# KGO at the NTCIR-12 Temporalia Task

## *Exploring Temporal Information in Search Queries*

**Xin Kang† & Yunong Wu†‡ & Fuji Ren†**

†Faculty of Engineering, Tokushima University

‡Department of R&D, Business Big Data Co., Ltd.

kang-xin@is.tokushima-u.ac.jp, wuyunong@brandbigdata.com, ren@is.tokushima-u.ac.jp

**Abstract**

We carefully develop a series of temporal features based on the general knowledge underlying Wikipedia, and construct a deep neural network with a softmax layer for disambiguating people's temporal intents in web search queries. We analyze the importance of different temporal features, and discuss the impact of neural network structures to the TID results.

## Introduction

Web search queries are short (Fig. 1). Table 1 shows some easy and difficult samples for TID. The temporal information about searching events (e.g. *iphone 6* and *memorial day*) which only exists in the human knowledge, must be explored for computers to understand people's temporal intents.

The KGO team carries forward their previous work (as TUTA1) in the NTCIR-11 TQIC subtask, by focusing on the development of temporal features and the construction of a deep neural network with a probabilistic interpretation, to solve the TID problem.

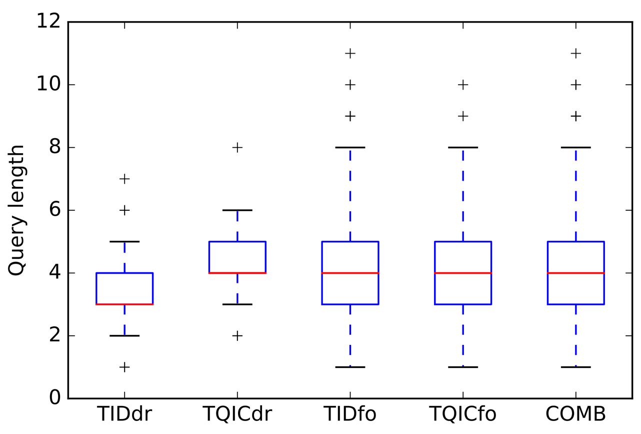



**Figure 1:** Query length dist.

| Pa | Re | Fu | At | Time | Query | Temp Feat |
|---|---|---|---|---|---|---|
| 1.0 | 0.0 | 0.0 | 0.0 | May 1, 2013 | when **was** electricity invented | UVT_VBD |
| 1.0 | 0.0 | 0.0 | 0.0 | May 1, 2013 | beer night **1974** | DIFF_past |
| 0.0 | 0.0 | 1.0 | 0.0 | May 1, 2013 | release date for **iphone 6** | DIFF_future |
| 0.0 | 0.0 | 0.6 | 0.4 | May 1, 2013 | **memorial day** | DIFF_future |

**Table 1:** Typical query examples in the TID Dry Run set.

## Temporal Feature Extraction

**Explicit Temporal Features** The **Uppermost Verb Tense** feature indicates the tense of a query, which could be directly obtained by picking tense information (part-of-speech tag) from the main (uppermost) verb in a query [2].

The **Time Gap** feature represents the difference between an explicit time expression like year *1974* and the query submission time, with year, month, day, season, and period normalized for time differentiation.

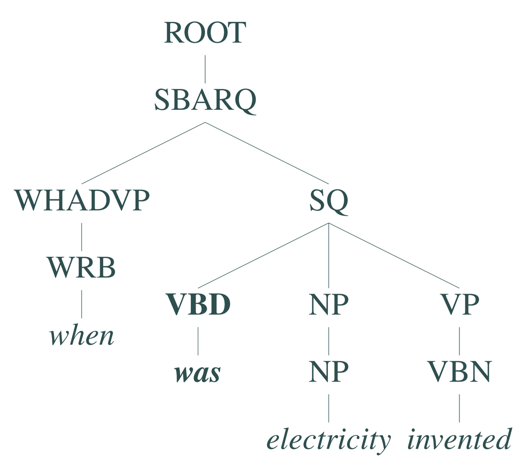



**Figure 2:** UVT feature extraction.

**Inexplicit Temporal Features** The **Temporal Named Entity** feature extracts temporal information of a named entity in query, by exploring the temporal knowledge from Wikipedia.

