Predicting Document Creation Times in News Citation Networks

Andreas Spitz
Heidelberg University
Heidelberg, Germany
spitz@informatik.uni-heidelberg.de

Jannik Strötgen
Max-Planck-Institute for Informatics
Saarbrücken, Germany
jannik.stroetgen@mpi-inf.mpg.de

Michael Gertz
Heidelberg University
Heidelberg, Germany
gertz@informatik.uni-heidelberg.de

ABSTRACT

For the temporal analysis of news articles or the extraction of temporal expressions from such documents, accurate document creation times are indispensable. While document creation times are available as time stamps or HTML metadata in many cases, depending on the document collection in question, this data can be inaccurate or incomplete in others. Especially in digitally published online news articles, publication times are often missing from the article or inaccurate due to (partial) updates of the content at a later time. In this paper, we investigate the prediction of document creation times for articles in citation networks of digitally published news articles, which provide a network structure of knowledge flows between individual articles in addition to the contained temporal expressions. We explore the evolution of such networks to motivate the extraction of suitable features, which we utilize in a subsequent prediction of document creation times, framed as a regression task. Based on our evaluation of several established machine learning regressors on a large network of English news articles, we show that the combination of temporal and local structural features allows for the estimation of document creation times from the network.

CCS CONCEPTS

• **Information systems** → *Web mining*; • **Computing methodologies** → *Supervised learning by regression*;

KEYWORDS

news, citation network, temporal evolution, document dating

ACM Reference Format:
Andreas Spitz, Jannik Strötgen, and Michael Gertz. 2018. Predicting Document Creation Times in News Citation Networks. In *WWW '18 Companion: The 2018 Web Conference Companion, April 23–27, 2018, Lyon, France.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3184558.3191633

1 INTRODUCTION

The question *When was this published?* does not only come to mind when browsing news websites that inconveniently neglect to include or update a publication timestamp, but is also a central aspect in the automated processing of news documents. For many news analysis tasks that rely on temporal information such as event detection or timeline generation, the extraction of temporal expressions is a key component. In the news domain, such an extraction relies heavily on the availability of accurate document creation times (DCT) since the majority of temporal expressions are given in relation to the reference time of the document [24]. Thus, knowledge of the DCT is a necessary precursor for subsequent temporal analyses.

Depending on the source of the document, the DCT may be difficult to obtain. While it is a simple task for articles obtained from RSS feeds, it is more challenging when articles are obtained via social media links or web crawls. In these cases, the creation time may be available from a variety of metadata fields, in the text of the article itself, or missing entirely. As a result, a number of methods have been developed to estimate the DCT of documents on the Web from metadata, available versions of the document, external web archives, and links to other documents (see, e.g., [21, 25]).

Due to the constantly changing structure of the Web, such metadata may not always be available. Even worse, the content of the news articles may change over time, often including an update of the timestamp to the last-modified date that discards the original date. Simply storing the data to circumvent this problem is often not possible due to the proprietary nature of news articles. As a result, estimating the DCT of news articles is a difficult problem even when the entire content and metadata is available, and becomes continuously more challenging as time progresses.

In this paper, we explore the premise that news citation networks, which encode the flow of knowledge between individual news articles, may be helpful tools in estimating the DCT of articles with unknown publication times based on their neighbourhood in the network. Since such networks encode the relational structure between articles but not their content, they are safe to store. Similar to scientific citation networks, news citation networks can be extracted from the references that are contained within digitally published news articles [22]. In contrast to scientific citations, however, the resulting networks of news citations are more sparse and thus pose a greater challenge for predictive tasks since very little adjacency information is available for each individual article.

Contributions. We construct a large news citation network from international English news articles and investigate its utility for the prediction of news article DCTs. We explore the structure and temporal evolution of such a network as well as the extraction of suitable machine learning features, before evaluating the prediction of DCTs as a regression task for six regression approaches.

2 RELATED WORK

Our work relates to the areas of document creation time estimation and news citation networks, which we survey in the following.

Estimating document creation times. The most straightforward way to estimate a document's last update time is to use HTTP header fields [1]. Since these are often either unavailable or unreliable, Toyoda and Kitsuregawa propose a novelty measure for identifying new documents in a series of unstable web snapshots by scoring incoming links from other web pages [26]. Similarly,

WWW '18 Companion, April 23–27, 2018, Lyon, France

ACM ISBN 978-1-4503-5640-4/18/04.
https://doi.org/10.1145/3184558.3191633

Nunes et al. exploit neighboring pages of web documents by using incoming links, outgoing links, and HTML source attributes to discern the last-modified date of web documents [17]. For online resources, the DCTFinder combines supervised learning and rules to detect the DCT of web pages by identifying the title and selecting the oldest date among the possible candidates [25]. In contrast, CarbonDate [21] exploits a variety of web features to determine the DCT of web pages, e.g., the first time the URI was shortened or tweeted, and the first time it appeared in a web archive. Again, the document creation time is estimated to be the oldest available date.

