DUT-NLP-CH @ NTCIR-12 Temporalia TID Subtask

Jiahuan Pei[1], Degen Huang[2], Jianjun Ma[3], Dingxin Song, Leyuan Sang

Dalian University of Technology

p_sunrise@mail.dlut.edu.cn[1], huangdg@dlut.edu.cn[2], majian@dlut.edu.cn[3]

ABSTRACT

We take the Temporal Intent Disambiguation subtask as a problem of classification and find explicit and implicit distinguishable features to feed machine learning classifiers. The best AvgCosin and AvgAbsLoss reach 0.8135 of and 0.1710. After the submission of the formal run, a post-task research has been done with AvgCosin and AvgAbsLoss rising to 0.8886 and 0.1286.

INTRODUCTION

- **Related works**
  Jones and Diaz [1] categorize queries into temporally unambiguous, temporally ambiguous and atemporal. Ren et al. [2] investigated the automatic detection of web queries and categorized users' temporal intents into hierarchical temporal classes.
- **Main challenges**

1. The lack of explicit temporal information
2. No available query logs
3. The changing of temporal intents
4. The ambiguity of temporal intents

DATASETS

- **DATA1** 52 dry run quires released by NTCIR-12
- **DATA2** 300 formal run quires from NTCIR-11 TQIC subtask
- **DATA3** 503 time-sensitive search queries extracted from SogouQ log data
- **TESTDATA** 300 testing queries from NTCIR-12 TID subtask

Table 1. Average distribution of four temporal intent classes

Past	Recency	Future	Atemporal	Total
0.13	0.16	0.07	0.64	1

APPROACH

- **Overview**
  We regard this task as a classification problem with probability output since each query has a distributional tagging vector of four temporal classes. Our overall method relies on well-designed features and well-established classifiers.
- **Basic**

1. Chinese Segmentation, POS tagging, Name Entity Recognizer and Parser by Stanford Corenlp tookit [3], Temporal Expression Recgnition by HeidelTime [4]
2. Feature selection based on preprocessed results
3. Classification provided by scikit-learn, a machine learning module in Python

FEATURE DESIGN

Table 2. Feature list

Group name	Feature No.	Feature name	Meaning or value
Time Gap	f1-f4	PAST_REF/RECENCY_REF/ FUTURE_REF/IMPLICIT_REF	0 or 1
Word-based Probability Distribution	f5-f8	$P_{past}/P_{recency}/P_{future}/P_{atemporal}$	0≤P≤1 sum(P)=1
Temporal Trigger Word	f9-f12	P_TIGGER/R_TIGGER/ F_TIGGER/A_TIGGER	1,2,...,N
Other textual Features	f13	CenterWord	1,2,...,N
	f14	posOfCenterWord	1,2,…,33
	f15	validQueryLength	1,2,…,N
	f16	numOfNER	1,2,…,N
	f17	numOfNotChWords	1,2,…,N
	f18	isNo,unFrag	0 or 1
Google Trends' Time Gap	f19-f22	$GT_{past}/GT_{recency}/GT_{future}/GT_{atemporal}$	0≤GT≤1 sum(GT)=1

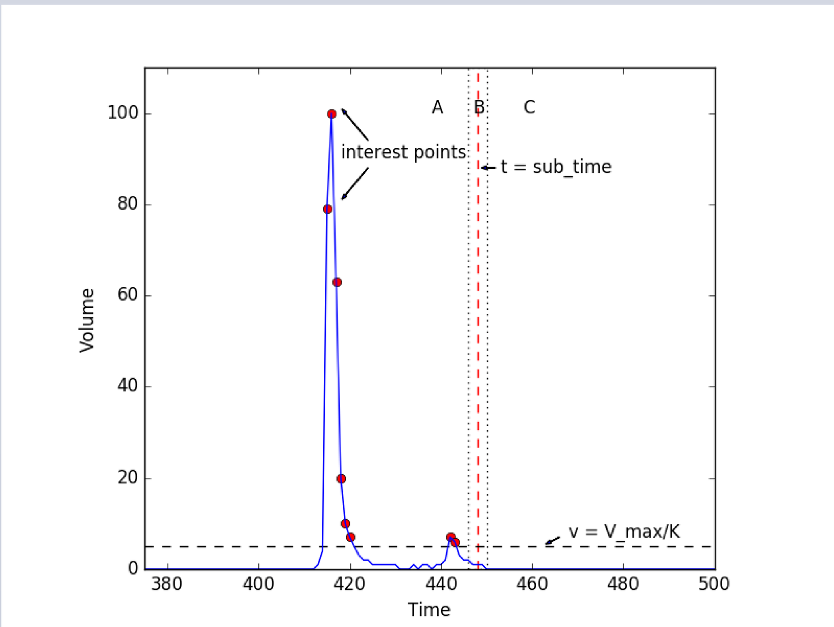
- **Temporal Trigger Word**

$$P(C_i|Query) = \frac{P(C_i)\prod_{w_i \in dict} P(w_i|C_i)^{TF(w_j,Query)}}{\sum_{k=1}^{4} P(C_k)\, P(w_j|C_k)^{TF(w_j,Query)}}$$

