THUIR@NTCIR-12 IMine

Zeyang Liu, Ye Chen, Rongjie Cai, Jiaxin Mao, Chao Wang, Cheng Luo, Xin Li, Yiqun Liu, Min Zhang, Huanbo Luan, Shaoping Ma

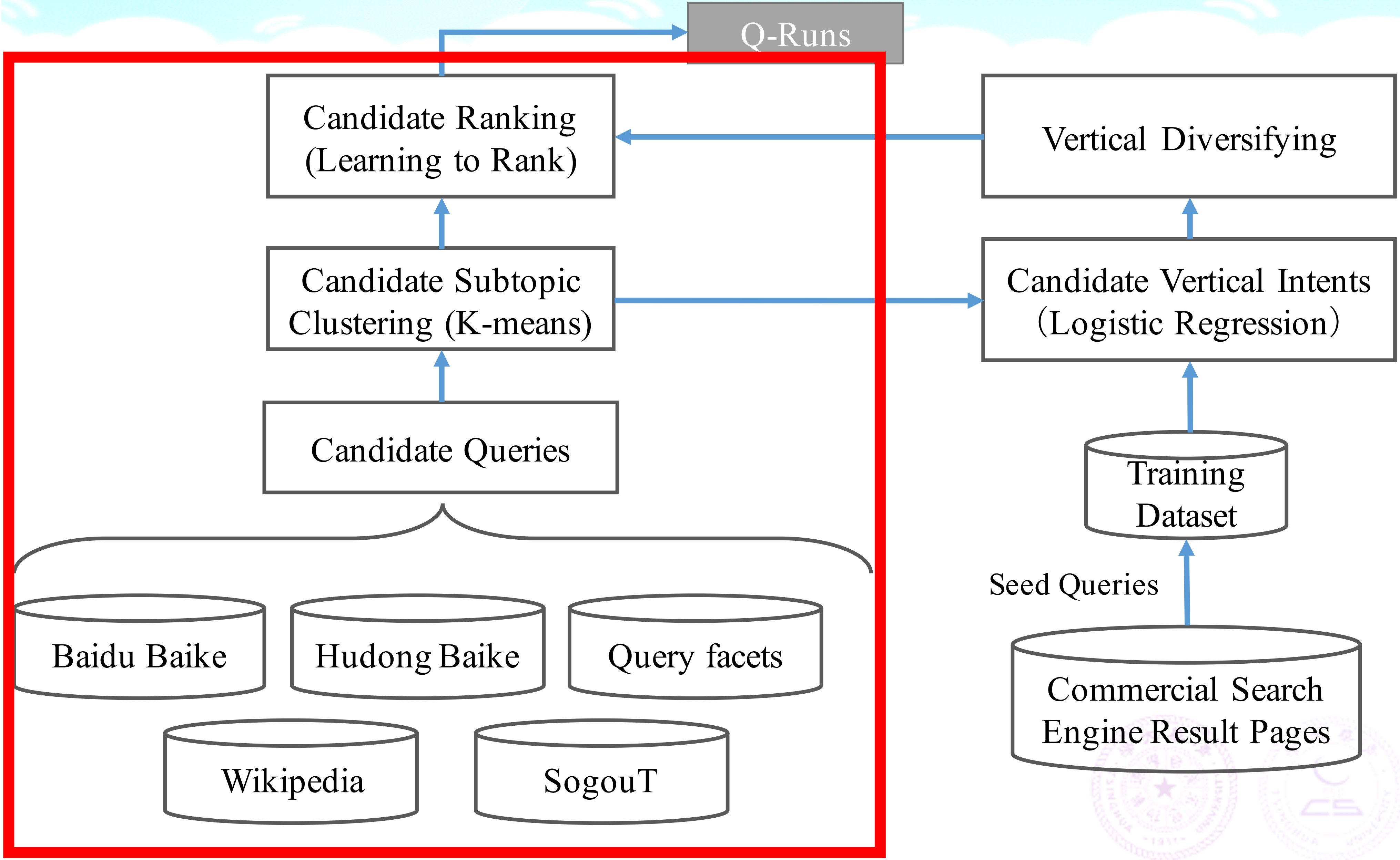
Tsinghua University

Jul 7th, 2016

NTCIR: Imine-2

- Search Intent and Task Mining
- Goal: explore and evaluate the technologies of understanding user intents behind the query and satisfying different user intents.
- Subtasks:
  - **Query Understanding(C)**: generate a diversified ranked list of subtopics with corresponding vertical intents
  - **Vertical Incorporating(C)**: return a diversified ranked list of not more than 100 results, including organic documents and virtual vertical results.

