Overview of the CLEF-2023 LongEval Lab on Longitudinal Evaluation of Model Performance

Rabab Alkhalifa[1,2][0000-0002-2875-5400], Iman Bilal[3][0000-0002-7972-9085], Hsuvas Borkakoty[4][0000-0003-3262-0127], Jose Camacho-Collados[4][0000-0003-1618-7239], Romain Deveaud[6][0000-0003-2676-7405], Alaa El-Ebshihy[9][0000-0001-6644-2360], Luis Espinosa-Anke[4,12][0000-0001-6830-9176], Gabriela Gonzalez-Saez[7][0000-0003-0878-5263], Petra Galuščáková[7][0000-0001-6328-7131], Lorraine Goeuriot[7][0000-0001-7491-1980], Elena Kochkina[1,5][0000-0003-0691-3647], Maria Liakata[1,3,5][0000-0001-5765-0416], Daniel Loureiro[4][0000-0001-5134-360X], Philippe Mulhem[7][0000-0002-3245-6462], Florina Piroi[9][0000-0001-7584-6439], Martin Popel[10][0000-0002-3628-8419], Christophe Servan[6,11][0000-0003-2306-7075], Harish Tayyar Madabushi[8][0000-0001-5260-3653], and Arkaitz Zubiaga[1][0000-0003-4583-3623]

[1] Queen Mary University of London
[2] Imam Abdulrahman Bin Faisal University, SA
[3] University of Warwick, UK
[4] Cardiff University, UK
[5] Alan Turing Institute, UK
[6] Qwant, France
[7] Univ. Grenoble Alpes, CNRS, Grenoble INP (Institute of Engineering Univ. Grenoble Alpes.), LIG, Grenoble, France
[8] University of Bath, UK
[9] Research Studios Austria, Data Science Studio, Vienna, AT
[10] Charles University, Prague, Czech Republic
[11] Paris-Saclay University, CNRS, LISN, France
[12] AMPLYFI, UK

Abstract. We describe the first edition of the LongEval CLEF 2023 shared task. This lab evaluates the temporal persistence of Information Retrieval (IR) systems and Text Classifiers. Task 1 requires IR systems to run on corpora acquired at several timestamps, and evaluates the drop in system quality (NDCG) along these timestamps. Task 2 tackles binary sentiment classification at different points in time, and evaluates the performance drop for different temporal gaps. Overall, 37 teams registered for Task 1 and 25 for Task 2. Ultimately, 14 and 4 teams participated in Task 1 and Task 2, respectively.

Keywords: Evaluation · Temporal Persistence · Temporal Generalisability

1 Introduction

Datasets collected across different time periods can vary in several aspects, including the language used, the data format, as well as other structural changes. Time is however a dimension that is often overlooked when conducting experiments with static datasets. As recent research has demonstrated, however, models trained on data pertaining to a particular time period struggle to keep their performance levels when applied on test data that is distant in time. This has been shown to be the case for information retrieval (IR) systems as well as for text classification models [3].

With the aim of tackling this challenge of making models persistent over time, the objective of the LongEval lab is twofold: (i) to explore the extent to which the evolution of evaluation datasets deteriorates performance of information retrieval and classification systems, and (ii) to propose improved methods that mitigate performance drop by making models more robust over time.

The LongEval lab took place as part of the Conference and Labs of the Evaluation Forum (CLEF) 2023, and consisted in two separate tasks: (i) Task 1, focused on information retrieval, and (ii) Task 2, focused on text classification for sentiment analysis. Both tasks provided labeled datasets enabling analysis and evaluation of models over longitudinally evolving data.

In what follows, we describe the datasets, experiment settings as well as final results for each of these two tasks.

2 Task 1 - Retrieval

The goal of the retrieval task is to explore the effect of changes in datasets on retrieval of text documents. More specifically, we focus on a setup in which the datasets are evolving. This means, that one dataset can be acquired from another by adding, removing (and replacing) a limited number of documents and queries. We explore two main scenarios and the setup of the task thus reflects the details of these two problems.

A single system in an evolving setup
We explore how one selected system behaves if we evaluate it using several collections, which evolve across the time. Specifically, we explore the effect of changes in datasets on retrieval performances in a **Web search** domain. In this domain, the documents, queries and also the perception of relevance naturally continuously evolves and Web search engines need to deal with this situation. The evaluation in this scenario is thus very specific and should take into account the evolving nature of the data. Evaluation should ideally reflect the changes in the collection and especially signal substantial changes that could lead to performance drop.This would allow to re-train the search engine model, exactly when it is really needed, and enable much more efficient overall training.

This problem emerges also with the popularity of neural networks. The stability of the performance of the neural networks seems to be lower than in the case of the statistical model. Moreover, the performance strongly depends on the data used for training the neural model. One objective of the task is to explore the behavior of the neural system in the evolving data scenario.

Comparison of multiple systems in an evolving setup
While in the first point, we explore a single system, comparison of this systems with multiple systems across evolving collections, should provide more information about systems stability and robustness.

2.1 Description of the task

The task datasets were created over sequential time periods, which allows doing observations at different time stamps t, and most importantly, comparing the performance across different time stamps t and t'. Two sub-tasks are organized as follows:
A) Short-term (ST) Persistence task that aim to assess the performance difference between t and t' when t' occurs right after or shortly after t
B) Long-term (LT) Persistence task that aim to examine the performance difference between two t and t'', when t'' occurs several months after t (and thus $|t'' - t| > |t' - t|$).
In addition to this, we provide Within-time (WT) dataset, which contains the same documents (but different queries) as the training data. This data are used as a control group and applied to measure a change against the training data.

2.2 Dataset

Data for this task were provided by the French search engine Qwant. They consist of the queries issued by the users of this search engine, cleaned Web documents, which were 1) selected to correspond to the queries, and 2) to add additional noise, and relevance judgments, which were created using a click model. The dataset is fully described in [5]. We provided training data, which included 672 train queries, with corresponding 9,656 assessments and 1,570,734 Web pages. In addition to this, the training data included the 98 heldout WT queries. All training and heldout data were collected during June 2022. Test data were split into two collections, each corresponding to a single sub-task. The data for the short-term persistence sub-task was collected over July 2022 and this dataset contains 1,593,376 documents and 882 queries. The data for the long-term persistence sub-task was collected over September 2022 and this dataset consists of 1,081,334 documents and 923 queries. All the datasets are freely available at Lindat/Clarin. As the data were initially collected by French search engine and are all in French, we also provide automatic English translations of both queries and documents.

Though online evaluation is more frequent in Web search scenarios, we focus on offline evaluation, which allows us to make the collection re-usable. However, we use two different relevance judgments: the judgments acquired by the click model, based on the raw clicks of the users; and manual relevance judgment on a pooled subset. This allows us to interconnect the advantages of offline and online evaluation approaches. As the manual evaluations are ongoing, in this paper we only report the relevance judgments acquired from the click model.

For evaluating both subtasks, we use the NDCG measure (calculated for each dataset), as well as the drop between the ST and LT collection against the training data (WT collection).

2.3 Submissions

In total 14 teams submitted their systems to the Retrieval task. 12 of these teams submitted the results into both Short-term and Long-term retrieval sub-tasks, two teams only submitted the results for the Short-term retrieval sub-tasks. As per the requirements, all participating teams needed to submit their systems also on the within-time dataset, which was created at the same dataframe as the training data, which allows measuring relative drop between the datasets. All teams, except one, which submitted 4 systems, decided to submit 5 systems. Together, with 4 baseline runs provided by the Université Grenoble Alpes (marked as UGA), this creates a pool of 73 systems available on the within-time (WT, corresponding to the Heldout queries runs on the Train corpus) and short-term (ST) collections and 63 systems available on the long-term collection.

