WIS @ the NTCIR-12 Temporalia-2 Task

Yue Zhao and Claudia Hauff
Web Information Systems
Delft University of Technology
Delft, the Netherlands
{y.zhao-1, c.hauff@tudelft.nl}

ABSTRACT

Users' time-related information can may be multi-faceted, leading to temporal intent ambiguity. Here, we present an overview of our submissions to the Temporalia-2's *Temporal Intent Disambiguation* subtask. Our approach focused on the question of whether temporal signals, extracted from publicly available, external data sources (in this case the Wikipedia page view stream), as features in a machine learning setup are beneficial for this task. Although we find that for some queries, temporal features can be highly beneficial for intent prediction, this is not the case for the majority of queries in Temporalia-2's dataset.

Team Name

WIS

Subtasks

Temporal Intent Disambiguation (TID) subtask (English)

Keywords

temporal intent, query intent disambiguation, time-series data, Wikipedia

1. INTRODUCTION

In the Web search setting, the temporal intent of users' queries affects relevance heavily [7]. Similarly, enabling an accurate disambiguation of a query's temporal intent has been shown to be important for the diversification of search results [14, 10]. Given these high-profile applications of temporal intent disambiguation, it is not surprising that a benchmark —Temporalia— was created around this task, which was first introduced at NTCIR-11 [5]. The task is currently in its second edition (thus, Temporalia-2) and part of the NTCIR-12 benchmark campaign. It contains two subtasks which are described in detail in [6].

Our group participated in the so-called *Temporal Intent Disambiguation* (TID) subtask, a refinement of last year's *Temporal Query Intent Classification* (TQIC) subtask [5]. While in TQIC participants were asked to *classify* a query (which was provided in the form of a general Web search query and its time of submission to the search engine) into one of four predefined temporal intents (*past*, *recency*, *future* and *atemporal*), in the revised TID subtask participants are now asked to *estimate* the probability of a query belonging to each of these four temporal intents.

Previous work indicates that the users' information needs may not be well expressed by query content and may change over time [12] and are thus strongly dependent on their issue time. To combat temporal ambiguity, large-scale time-series data sources have been employed in a number of works. One popular choice of aggregated time-series data is Google Trends[1] [1] which provides information on the popularity of search terms over time. While the data provided can be considered as highly reliable (it is an aggregate of billions of users searching the Web), it is also highly stylized and coarse-grained as (i) no absolute frequencies are available, (ii) it is unknown what data pre-processing & cleaning steps occurred, and, (iii) the aggregations occur at a month-by-month level.

We investigate here to what extent we can employ large-scale *publicly available* data as a source of time-series signals. The advantage of a public source lies not only in the reproducibility of the results, but also in the ability to conduct a deeper analysis of the benefits and issues of using time-series data for the TID subtask. As data source, we employ Wikipedia's pageview statistics[2], which provide us with the number of page views a Wikipedia page has received hour by hour since 2007.

Our methodology is inspired by [13], the most successful submission to last year's TQIC subtask. We build on it through the extraction of novel time-signal based features.

Besides presenting our work, we also offer a first error analysis of our runs, in particular with respect to the temporal signals we employ as part of our feature set.

2. APPROACH

In this section we first outline our feature extraction steps, before describing the machine learning framework we employed in this work to solve TID.

2.1 Feature Extraction

The TID subtask requires us to build a pipeline that, given a query and a query issue timestamp, estimates the temporal intent probabilities of four temporal classes. Based on this information, we extract two types of features: (i) query-content features, that is, features derived from the query alone, and, (ii) Wikipedia pageview-based features, that is temporal features derived from Wikipedia pages related to the concepts mentioned in the query.

[1] https://www.google.com/trends/

[2] https://dumps.wikimedia.org/other/pagecounts-raw/

For each query, we extracted a total of 356 features, 227 from query-content, 113 from related Wikipedia concepts and 16 from Wikipedia pageview data.

Query-Content Features.

The features we extract are largely in line with those proposed in [13] for last year's TQIC subtask. The main difference between [13] and our work is the features encoded for temporal expressions. We relied on Stanford's CoreNLP [8] and SUTime [2] packages for feature extraction.

