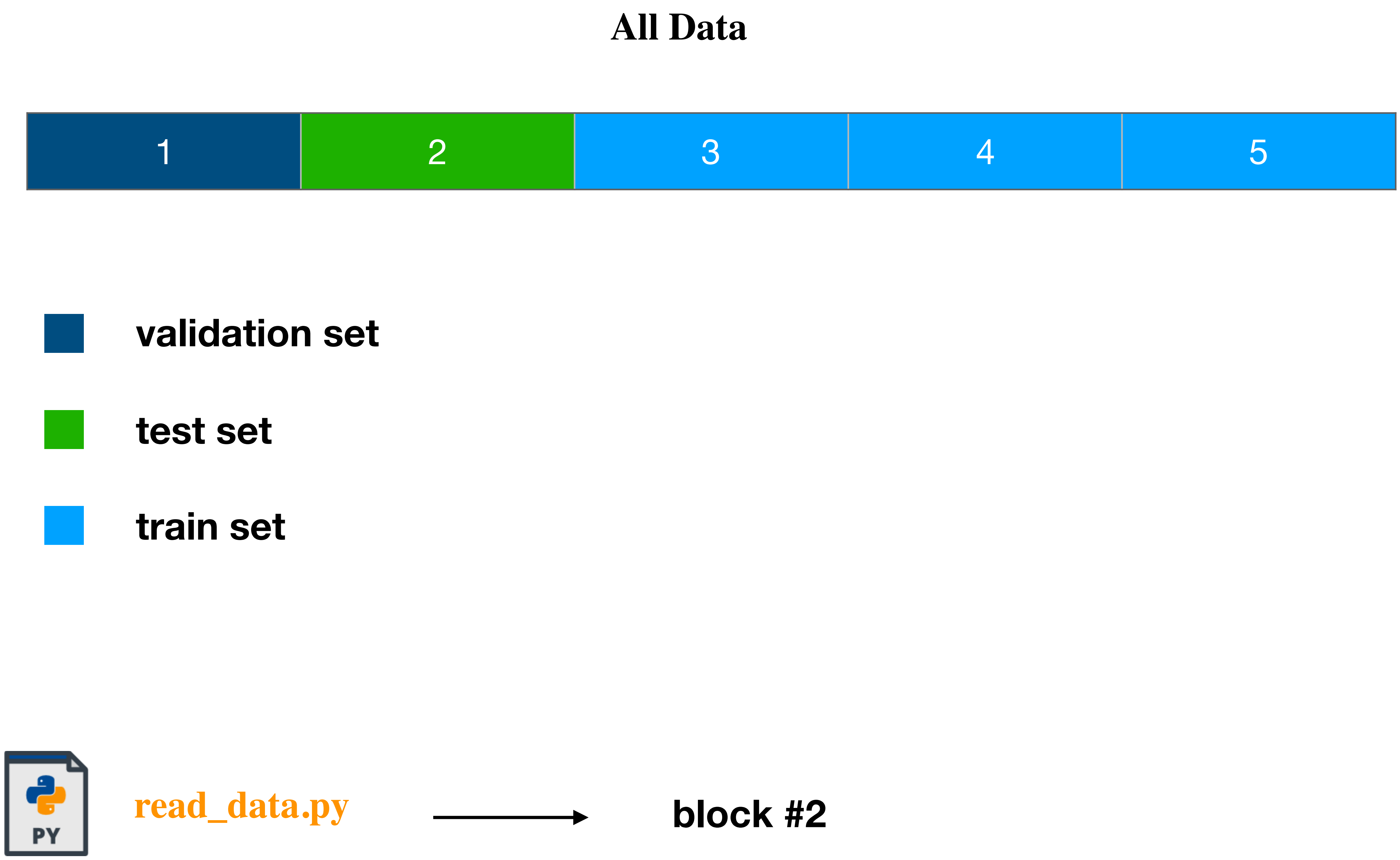
Data sets

