 enters in T2 within a *n* years' time window (*n*=5 in our implementation).

Our method for extracting these patterns iterates over years and technologies, and counts the number of publications associated with the most common spreading patterns. Even if this technique simply finds the most common sequences of technology spreading patterns, without any assumption on the causality of the propagation links, it allows us to identify valuable and interesting patterns. For example, the patterns linking artificial intelligence and information retrieval to the fields of medical imaging and bioinformatics show the variety of technologies that were adopted by these fields over time, such as Support Vector Machine, Neural Networks, Finite-state Machine, Gesture Recognition applications, MapReduce, and so on. Some other patterns highlight how a variety of techniques first adopted for image analysis (e.g., AdaBoost, Boltzman machine, Conditional Random Fields, Neural Networks, Non-Negative Matrix Factorisation) were then used also for speech recognition (and vice versa). A full analysis of these patterns is not within the scope of this paper and will be presented in future works.

2.3 Technology Propagation Forecasting

The forecasting of technology propagation is treated as M separate classification problems, one for each topic of interest. Pseudocode 1 highlights the steps of the process. For the m^{th} topic, the sequence of technology-topic matrices is processed to extract *examples* to be fed to the machine learning models. Each example consists of a pair $(features, label)$, represented respectively in light blue and pink in Figure 3, where $features$ is a vector characterising (in terms of topics) a technology in one or more years and $label$ is a boolean indicating whether the technology in the example will spread to the topic of interest in a window of future years equal to the $lookforward$ input parameter (set to 5 by default). For each one of the M classification problems, the examples are organised into training sets and a test set including all examples after a certain year (defined by the $split\_year$ parameter) for evaluating the precision, recall and F1 score of the classifiers.

In order to produce suitable examples for one of the M classification problems, the sequence of technology-topic matrices (TTM for short) is scanned by year (within an input range) and by technology; each element $ttm_{y,t}$ of TTM locates precisely one and only one topic distribution about technology t in year y within the sequence of technology-topic matrices. For each $ttm_{y,t}$ we compute its cumulative topic distribution $CTD_{y,t}$:

$$CTD_{y,t} = \sum_{i=first\_year}^{y} ttm_{i,t}$$

where $ttm_{i,t}, CTD_{y,t} \in \mathbb{R}^N$, $first\_year$ is the first year of the period under analysis (i.e. 1990), and N is the number of components pertaining to a topic distribution, i.e. the number of topics considered in the problem statement (200 by default). This is in principle different from the number of classifications problems M.

Given a topic m, an example is created from a $CTD_{y,t}$ if and only if the number of publications for that topic $CTD_{y,t}^{(m)}$ is less than $propagation\_threshold$, a threshold accounting for the presence (absence) of a technology in a topic (set empirically to 5 in this work). This condition ensures that the machine learning algorithms focus only on cases in which a technology has not yet propagated to topic m.

Features vectors are obtained by concatenating the CTD of the year y with the $CTDs$ of the previous years, according to the variable $lookback$ (set to 2 by default):

$$features_{y,t} = \left(CTD_{y-lookback,t}, \dots, CTD_{y-1,t}, CTD_{y,t}\right)$$

Whenever a past year is missing or beyond the dataset boundaries, a *zero-padding* strategy is adopted by using $CTD_{padding} = (0, 0, \dots, 0) \in \mathbb{R}^N$.

The $label$ related to an example built around $CTD_{y,t}$ is computed by looking at future values of the technology for topic

m within $[CTD_{y+1,t}, \ldots, CTD_{y+lookforward,t}]$: if at least one of the $CTD^{(m)}$ is equal or greater than $propagation\_threshold$, then the label associated to the example is positive, otherwise it is negative. Namely:

$$label_{y,t} = \bigvee_{j=y+1}^{y+lookforward} threshold\left(CTD_{j,t}^{(m)}\right)$$

where $CTD_{j,t}^{(m)}$ is the m^{th} component of $CTD_{j,t}$, i.e. the cumulative number of publications about technology t associated to topic m up to year y, and $threshold()$ is a function yielding $True$ when the argument is greater than $propagation\_threshold$, $False$ otherwise. In analogy to the window $lookback$, the window $lookforward$ defines the maximum number of future years in which the approach checks whether a technology propagates to a topic.