2.4 Absolute Scores

The overview of NDCG and MAP scores for each submitted run on different datasets (WT, ST, LT) is presented in Table 1. In this table, one column indicates, for each run, which language was used (English, French, or both), whether any neural approach (yes/no) was involved, and whether a combination of several approaches (yes/no) or a single approach was used.

From Table 1, we see that the systems which are the best for the WT data are also among the top for the ST and LT datasets. For instance, the best system in the WT according to the NDCG measure (FADERIC_Fr-BM25-S50-LS-S-F-SC-R20W6), is ranked best also for ST, and considering the systems that did get non-zero evaluation for the two tasks, the best system for NDCG in WT, SQUID_SEARCHERAI, is also the best on ST and LT datasets. This finding does not hold for the MAP measures: considering the systems that participated to the two tasks, the best system for MAP in WT, CLOSE_SBERT_BM25, is the second best on the ST dataset and the fourth best on the LT dataset. An explanation may come from the fact that the NDCG emphasizes on the top ranked documents of the runs.

We describe now the methods used in the top-3 runs, according to the NDCG evaluation measure, for each WT, ST and LT. For the WT Dataset Heldout queries, the top systems are:

1. CLOSE_SBERT_BM25 from the CLOSE team: The system uses query variant generated from GPT using dedicated prompts, and applies sentence BERT to rerank the initial BM25 results.

2. gwca_lightstem-phrase-qexp from de GWCA team: this systems uses a French stoplist and stemmer, a query expression is composed of the original text, phrases extracted from the query, and text generated using GPT 3.5.
3. SQUID_SEARCHERAI from the Squid team: this systems relies on Lucene indexing and searcher on French documents and queries. It uses several fields for the documents (title/url/body) with different boost values, and expands the queries with synonyms from GPT 3.5.

For the ST Dataset, the top-3 systems are:

1. FADERIC_Fr-BM25-S50-LS-S-F-SC-R20W6 from the FADERIC team. The matching is based on BM25, fine-tuned on the training set. The query processing use the Lucene *fuzzy* matching, able to allow partial match of words, and integrate synomyms expansion. A reranking fuses linearly the BM25 scores and BERT for the 20 top BM25 documents. Though the runs from the FADERIC team achieve the highest NDCG scores on the ST collection, unfortunately the scores achieved on the LT collection is zero, presumably due to an error.
2. FADERIC_Fr-BM25-S50-LS-S-F-R30 from the FADERIC team. This run is similar to the one above, the differences rely on the number of document reranked (here 30) and a different weight of BM25 score in the linear combination.
3. SQUID_SEARCHERAI from the Squid team, already described above.

For the LT Dataset, the top-3 systems are:

1. CLOSE_SBERT_BM25 from the CLOSE team, already described;
2. SQUID_W2V from the Squid team: this system relies on Lucene indexing and searcher on french documents and queries. It uses several fields for the documents (title/url/body) with different boost values, and expands the queries with word2Vec similar terms.
3. SQUID_SEARCHERAI from the Squid team, already described above.

Thus, the best approaches all rely to some extent on query expansion techniques, and integrate at one point or another embeddings or Large Language Models. The best results use French documents and queries. The effect of the translation provided by the lab has a clear impact. This remark is exemplified by the *UGA* baselines: the UGA_BM25_French outperforms the UGA_BM25_English default, and similarly the reranking using T5 French run (UGA_T5_French) outperforms its English counterpart (UGA_T5_English).

Considering the Figures **??** and **??**, we see that the shape of the distribution of the NDCG values are similar for the WT and ST datasets. However, the best systems have higher performances on WT than on ST: 13 runs on the WT dataset are above 0.4, while only 7 on the ST dataset.

Table 1: NDCG and MAP scores for three test datasets (WT, ST, LT). Results are sorted according to the NDCG scores achieved on the ST dataset.