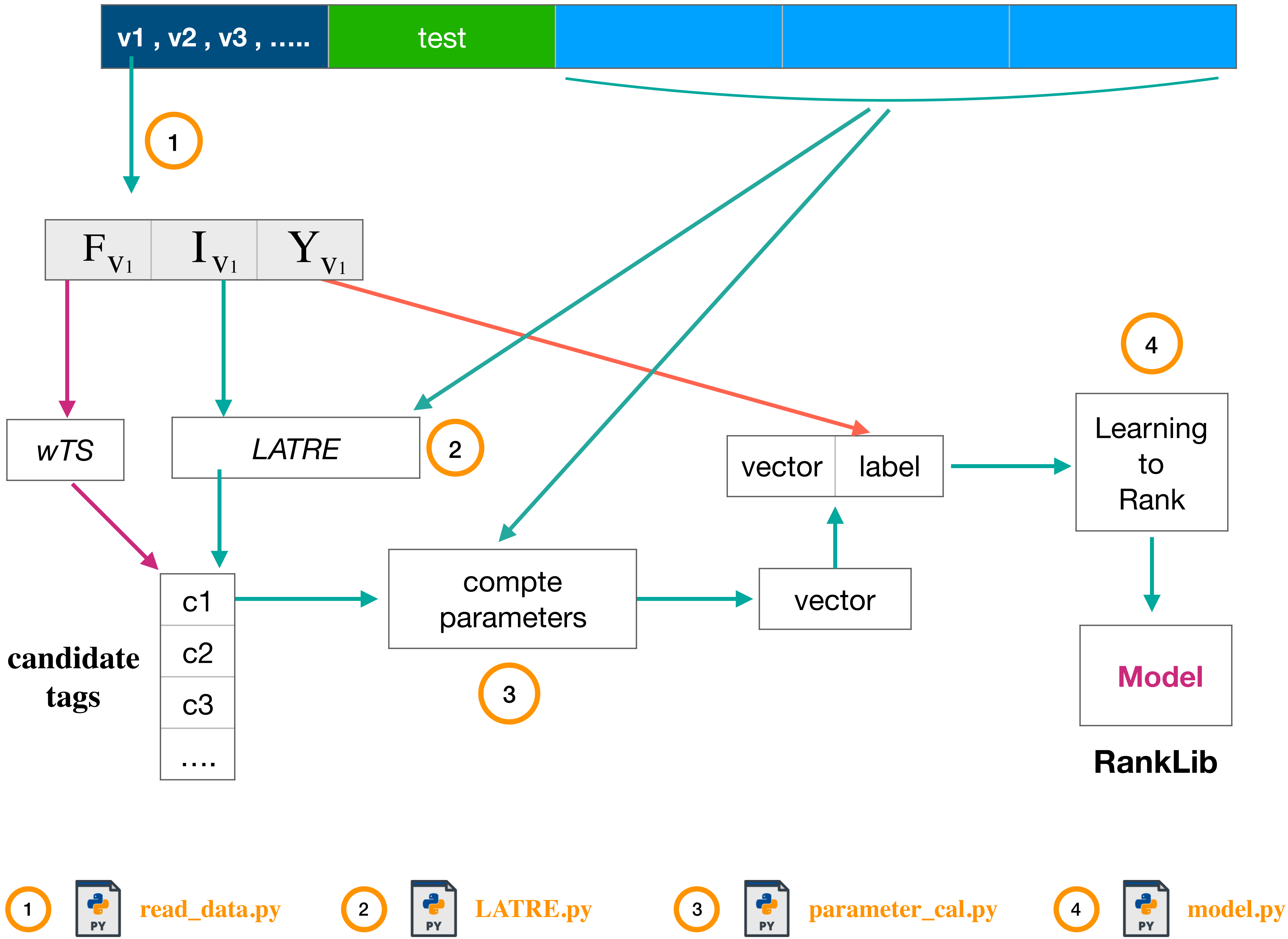
- YouTube
- LastFM
- BibSonomy
- YahooVideo