We use Random Forest as default since it yielded the best performance for this task, as it will be discussed in the evaluation. However, the current implementation of TTF allows us to choose between six different machine learning classifiers.

An important point is that TTF cannot be applied indiscriminately to all research areas. Some of them will tend to adopt specific technologies and thus be more predictable, while others can be inherently more erratic. For this reason, in a realistic setting, it is necessary to evaluate the method on historical data and to return only the most reliable predictions, involving the subset of topics yielding the best results.

```
Function TTF_forecast (ttm, year_range, split_year, lookback, lookforward, topics)
    for topic in topics do
        features, labels ⟵ [ ], [ ];
        for year in year_range do
            for technology in ttm[year] do
                example_features ⟵ [ctd(ttm[year − lookback][technology]), ...,
                                    ..., ctd(ttm[year][technology])];
                example_label ⟵ threshold(ctd(ttm[year + 1][technology]), ...
                                ..., ctd(ttm[year + lookforward][technology]));
                if example is valid then
                    features.append(example_features);
                    labels.append(example_label);
                end
            end
        end
        x_train, x_test, y_train, y_test ⟵ split(features, labels, split_year);
        if enough examples then
            train(x_train, y_train);
            predictions = predict(x_test);
            eval_metrics = evaluate(predictions, y_test);
        end
    end
    return predictions, eval_metrics;
end
```

Pseudocode 1. The pseudocode of the TTF forecasting step.

3 RELATED WORK

Technology forecasting is an established research area, which can be tracked all the way back to the 1930s and currently adopts a variety of modern data driven methods [19, 20]. It has traditionally focused on detecting emerging [16] or vacant [21] technologies and assessing their potential. Most approaches to technology forecasting aim at supporting human experts with comprehensive models representing technical, social and political information about technologies. In recent years, we have also witnessed the emergence of (semi-)automatic methods for identifying and assessing promising technologies [22, 23]. Since TTF is a fully automatic method, we will focus on this second category.

Most of these approaches detect and assess technologies by analysing scientific literature [22] and patents [23]. For example, Kim et al. [21] used Latent Dirichlet Allocation for clustering patent documents and detecting promising technologies. Similarly, Jun et al. [24] forecasted vacant technologies by exploiting a K- medoids clustering method based on support vector clustering applied on patents. Chen et al. [23] presented an approach that integrates bibliometric and patent analysis into the logistic growth curve model for hydrogen energy and fuel cell technologies with the aim of identifying the optimal patent strategy for the fuel cell industry. Bengisu and Nekhili [16] used a scientometric approach to determine the number of publications and patents for some emerging technologies and assess their trends. Several analyses on the diffusion of technologies showed that their growth could be approximated by a S-shaped curve. which is usually modelled and predicted using Bass, Gompertz, Logistic, Richards and other statistical models [25].

However, as discussed in the introduction, these methods cannot be applied on the task addressed by this paper, since they focus on forecasting the growth of a technology, rather than predicting its adoption by researchers working on a specific research topic. In the latter case, the technology would not be associated with enough publications in the topic of interest to allow the application of statistical techniques.

Characterising the research environment through scientific artefacts, epistemological concepts (e.g., claims, hypothesis, motivation, background, experiment), and other research concepts is becoming increasingly important for analysing research and ensuring reproducibility. For example, The Resource Identification Initiative portal [14] is a manually curated archive which collects and assigns IDs to a number of scientific objects, including applications, systems and prototypes. Bio2RDF [6] is an initiative which provides a large network of Linked Data for the Life Sciences and includes information about biological compounds, drugs and genes. Taverna [7] is a workflow management system for designing, representing and executing scientific workflows. In addition, Linked Data repositories, such as Scholarly.org [5], and projects fostering the Open Science movement, such as OpenAIRE [26], provide a variety of data about scientific papers, authors, venues, and so on. Another way to express scientific knowledge is by means of nano-publications [8, 9] which are units of information that contain scientific claims that can be uniquely identified and attributed to their authors. Similarly, micro-publications [10] usually serialised in the Web Ontology Language, represent scientific claims and how they support and/or challenge one another. The FORCE 11 Software citation group [13] is also contributing to this endeavour by producing a set of principles to foster the traceability of software in research. These initiatives are complementary to TTF: they can potentially provide input information to TTF and exploit its output to better characterise the flow of knowledge in research.