Lemmas and Named Entities are extracted by CoreNLP's tagging framework. For example, from the query "how did Amy Winehouse die"[3] the lemmas "how", "do", "amy", "winehouse" and "die", as well as the named entity "amy winehouse" are detected. Lemmas and named entities that appear only once in the set of TID queries (393 in total) are removed. A small set of stopwords (prepositions, pronouns and articles) are removed as well. We removed these terms as we consider them of little value to the task at hand. After these filtering steps, 197 features (unique lemmas and named entities) remain in our feature set. The weight of each feature is the count of the lemma or named entity in the corresponding query.

Verb Tenses can be good indicators of temporality [3]. Since the influence of the verbs appearing in the same query may differ, they can be represented by their uppermost verb tense (*UVB_tense*) and verb tense with lemma (*tense_lemma*). For example, the query "when was television invented" has three verb features: *UVB_VBD*, *VBD_be* and *VBD_invent*. The uppermost verb is the verb which is uppermost in the parse tree tagged by Stanford's CoreNLP. It represents the verb which is most related to the whole query content. Overall, 22 verb-tense based features are added to our set of features.

Temporal expressions (TEs) are extracted via SUTime, a library for the detection and normalization of time expressions. The relations between the TEs detected in a query and the query issue time are encoded in five features:

- ref_{past}: number of TEs referring to past times with respect to the query issue time;
- ref_{future}: number of TEs referring to future times with respect to the query issue time;
- $same_Y$: number of TEs referring to the same year as the query issue time.
- $same_{YM}$: number of TEs referring to the same year & month as the query issue time.
- $same_{YMD}$: number of TEs referring to the same year & month & day as the query issue time.

How many of the queries though do actually contain TEs? We answer this question in Table 1, where we list the number of queries with and without detected TEs — not only for TID, but also for the queries of last year's TQIC subtask. While last year more than 40% of the queries contained one or more TEs, in this year's data this is the case for less than 20% of the queries.

Even though SUTime is able to detect most TEs appearing in our queries correctly, we do observe query instances where TEs with "misleading" textual evidence cannot be detected (an example is query 199: "When to File 2014 Taxes", where "2014" is not recognized as a TE). In addition, most explicit temporal expressions have numerical lemmas to indicate their years. We encode the relation between numerical lemmas[4] and the query issue time in three features:

- $lemY_{past}$: number of numerical lemmas referring to past years with respect to the query issue time.
- $lemY_{future}$: number of numerical lemmas referring to future years with respect to the query issue time.
- $lemY_{same}$: number of numerical lemmas referring to same years with respect to the query issue time.

As a concrete example, the query "NBA playoffs 2012 2013" issued May 1, 2012 will result in the following non-zero features: $\{ref_{future} = 1, same_Y = 1, lem_{future} = 1, lem_{same} = 1\}$.

Wikipedia Pageview Features.

Table 1 shows the importance of external temporal signals as the vast majority of TID queries do not contain any temporal markers. Here, we experiment with Wikipedia pageview statistics, which provide us with an hour-by-hour overview of the number of page visits a Wikipedia page attracts. In Figures 1, 2, and 3 we show the similarity between the term/phrase trends as aggregated on Google Trends and as aggregated from Wikipedia's pageview counts for three typical queries from TID (each query is represented by its corresponding Wikipedia page). We observe similar temporal signals from both data sources. This (though admittedly anecdotal) evidence leads us to hypothesize that Wikipedia pageviews are a suitable approximation of Web search query popularities over time.

	#Queries overall	#Queries with TEs	#Queries without TEs
TQIC formal-run	300	127	173
TID dry-run	93	15	78
TID formal-run	300	57	243

Table 1: Number of TID and TQIC queries with & without one or more extracted TEs (based on SUTime).

Since we rely on the pageview statistics from Wikipedia, we first need to determine which concept or concepts (each concept is one Wikipedia page in our definition) the query refers to. We make use of DBpedia Spotlight [9] for concept detection and disambiguation. DBpedia Spotlight also provides the types of the detected concepts, which we incorporate in our feature set as well. For example, from the query "baseball playoffs" DBPedia Spotlight extracts two concepts ("Baseball" and "Playoffs") as well as two types ("Activity" and "Sport"). As a secondary approach we extract all noun phrases identified in a query (via CoreNLP) and employ the OpenSearch API[5] provided by MediaWiki

[3] Query #024 in the dry-run data of the TID subtask

[4] To avoid noise, we only consider numerical lemmas within ±20 years of query issue time.

