Time-aware online reputation analysis

Peetz, M.-H.

Publication date
2015
Document Version
Final published version

7

Cognitive Temporal Document Priors

Every moment of our life we retrieve information from our brain: we remember. We remember items to a certain degree, for a mentally healthy human being retrieving very recent memories is virtually effortless, while retrieving non-salient[1] memories from the past is more difficult [159]. Early research in psychology was interested in the rate at which people forget single items, such as numbers. Recently however, in psychology, researchers have become interested in how people retrieve events. Hertwig et al. [102] let users remember entities such as cities, names, and companies; entities are better remembered if they recently appeared in a major newspaper, and propose models of how people retrieve terms based on their findings. Similarly, Chessa and Murre [51, 52] record events and hits of web pages related to the event and fit models of how people remember, the so-called *retention function*. For online reputation analysis this is interesting: old events have less potential to impact the current reputation of a company. Since not all data can be manually annotated feasibly, more recent data should be favoured.

Modeling the retention of memory has a long history in psychology, resulting in a range of proposed retention functions. In information retrieval (IR), the relevance of a document depends on many factors. If we request recent documents, then how much we remember is bound to have an influence on the relevance of documents. Can we use the psychologists' models of the retention of memory as (temporal) document priors? Previous work in temporal IR has incorporated priors based on the exponential function into the ranking function [74, 75, 144, 157]—this happens to be one of the earliest functions used to model the retention of memory. But, many other such functions have been considered by psychologists to model the retention of memory—what about the potential of other retention functions as temporal document priors?

Inspired by the cognitive psychology literature on human memory and on retention functions in particular, we consider 7 temporal document priors. We propose a framework for assessing them, building on four key notions: *performance*, *parameter sensitivity*, *efficiency*, and *cognitive plausibility*, and then use this framework to assess those 7 document priors. For our experimental evaluation we make use of two (temporal) test collections: newspapers and microblogs. In particular, we answer the following questions

RQ4.1 Does a prior based on exponential decay outperform other priors using cognitive

[1] Salient memories are very emotional memories and traumatic experiences; human retrieval of such memories is markedly different from factual memories [199].

retention functions with respect to effectiveness?

RQ4.2 In how far do the proposed recency priors meet requirements, such as efficiency, performance, and plausibility?

We show that on several datasets, with different retrieval models, the exponential function as a document prior should not be the first choice. Overall, other functions, like the Weibull function, score better within our proposed framework for assessing temporal priors. The chapter is structured as follows. In Section 7.1 we give related work for models for cognitive information retention. Section 7.2 introduces the baselines and functions underlying the recency priors. The experiments are laid out in Section 7.3.2. Section 7.4 analyses the results and Section 7.5 concludes.

7.1 Memory Models

Modeling the retention of memory has been a long studied area of interest in cognitive psychology. Ebbinghaus [71] hypothesizes that retention decays exponentially and supports his hypothesis with a self-experiment. Schooler and Anderson [219] propose a

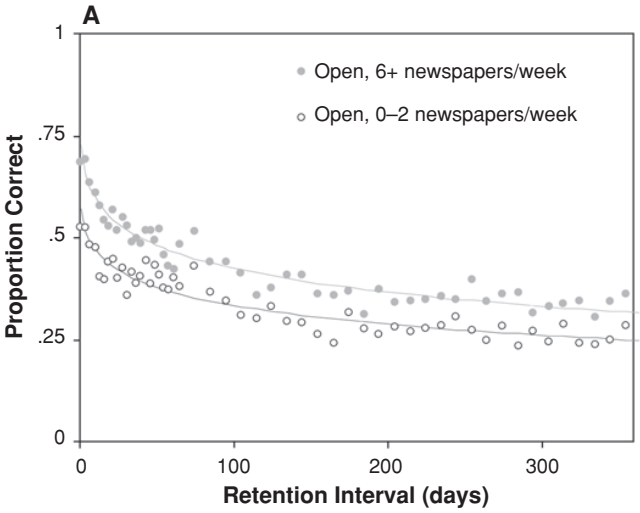


Figure 7.1: Retention curves for participants in a study on how they remembered news and fitted retention functions. Plotted separately are participants who read many newspapers (at least 6 a week) and those who read few newspapers (0–2 per week). Taken from Meeter et al. [159].

power law model for retention and learning and Rubin et al. [210] fit a power function to 100 participants. Wickens [271] analyses probability distributions for their suitability as retention models. Heathcote et al. [101] show that the exponential functions fit much better. Finally, Meeter et al. [159] perform a study with 14,000 participants and compare state-of-the-art memory models and how they fit the retention data. Over the

course of two years, participants were asked questions of the day's news and news in the past. Figure 7.1 shows how much people could remember over time. Chessa and Murre [52] use large-scale experiments to show that the Weibull function is a much better model and the power law can merely be an approximation. To model interest, Chessa and Murre [51, 52] record events and hits of web pages related to the event; interest decays similarly to memory. While this line of work mainly describes the outcome and the distributions of much we remember and forget, it is still open if we actually forget. Apart from pure failure in storing, decay theory [28] hypothesizes that memory fades away over time, while interference theory assumes that memory items compete and even though we store all items, we can only retrieve the new material [241]. In cognitive psychology we speak of memory retention models, while we speak of memory retention functions.