TTF characterises technologies with a distribution of research topics from an ontology. Representing entities, such as authors, venues, communities and citations according to their associated

[13] https://www.force11.org/group/software-citation-working-group

topic has proved to be very useful for generating analytics on research. For example, the Author-Conference-Topic (ACT) model [1] uses Latent Dirichlet Allocation and treats authors as probability distributions over topics, conferences and journals. Zhao et al. [27] proposed a topic-oriented community detection approach which combines both topic clustering and link analysis. Similarly, Racherla and Hu [28] identified topic-oriented communities by exploiting a topic similarity matrix and assigning a predefined research topic to each document and author. Differently from these approaches, we characterise topics through a formal ontology since this solution allows us to generate a structured set of unambiguous research topics linked by semantic relationships [18].

4 EVALUATION

We evaluated TTF on 1,118 technologies and 173 topics in the field of Computer Science during the 1990-2013 period[14]. The evaluation had two main purposes. First, to confirm the initial hypothesis, i.e., that it is possible to forecast technology propagation, at least for a certain subset of technologies and topics, by learning how technologies spread in the past. Secondly, to compare the performance of several machine learning algorithms on this task.

4.1 Experimental Setup

We selected as *training set* examples in the 1990-2004 period and as *test set* examples in the 2005-2008 period. We chose these intervals as they allowed us to label the examples in the test set using a $lookforward$ window of five years (2009-2013). We set the $propagation\_threshold$ threshold to 5 and $lookback$ to 2 years. Since we wanted to focus on predicting relatively new technologies, we considered only examples about technologies which existed for no more than 5 years and in which a technology has two or fewer publications in a topic. We simulated a realistic situation by assuming 2005 as current year and not using any information successive to that year to label the examples in the training set.

We selected the 173 topics which were associated with at least 30 positive examples in both the training and the test sets in the period under analysis and trained a classifier for each of them. Each topic classifier was trained on average on 5,136 ± 240 examples (for a total of 888,633 examples) and was evaluated on 679 ± 90 examples (for a total of 117,516 examples).

As discussed previously, it is not possible to apply to this task the standard models for forecasting the potential of a technology. We thus tested six machine learning algorithms on the technology-topic model that characterise the propagation patterns: Logistic Regression, Random Forest, Decision Tree, Support Vector Machine, Neural Network, and Gradient Boosting. The tuning of hyper-parameters used in each single model was performed by a twofold cross-validation over the training set as supported by the scikit-learn library[15].

[14] The evaluation data are available at http://rexplore.kmi.open.ac.uk/TTF

[15] http://scikit-learn.org

We measured the performance of the classifiers by computing precision, recall and F1 score. The overall performance over a number of topics was determined by computing the micro-average of these metrics. The significance of the results on the categorical variables precision and recall was assessed by using the chi-square test for tables of cross-categorised frequency data rxc (with Yates' correction for 2x2 tables), or, when more appropriate the McNemar's test for correlated proportions (recall). The existence of statistical differences among *n* F1 not-independent distributions was explored with non-parametric tests, Wilcoxon's for n=2 and Friedman's for n>2. Similarities in the behaviour of two F1 distributions were investigated using the within-subject ANOVA to compute the intra-class correlation coefficient ICC (0-1), which measures the item reliability, and the η^2 coefficient (0-1), also known as Cronbach's coefficient, which measures the tendency of the correlated members of two distributions to have approximately the same values. The closer ICC and η^2 to 1, the higher the similarity.