Furthermore, the textual content of documents is exploited in several approaches for document dating, where the goal is to assign the most likely creation time to an undated piece of text. Typically, such approaches focus on historic documents and thus work at the coarse temporal granularity of years. Often, temporal language models are exploited for this task [7, 10]. In contrast, Chambers infers document creation times based on the temporal expressions occurring in the documents, while Ge et al. propose an event-based model [8]. Based on the observation that parts of documents may have differing creation times, Zhao and Hauff address the estimation of creation times for sub-documents on blog pages [28].

In contrast to the above approaches, we focus exclusively on the estimation of document creation times in citation networks of news articles, whose texts contain references and temporal expressions, but no further external metadata. Since multiple versions or mentions of articles are not available in this setting, the prediction task is best approached as a pure regression problem.

News citation networks. News citation networks are conceptually similar to scientific citation networks (for an overview, see [18]). While scientific citation networks are well researched models of knowledge dynamics, news citation networks have so far received little attention. Based on various web document types such as news articles, blogs, and social media posts, Kim et al. analyze the structure of user citation networks [11]. Similarly, with a focus on online news, Spitz and Gertz investigate the evolution of citations in a network of German news articles [22]. In contrast, we focus exclusively on news citations and on a much larger network extracted from international English speaking news outlets.

3 DATA EXTRACTION AND EXPLORATION

Before we proceed to the prediction of DCTs from news citations, we investigate the underlying network structure.

3.1 News Citation Networks

Intuitively, a news citation network represents citations between news articles, much like a citation network between scientific publications. However, an important aspect is the limitation to *internal* references, i.e., the focus on references that are anchored in the article text and the exclusion of advertisements or navigational links. Formally, let V be a set of news articles. With $E \subseteq V \times V$, we denote the set of edges between these articles such that for two articles v and w, we have $(v, w) \in E$ iff the text of v contains a reference to w. The directed graph $G = (V, E)$ then represents the network of news citations. Each article $v \in V$ can be attributed further, for example, with a publication time, a text, or temporal expressions. For a more in-depth introduction to news citation networks, see [22].

Table 1: Overview of news outlets, with number of days d the outlet has been included, average number of articles per day $\langle a \rangle$, average number of temporal expressions per article $\langle t \rangle$, and percentage of incoming citations e_{in} and outgoing citations e_{out} that connect to a different news outlet.

short	news outlet	d	$\langle a \rangle$	$\langle t \rangle$	e_{in}	e_{out}
AJ	Al Jazeera	334	14.0	7.4	7.9	1.4
AP	Associated Press	548	0.6	7.6	0.0	0.0
AT	The Atlantic	334	7.2	10.5	16.7	50.6
BBC	British Bc. Corp.	730	8.1	6.5	19.1	8.0
CBC	Canadian Bc. Corp.	334	12.2	7.4	6.6	3.0
CBS	Columbia Bc. System	548	31.9	6.7	5.3	1.1
CDT	China Digital Times	244	1.2	10.3	0.5	28.2
CNN	Cable News Network	548	2.8	8.8	3.3	61.1
DM	Daily Mail	244	7.4	8.3	0.0	0.0
DT	Daily Telegraph (AU)	213	3.0	5.4	9.9	43.5
DW	Deutsche Welle	334	1.2	6.1	48.1	5.9
FOX	Fox News	548	2.7	9.8	0.0	0.0
TG	The Guardian	730	40.7	7.6	4.7	3.8
TH	The Herald	244	0.7	7.3	0.6	0.0
HK	Huffington Post (UK)	548	4.9	4.7	1.6	42.0
HU	Huffington Post (US)	548	6.8	8.1	9.5	86.3
IBT	Int. Business Times	669	29.3	6.4	0.4	15.2
TI	The Independent	730	35.4	5.7	6.1	5.5
LAT	LA Times	548	31.6	8.2	2.9	4.1
NPR	National Public Radio	334	0.4	8.4	63.6	58.5
NY	The New Yorker	548	3.0	13.2	33.5	30.6
NYT	New York Times	669	23.8	10.7	26.8	4.7
OBS	The Observer	213	18.8	5.9	0.2	9.0
CMP	S. China Morn. Post	122	19.2	7.8	4.5	0.0
SC	The Scotsman	244	2.0	5.3	5.8	3.6
SKY	Sky News	548	13.0	5.0	6.5	0.0
SMH	Sydney Morn. Herald	548	2.3	7.0	3.0	51.9
TEL	The Telegraph	730	28.9	6.5	7.1	2.4
EX	The Express	244	6.7	5.7	1.0	3.2
TS	Toronto Star	334	25.3	7.8	1.0	1.5
UPI	United Press Int.	334	15.1	6.9	1.6	32.0
USA	USA Today	669	1.3	9.2	0.0	0.0
VS	Vancouver Sun	334	0.4	6.4	5.6	38.7
WP	Washington Post	548	62.7	9.4	13.7	5.1