7.2 Methods

We introduce basic notation and well-known retrieval models into which the temporal document priors that we consider are to be integrated. We then describe several retention functions serving as temporal document priors.

We say that document D in document collection $\mathcal{D}$ has time *time*(D) and text *text*(D). Similarly, a query q has time *time*(q) and text *text*(q). We write $\delta_g(q, D)$ as the time difference between *time*(q) and *time*(D) with the granularity g. E.g., if *time*$(q') =$ 20th July, 2012 and *time*$(D') =$ 20th June, 2012, then the time difference between q' and D' is $\delta_{\text{day}}(q', D') = 30$, $\delta_{\text{month}}(q', D') = 1$, and $\delta_{\text{year}}(q', D') = 0.083$ for a granularity of a day, month, and year, respectively.

7.2.1 Baselines

In order to keep our experiments comparable with previous work, we use the query likelihood model (see Section 2.3), both as baseline and as retrieval algorithm for an initially retrieved set of documents. For smoothing we use Dirichlet smoothing (see Eq. 2.2). A variant to this baseline for recency queries has been proposed by Li and Croft [144] (see Section 2.6). Rather than having a uniform document prior $P(D)$, they use an exponential distribution as an approximation for the prior (see Eq. 7.4). We use different functions to approximate the prior.

(Temporal) Query modeling

Massoudi et al. [157] introduce a query modeling approach that aims to capture the dynamics of topics in Twitter. This model takes into account the dynamic nature of microblogging platforms: while a topic evolves, the language usage around it is expected to evolve as well. The fundamental idea is that the terms in documents closer to query submission are more likely to be a description of the query. To construct a new query model, we rank terms according to their temporal and topical relevance and select the top

k terms. Specifically,

$$\text{score}(w,q) = \log\left(\frac{|\mathcal{D}_{\text{time(q)}}|}{|\{D : w \in D, D \in \mathcal{D}_{\text{time(q)}}\}|}\right) \cdot \sum_{\{D \in \mathcal{D}_{\text{time(q)}} : w_q \in \textit{text}(q) \text{ and } w, w_q \in \textit{text}(D)\}} f(D,q,g), \tag{7.1}$$

where $f(D,q,g)$ is a retention function (introduced below) and $\mathcal{D}_{\textit{time}(q)}$ is the set of documents published before the time of query q. The original experiments were done using an exponential function (see Eq. 7.4). Standard query modeling uses a prior $f(D,q,g) = 1$. We refer to this as *query modeling*. We propose to use different priors as introduced in Section 7.2.2.

The set W_q consists of the top k terms w for query q, sorted by $\text{score}(w,q)$. The probability of term t given query q is:

$$P(w \mid q) = \begin{cases} \dfrac{\text{score}(w,q)}{\sum_{w' \in W} \text{score}(w',q)} & \text{if } w \in W, \\ 0 & \text{otherwise.} \end{cases} \tag{7.2}$$

We then use Kullback-Leibler (KL) divergence [155] to estimate the retrieval score of a document D for a query q:

$$\text{Score}(q,D) = -\sum_{w \in V} P(w \mid q) \log P(w \mid D), \tag{7.3}$$

where V is the vocabulary, i.e., the set of all terms that occur in the collection and $P(w \mid D)$ is the generative probability for a term as specified in Eq. 2.1.

7.2.2 Retention Functions

The main underlying assumption in this chapter is that in the setting of news and social media people search for something they could potentially remember. Something that is not too far away in the past. Daily news in old egypt are probably not as interesting as the news yesterday. Social chatter from last week is less interesting as social chatter from today. We assume that the prior for a time bin with respect to a the query time will be the rate at which people will probably have forgotten about the content. We will base our approach on the models found in the large-scale experiments by Chessa and Murre [52]. In the following we introduce a series of retention functions based on the models. The *memory chain models* (Eq. 7.4 and 7.5) build on the assumptions that there are different memories. The memory model introduced in Eq. 7.4 is equivalent to the exponential prior used in the IR literature. The Weibull functions (Eq. 7.6 and 7.7) are of interest to psychologists because they fit human retention behavior well. In contrast, the retention functions *linear* and *hyperbolic* (Eq. 7.9 and 7.10) have little cognitive background.