	small	medium	large
YouTube	2 to 5 tags/obj	6 to 9 tags/obj	10 to 74 tags/obj
LastFM	2 to 6 tags/obj	7 to 16 tags/obj	17 to 152 tags/obj

partition the data set to to ***small***, ***medium*** and ***large*** set

read_data.py → **block #1**


All Data
1
2
3
4
5
validation set
test set
train set
PY
read_data.py
block #2



v1 , v2 , v3 , ...
test
1
F V1
I V1
Y V1
wTS
LATRE
2
vector
label
4
Learning
to
Rank
candidate
tags
c1
c2
c3
....
compte
parameters
3
vector
Model
RankLib
1
PY
read_data.py
2
PY
LATRE.py
3
PY
parameter_cal.py
4
PY
model.py

2

LATRE.py

important functions	
`projectData(trainSet_tags, testItemSet):`	projects/filters the training data according to the tags in Io, and extracts rules from this projected data
`findsubsets_list(s):`	help to extract association rules
`associationRule(confidence_min=0.01, sup_min=2):`	exploits co-occurrence of tags by extracting association rules
`get_candidate_tag(initial_tags, confidence_min, sup_min):`	get candidate tags by given initial tags and min confidence …
`get_ranked_candidate(rules):`	return a list of tupel(pair) of candidate tag and the score of that Example: [(t1, score),(t2, score)]
`calculate_score(rules, tag):`	$\sum_{X \subseteq I_o} \theta(X \rightarrow c), \quad (X \rightarrow c) \in \mathcal{R}$

confidence

rules set

2 LATRE.py (Example)

		$\mathcal{I}$	$\mathcal{Y}$
$\mathcal{D}$	d_1	unicef children un united nations	$\emptyset$
	d_2	un climatechange summit environment	$\emptyset$
	d_3	climatechange islands environment	$\emptyset$
	d_4	children games education math	$\emptyset$
	d_5	education children unicef job	$\emptyset$
$\mathcal{T}$	t_1	unicef education haiti	?

Table 2. Projected training data for object t_1.

		$\mathcal{I}^t$	$\mathcal{Y}^t$
$\mathcal{D}_{t_1}$	$d_1^{t_1}$	unicef	children un united nations
	$d_4^{t_1}$	education	children games math
	$d_5^{t_1}$	unicef education	children job

- unicef $\xrightarrow{\theta=1.00}$ children
- {unicef$\wedge$education} $\xrightarrow{\theta=1.00}$ children
- education $\xrightarrow{\theta=0.50}$ math