Step 1. Get Wikipedia summary of *iphone 6*:

*The iPhone 6 and iPhone 6 Plus are smartphones designed and marketed by Apple Inc. The devices are part of the iPhone series and were* ***unveiled*** *on* ***September 9, 2014****, and* ***released*** *on* ***September 19, 2014****. The iPhone 6 and iPhone 6 Plus jointly serve as successors to the iPhone 5C and iPhone 5S.*

Step 2. Parse summary and extract the Time Gap features:

*September 9, 2014* → DIFF_future
*September 19, 2014* → DIFF_future

Step 3. Resolve the correlation between Time Gap $t$ and Named Entity $e$:

$$s(t,e) = \cos\left(\frac{1}{|C(t)|}\sum_{w\in C(t)} \mathrm{v}(w), \frac{1}{|C(e)|}\sum_{w\in C(e)} \mathrm{v}(w)\right), \quad (1)$$

where $C(t)$ and $C(e)$ are the contexts for $t$ and $e$ respectively, and $\mathrm{v}(w)$ is a 1000-dimensional semantic vector generated by a word2vec model [1]. Specifically, we extract the closest predicate to $t$ as its context $C(t)$, and extract words in query except $e$ as its context $C(e)$: $C$(*September 9, 2014*) = {*unveiled*}, $C$(*September 19, 2014*) = {*released*}, $C$(*iphone 6*) = {*release*, *date*, *for*}, and get the correlations:

$s$(*September 9, 2014*, *iphone 6*) = 0.2551,
$s$(*September 19, 2014*, *iphone 6*) = 0.3291.

The **Holiday** feature extracts temporal information of holidays in a query from a holiday database (HDB).

Step 1. Extract the holiday name: *memorial day*.

Step 2. Extract the country information: default *United States*.

Step 3. Extract the year information: default *query submission year*.

Step 4. Query holiday date from HDB: *May 27, 2013*.

Step 5. Date differentiate: DIFF_future.

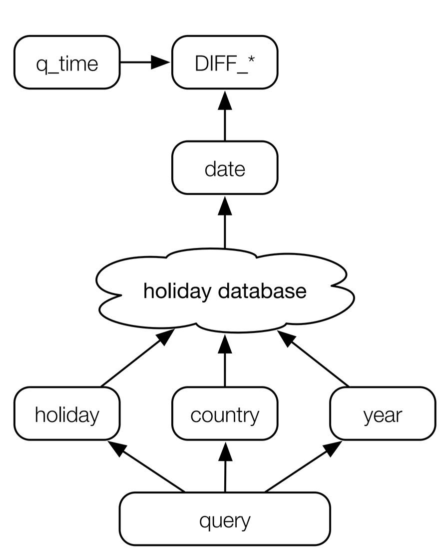



**Figure 3:** Holiday extraction.

**Other Features** The **People** and **Time** feature clusters people and events based on their semantic similarities in TextRazor Entity Extraction (to decrease the feature space and avoid over-fitting).

*what time does the* ***Super Bowl*** *start* → RECURRING_EVENT
*how did* ***Amy Winehouse*** *die* → DECEASED_PERSON, MEASURED_PERSON, PERSON

The **Lemma** feature normalizes words in a query. Part-of-speech tags in verb lemmas are kept for their tense information.