Memory chain model

The memory chain model [51] assumes a multi-store system of different levels of memory. An item is stored in one memory with the probability of μ,

$$f_{\text{MCM-1}}(D,q,g) = \mu e^{-a\delta_g(q,D)}. \tag{7.4}$$

The parameter a indicates how items are being forgotten. The function $f_{\text{MCM-1}}(D, q, g)$ is equivalent to the exponential decay in Li and Croft [144] when the two parameters (μ and a) are equal. In fact, as μ is document independent it does not change the absolute difference between document priors when used for query likelihood and $f_{\text{MCM-1}}(D, q, g)$ is essentially equal to the exponential function used in Li and Croft [144].

In the two-store system, an item is first remembered in short term memory with a strong memory decay, and later copied to long term memory. The item stays in two different memories. Each memory has a different decay parameter, so the item decays in both memories, at different rates. The overall retention function is

$$f_{\text{MCM-2}}(D, q, g) = 1 - e^{-\mu_1 \left(e^{-a_1 \delta_g(q,D)} + \frac{\mu_2}{a_2 - a_1} (e^{-a_2 \delta_g(q,D)} - e^{-a_1 \delta_g(q,D)}) \right)}, \quad (7.5)$$

where an overall exponential memory decay is assumed. The parameter μ_1 and μ_2 are the likelihood that the items are initially saved in short and long term memory, whereas a_1 and a_2 indicate the forgetting of the items. Again, t is the time bin.

Weibull function

Wickens [271] discusses different potential memory modeling functions. The prefered function is the Weibull function

$$f_{\text{basic Weibull}}(D, q, g) = \left(e^{-\frac{a\delta_g(D,q)}{d}^d} \right), \quad (7.6)$$

and its extension

$$f_{\text{extended Weibull}}(D, q, g) = b + (1 - b)\mu e^{\left(-\frac{a\delta_g(D,q)}{d} \right)^d}. \quad (7.7)$$

The parameters a and d indicate how long the item is being remembered: a indicates the overall volume of what can potentially be remembered whereas d determines the steepness of the forgetting function. The parameter μ determines the likelihood of initially storying an item, and b denotes an asymptote parameter.

Amended power function

The amended power function has also been considered as a rentention function [210]. The power function is ill-behaved between 0 and 1 and usual approximations start at 1. The *amended power function* is

$$f_{\text{power}}(D, q, g) = b + (1 - b)\mu(\delta_g(D, q) + 1)^a, \quad (7.8)$$

where a, b, and μ are the speed of decay, an asymptote parameter, and the initial learning performance.

Linear function

A very intuitive baseline is given by the linear function,

$$f_{\text{lin}}(D, q, g) = \frac{-(a \cdot \delta_g(q, D) + b)}{b}, \tag{7.9}$$

where a is the gradient and b is $\delta_g(q, \text{argmax}_{D' \in \mathcal{D}}\, \delta_g(q, D'))$. Its range is between 0 and 1 for all documents in $\mathcal{D}$.

Hyperbolic function

The hyperbolic discounting function [2] has been used to model how humans value rewards: the later the reward the less they consider the reward worth. Here,

$$f_{\text{hyp}}(D, q, g) = \frac{1}{-(1 + k \cdot \delta_g(q, D))}, \tag{7.10}$$

where k is the discounting factor.

7.3 Experimental Setup

Further, we detail a framework of requirements for priors in Section 7.3.1 and then proceed with a description of our experiments (see Section 7.3.2).

7.3.1 A Framework for Assessing Temporal Document Priors

We propose a set of four main criteria for assessing temporal document priors. Below, we evaluate how the priors follow several requirements. The most natural approach to evaluating new document priors is *performance*. *Parameter sensitivity* is an important criteria to avoid fluctuating performances. A further computational requirement is *efficiency*. Finally, we also propose *cognitive plausibility* as a criterion.

Performance

A document prior should improve the performance on a set of test queries for a collection of time-aware documents. A well-performing document prior improves on the standard evaluation measures across different collections and across different query sets. We use the *number of improved queries* as well as the *stability of effectiveness* with respect to different evaluation measures as an assessment for performance, where stability means that improved or non-decreasing performance over several test collections.

Sensitivity of parameters

A well-performing document prior is not overly sensitive with respect to parameter selection: the best parameter values for a prior are in a *region* of the parameter space and not a single value.

Table 7.1: Abbreviations of methods and their description.

Run id	Description
D	Query likelihood + smoothing
QM	Query modeling [157]
MCM-1	one store memory chain (Eq. 7.4)
MCM-2	two store memory chain (Eq. 7.5)
BW	basic Weibull (Eq. 7.6)
EW	extended Weibull (Eq. 7.7)
AP	amended power (Eq. 7.8)
L	linear (Eq. 7.9)
HD	hyperbolic discounting (Eq. 7.10)

Efficiency

Query runtime efficiency is of little importance when it comes to distinguishing between document priors: if the parameters are known, all document priors boil down to simple look ups. We use the *number of parameters* as a way of assessing the efficiency of a prior.