3

parameter_cal.py

- _sum
- sum_plus
- vote
- voye_plus
- ts
- tf
- wts
- wtf

Tag Co-occurrence	*Sum*	Let X be a set of tags and c a candidate tag. $X \rightarrow c$ is an association rule and $\theta(X \rightarrow c)$ is its *confidence*. *Sum* is defined as: $Sum(c, I_o, \ell) = \sum_{X \subseteq I_o} \theta(X \rightarrow c), \quad (X \rightarrow c) \in \mathcal{R}, \lvert X \rvert \leq \ell,$ (1) where $\mathcal{R}$ is a set of association rules computed offline over the training set $\mathcal{D}$, and ℓ is the size limit for the antecedent X. As in our previous work by Belém et al. [2011], we use the *LATRE* algorithm to generate these rules.
	Sum$^+$	$Sum^+(c, I_o, k_x, k_c, k_r) = \sum_{x \in I_o} \theta(x \rightarrow c) \times Stab(x, k_x) \times Stab(c, k_c) \times Rank(c, x, k_r),$ (2) where $Stab(x, k_x)$ is defined in Eq. (10), and k_x, k_c and k_r are tuning parameters. $Rank(c, x, k_r)$ is equal to $k_r/(k_r + p(c, x))$, where $p(c, x)$ is the position of c in the ranking of candidates according to the confidence of the corresponding association rule (whose antecedent is x).
	Vote	$Vote(c, I_o) = \sum_{x \in I_o} j, \text{ where } j = \begin{cases} 1 & \text{if } (x \rightarrow c) \in \mathcal{R} \\ 0 & otherwise \end{cases}$ (3)
	Vote$^+$	$Vote^+(c, I_o, k_x, k_c, k_r) = \sum_{x \in I_o} j \times Stab(x, k_x) \times Stab(c, k_c) \times Rank(c, x, k_r), \text{ where } j = \begin{cases} 1, & if \quad x \rightarrow c \in \mathcal{R} \\ 0, & otherwise \end{cases}$ (4)
Descriptive Power	*Term Spread (TS)*	$TS(c, o) = \sum_{F_o^i \in F_o} j, \text{ where } j = \begin{cases} 1 & \text{if } c \in F_o^i \\ 0 & \text{otherwise} \end{cases}$ (5)
	Term Frequency (TF)	$TF(c, o) = \sum_{F_o^i \in F_o} tf(c, F_o^i),$ (6) where $tf(c, F_o^i)$ is the number of occurrences of c in textual feature F_o^i of object o.
	Weighted Term Spread (wTS)	Let the *Feature Instance Spread* of a feature F_o^i associated with object o, $FIS(F_o^i)$, be the average TS over all terms in F_o^i. We define the *Average Feature Spread* $AFS(F^i)$ as the average $FIS(F_o^i)$ over all instances of F^i associated with objects in the training set $\mathcal{D}$. The wTS is defined as: $wTS(c, o) = \sum_{F_o^i \in F_o} j, \text{ where } j = \begin{cases} AFS(F^i) & \text{if } c \in F_o^i \\ 0 & \text{otherwise} \end{cases}$ (7)
	Weighted Term Frequency (wTF)	$wTF(c, o) = \sum_{F_o^i \in F_o} tf(c, F_o^i) \times AFS(F^i)$ (8)

3 **parameter_cal.py**

iff	Discriminative Power	*Inverse Feature Frequency (IFF)*	$$IFF(c) = log \frac{\lvert\mathcal{D}\rvert + 1}{f_c^{tag} + 1}, \quad (9)$$ where f_c^{tag} is the number of objects in the training set $\mathcal{D}$ that contain c *associated as a tag*.
stab		*Stability (Stab)*	$$Stab(c, k_s) = \frac{k_s}{k_s + \lvert k_s - log(f_c^{tag})\rvert}, \quad (10)$$ where the tuning parameter k_s represents the "ideal frequency" of a term in the data collection.

4 **model.py**

..... → vector | label ..... → outputs file in output directory ready for rankLib