				NDCG			MAP		
System	**Neural**	**Comb.**	**Language**	**WT**	**ST**	**LT**	**WT**	**ST**	**LT**
FADERIC_Fr-BM25-S50-LS-S-F-SC-R20W6	yes	no	French	0.4169	0.4239	0	0.2474	0.2665	0
FADERIC_Fr-BM25-S50-LS-S-F-R30	yes	no	French	0.4147	0.4145	0	0.2416	0.2546	0
SQUID_SEARCHERAI	yes	no	French	0.4279	0.4141	0.4177	0.2594	0.2554	0.2473
CLOSE_SBERT_BM25	yes	yes	French	0.4318	0.4128	0.4139	0.2675	0.2531	0.2432
gwca_lightstem-phrase-qexp	no	no	French	0.4294	0.4114	0.4161	0.2524	0.2475	0.2453
SQUID_W2V	yes	no	French	0.4232	0.4106	0.4174	0.2583	0.2497	0.2444
CLOSE_RERANKING	yes	yes	French	0.4166	0.4068	0.4062	0.2595	0.2508	0.2383
FADERIC_Fr-BM25-S50-LS-S-F-SC	no	no	French	0.4079	0.4034	0.4091	0.2376	0.2412	0.2384
FADERIC_Fr-BM25T-S50-LS-S-F	no	no	French	0.4044	0.4034	0.4071	0.2324	0.2414	0.235
SQUID_BasicSearcher	no	no	French	0.4149	0.3998	0.411	0.2522	0.2439	0.2425
SQUID_W2VRerank	yes	no	French	0.4154	0.3997	0.4105	0.2538	0.2442	0.242
gwca_lightstem-phrase	no	no	French	0.4052	0.3992	0.3988	0.2303	0.2375	0.2297
DARDS_BM25FRENCHBASE	no	no	French	0.3843	0.3924	0.3916	0.2083	0.2291	0.2207
semicolon_frenchAnalyzerFrStopWord	no	no	French	0.3869	0.3897		0.21	0.2273	
semicolon_frenchAnalyzerFrStopNum	no	no	French	0.3861	0.3895		0.2086	0.2277	
DARDS_BM25FRENCHBOOSTURL	no	no	French	0.3859	0.3866	0.3945	0.2151	0.2241	0.2243
gwca_lightstem-phrase-qexp-rerank3f	no	no	French	0.3872	0.3863	0.3942	0.2099	0.216	0.2168
gwca_lightstem-phrase-qexp-rerank2f	no	no	French	0.4059	0.3833	0.3905	0.2302	0.2117	0.2131
RAFJAM_BasicRuns	no	no	French	0.374	0.3804	0.3807	0.2018	0.2207	0.2123
gwca_word2vec-nostem	no	no	French	0.3843	0.3801	0.384	0.2083	0.2205	0.2176
CLOSE_QUEREXPANSION	yes	no	French	0.3725	0.3795	0.3736	0.2029	0.2213	0.2062
DARDS_BM25FRENCHRERANK100	no	no	French	0.3755	0.3756	0.3758	0.1982	0.2075	0.202
UGA_T5_French	yes	yes	French	0.3757	0.3717	0.3801	0.2223	0.2209	0.2207
SQUID_BOOST	no	no	French	0.3586	0.3693	0.3736	0.2024	0.2243	0.2172
DARDS_BM25FRENCHSPAM	no	no	French	0.3605	0.368	0.3643	0.1916	0.2126	0.2019
UGA_BM25_French	no	no	French	0.354	0.3541	0.3526	0.1904	0.2027	0.1936
seupd2223-JIHUMING-10_fr_fr_5gram	no	no	French, English	0.3413	0.3447	0.3533	0.1788	0.1926	0.192
seupd2223-JIHUMING-09_fr_fr_4gram	no	no	French, English	0.3364	0.3423	0.348	0.1763	0.1911	0.1888
seupd2223-hiball_BERT	yes	yes	English	0.3119	0.3418		0.1732	0.1991	
seupd2223-JIHUMING-08_fr_fr_3gram	no	no	French, English	0.3307	0.3384	0.3454	0.1725	0.1893	0.1881
seupd2223-JIHUMING-07_fr_fr	no	no	French, English	0.3271	0.3367	0.3443	0.1746	0.1883	0.1878
RAFJAM_PseudoRelQERuns	no	no	French	0.3516	0.3355	0.349	0.1971	0.1843	0.1872
FADERIC_En-BM25-S50-KS-S-F-SP-R30	yes	no	English	0.3031	0.3296	0.3262	0.1626	0.1931	0.1809
RAFJAM_SynQERuns	no	no	French	0.3193	0.3295	0.3231	0.1614	0.1876	0.1719
CLOSE_RERANKING_ENGLISH	yes	yes	English	0.3113	0.3285	0.3373	0.1822	0.1941	0.192
IRC_BM25+monoT5	yes	yes	English	0.3034	0.3256	0.3376	0.1642	0.19	0.1895
UGA_T5_English	yes	yes	English	0.2886	0.3202	0.3347	0.1576	0.1863	0.1936
RAFJAM_AllQERuns	no	no	French	0.3209	0.3172	0.3138	0.1652	0.1785	0.1676
IRC_BM25+colBERT	yes	yes	English	0.2883	0.3132	0.3209	0.1551	0.1769	0.1736
IRC_d2q+BM25	yes	no	English	0.2746	0.3072	0.3211	0.1347	0.168	0.1736
DARDS_BM25TRANSLATEDQUERIES	no	no	French, English	0.3072	0.304	0.3182	0.1525	0.1587	0.1644
semicolon_fusedRankAllEnglish	no	yes	English	0.2921	0.3032		0.1452	0.1608	
seupd2223-JIHUMING-12_fr_fr_4gram_ner	no	no	French, English	0.2868	0.298	0.3046	0.1369	0.1468	0.1433
IRC_E5_base	yes	no	English	0.2891	0.297	0.3131	0.1629	0.1599	0.1661
seupd2223-hiball_BASELINE	no	no	English	0.279	0.2955		0.1363	0.1576	
soup_kml	no	no	English	0.2705	0.2941	0.3042	0.1304	0.1559	0.1567
soup_kbase	no	no	English	0.2693	0.294	0.3021	0.1303	0.1551	0.1548
IRC_RRF(BM25+Bo1-XSqrA_M-PL2)	no	yes	English	0.2842	0.2939	0.3068	0.1355	0.1516	0.1557
soup_kngml	no	no	English	0.2698	0.2939	0.3039	0.1297	0.1558	0.1565
semicolon_Ngram34	no	no	English	0.2868	0.2938		0.1441	0.1557	
semicolon_porter2-1p4-eng	no	no	English	0.2739	0.2912		0.1303	0.1516	
soup_lng	no	no	English	0.2714	0.2899	0.2986	0.1338	0.1535	0.1526
HIBALL_AI-MERGED	no	no	English	0.2652	0.2887		0.1255	0.1506	
UGA_BM25_English	no	no	English	0.2689	0.2873	0.2992	0.1326	0.151	0.1536
seupd2223-hiball_RRF60	no	no	English	0.2664	0.2866		0.1247	0.1462	
soup_kmls	no	no	English	0.2739	0.2862	0.2988	0.1331	0.1492	0.152
QEVALS_LMDirichlet	no	no	French	0.2896	0.2819	0.2805	0.1572	0.1684	0.1633
QEVALS_BM25DFLT	no	no	French	0.2999	0.2806	0.285	0.1688	0.1694	0.1687
ows-bm25-10-variants-prompt-2	no	yes	English	0.256	0.2792	0.2872	0.1225	0.1432	0.1432
ows-pl2-10-variants-prompt-2	no	yes	English	0.2636	0.2776	0.2881	0.1285	0.1381	0.1393
QEVALS_BM25CSTM	no	no	French	0.2966	0.2776	0.2845	0.1653	0.1661	0.1681
ows-bm25-5-variants-prompt-2	no	yes	English	0.2556	0.2762	0.2838	0.1243	0.1401	0.1389
QEVALS_IB	no	no	French	0.3009	0.276	0.2833	0.1763	0.1634	0.1664
ows-lgd-10-variants-prompt-2	no	yes	English	0.2662	0.2759	0.2875	0.1275	0.1364	0.1384
ows-pl2-5-variants-prompt-2	no	yes	English	0.2631	0.2759	0.2876	0.1303	0.136	0.139
QEVALS_DFR	no	no	French	0.2976	0.2746	0.2824	0.1686	0.1626	0.1659
CLOSE_JSCLEANER_BM25	no	no	English	0.2647	0.2694	0.2803	0.1286	0.141	0.1419
NEON_1b	no	no	English	0.2269	0.2294	0.243	0.1338	0.139	0.1478
NEON_3b	no	no	English	0.2017	0.226	0.2387	0.1226	0.1384	0.1442
NEON_1a	no	no	English	0.2201	0.2241	0.2393	0.1287	0.1356	0.1446
NEON_2br	no	no	English	0.2177	0.2219	0.2282	0.1279	0.1319	0.1351
NEON_4b	no	no	English	0.2054	0.2187	0.2282	0.1213	0.1324	0.1351
HIBALL_AI-FIXED	no	no	English	0.0908	0.0923		0.0332	0.0319	
AVERAGE				**0.3203**	**0.3256**	**0.3234**	**0.1739**	**0.1850**	**0.1790**

2.5 Changes in the Scores

The main part of the task is to see the changes in the scores between the collections. All collections were created using the same approach and procedure and have a high overlap in terms of both queries and documents. In Table 2, we thus provide the relative drops between the collections ST and WT and between the collections LT and WT. The definition of the value “WT-ST” NDCG change is defined, for a run r as:

$$\frac{\mathrm{NDCG}_{WT}(r) - \mathrm{NDCG}_{ST}(r)}{\mathrm{NDCG}_{WT}(r)}$$

For “WT-LT” the formula is:

$$\frac{\mathrm{NDCG}_{WT}(r) - \mathrm{NDCG}_{LT}(r)}{\mathrm{NDCG}_{WT}(r)}$$

With such definitions, large negative values for columns “WT-ST” and “WT-LT” mean that the systems are able to generalize well on the new test collections, as the WT heldout queries are processed on the same document corpus as the training data, which is not the case of the ST and LT datasets.

What we see in Table 2 is that the systems that are the more robust to the evolution of test collection are not the top ones: for instance the NEON_3b run is almost at the bottom on Table 3 but does increase its NDCG values at ST, as well as at LT. We also see that the best systems according to NDCG at ST, FADERIC_Fr-BM25-S50-LS-S-F-SC-R20W6, FADERIC_Fr-BM25-S50-LS-S-F-R30 and SQUID_SEARCHERAI, are stable or decreasing their NDCG values at ST.