Query Understanding Framework



Candidate Mining

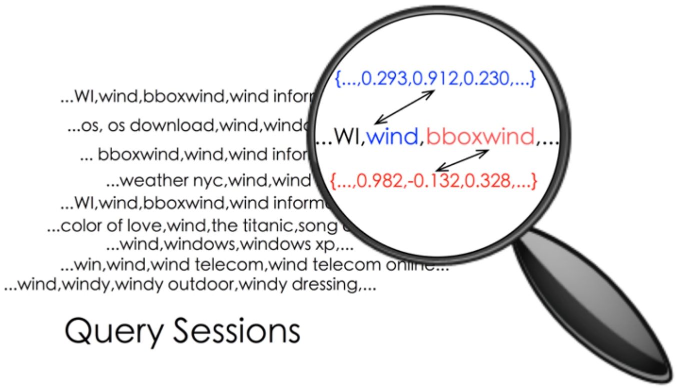
- Disambiguous items
  - Wikipedia, Baidu Baike and Hudong Baike
- Query Facets
  - Query completions and query suggestions
- Query Recommendations
  - Crawled from Sogou and Baidu
- Query reformulations
  - Extracted from SogouT dataset

Candidate Mining

- Query reformulations
  - SogouT dataset
  - Session detection [Catledge et al., 1995]
  - Find out sessions containing the task query
  - Queries in such sessions are regarded as query reformulations

Subtopic Candidate Clustering

- Background: Candidates from different resources are duplicated
- Goal: Find diversified candidates
- Cluster candidates with K-means algorithm
- Query vector representation
  - Word embedding trained based on SogouT dataset
  - Long query candidate: average word embedding of its' words
- Cluster subtopic candidates into n clusters
  - n = 5 or 10



Subtopic Candidate Clustering

- Subtopic candidate evaluation
  - clusters: $c_1, c_2, \cdots, c_n$
  - Inner distance

$$S_{inner}(q) = \frac{\sum_{q_k \in c_i, q_k \neq q} dist(q, q_k)}{|c_i| - 1}$$

  - Outer distance

$$S_{outer}(q) = \min\left\{\frac{\sum_{q_k \in c_j} dist(q, q_k)}{|c_j|}\right\} (j = 1, 2, \cdots, n, j \neq i)$$

  - Candidate score

$$S(q) = \frac{S_{outer}(q) - S_{inner}(q)}{\max\{S_{outer}(q), S_{inner}(q)\}}$$