features

label

```
0 qid:160 1:0.3333333333333333 2:0.3333333333333333 3:0.0008680555555555555 4:1 5:0.0026041666666666665 6:0 7:0 8:0 9:0.0 10:6.55953191458 11:0.0625 #z2DWGtmZA9I
0 qid:26617 1:0.3333333333333333 2:0.3333333333333333 3:0.002314814814814147 4:1 5:0.006944444444444444 6:0 7:0 8:0 9:0.0 10:7.7833073462 11:0.25 #z2DWGtmZA9I
0 qid:1467 1:0.3333333333333333 2:0.3333333333333333 3:0.0001929012345679012 4:1 5:0.0005787037037037037 6:0 7:0 8:0 9:0.0 10:5.78182734599 11:0.02777777777777776 #z2DW
0 qid:11046 1:0.3333333333333333 2:0.3333333333333333 3:0.0007936507936507937 4:1 5:0.002380952380952381 6:0 7:0 8:0 9:0.0 10:7.31330371695 11:0.14285714285714285 #z2DWG
0 qid:2268 1:0.10526315789473684 2:0.10526315789473684 3:0.0001629460648525338 4:1 5:0.0015479876160990710 6:0 7:0 8:0 9:0.0 10:6.50237350074 11:0.058823529411764705 #J-P1
0 qid:1582 1:0.10526315789473684 2:0.10526315789473684 3:9.23361034164358e-05 4:1 5:0.0008771929824561403 6:0 7:0 8:0 9:0.0 10:6.34822282091 11:0.05 #J-P1EGnujXA
0 qid:4961 1:0.14285714285714285 2:0.14285714285714285 3:6.07385811467444e-05 4:1 5:0.00042517006802721088 6:0 7:0 8:0 9:0.0 10:5.63154514294 11:0.023809523809523808 #MUz
1 qid:1344 1:0.10714285714285714 2:0.10714285714285714 3:0.00014172335600907027 4:1 5:0.0013227513227513227 6:2 7:2.495003006861588 8:2 9:2.495003006861588 10:7.09016016
0 qid:4162 1:0.07142857142857142 2:0.07142857142857142 3:3.188775510204081e-05 4:1 5:0.00044642857142857140 6:0 7:0 8:0 9:0.0 10:6.34822282091 11:0.05 #MUzOYRsXeEc
0 qid:176 1:0.07142857142857142 2:0.07142857142857142 3:2.429543245869776e-05 4:1 5:0.0003401360544217687 6:0 7:0 8:0 9:0.0 10:6.30170280527 11:0.047619047619047616 #MUz
0 qid:5523 1:0.07142857142857142 2:0.07142857142857142 3:2.237737200143215e-05 4:1 5:0.00031328320802005010 6:0 7:0 8:0 9:0.0 10:6.39701298508 11:0.05263157894736842 #MUz
```

tag id

object id

```
3 qid:1 1:1 2:1 3:0 4:0.2 5:0 # 1A
2 qid:1 1:0 2:0 3:1 4:0.1 5:1 # 1B
1 qid:1 1:0 2:1 3:0 4:0.4 5:0 # 1C
1 qid:1 1:0 2:0 3:1 4:0.3 5:0 # 1D
1 qid:2 1:0 2:0 3:1 4:0.2 5:0 # 2A
2 qid:2 1:1 2:0 3:1 4:0.4 5:0 # 2B
1 qid:2 1:0 2:0 3:1 4:0.1 5:0 # 2C
1 qid:2 1:0 2:0 3:1 4:0.2 5:0 # 2D
2 qid:3 1:0 2:0 3:1 4:0.1 5:1 # 3A
3 qid:3 1:1 2:1 3:0 4:0.3 5:0 # 3B
4 qid:3 1:1 2:0 3:0 4:0.4 5:1 # 3C
1 qid:3 1:0 2:1 3:1 4:0.5 5:0 # 3D
```