3.2 Sources, Extraction, and Annotation

We use a network of citations between English news articles. As described above, only the content of the news articles is considered for the extraction of text-anchored links to other news articles. The network is constructed from news articles that were collected between November 1, 2015 and October 31, 2017. Some outlets were added after 2015 and are thus present for a shorter period of time. Articles that do not give or receive citations are not included. In total, the network consists of 244, 631 articles (nodes) that are connected through 367, 225 citations (edges), and can be downloaded from our website[1]. The total number of 34 news outlets includes outlets from the UK, the US, Canada, Australia, Qatar, Germany, and China. To extract the contained temporal expressions, we tag all articles with

[1] All data and code is available at https://dbs.ifi.uni-heidelberg.de/resources/data/

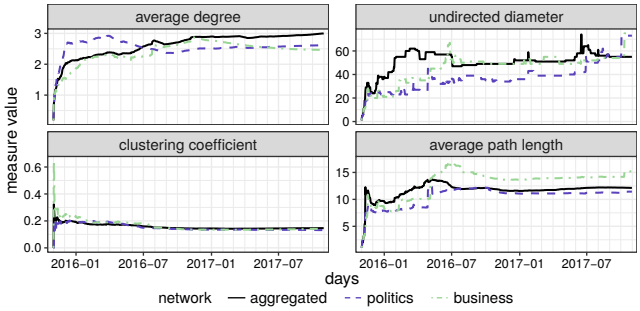

Figure 1: Evolution of network metrics for the entire network as well as the politics and business subnetworks.

HeidelTime in the news domain setting [23]. The articles contain a total of 1, 748, 813 temporal expressions of the type *date*: 41.5% have the granularity day, 19.7% the granularity month, and 38.8% the granularity year. In Table 1, we show an overview of the data set. The percentages of incoming and outgoing citations that refer to (or from) a different news outlet give an indication of the citation policy of the individual news outlets with respect to the competition.

3.3 Temporal Correlations

To obtain an impression of the temporal expressions contained in the articles, we consider the correlation of temporal expressions with day granularity to the publication dates. When comparing the temporal expressions of an article with the publication date of the article itself, we obtain a Pearson correlation of $\rho_{self} = 0.440$. If we compare the temporal expressions in an article with the publication dates of citing articles (i.e., along incoming edges in the network), this drops to $\rho_{in} = 0.400$, while the correlation with publication dates of articles at the end of outgoing edges is $\rho_{out} = 0.473$. We take this as an indication that using the correlation between temporal expressions and the dates of articles along outgoing edges is more beneficial, which conforms with the expectation that articles tend to contain temporal expressions that match the relevant dates of referenced articles. However, when compared to the much higher correlation between publication dates of articles along edges $\rho_{pub} = 0.934$, we expect the temporal information contained in the publication dates to be more useful for DCT prediction.

3.4 Evolution of Network Metrics

An interesting characteristic of evolving networks is the change in their metrics as nodes and edges are added. For many naturally occurring networks, typical characteristics are a long-tailed degree distribution, leading to a shrinking diameter [12] (i.e., a decreasing length of the longest shortest path) and an increased clustering coefficient [2] (i.e., a densification of the local neighbourhood) as the network evolves. In contrast, a news citation network for four German news outlets was observed to maintain constant clustering coefficient and constant diameter over a period of 300 days [22].