On average (last line of Table 2), the systems increase less their results on ST than on LT, which is surprising. This surprising point will need further explorations as it looks contradictory to what we were expecting. Another element worth noticing is that the NDCG changes WT-ST and WT-LT behave consistently: for most of the systems the absolute value for WT-ST is smaller than the absolute value of WT-LT.

2.6 Run Rankings

We have so far studied our first problem, which was a comparison of performance of a single system in an evolving setup. Next, we would like to study how do the submitted runs compare to each other, either in terms of the absolute NDCG scores achieved on the collections, or in terms of NDCG changes between the collections. For this, we display the ranking of runs according in all these tasks, see Table 3.

In addition, we also calculated the Pearson correlation between the rankings. The correlation between the rankings (in terms of NDCG scores) achieved on WT and ST is very high (0.95). The correlation between both WT and ST and between ST and LT rankings is slightly lower – 0.71 and 0.70, respectively. This corresponds with the high overlaps of the documents and also queries between WT and ST collections and slightly smaller overlaps of the LT collection.

Table 2: Changes in the NDCG scores. Table is sorted according to the highest change between the ST and WT collection.

	NDCG			NDCG Change	
System	**WT**	**ST**	**LT**	**WT-ST**	**WT-LT**
NEON_3b	0.2017	0.226	0.2387	-0.1205	-0.1835
IRC_d2q+BM25	0.2746	0.3072	0.3211	-0.1188	-0.1694
UGA_T5_English	0.2886	0.3202	0.3347	-0.1095	-0.1598
seupd2223-hiball_BERT	0.3119	0.3418		-0.0959	
soup_kbase	0.2693	0.294	0.3021	-0.0918	-0.1218
ows-bm25-10-variants-prompt-2	0.256	0.2792	0.2872	-0.0907	-0.1219
soup_kngml	0.2698	0.2939	0.3039	-0.0894	-0.1264
HIBALL_AI-MERGED	0.2652	0.2887		-0.0887	
FADERIC_En-BM25-S50-KS-S-F-SP-R30	0.3031	0.3296	0.3262	-0.0875	-0.0763
soup_kml	0.2705	0.2941	0.3042	-0.0873	-0.1246
IRC_BM25+colBERT	0.2883	0.3132	0.3209	-0.0864	-0.1131
ows-bm25-5-variants-prompt-2	0.2556	0.2762	0.2838	-0.0806	-0.1104
seupd2223-hiball_RRF60	0.2664	0.2866		-0.0759	
IRC_BM25+monoT5	0.3034	0.3256	0.3376	-0.0732	-0.1128
UGA_BM25_English	0.2689	0.2873	0.2992	-0.0685	-0.1127
soup_lng	0.2714	0.2899	0.2986	-0.0682	-0.1003
NEON_4b	0.2054	0.2187	0.2282	-0.0648	-0.1111
semicolon_porter2-1p4-eng	0.2739	0.2912		-0.0632	
seupd2223-hiball_BASELINE	0.279	0.2955		-0.0592	
CLOSE_RERANKING_ENGLISH	0.3113	0.3285	0.3373	-0.0553	-0.0836
ows-pl2-10-variants-prompt-2	0.2636	0.2776	0.2881	-0.0532	-0.0930
ows-pl2-5-variants-prompt-2	0.2631	0.2759	0.2876	-0.0487	-0.0932
soup_kmls	0.2739	0.2862	0.2988	-0.0450	-0.0910
seupd2223-JIHUMING-12_fr_fr_4gram_ner	0.2868	0.298	0.3046	-0.0391	-0.0621
semicolon_fusedRankAllEnglish	0.2921	0.3032		-0.0381	
ows-lgd-10-variants-prompt-2	0.2662	0.2759	0.2875	-0.0365	-0.0801
IRC_RRF(BM25+Bo1-XSqrA_M-PL2)	0.2842	0.2939	0.3068	-0.0342	-0.0796
RAFJAM_SynQERuns	0.3193	0.3295	0.3231	-0.0320	-0.0120
SQUID_BOOST	0.3586	0.3693	0.3736	-0.0299	-0.0419
seupd2223-JIHUMING-07_fr_fr	0.3271	0.3367	0.3443	-0.0294	-0.0526
IRC_E5_base	0.2891	0.297	0.3131	-0.0274	-0.0831
semicolon_Ngram34	0.2868	0.2938		-0.0245	
seupd2223-JIHUMING-08_fr_fr_3gram	0.3307	0.3384	0.3454	-0.0233	-0.0445
DARDS_BM25FRENCHBASE	0.3843	0.3924	0.3916	-0.0211	-0.0190
DARDS_BM25FRENCHSPAM	0.3605	0.368	0.3643	-0.0209	-0.0106
NEON_2br	0.2177	0.2219	0.2282	-0.0193	-0.0483
CLOSE_QUEREXPANSION	0.3725	0.3795	0.3736	-0.0188	-0.0030
NEON_1a	0.2201	0.2241	0.2393	-0.0182	-0.0873
CLOSE_JSCLEANER_BM25	0.2647	0.2694	0.2803	-0.0178	-0.0590
seupd2223-JIHUMING-09_fr_fr_4gram	0.3364	0.3423	0.348	-0.0176	-0.0345
RAFJAM_BasicRuns	0.374	0.3804	0.3807	-0.0172	-0.0180
FADERIC_Fr-BM25-S50-LS-S-F-SC-R20W6	0.4169	0.4239		-0.0168	
HIBALL_AI-FIXED	0.0908	0.0923		-0.0166	
NEON_1b	0.2269	0.2294	0.243	-0.0111	-0.0710
seupd2223-JIHUMING-10_fr_fr_5gram	0.3413	0.3447	0.3533	-0.0100	-0.0352
semicolon_frenchAnalyzerFrStopNum	0.3861	0.3895		-0.0089	
semicolon_frenchAnalyzerFrStopWord	0.3869	0.3897		-0.0073	
DARDS_BM25FRENCHBOOSTURL	0.3859	0.3866	0.3945	-0.0019	-0.0223
DARDS_BM25FRENCHRERANK100	0.3755	0.3756	0.3758	-0.0003	-0.0008
UGA_BM25_French	0.354	0.3541	0.3526	-0.0003	0.0040
FADERIC_Fr-BM25-S50-LS-S-F-R30	0.4147	0.4145		0.0005	
gwca_lightstem-phrase-qexp-rerank3f	0.3872	0.3863	0.3942	0.0024	-0.0181
FADERIC_Fr-BM25T-S50-LS-S-F	0.4044	0.4034	0.4071	0.0025	-0.0067
DARDS_BM25TRANSLATEDQUERIES	0.3072	0.304	0.3182	0.0105	-0.0359
UGA_T5_French	0.3757	0.3717	0.3801	0.0107	-0.0118
gwca_word2vec-nostem	0.3843	0.3801	0.384	0.0110	0.0008
FADERIC_Fr-BM25-S50-LS-S-F-SC	0.4079	0.4034	0.4091	0.0111	-0.0030
RAFJAM_AllQERuns	0.3209	0.3172	0.3138	0.0116	0.0222
gwca_lightstem-phrase	0.4052	0.3992	0.3988	0.0149	0.0158
CLOSE_RERANKING	0.4166	0.4068	0.4062	0.0236	0.0250
QEVALS_LMDirichlet	0.2896	0.2819	0.2805	0.0266	0.0315
SQUID_W2V	0.4232	0.4106	0.4174	0.0298	0.0138
SQUID_SEARCHERAI	0.4279	0.4141	0.4177	0.0323	0.0239
SQUID_BasicSearcher	0.4149	0.3998	0.411	0.0364	0.0094
SQUID_W2VRerank	0.4154	0.3997	0.4105	0.0378	0.0118
gwca_lightstem-phrase-qexp	0.4294	0.4114	0.4161	0.0420	0.0310
CLOSE_SBERT_BM25	0.4318	0.4128	0.4139	0.0441	0.0415
RAFJAM_PseudoRelQERuns	0.3516	0.3355	0.349	0.0458	0.0074
gwca_lightstem-phrase-qexp-rerank2f	0.4059	0.3833	0.3905	0.0557	0.0380
QEVALS_BM25CSTM	0.2966	0.2776	0.2845	0.0641	0.0408
QEVALS_BM25DFLT	0.2999	0.2806	0.285	0.0644	0.0497
QEVALS_DFR	0.2976	0.2746	0.2824	0.0773	0.0511
QEVALS_IB	0.3009	0.276	0.2833	0.0828	0.0585
AVERAGE	**0.3226**	**0.3273**	**0.3359**	**-0.0195**	**-0.0376**