Cognitive plausibility

We define the cognitive plausibility of a document prior (derived from a retention function) as how well the underlying retention function fitted in large scale human experiments [159]. This conveys an experimental, but objective, view on cognitive plausibility. We also use a more subjective definition of plausibility in terms of *neurobiological background* and how far the retention function has a biological explanation.

7.3.2 Experiments

In our experiments we seek to understand in how far the recency priors introduced in Section 7.2 meet the requirements mentioned above. Since the exponential decay is the most commonly used decay function, we want to understand if it is also the most effective. For our experiments to answer the questions we use three collections: the news collections TREC-2 and TREC-{6,7,8} (see Section 3.1), and Tweets2011, a collection of tweets (see Section 3.3). We use all topics, but also focus on the subset *recent-2* for TREC-2 and TREC-{6,7,8}. To ensure comparability with previous work, we use different models for the different datasets. On the news dataset, we analyse the effect of different temporal priors on the performance of the baseline, query likelihood with Dirichlet smoothing (D). We optimize the parameters for the different priors on TREC-6 using grid search. On the Tweets2011 dataset, we analyse the effect of different temporal priors incorporated in the query modeling (QM). We do not have a training set and we evaluate using leave-one-out cross-validation. Table 7.1 lists the models whose effectiveness we examine below.

We optimize parameters with respect to mean average precision (MAP). MAP, precision at 10 (P@10), R-precision (Rprec) and mean reciprocal rank (MRR) are the quan-

Table 7.2: Parameter values for document priors based on retention functions, as fitted on the news training data and as fitted on human data (last column). For cells marked with *, the function was fitted to data with a granularity of milliseconds, otherwise months.

function	parameter	TREC-6 optimized	Tweets2011 optimized	reported values
MCM-1 (Eq. 7.4)	r	0.0013	0.2	0.00142* [210]
	μ	1	0.9	3800* [210]
MCM-2 (Eq. 7.5)	μ_1	0.7	0.3	0.49–1.29 [159]
	a_1	0.007	0.004	0.018–0.032 [159]
	μ_2	0.6	0.7	0.01–0.018 [159]
	a_2	0.4	0.4	0–0.0010
basic Weibull (Eq. 7.6)	a	0.00301	0.3–0.9	–
	d	0.087	0.4	–
extended Weibull (Eq. 7.7)	a	0.009	0.1	0.0017–0.0018 [159]
	d	0.7	0.02–0.04	0.087–0.2 [159]
	b	0.1	0.1	0–0.25 [159]
	μ	0.7	0.7	1 [159]
amended power (Eq. 7.8)	a	0.03	0.9	840.56* [210]
	b	0.01	0.02	0.33922*[210]
	μ	0.6	1	17037* [210]
linear (Eq. 7.9)	a	0.4	1.0	–
	b	0.05	1.0	–
hyperbolic (Eq. 7.10)	k	0.0007–0.0009	0.5	–

titative evaluation measures. For the Tweets2011 collection we do not use the official metric for TREC 2011 (sorting by time and then precision at 30), but the metric to be used for TREC 2012; the previously used metric proved to be sensitive to good cut-off values [4]. The parameter values found are listed in Table 7.2.

We use the Student's t-test to evaluate the significance for all but the small recency query sets from the news data. We denote significant improvements with $^{\blacktriangle}$ and $^{\triangle}$ ($p < 0.01$ and $p < 0.05$, respectively). Likewise, $^{\triangledown}$ and $^{\blacktriangledown}$ denote a decline.

7.4 Analysis

In this section we seek to understand in how far document priors based on retention functions fulfil the requirements set out above. We first examine the retrieval effectiveness of the approaches. After that we use our framework for assessing the document priors.

Table 7.3: Results on news data, TREC-7 and TREC-8. All priors are based on the baseline D, e.g.; MCM-1 refers to D+MCM-1. Significant changes w.r.t. the baseline (D) and the exponential prior (D+MCM-1). The earlier is shown in superscripts and the latter is shown in brackets.