4.2 Results

Figure 4 and Figure 5 show respectively the precision and the recall obtained by the six algorithms on the first *n* topics, ordered by the number of positive labels in the test set. The classifiers perform better on the topics associated to a higher number of technology propagation events, so the performance decrease with the number of topics.

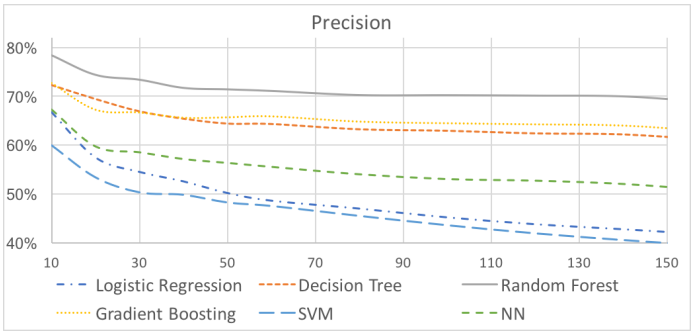

Figure 4. Average precision of the six machine learning approaches on the first *n* topics.

The first part of the analysis considered the ensemble of the six algorithms, arranging in two 6x2 contingency tables (5 degrees of freedom) the values of precision and recall. The chi-square test evidenced a highly significant difference among the 6 approaches both for precision and recall ($p<0.0001$). We then zoomed the analysis on the three top performers: Random Forest, Decision Tree, and Gradient Boosting. Random Forest yielded the best result in terms of precision. For the first 20 topics, its precision was over 74.4%, significantly higher ($p<0.0001$) than the value of 69.4% obtained with Decision Tree and 67.2% with Gradient Boosting. Also, considering the first 100 topics, Random Forest scored best, with 70.2% versus 62.9% of Decision Tree and 64.4% of Gradient Boosting ($p<0.0001$). Conversely, Gradient

Boosting performed best in terms of recall. For the first 20 topics, it scored 47.2%, significantly higher than the value of 44.7% for Random Forest ($p=0.04$) and the value of 42.5% for Decision Tree ($p<0.0001$). For the first 100 topics, the Gradient Boosting recall was 35.1%, again significantly higher ($p<0.0001$) than 32% for Random Forest and 31.5% for Decision Tree.

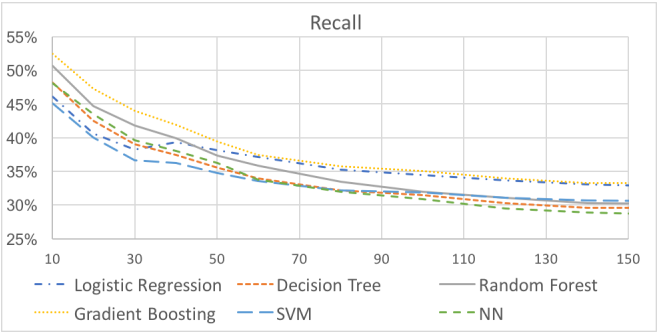

Figure 5. Average recall of the six machine learning approaches on the first *n* topics.

The F1 distributions of the six algorithms resulted significantly different when compared with Friedman's test for multiple correlated distributions ($p<0.0001$). Zooming again on the three top performers, also based on the results obtained for precision and recall, we carried out three one-to-one direct comparisons: Random Forest vs Gradient Boosting, Random Forest vs Decision Tree and Gradient Boosting vs Decision Tree. For the first comparison, we obtained ICC=1 and $\eta^2=0.99$; the excellent agreement between the two algorithms was confirmed by Wilcoxon's test ($p=0.11$, no significant differences). Decision Tree was instead significantly different from both Random Forest and Gradient Boosting ($p=0.003$), with lower values of ICC and η^2 (ICC= 0.9 and 0.86 respectively and $\eta^2=0.94$ and 0.92).

4.3 Discussion

The evaluation confirms our initial hypothesis: it is indeed possible to learn from historical spreading patterns and forecast technology propagation, at least for the set of topics that are more involved in technology propagation events. In particular, the good value of precision indicates we may be able to safely produce sound suggestions to researchers in those fields. The recall seems to confirm the intuition that is not possible to forecast all propagation events only on the basis of previous propagation patterns.