Since the sparsity of the data and the network's structure are of interest to our subsequent prediction task, we show the results of a similar analysis on the larger international news network in Figure 1. We observe that the findings hold for the larger network. While there are some spikes in the diameter, it is largely constant, as are the clustering coefficient and the average path lengths for the

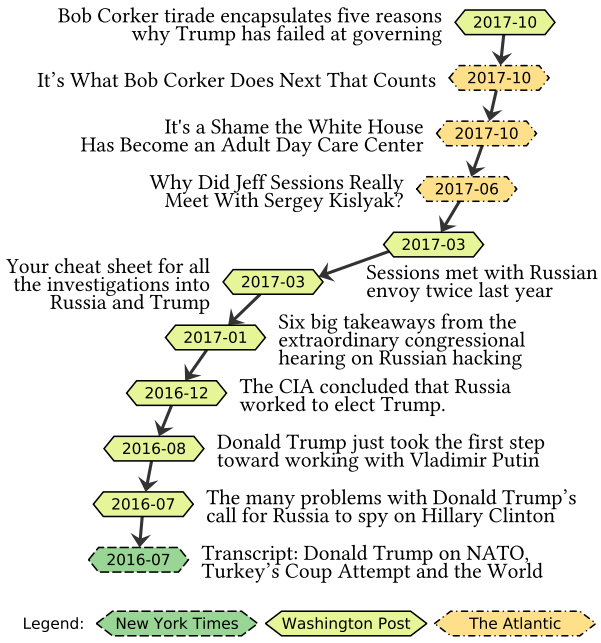

Figure 2: Example of a news citation chain concerning Russian involvement in the 2016 U.S. presidential election.

entire network. The politics subnetwork is similar and dominates the combined network, while the business subnetwork is smaller and less regular. Overall, we observe long chains of article citations that lend themselves to the exploration of evolving news stories.

3.5 Exploration of Citation Chains

The low density and the high diameter of the network suggest the emergence of long citation chains as the network evolves. Naturally, such citation chains are not only of interest for estimating article publication times, but also for investigating the spread of information in the network. While an in-depth analysis of such information propagation is beyond the scope of this paper, we show an example of a medium-length citation chain in Figure 2. As the article headlines indicate, there is a propagation of information as the story evolves over more than a year. Note that the figure shows only a single citation chain, which overlaps and intersects with other chains in the entire network. In the following, we use this network structure to derive topological and temporal network features as an aid in the prediction of publication dates.

4 PUBLICATION TIME PREDICTION

We next describe our approach used to predict article publication times by exploiting the temporal citation network.

4.1 Feature Extraction

To train the regressors, we use a set F of 27 features, which can be grouped into three categories: features derived from the topology of the citation network, features derived from the publication times of adjacent articles in the network, and features derived from temporal expressions in adjacent articles. The 28th variable is the publication time of the article itself, which we denote with T and use as response

variable in the subsequent experiments. To encode all temporal features, we use an integer value representing POSIX time.

Network topology features. To utilize the structure of the news citation network, we extract purely topological features. That is, we rely on the connectivity information of the network. For definitions and derivations of the network metrics, see [16], for example. The degree captures the most basic connectivity information, namely the number of adjacent edges. Since the network is directed, we include the outgoing degree deg_{out}, the incoming degree deg_{in}, and the undirected (total) degree deg_{all} as features for each node. As a description of the neighbourhood of a node, we utilize the undirected local clustering coefficient cc as a feature, which captures the degree to which the neighbours of a given node are interconnected. Finally, we include a number of centrality measures, namely the betweenness centrality c_{btw}, the page rank centrality c_{pr}, and the incoming and outgoing closeness centralities $c_{cl,in}$ and $c_{cl,out}$. For the computation of these network features, we use the `igraph` package [6] in R with default parameter settings.

Temporal network features. Moving beyond mere topological information, we combine the network connectivity with the publication times of adjacent articles. To this end, let T_{in} denoted the set of publication dates of articles that reference a given article v, and let T_{out} denote the publication times of articles that v references. Then, we derive a set of features from the relations between those outgoing and incoming dates. Specifically, we use the maximum and minimum publication date of articles that are referenced by article v and denote them with $max(T_{out})$ and $min(T_{out})$. We also compute the mean $\mu(T_{out})$ and standard deviation $\sigma(T_{out})$ of these publication times, along with the time span $span(T_{out}) = max(T_{out}) - min(T_{out})$ between them. Similarly, for articles that include references to v, we compute $max(T_{in})$, $min(T_{in})$, $\mu(T_{in})$, $\sigma(T_{in})$, $span(T_{in})$ from the set of their publication dates T_{in}. Intuitively, the publication date of an article should be located in the interval of referenced and referencing articles. Therefore, we also construct the set of pairwise distances between the incoming and outgoing adjacent articles as

$$Dist = \bigcup_{\substack{t_i \in T_{in} \\ t_o \in T_{out}}} t_i - t_o$$

and derive from them the minimum distance $min(Dist)$, the maximum distance $max(Dist)$, as well as the average distance $\mu(Dist)$ and the standard deviation of the distance $\sigma(Dist)$. For a conceptual visualization of these 14 features, see Figure 3.