	all queries			recency-2 queries			non-recency-2 queries		
Run	MAP	P@10	Rprec	MAP	P@10	Rprec	MAP	P@10	Rprec
D	0.2220	**0.3770**	0.2462	0.2030	**0.3667**	0.2251	0.2281	0.3803	0.2529
MCM-1	0.2223	0.3750	0.2473	0.2057$^{\triangle}$	0.3625	0.2279	0.2275	0.3789	0.2534
MCM-2	0.2253	0.3640$^{\triangle}_{\triangledown}$	0.2560	**0.2108**$^{\triangle}$	0.3542	**0.2428**$^{\blacktriangle}_{\blacktriangle}$	0.2299	0.3671$^{\triangledown}_{\triangledown}$	0.2602
BW	**0.2270**	0.3730	0.2603	0.2079$^{\triangle}_{\triangle}$	0.3625	0.2339$^{\blacktriangle}_{\triangledown}$	**0.2331**	0.3763	0.2687
EW	0.2268	0.3720	**0.2611**	0.2086$^{\triangle}$	0.3583	0.2346$^{\blacktriangle}$	0.2326	0.3763	**0.2695**
AP	0.2222	0.3760	0.2462	0.2032	**0.3667**	0.2251	0.2281	0.3789	0.2528
L	0.2157$^{\blacktriangledown}$	0.3740	0.2468	0.1855$^{\blacktriangledown}$	0.3458	0.2123	0.2253	**0.3829**	0.2577
HD	0.2224	**0.3770**	0.2462	0.2042	0.3583	0.2261	0.2281	**0.3829**	0.2525

7.4.1 Retrieval Effectiveness

We begin with an analysis of the retrieval effectiveness of the document priors. We ask:

RQ4.1 Does a prior based on exponential decay outperform other priors using cognitive retention functions with respect to effectivity?

We analyse the retrieval effectiveness of the document priors on the news data, follow-up with the microblog data and conclude with a cross-collection discussion.

News data

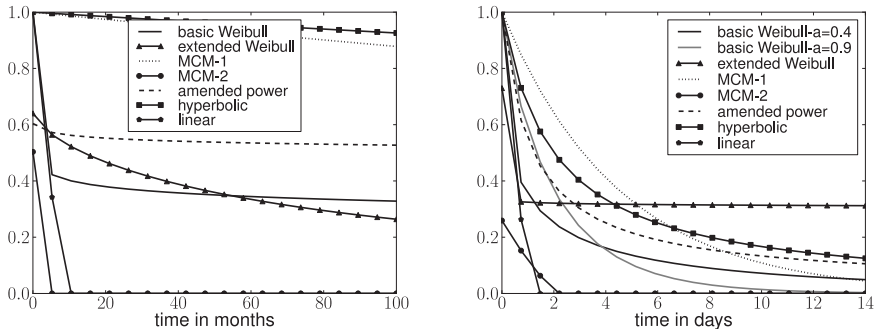
We compare the retrieval performance of our document priors on the TREC-2 and TREC-{7,8} datasets. Table 7.3 shows the results for the TREC-{7,8} dataset. We observe significant improvements (in terms of MAP and Rprec) for recency-2 queries using the basic Weibull function (BW) function as a document prior over the baseline without any prior and using MCM-1 (which is equivalent to the exponential prior [144]). We also see statistically significant improvements in terms of Rprec using the MCM-2 function, over both the baseline and using MCM-1. There are interesting differences between the two functions; first, using MCM-2 also yields the worst precision at 10 (by far), for both recency-2 and non-recency-2 queries; second, while using MCM-2 yields the highest MAP for recency-2 queries, the change is not significant. A per query analysis shows that the changes for MCM-2 are due to changes on very few queries, while for the majority of queries the average precision decreases. Using the basic Weibull function as document prior, however, has very small positive changes for more than half of the queries, thus proving to have more stable improvements.

Table 7.4 shows the results for the TREC-2 dataset. The improvements using the temporal priors over the baseline D are not significant. We can see, however, that functions that work well on the recency-2 query set (D+MCM-1, D+EW), yield significantly

Table 7.4: Results on news data, TREC-2. Indicated significance also holds for D + MCM-1. All priors are based on the baseline D, e.g.; MCM-1 refers to D+MCM-1.

	all queries			recency-2 queries			non-recency-2 queries		
Run	MAP	P@10	Rprec	MAP	P@10	Rprec	MAP	P@10	Rprec
D	0.1983	0.3430	0.2287	0.2719	0.4000	0.2913	**0.1799**	0.3287	0.2130
MCM-1	**0.1985**	0.3400	**0.2289**	0.2730	0.4050	0.2937	**0.1799**	0.3238$^{\triangledown}$	0.2127
MCM-2	0.1961	0.3330	0.2240$^{\triangledown}$	0.2731	**0.4150**	**0.2952**	0.1769$^{\blacktriangledown}$	0.3125$^{\blacktriangledown}$	0.2063$^{\blacktriangledown}$
BW	0.1984	0.3420	0.2287	0.2727	0.4050	0.2915	0.1798	0.3263	0.2130
EW	0.1983	0.3400	0.2277	**0.2749**	**0.4150**	0.2927	0.1792	0.3213$^{\triangledown}$	0.2114
AP	0.1983	**0.3430**	0.2283	0.2717	0.4050	0.2915	0.1799	**0.3275**	0.2125
L	0.1961$^{\triangledown}$	0.3410	0.2288	0.2671	0.3950	0.2902	0.1783$^{\triangledown}$	**0.3275**	**0.2135**
HD	0.1984	0.3410	0.2284	0.2730	0.4050	0.2915	0.1798	0.3250	0.2127

worse performance on the non-recency set. The only stable performance comes with the use of the functions MCM-1 and basic Weibull. We can see that using BW as a