[5] https://www.mediawiki.org/wiki/API:Opensearch

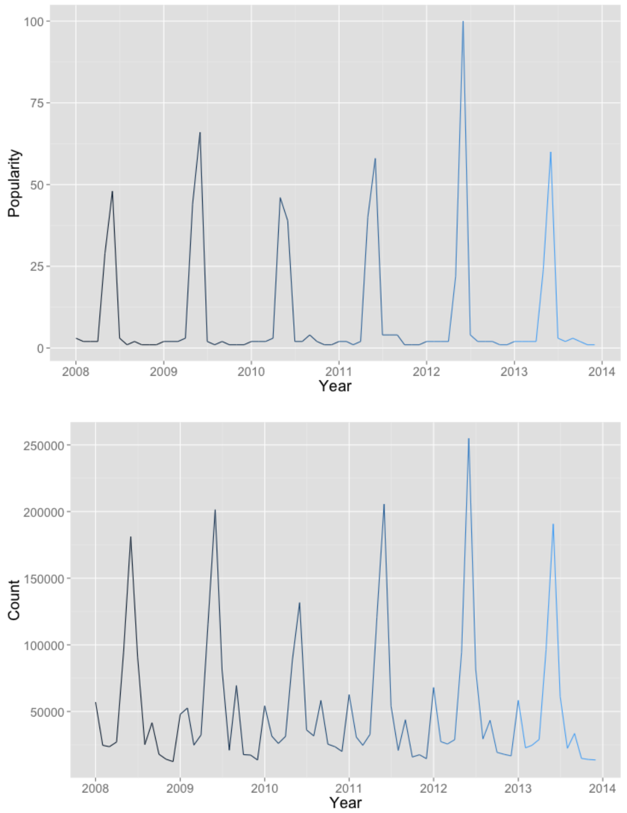

Figure 1: Comparison of Google Trends output (top) for the query "French Open" (a periodic event) and Wikipedia pageview output (bottom) of its related Wikipedia concept "French Open". To match the granularity of Google Trends, the hourly Wikipedia pageviews are aggregated on a monthly basis.

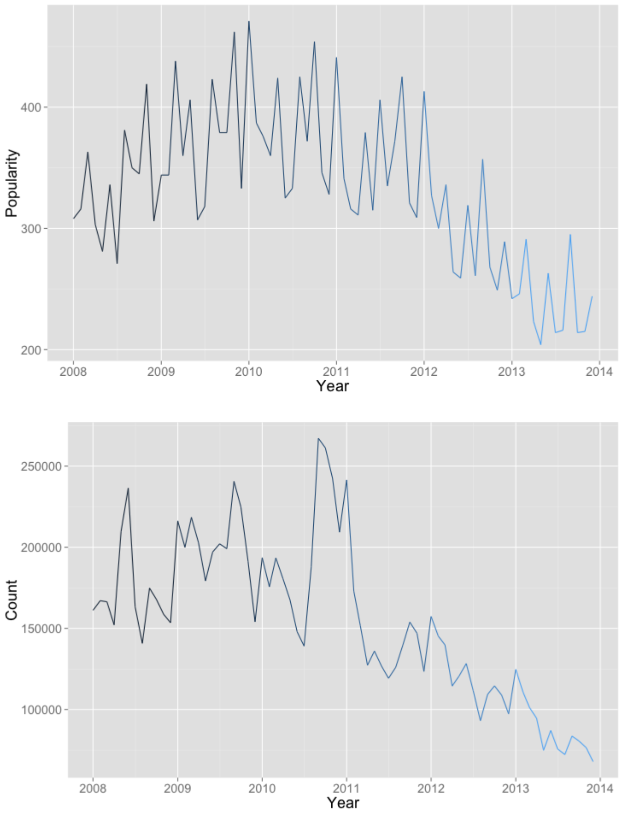

Figure 2: Comparison of Google Trends output (top) for the query "free dictionary" (an atemporal query) and Wikipedia pageview output (bottom) of its related Wikipedia concept "Dictionary". To match the granularity of Google Trends, the hourly Wikipedia pageviews are aggregated on a monthly basis.

to tag noun phrases with their corresponding Wikipedia concepts. Although this method is conceptually simpler than DBpedia Spotlight, for some queries it leads to a better match between query content and detected concept(s). An example of this phenomenon is the query "history of lesbianism"[6], which has a separate Wikipedia page, which is recognized correctly, through our non-phrase based search, whereas DBPedia Spotlight assigns the following concepts to the query: "History" and "Lesbian". Eventually, in our submitted runs we combine both methods, treating *all* detected Wikipedia concepts by either of those methods as a query's *related* Wikipedia concepts. The *best-match* concept among the related set is the concept whose surface form is most similar to the entire query string (based on cosine similarity). All related Wikipedia concepts and corresponding types (as tagged by DBPedia Spotlight) that appear in at least two queries are added to the feature set. Overall, 113 features are retained — 48 Wikipedia concepts and 65 corresponding types.