Table 3: Ranking of the submitted systems in terms of NDCG scores (columns 2-4), absolute changes in NDCG scores between WT and ST dataset (column 5), absolute changes in NDCG scores between WT and LT dataset (column 6). Column 7 shows the sum of the Borda count applied to ranking on ST dataset and Borda count of ranking change between ST and WT dataset. Column 8 shows the same value, but for the LT dataset. The darker color means better performance.

System	Ranking NDCG WT	Ranking NDCG ST	Ranking NDCG LT	Ranking NDCG Change ST-WT	Ranking NDCG Change LT-WT	Perf(ST) + Change (ST-WT)	Perf(LT) + Change (LT-WT)
seupd2223-hiball_BERT	34	29	64	4	62	113	0
UGA_T5_English	46	37	30	3	3	106	93
FADERIC_En-BM25-S50-KS-S-F-SP-R30	38	33	31	9	22	104	73
IRC_d2q+BM25	52	40	33	2	2	104	91
FADERIC_Fr-BM25-S50-LS-S-F-SC-R20W6	5	1	62	42	62	103	2
DARDS_BM25FRENCHBASE	18	13	13	34	34	99	79
IRC_BM25+colBERT	47	39	34	11	8	96	84
IRC_BM25+monoT5	37	36	28	14	9	96	89
soup_kbase	58	47	42	5	7	94	77
FADERIC_Fr-BM25-S50-LS-S-F-R30	9	2	63	51	62	93	1
SQUID_BOOST	25	24	20	29	29	93	77
CLOSE_RERANKING_ENGLISH	35	35	29	20	18	91	79
soup_kml	56	46	40	10	5	90	81
soup_kngml	57	49	41	7	4	90	81
CLOSE_QUEREXPANSION	23	21	19	37	41	88	66
DARDS_BM25FRENCHSPAM	24	25	21	35	39	86	66
RAFJAM_BasicRuns	22	19	16	41	36	86	74
FADERIC_Fr-BM25T-S50-LS-S-F	13	9	8	52	40	85	78
HIBALL_AI-MERGED	62	53	64	8	62	85	0
semicolon_frenchAnalyzerFrStopNum	16	15	64	46	62	85	0
semicolon_frenchAnalyzerFrStopWord	15	14	64	47	62	85	0
seupd2223-JIHUMING-07_fr_fr	31	31	27	30	26	85	73
RAFJAM_SynQERuns	33	34	32	28	37	84	57
seupd2223-JIHUMING-08_fr_fr_3gram	30	30	26	33	28	83	72
DARDS_BM25FRENCHBOOSTURL	17	16	11	48	33	82	82
seupd2223-hiball_BASELINE	51	45	64	19	62	82	0
FADERIC_Fr-BM25-S50-LS-S-F-SC	10	8	7	57	42	81	77
ows-bm25-10-variants-prompt-2	66	59	49	6	6	81	71
SQUID_SEARCHERAI	3	3	1	63	52	80	73
CLOSE_RERANKING	6	7	9	60	53	79	64
semicolon_fusedRankAllEnglish	43	42	64	25	62	79	0
seupd2223-JIHUMING-09_fr_fr_4gram	29	28	25	39	32	79	69
seupd2223-JIHUMING-12_fr_fr_4gram_ner	48	43	39	24	24	79	63
seupd2223-hiball_RRF60	60	55	64	13	62	78	0
soup_lng	55	52	45	16	13	78	68
SQUID_W2V	4	6	2	62	49	78	75
semicolon_porter2-1p4-eng	53	51	64	18	62	77	0
UGA_BM25_English	59	54	43	15	10	77	73
gwca_lightstem-phrase-qexp-rerank3f	14	17	12	53	35	76	79
NEON_3b	72	69	59	1	1	76	66
CLOSE_SBERT_BM25	1	4	4	67	58	75	64
gwca_lightstem-phrase	12	12	10	59	50	75	66
gwca_lightstem-phrase-qexp	2	5	3	66	54	75	69
DARDS_BM25FRENCHRERANK100	21	22	18	50	43	74	65
seupd2223-JIHUMING-10_fr_fr_5gram	28	27	22	45	31	74	73
ows-bm25-5-variants-prompt-2	67	62	52	12	12	72	62
SQUID_BasicSearcher	8	10	5	64	47	72	74
IRC_E5_base	45	44	37	31	19	71	70
IRC_RRF(BM25+Bo1-XSqrA_M-PL2)	50	48	38	27	21	71	67
UGA_BM25_French	26	26	23	49	45	71	58
gwca_word2vec-nostem	19	20	15	56	44	70	67
SQUID_W2VRerank	7	11	6	65	48	70	72
UGA_T5_French	20	23	17	55	38	68	71
soup_kmls	54	56	44	23	16	67	66
ows-pl2-10-variants-prompt-2	64	61	46	21	15	64	65
semicolon_Ngram34	49	50	64	32	62	64	0
gwca_lightstem-phrase-qexp-rerank2f	11	18	14	69	56	59	56
ows-pl2-5-variants-prompt-2	65	65	47	22	14	59	65
NEON_4b	71	72	61	17	11	57	54
ows-lgd-10-variants-prompt-2	61	64	48	26	20	56	58
DARDS_BM25TRANSLATEDQUERIES	36	41	35	54	30	51	61
RAFJAM_AllQERuns	32	38	36	58	51	50	39
RAFJAM_PseudoRelQERuns	27	32	24	68	46	46	56
CLOSE_JSCLEANER_BM25	63	67	56	40	25	39	45
NEON_2br	70	71	60	36	27	39	39
NEON_1a	69	70	58	38	17	38	51
NEON_1b	68	68	57	44	23	34	46
HIBALL_AI-FIXED	73	73	64	43	62	30	0
QEVALS_LMDirichlet	44	57	55	61	55	28	16
QEVALS_BM25DFLT	40	58	50	71	59	17	17
QEVALS_BM25CSTM	42	60	51	70	57	16	18
QEVALS_IB	39	63	53	73	61	10	12
QEVALS_DFR	41	66	54	72	60	8	12

The correlation between the ranking according to the NDCG score achieved on the WT dataset and the ranking of the performance change is negative. The Pearson correlation is -0.65 for the ST dataset and -0.51 on the LT dataset. This means that the better the system initially performs, harder it is to improve it. Not surprisingly, there is thus also a negative correlation between the ranking achieved on the ST dataset and the ranking of the change between the ST and WT dataset (-0.42). However, there is no such correlation (0.05) between the ranking achieved on the LT dataset and ranking of the change between the WT and LT datasets.