Our study suggests that Random Forest performs best in term of precision, while Gradient Boosting offers an advantage in terms of recall. It is possible that a higher number of examples might have favoured the neural network approach or other similar approaches that need very large example sets. However, currently it is still a challenge to generate large datasets of technology data. Since, the main aim of TTF is to forecast a set of propagations with high precision, we adopted Random Forest as default.

Table 1 shows the best research areas in term of F1 score when adopting Random Forest. Naturally, TTF tends to work best on topics associated with a large set of publications and technologies. However, the performance also depends on the inherent nature of topics and their behaviour in term of adopted technologies. For example, research areas about computer networks, sensors, and peer-to-peer systems perform well since they are usually associated with a coherent set of technologies. Similarly, research areas associated to the image processing field also yield good results, receiving a constant flow of approaches and formats to elaborate and describe images.

Table 1. Performance of Random Forest on the first 24 topics, with at least 50 positive labels, ordered by F1 score.

Topics	Prec.	Rec.	F1	Topics	Prec.	Rec.	F1
information retrieval	92.6%	66.8%	77.6%	wireless networks	64.7%	47.8%	55.0%
database systems	82.6%	65.9%	73.3%	sensor networks	71.9%	43.6%	54.3%
world wide web	88.6%	56.1%	68.7%	software engineering	70.6%	44.0%	54.2%
artificial intelligence	83.6%	55.2%	66.5%	distributed com. sys.	67.5%	45.0%	54.0%
computer architecture	68.3%	63.3%	65.7%	quality of service	59.6%	48.6%	53.5%
computer networks	82.1%	54.0%	65.2%	imaging systems	100.0%	35.8%	52.8%
image coding	96.8%	46.9%	63.2%	data mining	60.8%	45.3%	52.0%
P2P networks	78.9%	50.8%	61.9%	computer vision	92.3%	36.0%	51.8%
telecom. traffic	70.8%	48.1%	57.3%	Program. languages	65.3%	42.0%	51.2%
wireless telecom. sys.	74.4%	46.4%	57.1%	problem solving	69.0%	39.7%	50.4%
sensors	78.8%	43.7%	56.2%	semantic web	77.8%	37.1%	50.2%
web services	83.3%	42.2%	56.0%	image quality	74.2%	37.7%	50.0%

Table 2. Example of topics correctly forecasted by TTF.

Extreme learning machine: infor. retrieval, wireless telecommunication systems, signal processing, security of data, data mining, computer vision, robotics, image reconstruction, speech recognition, object recognition, [...]
Markov Logic Network : information retrieval, software engineering, problem solving, signal processing, image processing, data mining, KB systems, computer vision, theorem proving , image segmentation, [...]
Latent Dirichlet allocation: object recognition, software engineering, computer networks, security of data, ontology, semantic web, robotics, imaging systems, data reduction, image compression, object recognition, [...]
Web Ontology language: multimedia systems, image processing, parallel processing systems, computer vision, robotics, network architecture, security systems, computer aided design, genes (biology), e-learning, [...]
SKOS: software engineering, bioinformatics, computer networks, ML systems, information systems, computer programming languages, linguistics, object oriented programming, software design , e-learning, [...]
Semantic Web Rule Language: quality of service, object oriented programming, software design, mobile telecommunication systems, mobile devices, e-commerce, social networks, computer aided soft. engineering, [...]
FOAF: software engineering, computer networks, machine learning systems, security of data, distributed computer systems, web services, mobile computing, software design, multimedia systems, P2P networks, [...]

Table 2 shows, as an example, seven technologies taken in consideration in the 2005-2008 period and the set of topics where they did propagate after a few years, as correctly forecasted by TTF. For example, TTF correctly suggested that the novel (at that time) Extreme Learning Machine, which is a feedforward neural network with a single layer of hidden nodes, was deemed to be

applied to topics such as speech recognition, computer vision, and robotics. Similarly, it was able to forecast that Latent Dirichlet Allocation was going to be adopted by the Semantic Web area, as well as by communities working on data reduction, image compression, data security, and so on. TTF also anticipated the strong propagation of Semantic Web formats such as Web Ontology Language, SKOS and Semantic Web Rule Language to several research areas, such as bioinformatics, social network, electronic commerce, e-learning, and so on.