*when* → when
*was* → VBD_be
*electricity* → electricity
*inventented* → VBN_invent

## Neural Network Construction

We construct a **deep neural network** model to generate the probabilistic temporal predictions

$$\tilde{p}_j = p(y=j|\boldsymbol{x};\boldsymbol{W}) = \frac{\exp(\boldsymbol{x}^T\boldsymbol{W}_j)}{\sum_{j'=1}^{J}\exp(\boldsymbol{x}^T\boldsymbol{W}_{j'})}, \quad (2)$$

with cross entropy as its cost function

$$H(p,\tilde{p}) = -\sum_{j=1}^{J} p_j \log \tilde{p}_j. \quad (3)$$

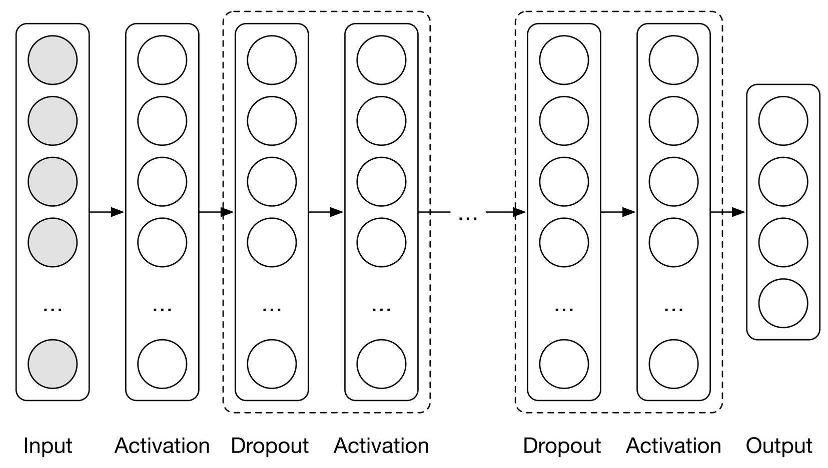



**Figure 4:** Deep neural network with dropout for TID. Hidden activations are selected from softplus, relu, tanh, sigmoid, hard sigmoid, and linear.

## Experiments

**Experiment Setup** We incorporate the TQIC data (400 samples) into the TID Dry Run set (93 samples) for **training** deep neural network, assuming that in TQIC there is no possibility mass for temporal labels except the tagged one, and **test** on the TID Formal Run set (300 samples). Parameters are **selected** through 5-fold cross-validation, based on the **averaged per-class absolute loss**

$$\mathrm{loss}(p,\tilde{p}) = \frac{1}{J}\sum_{j=1}^{J}|p_j - \tilde{p}_j|, \quad (4)$$

and the **cosine similarity**

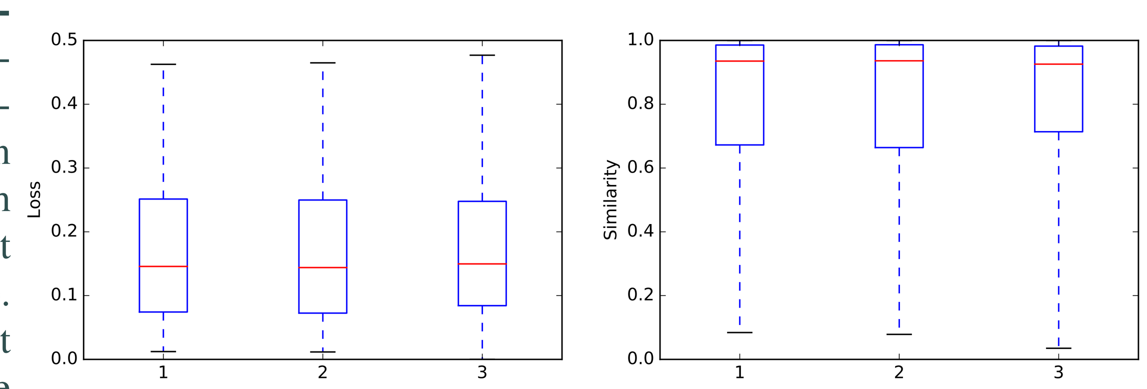
$$\mathrm{sim}(p,\tilde{p}) = \frac{\sum_{j=1}^{J} p_j \times \tilde{p}_j}{\sqrt{\left(\sum_{j=1}^{J} p_j \times p_j\right)\left(\sum_{j=1}^{J} \tilde{p}_j \times \tilde{p}_j\right)}}. \quad (5)$$