Sixteen temporal features, derived from pageview time-series counts are extracted for each query's best-match concept:

- *Sparsity*: indicates whether time-series data exists or not, and whether time-series data is sparse or not;
- *Seasonality* [11]: represented by the cosine similarity between the time-series data itself and its seasonal component generated through the Holt-Winter decomposition [4];
- *Autocorrelation*: measures the periodicity of the time-series data by comparing the past 12 months of data to the same time period a year earlier;
- $\{ref_{view\_D}, ref_{view\_MD}\}$: difference between the query issue month (month/day combination) and the month (month/day combination) the concept had the most pageviews in our Wikipedia pageview traces;
- finally, the mean, standard deviation and median of the concept's time-series data are also computed.

Note, that we only rely on the pageview time-series data of the best-match concept as we consider it to be the best representative of the entire query.

We make use of the pageview statistics provided by Wiki Trends[7], that aggregate the hourly view into a day-by-day

[6] Query #016 of the `TQIC` dataset.

[7] http://www.wikipediatrends.com/

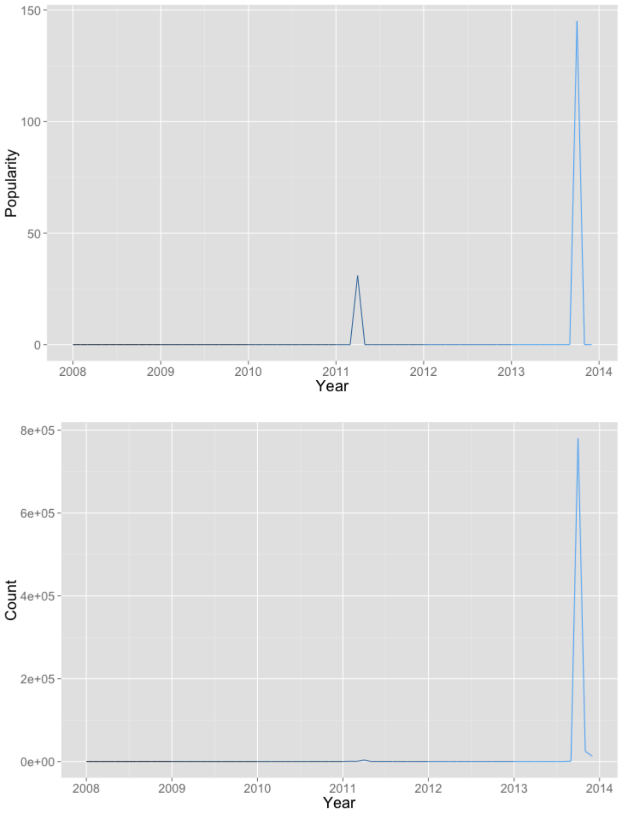

Figure 3: Comparison of Google Trends output (top) for the query "History of Government Shutdowns" (a sudden event) and Wikipedia pageview output (bottom) of its related Wikipedia concept "Government shutdown in the United States". To match the granularity of Google Trends, the hourly Wikipedia pageviews are aggregated on a monthly basis.

granularity and offer statistics from January 2008 onwards. For each query, we only employ the time-series data that was generated *before* the query issue time.

2.2 Intent Disambiguation

The machine learning methods leveraged for temporal query intent disambiguation are regression with multiple dependent variables and probabilistic classification.

Regression with Multiple Dependent Variables is the common choice for estimating tags with several values (e.g. the probability distribution over 4 temporal intent classes) simultaneously. Since the tag of each sample in both training data and test data is a probability distribution over 4 temporal intent classes, the advantage of this method is that the training data and the test data can be fed to the model directly. The disadvantage of this method is that the regression method may not fit the problem very well, since the `TID` subtask is in essence a probabilistic classification task.