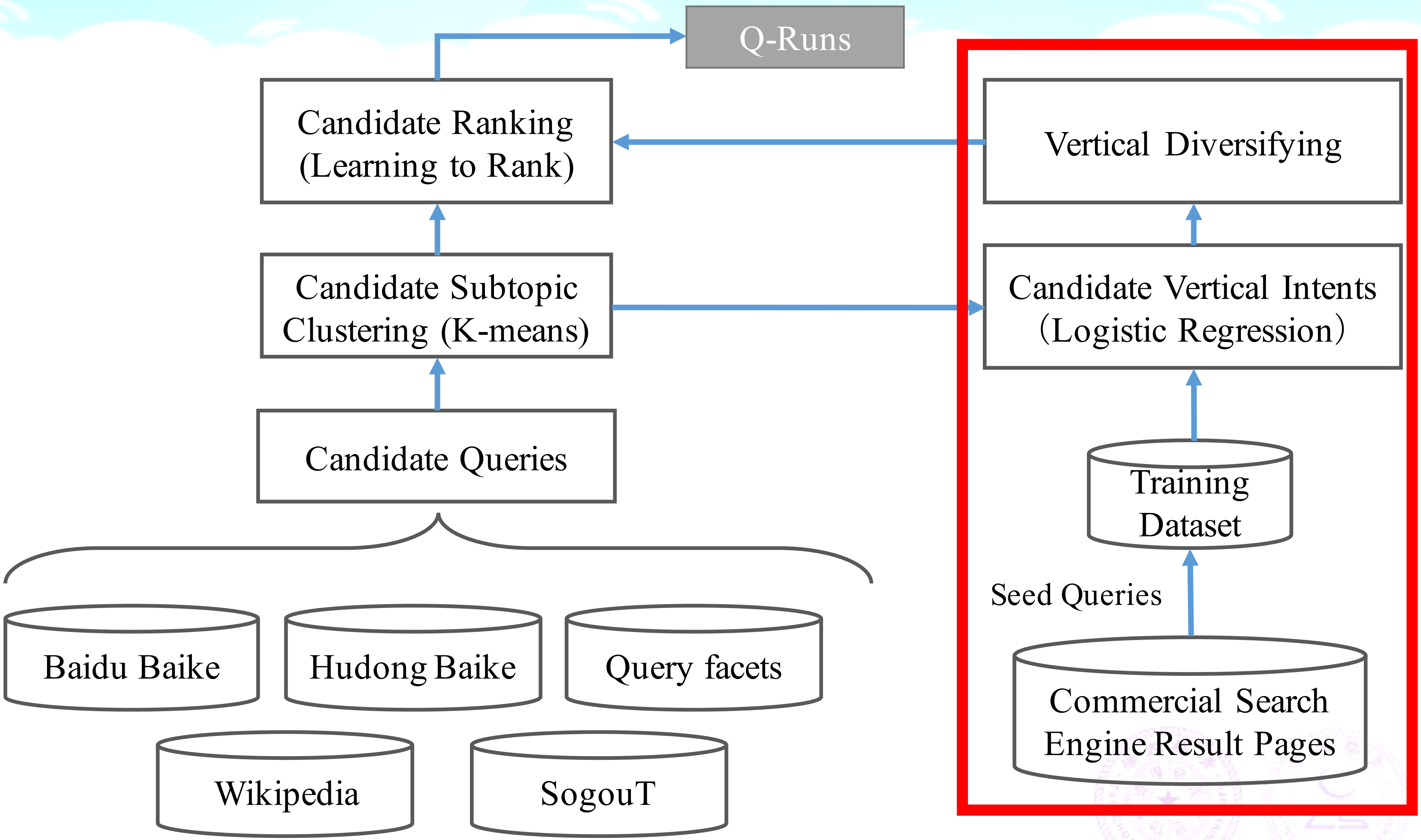
Candidate Ranking

- Background: Candidates are noisy
- Goal: Find high quality subtopic candidates
- Rank candidates with Learning To Rank algorithm
- Features:
  - Text similarity: length difference, Jaccard similarity, edit distance…
  - Word embedding: average, medium, top 3 average of cosine similarities
  - Search Result Similarity: number of shared results…
- Metric to optimize: NDCG@10
- Training set: Ranked Subtopics from NTCIR-11 Imine

Candidate Ranking

Method	No Normalization	Normalized by Sum	Normalized by Z-score
MART	0.5133	0.5426	0.5043
RankNet	0.5365	0.5328	0.6221
RankBoost	**0.6618**	0.6560	0.5467
AdaRank	0.5428	0.5560	0.5468
Coordinate Ascent	0.6157	0.5839	0.6352
LambdaRank	0.5349	0.4943	0.5796
LambdaMART	0.5306	0.5386	0.5199
ListNet	0.5543	0.5829	0.5783

Query Understanding Framework



Vertical Predicting

- Training Query
  - 1212 Seed Urls generated from an Open Directory Project (ODP)
  - News, videos, shopping, communities and games

Category	Subcategory	URL	Description
Video	Movie	http://movie.youku.com/	an online video site
		http://www.iqiyi.com/dianying/	an online video site
	Television	http://cctv.cntv.cn/	the official website of CCTV
		http://www.brtn.cn/	the official website of Beijing Television Station
	News of Movie	http://ent.sina.com.cn/film/	a news website about moives and celebrities
		http://ent.163.com/	a news website about moives and celebrities

Vertical Predicting

- Random walk
  - Based on a click-through bipartite graph
  - Expand candidate queries based on the seed urls
- Vertical Distribution
  - Vertical information collected from search engine result page
  - Vertical types extracted from CSS styles