(a) TREC-6 (b) Tweets2011

Figure 7.2: The temporal document prior instantiated with parameters optimised on different datasets. The x-axis shows the weight of the prior.

document prior improves the average precision of few recency queries, without decreasing the average precision of the other recency queries very much. More importantly, though, it improves the average precision of the recency-2 queries without harming the non-recency-2 queries.

Figure 7.2a shows the slopes of the different document priors. The similarity between MCM-2 and basic Weibull is apparent, both drop to a more or less stable function at the same time. The basic Weibull function, however, features a more gradual change. We also find that the hyperbolic and MCM-1 functions are very similar. The two functions that have a very similar slope to the basic Weibull are the amended power and the extended Weibull, but using them does not change the performance much. The main

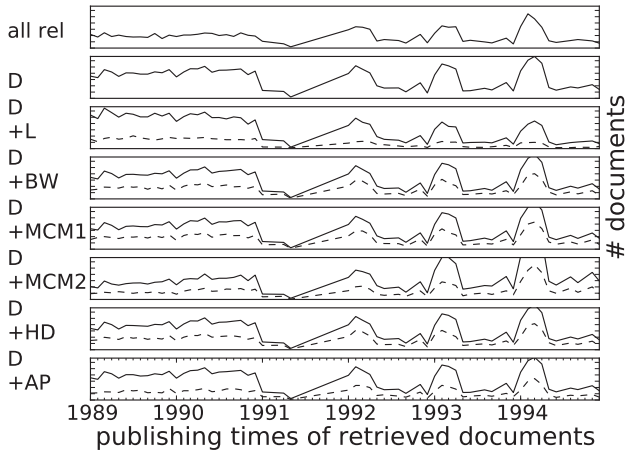


Figure 7.3: Distribution of retrieved (cut-off: 100) documents. The solid line is the distribution for all documents, the dashed line for documents retrieved for queries where improvements could be found.

difference of the slope of the functions to the slope of the basic Weibull is close to 0: the steeper the function at the beginning, the better the performance.

Figure 7.3 shows the temporal distribution of the top 100 retrieved documents for different approaches on the TREC-{7,8} test set. The topmost distribution shows the distribution for all relevant documents, which has only very few documents old documents. The second distribution is the distribution for the baseline, D. This baseline ranks older documents high. Using a linear retention function as document prior (D+L), the system retrieves even more old documents and fewer recent documents and it does not outperform the baseline for queries with recent documents. The distribution for D+MCM2 is the opposite and performs well for very recent queries, while D+MCM1 and D+BW reduce the number of old retrieved documents; thus performing fairly well on queries with old documents, and retrieve recent documents.

Microblog data

We compare the retrieval performance of the different priors on the Tweets2011 dataset. Table 7.5 shows the results for the Tweets2011 dataset. Query modeling (QM) with the MCM-1 function does not yield significant improvements. QM with basic Weibull (BW), amended power (AP), linear (L) and hyperbolic discounting (HD) does yield significant improvements in the mean reciprocal rank over the baseline QM. The increase is up to 15% for AP and BW. The MAP improves as well, but not significantly. Filtering improves the results for all approaches and while MRR increases by more than 7%, this is not significant. We can see similar effects on the filtered results: the prior does therefore not have the role of a filter.

Table 7.5: Results on microblog data, Tweets2011.

	unfiltered			filtered		
Run	MAP	P@10	MRR	MAP	P@10	MRR
QL	0.2731	0.3898	0.6133	0.2873	**0.5408**	0.7264
QM	0.2965	0.4061	0.6624	**0.3140**	0.5367	0.7559
QM+MCM-1	0.3101	0.4143	0.7682	0.3062	0.5306	0.7944
QM+MCM-2	0.2903	0.4102	0.7192	0.2912	0.5265	0.7675
QM+BW	0.3058	**0.4286**	**0.7801**$^{\Delta}$	0.3057	**0.5408**	0.7971
QM+EW	0.3038	0.4224	0.7251	0.3024	0.5224	0.7644
QM+AP	0.3100	0.4327	**0.7801**$^{\Delta}$	0.3103	**0.5408**	0.8046
QM+L	**0.3129**	0.4245	0.7700$^{\Delta}$	0.3082	0.5286	**0.8144**
QM+HD	0.3080	0.4286	0.7698$^{\Delta}$	0.3081	**0.5408**	0.7944

Table 7.4 shows a query level comparisons of the reciprocal rank between QM and QM+BW, for filtered and unfiltered queries. The comparisons are similar for the other functions. We have less queries with an increase in reciprocal rank in the filtered case.