Temporal expression features. Similar to the extraction of features from publication times, we can also consider the temporal expressions contained in adjacent articles (recall that temporal expressions within the article itself are useless prior to estimating the DCT). In the following, temporal expressions with a granularity of months or years are represented by the mean value of the interval. Based on our findings in Section 3.3, we conjecture that the temporal expressions in referenced articles are less useful for determining the DCT of the referencing article. However, the temporal expressions that are located within the text of referencing articles are likely related to the publication time of the referenced article. Thus, we denote with X_{in} the set of all temporal expressions in articles that reference a given article v. Based on this set, we derive the same

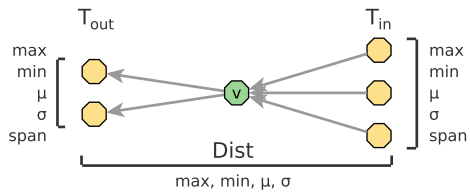

Figure 3: Conceptual overview of the temporal features.

types of features as we obtained for the publication times, namely the maximum and minimum of incoming temporal expressions $max(X_{in})$ and $min(X_{in})$, as well as their mean $\mu(X_{in})$, standard deviation $\sigma(X_{in})$, and time span $span(X_{in})$.

Feature imputation. Due to the sparseness of the network, many articles are lacking incoming or outgoing edges, which means that not all features can be computed for all articles. As a result, 30.8% of feature values are missing, which is quite substantial. Furthermore, 89.6% of the articles have at least one missing feature value. Therefore, discarding articles with incomplete feature values is not viable and we have to impute missing values. In the following, we impute values by the mean of a given feature. More involved imputation approaches are available that could potentially further improve the results, such as multiple imputation by chained equation [27]. However, given the already promising results for imputation by mean (see Section 4.3), we do not explore these here.

4.2 Regression Methods

Using the above set of 27 features, we train six different regression methods along with a baseline. In the following, we briefly introduce the methods and discuss their relevant parameters. Where necessary or beneficial, the features are standardized (i.e., shifted by their mean and normalized by their standard deviation). We use the R software environment for all implementations.

Baseline (BASE). As baseline, we include a predictor that averages the publication times of adjacent articles on incoming edges and outgoing edges. That is, we compute $T_{base} = 0.5[\mu(T_{in}) + \mu(T_{out})]$ such that the mean publication times of articles along all incoming and all outgoing edges are averaged with equal weight.

Linear regression (LR). As a first regression approach, we utilize multiple linear regression on all available features. That is, we fit a linear regression model for regression coefficients β_i as

$$T \sim \beta_0 \sum_{i=1}^{|F|} \beta_i F_i + \varepsilon$$

where ε denotes the error terms. We obtain a fit through QR factorization using the default `lm` implementation in R.

Bayesian regression (BAY). To compare the traditional linear regression to a more advanced method, we also include Bayesian regression as implemented in the `bayesreg` package based on methods by Makalic and Schmidt [14]. Specifically, we use Bayesian ridge regression with a Laplace model since the Gaussian and Student-t models yield identical results to traditional linear regression.

Random forest (RF). As a representative of decision tree learning, we train a random forest as implemented in the `randomForest` package [13], which is based on the implementation by Breiman and Cutler [3]. We set the forest size to 500 trees.

Table 2: Mean absolute error in days for predictions by the six regressors and the baseline. Shown are the values for all articles (all), articles with only incoming edges (in) or outgoing edges (out), and articles with both (in+out).

	BASE	LR	BAY	NN	RF	GB	SVM
all	66.72	60.46	59.61	26.88	24.98	**22.66**	26.19
in	88.88	66.48	87.55	34.03	32.25	**27.49**	32.29
out	87.32	59.54	40.24	32.52	30.10	**26.68**	30.77
in+out	18.68	55.45	54.95	12.62	**11.23**	12.76	14.31

Gradient boosting (GB). As a second tree-based learner, we use gradient boosting on decision trees from the `gbm` package [19]. Here, since our loss function is the mean absolute error, the Laplace distribution works best. We set the number of trees to $n = 20,000$, the shrinkage to $\lambda = 0.001$ and the tree depth to $K = 5$.

Support vector machine (SVM). For support vector machines, we utilize the package `e1071` [15], which serves as an interface to the `libsvm` library [5]. The radial kernel performs best, so we exclude the results for the linear and polynomial kernels. For training the SVM, we use ϵ-regression with a threshold of $\epsilon = 0.1$.