**Experiment Result** 3 Formal Runs with configurations in Table 2 are submitted, with the **averaged per-class absolute loss** and **cosine similarity** evaluations shown in Fig. 5a and 5b.

| Run | Criteria | Neural Network Parameters | | | | |
|---|---|---|---|---|---|---|
| | | $L$ | $n^{(*)}$ | $a^{(*)}$ | $b$ | $N$ |
| 1 | sim-$L$ | 2 | 32, 16 | relu, hard sigmoid | 256 | 385 |
| 2 | loss-$L$ | 2 | 32, 16 | relu, hard sigmoid | 256 | 392 |
| 3 | sim+$L$ | 3 | 64, 32, 16 | softplus, hard sigmoid, linear | 256 | 578 |

**Table 2:** Configurations of $L$ (the number of hidden layers), $n^{(l)}$ (the number of neurons in layer $l$), $a^{(l)}$ (the activation function from in layer $l$), $b$ (the batch size), and $N$ (the number of training epochs) in 3 Runs.

**Network Structure Analysis** **Run-1** gets the best errors. **Run-2** with fewer layers renders the best mean loss of 0.1676. **Run-3** with more layers renders the best mean similarity of 0.8136. **Run-3** also gets the most accurate predictions and the most medium-quality predictions.




(a) Averaged per-class absolute loss.

(b) Cosine similarity.

**Figure 5:** Evaluation of 3 Formal Runs.

**Feature Analysis** Fig. 6 evaluates the association between temporal features and temporal labels in the training corpus, with the normalized point-wise mutual information (npmi)

$$\mathrm{npmi}(x_i;y_j) = -\log\frac{p(x_i,y_j)}{p(x_i)p(y_j)}/\log p(x_i,y_j), \quad (6)$$

in which $x_i$ and $y_j$ represent feature and label respectively.

- **UVT** centers around 0 (too many UVT_NULL's)
- **VT** diverges from 0 for Past, Future, and Atemporal
- **TG** diverges from 0 (sensitive feature)
- **NE** diverges from 0 (sensitive feature)
- **LM** spreads in a wide range (needs finer investigation)

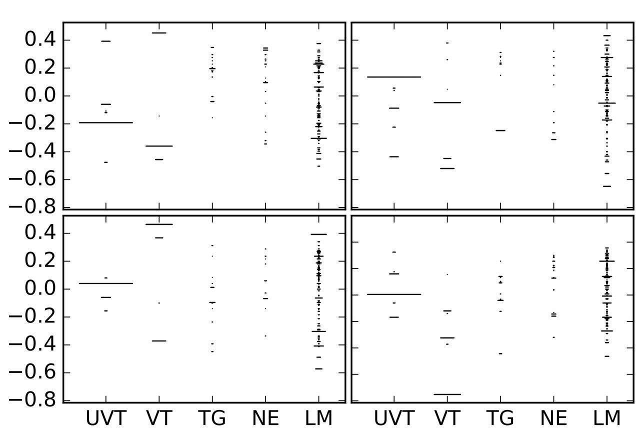



**Figure 6:** npmi between temporal labels (Past, Recent, Future, Atemporal) and temporal features (Uppermost Verb Tense, Verb Tense, Time Gap, Named Entity, LeMma).

## Conclusions

- Exploring the temporal information from Wikipedia and resolving the temporal correlation with vectorized context-similarity.
- Deriving the abstract temporal features and generating the probabilistic temporal predictions with a deep neural network.
- Run-2 with fewer layers achieved the best loss score and Run-3 with more layers rendered the highest similarity score.
- Examining the association between temporal features and temporal labels suggested some directions for improvement.

## References

[1] Idio. Enwiki word2vec model 1000 dimensions. 2015.

[2] H. Yu, X. Kang, and F. Ren. Tuta1 at the ntcir-11 temporalia task. 2014.