Figure 7.2b shows the slope of the different functions for the optimized parameters. We can see that the functions that help significantly are the functions that share the same rapid decrease on the first day with a continuous, slower, decrease on the second and third day. For the other functions, on the one hand MCM-2 decreases similarly on the first day, but not on the following days: QM+MCM-2 even decreases the MAP and P@10. MCM-1 decreases slowly and continues to decrease instead of settling. The changes in performance with respect to the metrics used are therefore not as visible as for example using QM-HD: here, the slope of HD decreases similarly to MCM-1, but then settles, while MCM-1 continues to fall. Queries for which the HD function increases the average precision, are queries that were cast in the second week of the collection period with more days of tweets to return and to process. QM+BW and QM+AP display significant increases in MRR, but neither of them decreases MAP and P@10; the two models have a very similar slope.

7.4.2 Assessing the Document Priors

We now step back and assess the temporal document priors based on the framework introduced in Section 7.3. We ask:

RQ4.2 In how far do the proposed recency priors meet requirements, such as efficiency, performance, and plausibility?

We look at the performance, parameter sensitivity, efficiency, and cognitive plausibility.

Performance

As described above, using the BW retention function as prior performs significantly better, better, or similar to MCM-1 over three data sets. Other retention functions either do

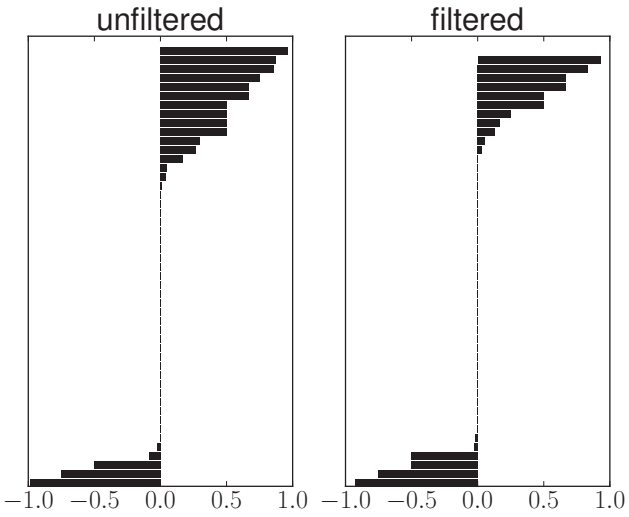


Figure 7.4: Per query comparision (y-axis) of the difference in the reciprocal rank (x-axis) between QM and QM+BW.

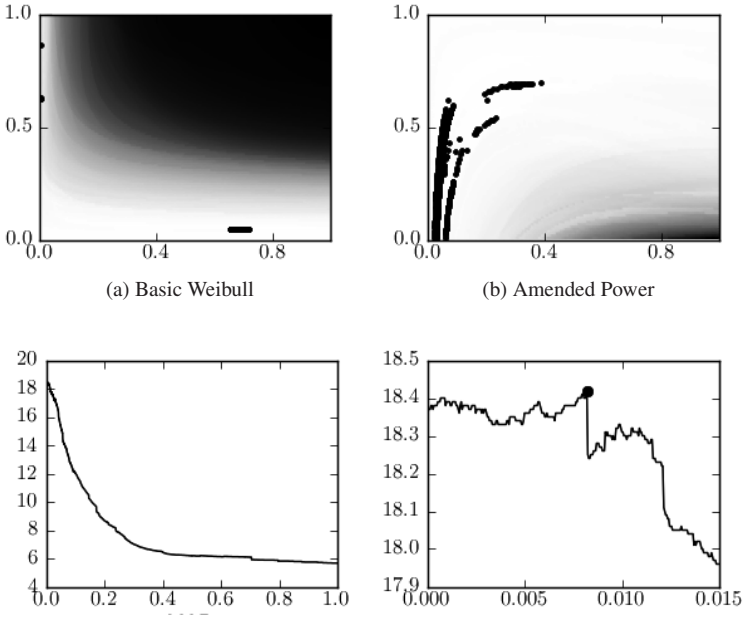
not show significant improvements or they improve on one subset while they decrease the performance on others. On a query level, BW, EW, and HD improve the greatest number of queries over MCM-1 and as Figure 7.3 shows, all three priors lead to retrieving more recent documents.