Neural network (NN). Given the construction of features, recurrent neural networks are not particularly applicable to the given problem (while dates along edge sequences could be exploited, the sparseness of the network is too pronounced for the extraction of sufficient training data). Thus, we use a classic feedforward neural network from the `neuralnet` package [9]. We use one node with linear output to obtain a regression model, and a single hidden layer with 14 nodes (i.e., mean number of nodes in the input and output layer). We rely on resilient backpropagation [20] for training the network with one repetition and a convergence threshold of 1.0. We increase the number of steps to 10^7 to obtain convergence.

4.3 Evaluation Results

We perform 10-fold cross validation for all regression methods. We use the mean absolute error (MAE) as evaluation metric instead of the commonly used root mean square error, since (1) giving more weight to larger errors to penalize outliers does not seem sensible, and (2) MAE is easier to interpret for temporal distances (in days).

In Table 2, we show the resulting MAE scores for all six methods on the entire data set (denoted by *all*). To analyze the impact of missing data, we also show results for subsets of the data. Specifically, we give the results for articles that only receive (*in*) or give references (*out*), and the subset of articles that both receive and give references (*in+out*). The three sets have roughly equal size in our data ($\sim$ 30%). Note that all articles have at least one incident edge or they would not be part of the network. For the entire data, the baseline performs worst with an average prediction error of two months. Linear regression and Bayesian regression are only slightly better, while the error of the neural network, SVM, and random forest regressors are less than one month. Gradient boosing performs slightly worse on the *in+out* set but best overall. All methods perform better on the smallest subset of articles that have both incoming and outgoing references, although the baseline is so good in this special case that linear regression and Bayesian regression do not outperform it. For the much harder cases of articles that only have incoming or outgoing edges, all methods outperform the baseline. Furthermore, the performances of all methods on the *in* set are slightly worse than on the *out* set, indicating that the direction of references does not play a major role. Bayesian regression is the only exception and benefits more from outgoing edges than from incoming edges. Overall, GB has the best performance.

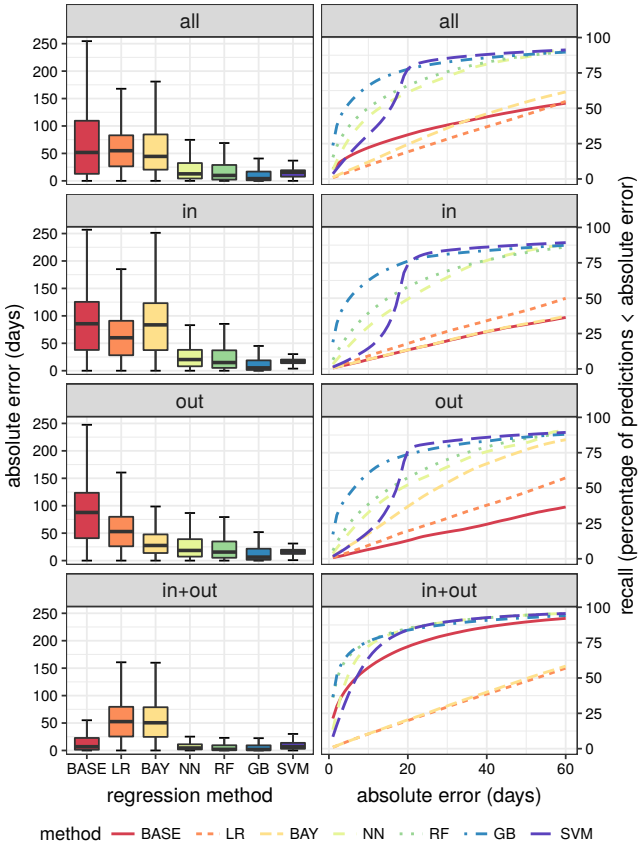

Figure 4: Results for the five regressors and the baseline, for all articles (all), those with only incoming edges (in) or outgoing edges (out), and both (in+out); left: distribution of the absolute error in days; right: recall for sliding absolute error.

To analyze the overall spread of the prediction quality, we show the distributions of the absolute error in Figure 4 (left). We find that the mean values in Table 2 correlate well with the median values and the overall distribution. The results of the SVM have a small spread but a higher median value than the RF and GB results, leading to a worse overall performance. In Figure 4 (right), we show the recall by increasing absolute error. We find that the SVM initially performs worse than GB, but peaks at an error of three weeks, where over 80% of the results are included, and then performs slightly better than gradient boosting.