We also provided the normalized results to the participants. The normalization was done according to [7] and the mean and standard deviation of the scores of all submitted runs were calculated. These scores were then used to calculate the score in normal distribution and this score was subsequently shifted using CDF into 0-1 space. However, the correlation of the original ranking and ranking according to the normalized values is highly correlated: 0.93, 0.95, and 0.88 for WT, ST and LT datasets, respectively. We thus further do not work with the normalized results.

Last, we calculated a combination of both rankings (ranking in terms of absolute values and ranking in terms of change). For this, we first calculated a Borda count of the ranking in terms of absolute values and Borda count of the ranking in terms of relative change and then we simply summed these two Borda counts: these results are displayed in two last columns in the Table 3. As the correlation between the absolute performance and performance change is negative, the best performing runs in terms of this measure are often mediocre in one measure and well performing in the another – for instance seupd2223-hiball_BERT run achieves high performance change, while it is mediocre in terms of NDCG achieved on ST dataset.

2.7 Discussion and conclusion

This task was a first attempt at collectively investigate the impact of the evolution of the data on search system's performances. Having 14 participating teams submitting runs confirmed that this topic was of interest to the community.

The dataset released for this task consisted in a sequence of test collections corresponding to different times. The collections were composed of documents and queries coming from Qwant, and relevance judgment coming from a click model and manual assessment. While the manual assessment is ongoing at the time of the paper's publication, performances of participants' submitted runs were measured using the click logs.

The results show that the best approaches were based on query expansion techniques, and embeddings or Large Language Models. The effect of the translation of the documents and queries provided by the lab has a clear impact: the best results were obtained on the original French data.

Since each subset had substantial overlaps, the correlations between systems rankings was pretty high. As for the robustness of the systems towards dataset changes, we observed that the systems that are the more robust to the evolution of test collection were not the best performing ones.

Further evaluations will be carried out in the near future with the manual assessment of the pooled sets. A thorough analysis of the results will be necessary to study the impact of queries on the results (their nature, topic, difficulty, etc.). Further analysis work will be necessary to fully establish the robustness of the systems and the specific impact of dataset evolution on the performances.

3 Task 2 - Classification

As the meanings of words and phrases evolve over time, sentiment classifiers may struggle to accurately capture the changing linguistic landscape [4], resulting in decreased effectiveness in capturing sentiments expressed in text. Recent research shows that this is particularly the case when one is dealing with social media data [3]. Understanding the extent of this performance drop and its implications is crucial for maintaining accuracy and reliable sentiment analysis models in the face of linguistic drift. The objective of this task aimed to quantitatively measure the performance degradation of sentiment classifiers over time, providing insights into the impact of language evolution on sentiment analysis tasks and identifying strategies to mitigate the effects of temporal dynamics. Participants of this task were invited to submit classification outputs of their systems that attempted to mitigate the temporal performance drop.

The aim of Task 2 was ultimately to answer the following research questions:

- **RQ1:** ***What types of models offer better short-term temporal persistence?***
- **RQ2:** ***What types of models offer better long-term temporal persistence?***
- **RQ3:** ***What types of models offer better overall temporal persistence?***

To assess the extent of the performance drop of models in shorter and longer temporal gaps, we provided training data pertaining to a specific year (2016), as well as test datasets pertaining to a close (2018) and a more distant (2021) year. In addition to measuring performance in each of these years separately, this setup enabled evaluating relative performance drops by comparing performance across years.

3.1 Description of the task

In this section, we introduce the task of temporal persistence classification, as the focus of a recent shared task [1]. The goal of this task was to develop classifiers that can effectively mitigate performance drops over short and long periods of time compared to a test set from the same time frame as the training data.

The shared task was in turn divided into two sub-tasks:

Sub-Task 1: Short-Term Persistence: In this sub-task, participants were asked to develop models that demonstrated performance persistence over short periods of time. Specifically, the performance of the models was expected to be maintained within a temporal gap of two years between the training and test data.

Sub-Task 2: Long-Term Persistence: This sub-task focused on developing models that demonstrated performance persistence over a longer period of time. The classifiers were expected to mitigate performance drops over a temporal gap of five years between the training and test data.

By providing a comprehensive training dataset, two practice sets, and three testing sets, the shared competition aimed to stimulate the development of classifiers that can effectively handle temporal variations and maintain performance persistence over different time distances. Participants were expected to submit solutions for both sub-tasks, showcasing their ability to address the challenges of temporal variations in performance.

3.2 Dataset

In this section, we present the process of constructing our final annotated corpus for the task. The large-scale dataset TM-Senti was originally described in [8], from which we extract samples that we use in this shared task. TM-Senti was chosen for the task as it provided a sufficiently longitudinal dataset (covering multiple years) and for using a consistent data collection and annotation strategy, which means that only the temporal evolution of data changes with other potentially confounding factors removed.

Temporal granularity. In the shared task, the **training** set covered a time period with a gap of 2 years, from 2014 to 2016. For the practice sets, within and distance time sets were introduced. The Practice-2016 set had a time gap of 0 years from the training data, given that it overlapped with the training period. In addition, the Practice-2018 set was also provided as a distant test set to practice with, having a temporal gap of two years from the training data.

For the test sets, the within set had a time gap of 0 years, covering the same period as the within Practice-2016 set. The Test-short set had a time gap of 2 years, coinciding with the distant Practice-2018 set. Lastly, the Test-long set had a time gap of 5 years, representing a long-term evaluation scenario.

By using these different time gaps, the shared task aimed to assess the models' performance persistence over varying temporal distances from the training data.

Un-labelled data. The data was sampled from Twitter using the Twitter academic API. Then, duplicates and near duplicates were removed. We also enforced a diversity of users and removed tweets from most frequent users with bot-like behaviour. Finally, user mentions were replaced by '@user' for anonymization, except for verified users that remained unchanged. For all these preprocessing steps, we relied on the same pipeline and script used by [6].

Test set annotation. The test set was annotated using Amazon Mechanical Turk (AMT)[13]. AMT candidate workers were filtered based on them successfully passsing two *qualification tasks*. The first, built-in in the system, seeks to find workers with certain experience and located in English-speaking countries to ensure, to a certain extent, high command of the English language and high familiarity with AMT. The second qualification task consisted in presenting each candidate annotator with 5 tweets, and only workers that correctly annotated 3 or more were allowed to proceed to the actual annotation task.

In total, we annotated 4,032 tweets, divided into 1874 for positive, 741 neutral and 1417 negative. Each tweet was annotated by 5 different workers, and the tweet's final label was decided by computing the *mode* of the array of annotations. Table 4 shows instances of the dataset, with labels and number of agreements between 5 and 3. In terms of overall statistics, 8.5% of the tweets were annotated with full agreement, 22.8% with 4 annotators agreeing, 46% with 3 agreements, and the remaining 22.5% with 2 agreements, which were mostly decided between positive and neutral, and negative and neutral.

Table 4: Tweets where 5, 4 and 3 annotators agreed. Tweets labeled as neutral tend to be factual or posing questions, whereas high agreement positive and negative tweets tend to be more emotional, occasionally backed by the use of stronger words.