Parameter sensitivity

We first examine the issue of parameter sensitivity on news data. Figure 7.5 shows heatmaps for the different functions for parameter optimisation TREC-6. We can see in Figure 7.5d that D+MCM-1 is very unstable with respect to the optimal value for r, especially when we look at the surrounding parameters. The models D+BW and D+AP have more optimal points and are more stable with respect to those points. We observe similar effects for D+EW. When we examine parameter sensitivity on Tweets2011, we look at the optimal parameters selected for each fold in a cross-validation. We find stable parameters for all priors but the Weibull function. The Weibull function fluctuates between 0.3 and 0.4, with one exception being 0.9 (see Figure 7.2b): the fluctuation is not very strong and shows that this is not only a locally best parameter, but that there is a stable parameter subspace. However, as Efron [73] points out, the recency of the information need for the Tweets2011 varies wildly.

Efficiency

The only difference in efficiency for using the different priors is the number of parameters needed for prior optimization. A parameter sweep for four parameters (for MCM-2 and



(a) Basic Weibull

(b) Amended Power

(c) MCM-1

(d) MCM-1, subset

Figure 7.5: Optimisation of parameters, for MAP. For Figure 7.5a and 7.5b, the lighter the color, the higher the MAP. Black dots indicate the parameter combination with highest MAP.

EW) is feasible but time-consuming: for a prior that is part of a system with its own parameters, the minimal number of parameters (MCM-1, BW, L, and HD) should be optimized.

Cognitive plausibility

Previous work [159] fitted retention functions to how participants remember news (see Figure 7.1). They report that the MCM-2 and EW functions fit best while MCM-1, as a less general case of MCM-2, obviously fits worse. Chessa and Murre [52] find that the AP retention function does not fit well enough to be more than an approximation. To the best of our knowledge, the linear and hyperbolic discounting function were not fitted on retention data.

7.4.3 Discussion

Table 7.6 summarizes how the different priors fulfill the requirements listed in Section 7.3.1. Priors using the BW, AP, and HD retention functions show stable performance across different collections, on a query level as well as on a general level, with BW performing well and being stable. We find that all three functions have a stable parameter

Table 7.6: Assessing temporal document priors; # improved queries is w.r.t. MCM-1.

Condition	MCM-1	MCM-2	BW	EW	AP	L	HD
# improved queries (recency-2)	n/a	14 (58%)	5 (20%)	16 (67%)	5 (20%)	2 (8%)	6 (25%)
# improved queries (non-recency-2)	n/a	27 (35%)	35 (46%)	26 (34%)	38 (50%)	36 (47 %)	33 (43%)
# improved queries (Tweets2011)	n/a	16 (32%)	17 (34%)	22 (44%)	0 (0%)	17 (34 %)	21 (42%)
MAP	+	–	+	0	0	–	0
P10	–	–	0	–	0	0	0
Rprec	0	±	+	±	0	0	0
MRR	0	0	+	0	+	+	+
Sensitivity of parameters	–	–	+	–	+	+	+
Efficiency: # parameters	2	4	2	4	3	2	1
Plausibility: fits human behaviour	+	++	+	++	+	n/a	n/a
Plausibility: neurobiological explanation	+	+	–	+	–	–	–

selection process for at least the news dataset. However, AP with three parameters is too inefficient, while BW and HD with two and one parameter converge to a result much faster. From a cognitive psychological perspective, we know that BW has a neurobiological explanation and fits humans fairly well. The exponential function (MCM-1) as prior does not fulfil the requirements as well as other functions. This prior does have good results, but is not particularly stable when it comes to parameter optimisation. Furthermore, the significant results from the news dataset do not carry over to the microblog dataset. Based on this assessment, we propose to use the basic Weibull retention function for temporal document priors.

7.5 Conclusion

The goal of this chapter was to understand the feasibility of memory retention functions as recency priors to retrieve recent documents. Answering **RQ4.1**, we first looked at the effectiveness of the document priors. We showed how functions with a cognitive

motivation yield similar, if not significantly better results than others on news and microblog datasets. In answer to **RQ4.2**, we introduced different requirements a recency prior should fulfill based on effectiveness, parameter sensitivity, efficiency, and plausibility in a psychological sense. With respect to all priors, the Weibull function in particular, was found to be stable, easy to optimize, and motivated by psychological experiments, therefore scoring best in the requirement framework. We found the frequently used exponential prior [144] to be inferiour.

The findings are timely for the age of social media data where old media becomes more and irrelevant. The findings are therefore interesting for researchers working on new models incorporating recency priors, be it for information retrieval, filtering, or topic modeling. Researchers working on evaluation can find the findings interesting since annotation of older documents will, following the findings from [159], not be as accurate. Future work in evaluation can use cognitive temporal priors as a model for how available certain news events are in the annotators mind. We believe that the memory functions are personal and parameters need to be fit to specific users. Future models adapting parameters of the models to specific users may prove to be more accurate.