5 CONCLUSIONS

In this paper, we presented the Technology-Topic Framework, a novel approach that characterises technologies in terms of relevant research topics over time and forecasts technology propagation across research areas. An implementation of the system was evaluated on a set of 1,118 technologies in the fields of Semantic Web and Artificial Intelligence, yielding a precision of 74.4% and a recall of 47.7% for the first 20 research areas when using Random Forest. These results confirm that is possible to use historical propagation patterns for forecasting technology spreading.

Nonetheless, TTF presents some limitations that we intend to address in future works. First, it trains each classifier sequentially, and therefore it is not very scalable. This issue could be solved by parallelising the training phase or by adopting multi-class/multi-label classification. In the second instance, the current method for associating technologies to research papers is purely syntactic. While this solution has been used by several technology forecasting systems [16], we think that a semantic characterisation of research technologies could possibly yield better results. Therefore, we intend to create an ontology of technologies and exploit it for mapping technologies to papers. In the third instance, TTF only focus on technology spreading patterns and does not take in account other potentially significant factors, such as the persons championing a technology, the cost of adoption, the presence of usable implementations, and the socio-political context. We plan to enrich the forecasting model by considering text generated features and possibly derive additional features from external knowledge bases and social media. The aim is to combine all these knowledge sources to forecast technology propagation events even when they do not match a previously observed pattern. We also intend to expand the scope of our work, by including in the analysis a more varied set of research fields, such as Biology, Social Science and Engineering. Finally, we intend to create a web application enabling researchers to explore technology patterns and to receive tailored suggestions of technologies that may support their research work.

ACKNOWLEDGMENTS

We thank Elsevier BV for providing us a Scopus dump.

REFERENCES

[1] Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L. and Su, Z.: Arnetminer: extraction and mining of academic social networks. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 990-998). ACM. (2008)

[2] Monaghan, F., Bordea, G., Samp, K., Buitelaar, P.: Exploring Your Research: Sprinkling some Saffron on Semantic Web Dog Food. In: Semantic Web Challenge at the International Semantic Web Conference. (2010)

[3] Osborne, F., Motta, E., Mulholland, P.: Exploring scholarly data with Rexplore. In The Semantic Web–ISWC 2013 (pp. 460-477). Springer (2013)

[4] Shotton, D.: Semantic publishing: the coming revolution in scientific journal publishing. Learned Publishing, 22(2), pp.85-94. (2009)

[5] Nuzzolese, A.G., Gentile, A.L., Presutti, V. and Gangemi, A.: Semantic web conference ontology - A refactoring solution. In International Semantic Web Conference (pp. 84-87). Springer International Publishing. (2016)

[6] Belleau, F., Nolin, M.A., Tourigny, N., Rigault, P. and Morissette, J.: Bio2RDF: towards a mashup to build bioinformatics knowledge systems. Journal of biomedical informatics, 41(5), pp.706-716. (2008)

[7] Wolstencroft, K., Haines, R., Fellows, D., Williams, A., Withers, D., Owen, S., Soiland-Reyes, S., Dunlop, I., Nenadic, A., Fisher, P. and Bhagat, J.: The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud. Nucleic acids research, p.gkt328. (2013)

[8] Groth, P., Gibson, A. and Velterop, J.: The anatomy of a nanopublication. Information Services & Use, 30(1-2), pp.51-56. (2010)

[9] Kuhn, T., Chichester, C., Krauthammer, M., Queralt-Rosinach, N., Verborgh, R., Giannakopoulos, G., Ngomo, A.C.N., Viglianti, R. and Dumontier, M.: Decentralized provenance-aware publishing with nanopublications. PeerJ Computer Science, 2, p.e78. (2016)