→ LETOR format

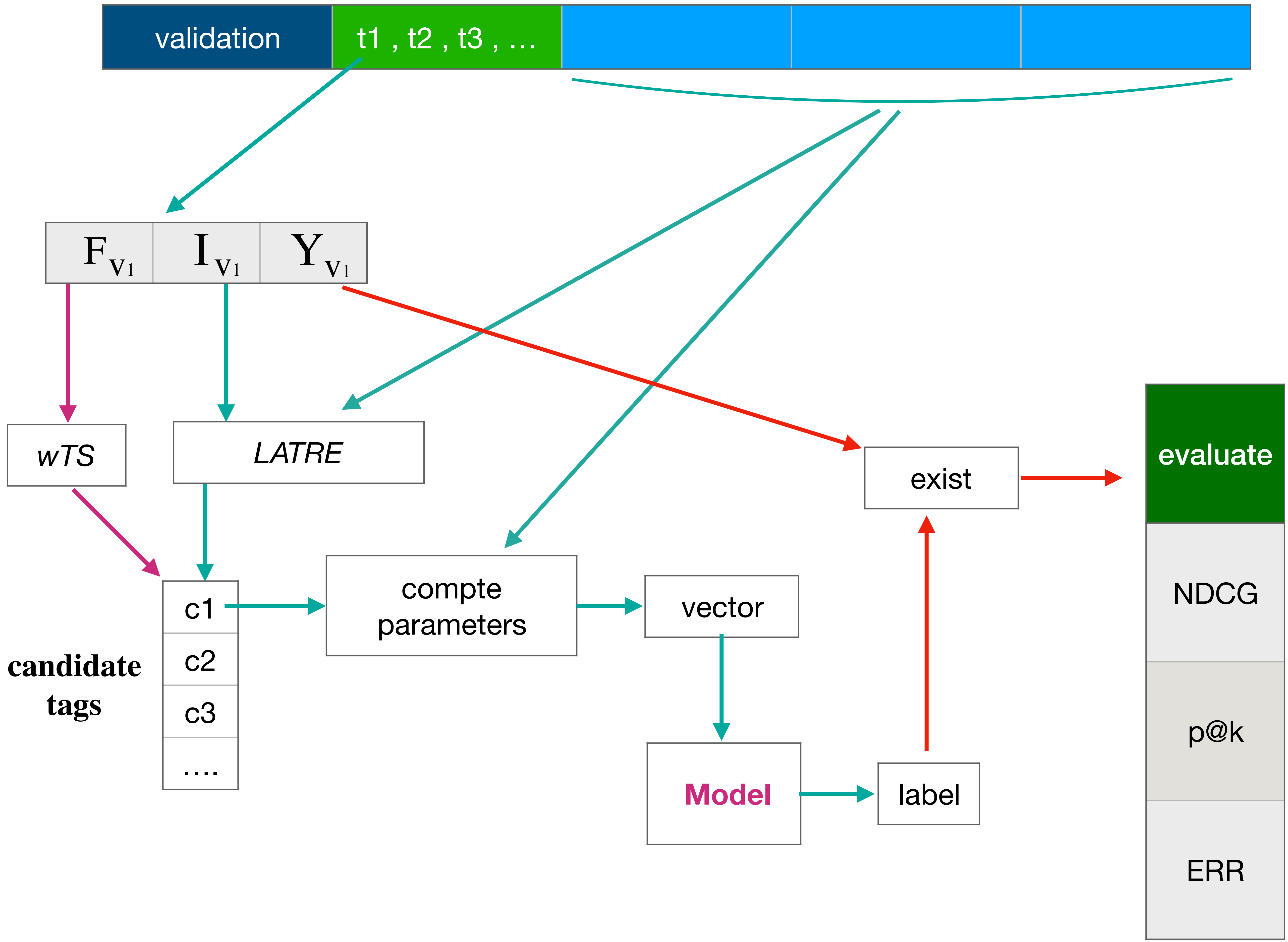
gather_all_ourputs.py

gather all outputs together

```
1 qid:1344 1:0.10714285714285714 2:0.10714285714285714 3:0.00014172335600907027 4:1 5:0.0013227513227513227 6:2 7:2.495003006861588 8:2 9:2.495003006861588 10:7.09016016
1 qid:2417 1:0.17647058823529413 2:0.17647058823529413 3:0.0007414730598121602 4:1 5:0.004201680672268907 6:2 7:2.495003006861588 8:2 9:2.495003006861588 10:7.3133037169
1 qid:7445 1:0.04838709677419355 2:0.04838709677419355 3:5.2029136316337144e-05 4:1 5:0.001075268817204301 6:1 7:1.3388401785714283 8:1 9:1.3388401785714283 10:8.0064508
1 qid:7266 1:0.03225806451612903 2:0.03225806451612903 3:1.0839403399236905e-06 4:1 5:3.360215053763441e-05 6:1 7:1.3388401785714283 8:1 9:1.3388401785714283 10:5.958758
1 qid:940 1:1.0 2:1.0 3:0.0125 4:1 5:0.0125 6:0 7:0 8:0 9:0.0 10:6.34822282091 11:0.05 #ABqBMw9aY08
1 qid:1092 1:0.8571428571428571 2:0.8571428571428571 3:0.0009003601440576229 4:1 5:0.0010504201680672268 6:1 7:1.1561628282901595 8:1 9:1.1561628282901595 10:5.837397197
1 qid:874 1:0.11475409836065574 2:0.11475409836065574 3:5.972110245155126e-06 4:1 5:5.2042674993494666e-05 6:2 7:2.495003006861588 8:2 9:2.495003006861588 10:5.233862175
1 qid:4607 1:0.3888888888888889 2:0.3888888888888889 3:0.0010288065843621398 4:1 5:0.0026455026455026454 6:1 7:1.1561628282901595 8:1 9:1.1561628282901595 10:7.313303716
1 qid:2417 1:0.17647058823529413 2:0.17647058823529413 3:0.0004943153732081067 4:1 5:0.0028011204481792713 6:0 7:0 8:0 9:0.0 10:7.31330371695 11:0.14285714285714285 #9uD
1 qid:3790 1:0.13333333333333333 2:0.13333333333333333 3:0.0009876543209876541 4:1 5:0.007407407407407406 6:0 7:0 8:0 9:0.0 10:8.00645089751 11:0.3333333333333333 #2Bvu1