4.4 Feature Importance

For an analysis of the importance of individual features, we rely on the tree-based methods, which provide the total sum of residuals that are computed in each split during the training process and allow us to measure the gain in node purity for splits on any given feature. In Figure 5, we show this feature importance obtained from the 10-fold cross validation as relative values. For RF, it is clear that

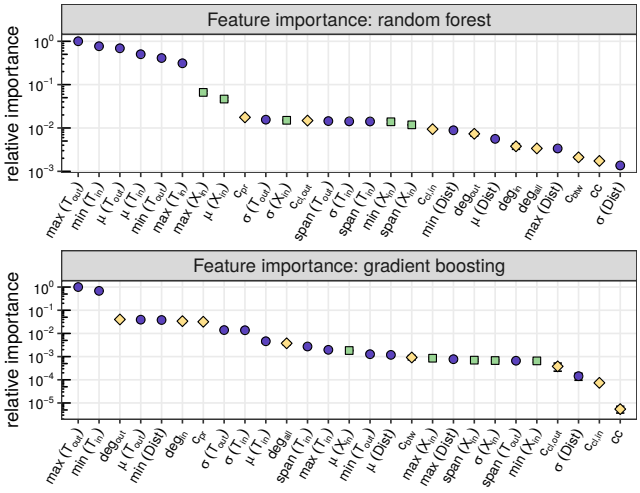

Figure 5: Relative importance for feature types: network topology (yellow), temporal expression (green), and temporal network (purple). Error bars are one standard deviation.

most features play a minor role and that six features account for the majority of selected splits, all of which are temporal network features. Two temporal expression features are the next most important, while topological features play a minor role. For GB, just two temporal network features account for the bulk of splits, which are the same as the top two features for RF. Overall, temporal expression features are less important for GB, while topological features have a higher importance, especially those that are degree-based. We see this distribution of feature importances as an indication that the temporal information contained in the local neighbourhood of articles is most valuable for DCT prediction.

5 SUMMARY AND OUTLOOK

In this paper, we created and analyzed a large-scale network of citations between English news articles covering two years. We investigated the task of predicting the document creation time of news articles from the network structure and the publication times of adjacent articles in the network as a regression problem. Despite the sparseness of the network, we found that document publication times in such a setting can be predicted reliably with an average error of slightly over three weeks. Overall, we observed the most challenging aspect to be the sparseness of the data since the predictive performance increased strongly for articles that both contain and receive references. As a result, we conjecture that denser news citation networks constructed from more news outlets stand to support better predictions. Finally, an analysis of feature importance for the two best-performing regressors showed that features derived from the network structure with the publication times of adjacent articles have the largest impact, indicating that the knowledge of the topological structure and the publication times is sufficient for obtaining high-quality predictions.

Future work. Given the individual performance profiles of the regressors, the construction of an ensemble classifier warrants further investigation. Similarly, the application of convolutional neural networks on citation chain features stands to further improve the predictive performance of the proposed approach.

REFERENCES

[1] Einat Amitay, David Carmel, Michael Herscovici, Ronny Lempel, and Aya Soffer. 2004. Trend Detection Through Temporal Link Analysis. *J. Am. Soc. Inf. Sci. Technol.* 55, 14 (Dec. 2004), 1270–1281. DOI: http://dx.doi.org/10.1002/asi.20082

[2] Béla Bollobás and Oliver M Riordan. 2003. Mathematical Results on Scale-free Random Graphs. *Handbook of Graphs and Networks: from the Genome to the Internet* (2003), 1–34.

[3] Leo Breiman. 2001. Random Forests. *Machine Learning* 45, 1 (2001), 5–32. DOI: http://dx.doi.org/10.1023/A:1010933404324

[4] Nathanael Chambers. 2012. Labeling Documents with Timestamps: Learning from Their Time Expressions. In *ACL*. http://dl.acm.org/citation.cfm?id=2390524.2390539

[5] Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A Library for Support Vector Machines. *ACM TIST* 2, 3 (2011), 27:1–27:27. DOI: http://dx.doi.org/10.1145/1961189.1961199

[6] Gabor Csardi and Tamas Nepusz. 2006. The igraph Software Package for Complex Network Research. *InterJournal, Complex Systems* 1695, 5 (2006), 1–9.

[7] Franciska M.G. de Jong, Henning Rode, and Djoerd Hiemstra. 2005. Temporal Language Models for the Disclosure of Historical Text. In *AHC*.