#agree	Tweet	Label
5	I say this a lot But I m just so in love with Evan	pos
	Online classes r a joke	neg
	Shout out to me for living 17 minutes away from school	neu
4	Honestly just a Hi from you already makes my day	pos
	Been one of them weeks and I just want to burst out crying	neg
	What s your fave throwback song to jam out to on Thursdays...	neu
3	Not a good idea to mix everything but great night	pos
	just had the worst nightmare I don t want to go back to sleep	neg
	Waiting to find a man that can dance like Chris Brown	neu

Data preprocessing we preprocess our dataset to ensure its quality with respect to the following criteria:

- Diversity: All retweets and replies are eliminated.
- Consistency: We prioritise posts written in English and impose a length restriction such that all posts contain at least 5 words and are at most 140 character long.
- Fluency: Posts containing URL links are eliminated. In addition, we select posts which contain at least one stop word as a proxy for fluency.

[13] https://www.mturk.com/

Before sampling, all emojis and emoticons are deleted from the body of text.

Data sampling. In the second stage, we sample from the preprocessed data previously obtained. As we aim for a well-balanced annotated set, the sampling strategy is defined in terms of: 1) sentiment distribution, 2) time span and 3) post length. For 1), we use the distant labels provided by [8] to obtain a balanced distribution between the negative and positive classes. For 2), we sample an equal number of posts for each month within the specified temporal window in each dataset. Finally for 3), we partition the data into four bins with respect to the word length of each post (i.e., each post falls into one of the following bins: [5,10), [10,15), [15,20) and [20, 20+]) and uniformly sample from each bin.

The resulting distribution of data is shown in Table 5.

Table 5: Dataset statistics summary of training, practice and testing sets.

Dataset	**Time Period**	**Size**
Training	Feb 2014 - Dec 2016	49608
Practice-2016 [within]	Jan 2016 - Dec 2016	1344
Practice-2018 [distant]	Jan 2018 - Dec 2018	1344
Test-within	Jan 2016 - Dec 2016	908
Test-short	Jan 2018 - Dec 2018	908
Test-long	Jan 2021 - Aug 2021	908

3.3 Evaluation

The performance of the submissions was evaluated in two ways:

1. **Macro-averaged F1-score:** This metric measured the overall F1-score on the testing set for the sentiment classification sub-task. The F1-score combines precision and recall to provide a balanced measure of model performance. A higher F1-score indicated better performance in terms of both positive and negative sentiment classification.

$$F - macro = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{1}$$

2. **Relative Performance Drop (RPD):** This metric quantified the difference in performance between the "within-period" data and the short- or long-term distant testing sets. RPD was computed as the difference in performance scores between two sets. A negative RPD value indicated a drop in performance compared to the "within-period" data, while a positive value suggested an improvement.

$$RPD = \frac{f_{score_{t_j}} - f_{score_{t_0}}}{f_{score_{t_0}}} \tag{2}$$

Where t_0 represents performance when time gap is 0; t_j represents performance when time gap is short or long as in was introduced in previous work [2].

The submissions were ranked primarily based on the macro-averaged F1-score. This ranking approach emphasized the overall performance of the sentiment classification models on the testing set. The higher the macro-averaged F1-score, the higher the ranking of the submission.

3.4 Results

Our shared task consisted of two subtasks: Short-term persistence (Sub-task A) and Long-term persistence (Sub-task B). Sub-task A focused on developing models that demonstrated performance persistence within a two-year gap from the training data, while Sub-task B required models that exhibited performance persistence over a longer period, surpassing the two-year gap. Additionally, an unlabeled corpora covering all periods of training, development, and testing was provided to teams interested in data-centric approaches. Along with the data, participating teams received python-based baseline code, and evaluation scripts [14]. The shared task progressed through two phases and results are discussed in the following paragraphs.

3.5 Practice phase

The initial phase was the practice phase, where participants received three distantly annotated sets, training set, within time practice set and short-term practice set. The training set was used for model training, while the two labeled practice set allowed participants to refine their systems before the subsequent phase. Moreover, we limited the sharing practice sets to within-time (Practice-2016) and single distance practice sets the short-term set (Practice-2018). This decision was made because participants were requested to take part in both subtasks and reduce over-fitting. The results of this phase were not considered in final models ranking.

Table 6: Performance comparison for practice set

Team Name	F1 Score Within	F1 Score Short	Overall Drop	Overall Score
Pablojmed	0.8244 (1)	0.7976 (1)	-0.0325 (2)	0.811
saroyehun	0.8170 (2)	0.7917 (2)	-0.0310 (1)	0.8043
Baseline	*0.7879 (3)*	*0.7611 (3)*	*-0.0340 (3)*	*0.7745*

As it can be seen from Table 6, **Pablojmed** showcased outstanding performance, surpassing the **Baseline** model with the highest scores in F1 Score

[14] https://clef-longeval.github.io/

Within (0.8244) and F1 Score Short (0.7976), as well as the highest Overall Score (0.811). **saroyehun** also demonstrated remarkable performance achieving the lowest Overall Drop (-0.0310), as well as outperforming the **Baseline** model in F1 Score Within (0.8170) and F1 Score Short (0.7917). The results highlight the potential of both **Pablojmed** and **saroyehun**'s submissions for enhancing the baseline model's results.

3.6 Evaluation phase

During the evaluation phase, participants were provided with three human-annotated testing sets, namely Test-within, Test-short and Test-long (See 3.2 for datasets details). The performance of participants on this phase was used to determine the overall rankings on the task.

Table 7: Performance comparison for evaluation set.

Team Name	**F1 Score Within**	**F1 Score Short**	**F1 Score Long**	**RPD Within-Short**	**RPD Within-Long**	**Overall Drop**	**Overall Score**
Pablojmed	0.7377 (2)	0.6739 (3)	<u>0.6971</u> (1)	-0.0866 (5)	-0.0550 (3)	-0.0708 (4)	<u>0.7029</u>
Baseline	*<u>0.7459</u>* (1)	*<u>0.6839</u>* (1)	*0.6549* (4)	*-0.0830* (4)	*-0.1220* (5)	*-0.1025* (5)	*0.6949*
Cordyceps	0.7246 (3)	0.6771 (2)	0.6751 (3)	<u>-0.0656</u> (1)	-0.0683 (4)	-0.0669 (3)	0.6923
saroyehun	0.7203 (4)	0.6674 (4)	0.6874 (2)	-0.0735 (2)	-0.0457 (2)	-0.0596 (2)	0.6917
pakapro	0.5033 (5)	0.4648 (5)	0.4910 (5)	-0.0765 (3)	<u>-0.0243</u> (1)	<u>-0.0504</u> (1)	0.4863

Short-term temporal persistence: From Table 7, we can see that still the **Baseline** model is the best for achieving the highest short-term F1 Score (0.6839) among all the teams, indicating that *RoBERTA* architecture has a better performance in capturing short-term patterns compared to the other models. In same time, **Cordyceps** obtained the lowest short-term RPD value (-0.0656), suggesting a smaller drop in performance compared to the **Baseline** model. This indicates that **Cordyceps** may offer better short-term temporal persistence despite not having the highest Short-term F1 Score.

Long-term temporal persistence: In term of long-term persistence, **Pablojmed** achieved the highest f score (0.6971), indicating better performance in capturing long-term patterns compared to the other models. However, when considering the long-term RPD measure, **pakapro** obtained the lowest value (-0.0243), suggesting a smaller drop in performance compared to the other models. This suggests that pessimistic models as in **pakapro** may provide a relatively stable long-term temporal persistence despite not having the highest long-term F1 Score. Although **Pablojmed** obtained the highest F1 Score Long (0.6971), the model that offers better long-term temporal persistence, considering RPD, is pakapro. Despite its lower F1 Score Long (0.4910), **pakapro** achieved the smallest long-term RPD (-0.0243) compared to the other models. This suggests that **pakapro** maintains its performance more consistently over a longer period, indicating better long-term temporal persistence.