- **Time Gap Features from Google Trends**

1. **Preprocessing**
   Downloading search volume of queries from Google Trends
2. **Time-series Prediction**
   ARMA model to predict the "future" volume after submission time.
3. **Extraction**
   Extracted the 11 features mentioned in [2] and use SVC(Gaussian kernel) predict the probability of each categories.

Fig 1. An example of a query



RESULTS & IMPROVEMENT

Table 3. Results of formal runs

Run	AvgCosin	AvgAbsLoss
1	0.8135	0.1728
2	0.8066	0.1854
3	0.8116	0.1710

Table 4. Comparison among different models

Model		AvgCosin	AvgAbsLoss
SVC	linear	0.8639±0.0011	0.1640±0.0003
	rbf	0.8658±0.0016	0.1637±0.0005
	poly	0.6657±0.0043	0.2543±0.0013
NuSVC	linear	0.8692±0.0008	0.1635±0.0005
	rbf	0.8724±0.0020	0.1611±0.0006
	poly	0.8280±0.0056	0.1875±0.0028
RF	balanced	0.8544±0.0062	0.1700±0.0038
	unbalanced	0.8862±0.0019	0.1262±0.0014
LR	$l1$	0.8623	0.1674
	$l2$	0.8453	0.1782
GNB		0.8433	0.1606
MNB		0.7416	0.2145
LDA		0.8812	0.1380
DT		0.8517	0.1702

- **Formal Run Results**
  Table 3 shows the results we submitted as formal runs.
- **A Posteriori Improvement**
  For the posteriori research after submission, we made some comparative experiments on different models; Table 4 shows the results of different models with default parameter settings based on two basic features: time gap feature and temporal trigger word feature. DATA1 and DATA2 were the training data, and 300 formal run data were test data.
- **Findings**

1. Linear SVC model, LR model with penaly $l1$, RF model with balanced class weight outperform Run1, Run2, Run3 respectively.
2. Class weight is an important factor for this task. For instance, RF model with class weight of {Past: 0.13, Recency: 0.16, Future: 0.07, Atemporal: 0.64} performs much better than its balanced model.
3. LR model, GNB model, LDA model and DT model are stable. Based on this finding, we chose LDA model to do feature selection and got the result (see Table 5).

Table 4. Comparison among different models

Run	Composition	AvgCosin	AvgAbsLoss
4	baseline	0.8812	0.1380
5	baseline+f7	0.8825	0.1343
6	baseline+f7+f20	0.8831	0.1339
7	baseline+f7+f14	0.8841	0.1335
8	baseline+f7+f14+f13	**0.8886**	**0.1286**

CONCLUSIONS

This paper introduces the approach DUT-CH group addressed for Temporalia task at the NTCIR-12. We participated in TID Chinese subtask aiming at predicting the distribution of four temporal intents. For the formal run of TID Chinese subtask, we adopt all the designed features to linear SVC model, Logistic Regression model with $l1$ penalty and Random Forest model with balanced class weight. After the submission of the formal run, we did further experiments to compare different models and feature combinations and finally got a better and more stable result by LDA model with features including selected time gap, word-based probability distribution vector, temporal trigger word, Google Trends' time gap, center word and its POS.

REFERENCES

[1] R. Jones and F. Diaz. Temporal profiles of queries. *ACM Transactions on Information Systems (TOIS)*, 25(3): 14, 2007.

[2] P. Ren, Z. Chen, X. Song, B. Li, H. Yang and J. Ma. Understanding temporal intent of user query based on time-based query classification. In: *Natural Language Processing and Chinese Computing*, pages 334-345, 2013

[3] 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.

[4] H. Li, J. Strötgen, J. Zell and M. Gertz. Chinese Temporal Tagging with HeidelTime. In *EACL*, pages 133-137, 2014.

ACKNOWLEDGMENTS

This work has been supported by the National Nature Science Foundation of China (No. 61173100 61272375) and National Social Science Foundation of China (No. 15BYY175).