[10] Schneider, J., Ciccarese, P., Clark, T. and Boyce, R.D.: Using the Micropublications ontology and the Open Annotation Data Model to represent evidence within a drug-drug interaction knowledge base. 4th International Conference on Linked Science, Vol 1282 (pp. 60-70). CEUR-WS. org. (2014)

[11] Kitano, H.: Artificial intelligence to win the nobel prize and beyond: Creating the engine for scientific discovery. AI magazine, 37(1), pp.39-50. (2016)

[12] Osborne, F., de Ribaupierre, H. and Motta, E.: TechMiner: Extracting Technologies from Academic Publications. In Knowledge Engineering and Knowledge Management: 20th International Conference, EKAW 2016, Bologna, Italy (pp. 463-479). Springer International Publishing. (2016)

[13] Ronzano, F. and Saggion, H.: October. Dr. inventor framework: Extracting structured information from scientific publications. In International Conference on Discovery Science (pp. 209-220). Springer (2015)

[14] Bandrowski, A., Brush, M., Grethe, J.S., Haendel, M.A., Kennedy, D.N., Hill, S., Hof, P.R., Martone, M.E., Pols, M., Tan, S.C., Washington, N.: The Resource Identification Initiative: A cultural shift in publishing. Journal of Comparative Neurology, 524(1), pp.8- 22. (2016)

[15] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R. and Hellmann, S.: DBpedia-A crystallization point for the Web of Data. Journal of Web Semantics, 7(3), pp.154-165. (2009)

[16] Bengisu, M. and Nekhili, R.: Forecasting emerging technologies with the aid of science and technology databases. Technological Forecasting and Social Change, 73(7), pp.835-844. (2006)

[17] Osborne, F., Salatino, A., Birukou, A. and Motta, E.: Automatic Classification of Springer Nature Proceedings with Smart Topic Miner. In International Semantic Web Conference (pp. 383-399). Springer (2016)

[18] Osborne, F. and Motta, E.: Klink-2: integrating multiple web sources to generate semantic topic networks. In International Semantic Web Conference (pp. 408-424). Springer International Publishing. (2015)

[19] Coates, V., Farooque, M., Klavans, R., Lapid, K., Linstone, H.A., Pistorius, C. and Porter, A.L.: On the future of technological forecasting. Technological forecasting and social change, 67(1), pp.1-17. (2001)

[20] Dror, I.: Forecasting technologies within their socioeconomic framework. Technological Forecasting and Social Change, 34(1), pp.69-80. (1988)

[21] Kim, G.J., Park, S.S. and Jang, D.S.: Technology forecasting using topic-based patent analysis. (2015)

[22] Daim, T.U., Rueda, G., Martin, H. and Gerdsri, P.: Forecasting emerging technologies: Use of bibliometrics and patent analysis. Technological Forecasting and Social Change, 73(8), pp.981-1012. (2006)

[23] Chen, Y.H., Chen, C.Y. and Lee, S.C.: Technology forecasting and patent strategy of hydrogen energy and fuel cell technologies. International Journal of Hydrogen Energy, 36(12), pp.6957-6969. (2011)

[24] Jun, S., Sung Park, S. and Sik Jang, D.: Technology forecasting using matrix map and patent clustering. Industrial Management & Data Systems, 112(5), pp.786-807. (2012)

[25] Marinakis, Y.D., 2012. Forecasting technology diffusion with the Richards model. Technological Forecasting and Social Change, 79(1), pp.172-179.

[26] Bardi, A., Castelli, D. and Manghi, P.: OpenAIRE Initiative: Providing Access, Monitoring and Contextualizing Open Access Publications. (2015)

[27] Zhao, Z., Feng, S., Wang, Q., Huang, J. Z., Williams, G. J., Fan, J.: Topic oriented community detection through social objects and link analysis in social networks. Knowledge-Based Systems, 26, 164-17. (2012)

[28] Racherla, P., Hu, C.: A social network perspective of tourism research collaborations. Annals of Tourism Research, 37(4), 1012-1034. (2010)