Overall temporal persistence: Considering the overall scores, **Pablojmed** achieved the highest overall score (0.7029) with (-0.0708) overall RPD, indicating better overall temporal persistence compared to the other models. However, **pakapro** offers better overall temporal persistence based on the Overall Drop metric. Indicating that **pakapro**'s approach may be more persistent over time in our case despite its low F1 Scores. Overall, the best model is **Pablojmed** demonstrating better overall F score and higher temporal persistence than **Baseline model**. Additionally, the **Baseline model** performed best in short-term temporal persistence, and **pakapro** shows promise for long-term temporal persistence despite not having the highest long-term F1 Score.

Systems temporal ranking: The **Baseline model**, ranks first in within-time and short-term F1 Score but drops to fourth place in long-term F1 Score. **Pablojmed** and **Cordyceps** interchange the second and third positions in both the within-time F1 Score and short-term F1 Score categories. This suggests a relatively consistent ranking between these two models within these specific categories. **saroyehun** consistently ranks fourth in both within-time F1 Score and short-term F1 Score. **pakapro** shows worst performance among all and ranks fifth in all three F scores demonstrate consistent performance across different timeframes compared to the other models.

It is important to note that ranking consistency varies across the different measures. We can see that low RPD does not indicate better performance rather stable metric over different sets. For example, if we look at the RPD metric, we see that **pakapro** achieves the best ranking in long-term and Overall Drop. This indicates a lower drop in performance over longer time-frames. However, when considering the F1 Score, **pakapro** ranks fifth in all three categories: F1 Score Within, F1 Score Short, and F1 Score Long. This demonstrates that a low RPD does not necessarily indicate better performance in terms of F1 Score.

In all cases, submitted systems demonstrated their highest performance when evaluated using the within-time held-out set. Moreover, the overall performance of participating teams seems to have dropped between the practice phase and the final evaluation phase. Given that participants are likely to have submitted their best models from the practice phase, it might be the case that this drop is a result of participants having overoptimism on the practice set.

3.7 Discussion

Only two out of the four teams have submitted technical reports for their used models. In the following, we delve into the discussion and interpretation of the findings concerning the three research questions we raised in relation to our classification task. These interpretations are solely based on the evaluation matrix, which is further explained in Section 3.3.

- Regarding **RQ1**, which aimed to identify the types of models offering better short-term temporal persistence, we observed that the **Baseline model**

achieved the highest short-term F1 Score among all the teams. This indicates its strong performance in maintaining consistency over a shorter time frame compared to its initial performance using within-time set. Additionally, when examining the short-term RPD values, we found that **Cordyceps** exhibited the smallest drop in performance compared to the **Baseline model**.
- Regarding **RQ2**, which investigated the models offering better long-term temporal persistence, we observed that **Pablojmed** achieved the highest F1 Score for the long-term. This indicates its superior ability to maintain performance over an extended period. Notably, **pakapro** demonstrated a smaller long-term RPD compared to the other models, suggesting its potential for maintaining performance stability over time.
- Regarding **RQ3**, this research question aimed to identify the models offering better overall temporal persistence. In this regard, **Pablojmed** ranked as the top performing system, achieving the highest overall score. Its relatively low overall RPD further supports its consistency across different time frames. Interestingly, **pakapro** demonstrated promising results for long-term temporal persistence, despite not achieving the highest long-term F1 Score.

By delving into the evaluation matrix results, we provided insights into the performance trends observed among the participating systems. However, it is essential to acknowledge that the absence of the submission from a certain number of systems may have influenced the overall interpretation of the findings. To address this limitation, we made our leaderbored available for future submissions in Codalab [15]. This should ensure more robust and unbiased assessment for the temporal persistence of text classifiers within the research community.

3.8 Conclusion

Overall findings highlight the importance of evaluating temporal persistence in model performance. The identified models showcase varying levels of persistence in both short-term and long-term persistence. These insights provide valuable guidance for future research and development efforts aimed at improving temporal consistency in machine learning models. In future shared tasks, we aim to incorporate evolving training sets as well as expanding out temporal persistence investigation to more tasks including stance detection and topic categorization.

Acknowledgements This work is supported by the ANR Kodicare bi-lateral project, grant ANR-19-CE23-0029 of the French Agence Nationale de la Recherche, and by the Austrian Science Fund (FWF, grant I4471-N). This work is also supported by a UKRI/EPSRC Turing AI Fellowship to Maria Liakata (grant no. EP/V030302/1) and The Alan Turing Institute (grant no. EP/N510129/1) through project funding and its Enrichment PhD Scheme for Iman Bilal. This work has been using services provided by

[15] https://codalab.lisn.upsaclay.fr/competitions/12762

the LINDAT/CLARIAH-CZ Research Infrastructure (https://lindat.cz), supported by the Ministry of Education, Youth and Sports of the Czech Republic (Project No. LM2018101) and has been also supported by the Ministry of Education, Youth and Sports of the Czech Republic, Project No. LM2018101 LINDAT/CLARIAH-CZ.

References

1. Alkhalifa, R., Bilal, I., Borkakoty, H., Camacho-Collados, J., Deveaud, R., El-Ebshihy, A., Espinosa-Anke, L., Gonzalez-Saez, G., Galuščáková, P., Goeuriot, L., Kochkina, E., Liakata, M., Loureiro, D., Tayyar Madabushi, H., Mulhem, P., Piroi, F., Popel, M., Servan, C., Zubiaga, A.: Longeval: Longitudinal evaluation of model performance at clef 2023. In: Kamps, J., Goeuriot, L., Crestani, F., Maistro, M., Joho, H., Davis, B., Gurrin, C., Kruschwitz, U., Caputo, A. (eds.) Advances in Information Retrieval. Springer Nature Switzerland, Cham (2023)
2. Alkhalifa, R., Kochkina, E., Zubiaga, A.: Opinions are made to be changed: Temporally adaptive stance classification. In: Proceedings of the 2021 Workshop on Open Challenges in Online Social Networks. pp. 27–32 (2021)
3. Alkhalifa, R., Kochkina, E., Zubiaga, A.: Building for tomorrow: Assessing the temporal persistence of text classifiers. arXiv preprint arXiv:2205.05435 (2022)
4. Alkhalifa, R., Zubiaga, A.: Capturing stance dynamics in social media: open challenges and research directions. International Journal of Digital Humanities pp. 1–21 (2022)
5. Galuščáková, P., Deveaud, R., Gonzalez-Saez, G., Mulhem, P., Goeuriot, L., Piroi, F., Popel, M.: Longeval-retrieval: French-english dynamic test collection for continuous web search evaluation (2023)
6. Loureiro, D., Barbieri, F., Neves, L., Espinosa Anke, L., Camacho-collados, J.: TimeLMs: Diachronic language models from Twitter. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations. pp. 251–260. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.acl-demo.25, https://aclanthology.org/2022.acl-demo.25
7. Urbano, J., Lima, H., Hanjalic, A.: A New Perspective on Score Standardization. In: International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1061–1064 (2019)
8. Yin, W., Alkhalifa, R., Zubiaga, A.: The emojification of sentiment on social media: Collection and analysis of a longitudinal twitter sentiment dataset. arXiv preprint arXiv:2108.13898 (2021)