1 qid:12541 1:0.7777777777777778 2:0.7777777777777778 3:0.005401234567901234 4:1 5:0.006944444444444444 6:0 7:0 8:0 9:0.0 10:7.1955206813 11:0.125 #3nIrACY08LM
```

```
0 qid:317 1:0.5476190476190477 2:0.5476190476190477 3:2.7507486820325882e-05 4:1 5:5.0231062889290733e-05 6:0 7:0 8:0 9:0.0 10:5.01071862396 11:0.012658227848101266 #x
0 qid:5791 1:0.04 2:0.04 3:6.349206349206348e-06 4:1 5:0.00015873015873015873 6:0 7:0 8:0 9:0.0 10:7.31330371695 11:0.14285714285714285 #YVSW1TbCyJs
0 qid:26925 1:0.034482758620689655 2:0.034482758620689655 3:2.1233225751656187e-05 4:1 5:0.0006157635467980296 6:0 7:0 8:0 9:0.0 10:8.29413296996 11:0.5 #x2P70C0UO24
0 qid:1794 1:0.2 2:0.2 3:0.0003703703703703704 4:1 5:0.0018518518518518517 6:0 7:0 8:0 9:0.0 10:7.09016016564 11:0.1111111111111111 #d234Xpg5aNc
0 qid:5446 1:0.08695652173913043 2:0.08695652173913043 3:2.7005130974885227e-05 4:1 5:0.00031055900621118014 6:0 7:0 8:0 9:0.0 10:6.34822282091 11:0.05 #CM20rsiYMJQ
0 qid:4676 1:0.022222222222222223 2:0.022222222222222223 3:7.482229704451927e-07 4:1 5:3.367003367003367e-05 6:0 7:0 8:0 9:0.0 10:6.90783860884 11:0.09090909090909091
0 qid:9654 1:0.05128205128205128 2:0.05128205128205128 3:2.9884645269260652e-05 4:1 5:0.0005827505827505828 6:0 7:0 8:0 9:0.0 10:6.90783860884 11:0.09090909090909091 #
0 qid:372141 1:0.025 2:0.025 3:4.734848484848486e-06 4:1 5:0.00018939393939393942 6:0 7:0 8:0 9:0.0 10:8.29413296996 11:0.5 #kiLWytE_N8A
0 qid:7266 1:0.06 2:0.06 3:3.6363636363636366e-06 4:1 5:6.0606060606060605e-05 6:0 7:0 8:0 9:0.0 10:5.95875805415 11:0.03333333333333333 #YVSW1TbCyJs
```

$\longrightarrow$ rankLib $\xrightarrow{\text{Random Forest (RF)}}$ **Model**


validation
t1 , t2 , t3 , …
F_{V_1}
I_{V_1}
Y_{V_1}
wTS
LATRE
exist
evaluate
compte parameters
vector
c1
c2
c3
….
candidate tags
NDCG
p@k
Model
label
ERR