[8] Tao Ge, Baobao Chang, Sujian Li, and Zhifang Sui. 2013. Event-Based Time Label Propagation for Automatic Dating of News Articles. In *EMNLP*. http://www.aclweb.org/anthology/D13-1001

[9] Frauke Günther and Stefan Fritsch. 2010. neuralnet: Training of Neural Networks. *The R Journal* 2, 1 (2010), 30–38. https://journal.r-project.org/archive/2010/RJ-2010-006/index.html

[10] Nattiya Kanhabua and Kjetil Nørvåg. 2009. Using Temporal Language Models for Document Dating. In *ECML PKDD*. DOI: http://dx.doi.org/10.1007/978-3-642-04174-7_53

[11] Minkyoung Kim, Lexing Xie, and Peter Christen. 2012. Event Diffusion Patterns in Social Media. In *ICWSM*. http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/view/4595

[12] Jure Leskovec, Jon M. Kleinberg, and Christos Faloutsos. 2005. Graphs Over Time: Densification Laws, Shrinking Diameters and Possible Explanations. In *KDD*. DOI: http://dx.doi.org/10.1145/1081870.1081893

[13] Andy Liaw and Matthew Wiener. 2002. Classification and Regression by randomForest. *R News* 2, 3 (2002), 18–22. http://CRAN.R-project.org/doc/Rnews/

[14] Enes Makalic and Daniel F Schmidt. 2016. High-Dimensional Bayesian Regularised Regression with the BayesReg Package. *arXiv preprint* (2016). https://arxiv.org/abs/1611.06649

[15] David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, and Friedrich Leisch. 2017. *e1071: Misc Functions of the Department of Statistics, Probability Theory Group*. https://CRAN.R-project.org/package=e1071

[16] Mark Newman. 2010. *Networks: An Introduction*. Oxford University Press.

[17] Sérgio Nunes, Cristina Ribeiro, and Gabriel David. 2007. Using Neighbors to Date Web Documents. In *WIDM*. DOI: http://dx.doi.org/10.1145/1316902.1316924

[18] Filippo Radicchi, Santo Fortunato, and Alessandro Vespignani. 2012. Citation Networks. In *Models of Science Dynamics: Encounters Between Complexity Theory and Information Sciences*, Andrea Scharnhorst, Katy Börner, and Peter van den Besselaar (Eds.). Springer. DOI: http://dx.doi.org/10.1007/978-3-642-23068-4_7

[19] Greg Ridgeway. 2006. *gbm: Generalized Boosted Regression Models*. https://cran.r-project.org/package=gbm

[20] Martin Riedmiller. 1994. Advanced Supervised Learning in Multi-layer Perceptrons–From Backpropagation to Adaptive Learning Algorithms. *Comput Stand Interfaces* 16, 3 (1994), 265–278. DOI: http://dx.doi.org/10.1016/0920-5489(94)90017-5

[21] Hany M. SalahEldeen and Michael L. Nelson. 2013. Carbon Dating the Web: Estimating the Age of Web Resources. In *WWW Companion*. DOI: http://dx.doi.org/10.1145/2487788.2488121

[22] Andreas Spitz and Michael Gertz. 2015. Breaking the News: Extracting the Sparse Citation Network Backbone of Online News Articles. In *ASONAM*. DOI: http://dx.doi.org/10.1145/2808797.2809380

[23] Jannik Strötgen and Michael Gertz. 2013. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation* 47, 2 (2013), 269–298. DOI: http://dx.doi.org/10.1007/s10579-012-9179-y

[24] Jannik Strötgen and Michael Gertz. 2016. *Domain-Sensitive Temporal Tagging*. Morgan & Claypool.

[25] Xavier Tannier. 2014. Extracting News Web Page Creation Time with DCTFinder. In *LREC*. http://www.lrec-conf.org/proceedings/lrec2014/summaries/3.html

[26] Masashi Toyoda and Masaru Kitsuregawa. 2006. What's Really New on the Web?: Identifying New Pages from a Series of Unstable Web Snapshots. In *WWW*. DOI: http://dx.doi.org/10.1145/1135777.1135815

[27] Stef Van Buuren, Jaap PL Brand, Catharina GM Groothuis-Oudshoorn, and Donald B Rubin. 2006. Fully Conditional Specification in Multivariate Imputation. *J Stat Comput Simul* 76, 12 (2006), 1049–1064. DOI: http://dx.doi.org/10.1080/10629360600810434

[28] Yue Zhao and Claudia Hauff. 2015. Sub-document Timestamping of Web Documents. In *SIGIR*. DOI: http://dx.doi.org/10.1145/2766462.2767803