In **probabilistic classification**, the output is a distribution over the set of classes (here: the set of intents). In order to employ probabilistic classification, we transform the training data. Each sample in the training data of the form $<query>$ $(P_{past} = x_1, P_{recency} = x_2, P_{future} = x_3, P_{atemporal} = x_4)$ with $x_i \in [0, 1], \sum_i x_i = 1$ is transformed into 100 samples (same feature set) and a single intent setting: $10 \times x_i$ samples for intent i.

We rely on the *Scikit-learn*[8] toolkit for our experimental work. Specifically, we use Ridge for our regression-based experiments and support vector machines (SVM) with an RBF kernel for our classification experiments. We chose those approaches as they performed best on the training data.

3. RESULTS

3.1 Submitted Runs

We submitted a total of three runs for the `TID` subtask.

WIS-TID-E-1 : In this run, we only employ the 227 query-content features. Since the number of training queries (73 samples) is smaller than the number of features, we employ principal component analysis (PCA) with 50 components (found via parameter grid search on the training data with the Mean Absolute Error as target metric) to avoid overfitting. The intent is predicted via our Ridge regressor.

WIS-TID-E-2 : The setup of this run is the same as for WIS-TID-E-1, however now, we now also include the Wikipedia pageview-based features.

WIS-TID-E-3 : Having used the regressor setup in the two previous runs, we now employ the classifier setup with query-content features only and PCA (100 components). The SVM parameters are $C = 1$ and $gamma = 0.001$ (parameter grid search on the training data).

The result of the three submitted runs are shown in Table 2. Our classifier with only query-content features performs significantly (t-test[9], $p < 0.01$) better than both regressor-based runs with respect to MAE. The temporal signals we have incorporated in WIS-TID-E-2 have not had the intended effect — this run has the highest error across a range of metrics.

3.2 Ablation Study

In order to understand the influence of different kinds of features in the three runs of experiments, an ablation study is introduced. The features from query contents are separated into 3 classes, which are $Lemma\&NN$, TE and $Verb$. Wikipedia concepts and types are treated as 1 class. The features of time-series data are separated into 5 classes (i.e. *Sparsity*, *Seasonality*, *Autocorrelation*, *Ref* and *Stats*) as mentioned in Section 2.1.

Based on Figure 3, it is obvious that the official evaluation metrics show the similar trends in the ablation study. It can be found that:

- The usage of time-series data reduce the influence of features derived from query content.

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

[9] One-sided t-test on the difference between the MAE of the classifier and two regressors with the hypothesis that the mean of differences is 0

			Per-Class Absolute Error			
Runs	**Cos Sim**	**MAE**	*Past*	*Recency*	*Future*	*Atemporal*
WIS-TID-E-1	0.792	0.215	0.211	0.154	0.204	0.291
WIS-TID-E-2	0.773	0.219	0.205	0.159	0.206	0.306
WIS-TID-E-3	0.791	0.197	0.151	0.146	0.204	0.288

Table 2: Result overview of our submitted runs according to the official evaluation metrics.

MAE	**WIS-TID-E-1**	**WIS-TID-E-2**	**WIS-TID-E-3**
Baseline	0.215	0.219	0.197
- *Lemma&NN*	+0.0128	+0.0034	+0.0275
- *TE*	+0.0075	+0.0053	+0.0119
- *Verb*	+0.0052	-0.0007	-0.0103
- *Wiki&Type*	–	-0.0036	–
- *Sparsity*	–	-0.0043	–
- *Season*	–	-0.0011	–
- *AutoCor*	–	+0.0003	–
- *Ref*	–	-0.0002	–
- *Stats*	–	+0.0003	–
Cos Sim	**WIS-TID-E-1**	**WIS-TID-E-2**	**WIS-TID-E-3**
Baseline	0.792	0.773	0.791
- *Lemma&NN*	-0.0437	-0.0085	-0.0614
- *TE*	-0.0208	-0.0186	-0.0314
- *Verb*	-0.0118	+0.0016	+0.0252
- *Wiki&Type*	–	+0.0148	–
- *Sparsity*	–	+0.0147	–
- *Season*	–	+0.0048	–
- *AutoCor*	–	-0.0003	–
- *Ref*	–	+0.0006	–
- *Stats*	–	-0.0010	–

Table 3: Ablation study of our submitted runs according to the official evaluation metrics.

- Features derived from query content also have different influence in different models. For example, *Verb* features provide no help in the classification model, while *Lemma&NN* play a more important role in the classification model.
- Not all the features derived from time-series data are helpful. In this case, only features of autocorrelation *AutoCor* and statistics *Stats* are helpful.