Vertical Predicting

- Random walk
  - Based on a click-through bipartite graph
  - Expand candidate queries based on the seed urls
- Vertical Distribution
  - Vertical information collected from search engine result page
  - Vertical types extracted from CSS styles
  - Presentation score

$$P\text{-}score = \begin{cases} \dfrac{1}{\log(1+R_i)} & \text{if exists} \\ 0 & \text{else} \end{cases}$$

Vertical Predicting

- Model Construction
  - Logistic Regression
  - Six prediction models for each type of verticals
  - Input: Query representation based on word embedding
  - Output: presentation score of one specific type of vertical
  - Vertical type with the highest score is chosen to be the vertical intent

Vertical Diversification

- Background: strong effect of web vertical bias
  - Web verticals occupy a large portion of search results in practical
  - The prediction performance is limited

Vertical Diversification

- Background: strong effect of web vertical bias
  - Web verticals occupy a large portion of search results in practical
  - The prediction performance is limited
- Empirical rules for queries with web vertical intent
  - If the top result is a vertical result: replaced with the corresponding vertical intent
  - If there are two verticals in the top 3 positions: replaced with the highest ranked vertical

Query Understanding Results

RUNNAME	SYSTEM DESC.	D#-nDCG	V-score	QU-score
THUIR-QU-1A	Cluster all subtopic candidates into 10 clusters and select the candidate with the highest S(q) from each cluster.	0.5204	0.5579	0.5392
THUIR-QU-2A	Cluster all subtopic candidates into 5 clusters and select two candidates with the highest two S(q) from each cluster.	0.5550	0.5506	0.5528
THUIR-QU-1B	Rerank the 10 subtopics generated by THUIR-QU-1A with learning to rank algorithm.	0.5368	0.5763	**0.5565**
THUIR-QU-2B	Rerank the 10 subtopics generated by THUIR-QU-2A with learning to rank algorithm.	0.5436	0.5686	**0.5561**
THUIR-QU-3A	Cluster all subtopic candidates into 5 clusters and select the candidate with the highest ten S(q).	0.4973	0.5942	0.5458

Vertical Incorporating

- Retrieval Models for Organic Results:
  - Probabilistic model based on BM25 and our previous proposed word pair model
  - Relevance score for subtopic
    - $R(q, D) = W_{BM25} + \alpha \cdot W_{wp}$
    - $W_{BM25} = \sum_{i=1}^{m} log \frac{N-n(q_i)+0.5}{n(q_i)+0.5} \cdot \frac{f(q_i,D)\cdot(k_1+1)}{f(q_i,D)+k_1\cdot(1-b+b\cdot\frac{|D|}{avgdl})}$
    - $W_{wp} = \sum_{i=1}^{m} log \frac{N-n(q_i q_{i+1})+0.5}{n(q_i q_{i+1})+0.5} \cdot \frac{f(q_i q_{i+1},D)\cdot(k_1+1)}{f(q_i q_{i+1},D)+k_1\cdot(1-b+b\cdot\frac{|D|}{avgdl})}$
  - Relevance score for query
    - $R(Q, D) = \sum_{i=1}^{10} R(q_i, D) \times S(q_i)$

Vertical Incorporating

- Vertical Result Ranking
  - Vertical importance based on subtopic candidate score

$$I(v) = \alpha \cdot S\text{-}score(v)$$

  - v: a type of vertical intent
  - $S - score(v)$: the score of the subtopic which contains this vertical intent
  - α: importance weight, different for different types of verticals
  - Combination of $R(Q, D)$ and I(v)

Vertical Incorporating

- Vertical Result Ranking
  - Vertical importance based on subtopic candidate score

$$I(v) = \alpha \cdot S\text{-}score(v)$$

  - v: a type of vertical intent
  - $S - score(v)$: the score of the subtopic which contains this vertical intent
  - α: importance weight, different for different types of verticals
  - Combination of $R(Q, D)$ and I(v)

Vertical Incorporating Results

RUNNAME	D#-nDCG (unclear topics)	nDCG (clear topics)	D#-nDCG+nDCG (all topics)
THUIR-QU-1A	0.6677	0.5756	**0.6594**
THUIR-QU-2A	0.6664	0.5652	0.6573
THUIR-QU-1B	0.6594	0.5416	0.6488
THUIR-QU-2B	0.6632	0.5442	0.6525
THUIR-QU-3A	0.6429	0.5506	0.6346

Thank you!

chenye617@gmail.com
www.thuir.cn