4. ERROR ANALYSIS

In our error analysis, we investigate (i) the effects of the temporal features on a query-by-query basis by comparing WIS-TID-E-1 and WIS-TID-E-2 (which only differ in their feature set), and (ii) the effect of the choice of predictor by comparing WIS-TID-E-1 and WIS-TID-E-3 (which only differ in their choice of predictor).

Impact of Temporal Features.

In Figure 4 we plot the difference in MAE on a query-by-query basis for the formal-run data of `TID`.

For a large number of queries, the impact of the temporal features is small — 280 out of 300 queries exhibit an MAE difference of less than 0.1. We manually investigated those remaining 20 queries with a difference of ≥ 0.1. We found that:

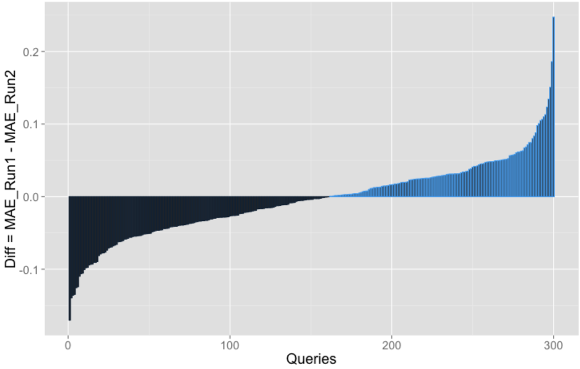

Figure 4: Overview of the difference on MAE of WIS-TID-E-1 and WIS-TID-E-2.

A common problem for WIS-TID-E-2 are queries, whose ground truth has a large *Atemporal* intent probability; those are often estimated as having large probabilities for either the *Recency* or the *Future* intent.

WIS-TID-E-2 performs better on queries that either have no clear trends in the time-series data or refer to recurrent events — Table 4 lists the ten queries WIS-TID-E-2 outperforms WIS-TID-E-1 on.

#069 estimate on the Debt Crisis in Greece
#165 sovereign debt crisis
#021 PS4 Release Date
#131 The first walk into outer space was taken by a Soviet
#012 summer days
#085 timetable bus suzhou
#113 NBA Finals
#104 snake dishes have become popular in recent years
#013 The Following Recap
#092 baseball playoffs

Table 4: List of the ten queries where WIS-TID-E-2 outperforms WIS-TID-E-1 with more than 0.1 **MAE difference.**

Impact of Predictor Choice.

Comparing WIS-TID-E-1 (regressor) and WIS-TID-E-3 (classifier), we find the classifier to outperform the regressor on nearly all evaluation metrics. The MAE difference query-by-query is plotted in Figure 6. For 200 of the 300 `TID`

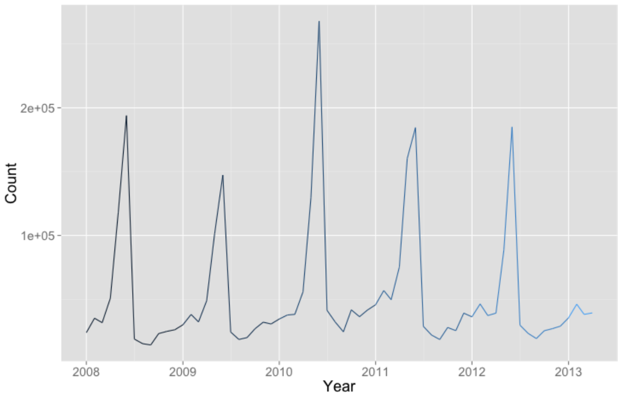

Figure 5: Wikipedia daily page views of the concept "NBA Finals" corresponding to the query "NBA Finals".

formal run queries, the classifier-based predictions have a lower MAE. The reason for this discrepancy can be found in the ground truth, which has discrete probabilities with one or two intent types usually set to zero, which the regressor cannot fit very well.

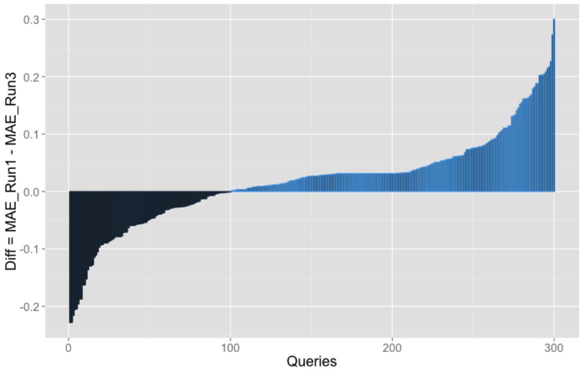

Figure 6: Overview of the difference in MAE of WIS-TID-E-1 and WIS-TID-E-3.

In Table 5 we list the five queries, WIS-TID-E-3 performs considerably better and worse than WIS-TID-E-1. Evidently, the verb tense plays an important role. The five queries WIS-TID-E-3 performs worse than WIS-TID-E-1 all have "be" or "do" in their query string, whose *Atemporal* intents in the ground truth are estimated as *Future* intents in the predictions. It shows the similar trend to our ablation study in Section 3.2.

5. CONCLUSION

In this report, we have described our approach to the `TID` subtask at Temporalia-2. Our focus in this work lies in the incorporation of temporal features from publicly available time-series data sources (specifically Wikipedia pageviews). In contrast to our hypothesis, temporal features — at least in our current pipeline — overall are not beneficial for this task. Our error analysis though provides reasons to continue in this direction, as there are indeed queries that benefit from temporal features. In the future, we will investigate to what extent we can leverage the time-series data of not just the best-match Wikipedia concept, but all related concepts to some degree. We will also conduct a more in-depth evaluation of our different pipeline components and the impact of their accuracy on the overall intent disambiguation task.

++++ WIS-TID-E-1 outperforms WIS-TID-E-3 ++++
#106 Barack Obama is the 44th US president
#100 science fiction is a popular
#117 is it easy to find a job in hong kong
#175 How college is different from high school
#009 When Does Time Change
++++ WIS-TID-E-3 outperforms WIS-TID-E-1 ++++
#069 estimate on the Debt Crisis in Greece
#043 The original building was built in 1710
#219 When Did WW2 Start
#026 when does fall start
#165 sovereign debt crisis

Table 5: List of the top five queries WIS-TID-E-3 performs worse (top) or better (bottom) than WIS-TID-E-1. Only queries with more than 0.1 MAE difference are included.

6. REFERENCES

[1] H. A. Carneiro and E. Mylonakis. Google trends: a web-based tool for real-time surveillance of disease outbreaks. *Clinical infectious diseases*, 49(10):1557–1564, 2009.

[2] A. X. Chang and C. D. Manning. Sutime: A library for recognizing and normalizing time expressions. In *LREC*, pages 3735–3740, 2012.

[3] M. Enç. Towards a referential analysis of temporal expressions. *Linguistics and Philosophy*, 9(4):405–426, 1986.

[4] P. Goodwin et al. The holt-winters approach to exponential smoothing: 50 years old and going strong. *Foresight*, 19:30–33, 2010.

[5] H. Joho, A. Jatowt, R. Blanco, H. Naka, and S. Yamamoto. Overview of NTCIR-11 temporal information access (Temporalia) task. In *NTCIR-11*, 2014.

[6] H. Joho, A. Jatowt, R. Blanco, H. Yu, and S. Yamamoto. Overview of NTCIR-12 temporal information access (temporalia-2) task. In *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies*, 2016.

[7] A. Kulkarni, J. Teevan, K. M. Svore, and S. T. Dumais. Understanding temporal query dynamics. In *WSDM '11*, pages 167–176, 2011.

[8] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60, 2014.

[9] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8. ACM, 2011.

[10] T. N. Nguyen and N. Kanhabua. Leveraging dynamic query subtopics for time-aware search result diversification. In *Advances in Information Retrieval*, pages 222–234. Springer, 2014.

[11] M. Shokouhi. Detecting seasonal queries by time-series analysis. In *SIGIR '11*, pages 1171–1172, 2011.

[12] K. Spärck-Jones, S. E. Robertson, and M. Sanderson. Ambiguous requests: implications for retrieval tests, systems and theories. In *ACM SIGIR Forum*, volume 41, pages 8–17. ACM, 2007.

[13] H.-T. Yu, X. Kang, and F. Ren. Tuta1 at the ntcir-11 temporalia task. In *Proceedings of the 11th NTCIR Conference on Evaluation of Information Access Technologies*, 2014.

[14] K. Zhou, S. Whiting, J. M. Jose, and M. Lalmas. The impact of temporal intent variability on diversity evaluation. In *ECIR '13*, pages 820–823. 2013.