Analysis of Similarity Measures between Short Text for the NTCIR-12 Short Text Conversation Task

Kozo Chikai

Graduate school of information science and technology,
Osaka University

Yuki Arase

Graduate school of information science and technology,
Osaka University

OSAKA UNIVERSITY
Live Locally, Grow Globally

AGENDA

① Introduction

➢Overview of STC task and goal of our study

② Methods & System Design

- WTMF model
- Word2vec
- Random Forest and its learning method

③ Experiment

④ Result & Discussion

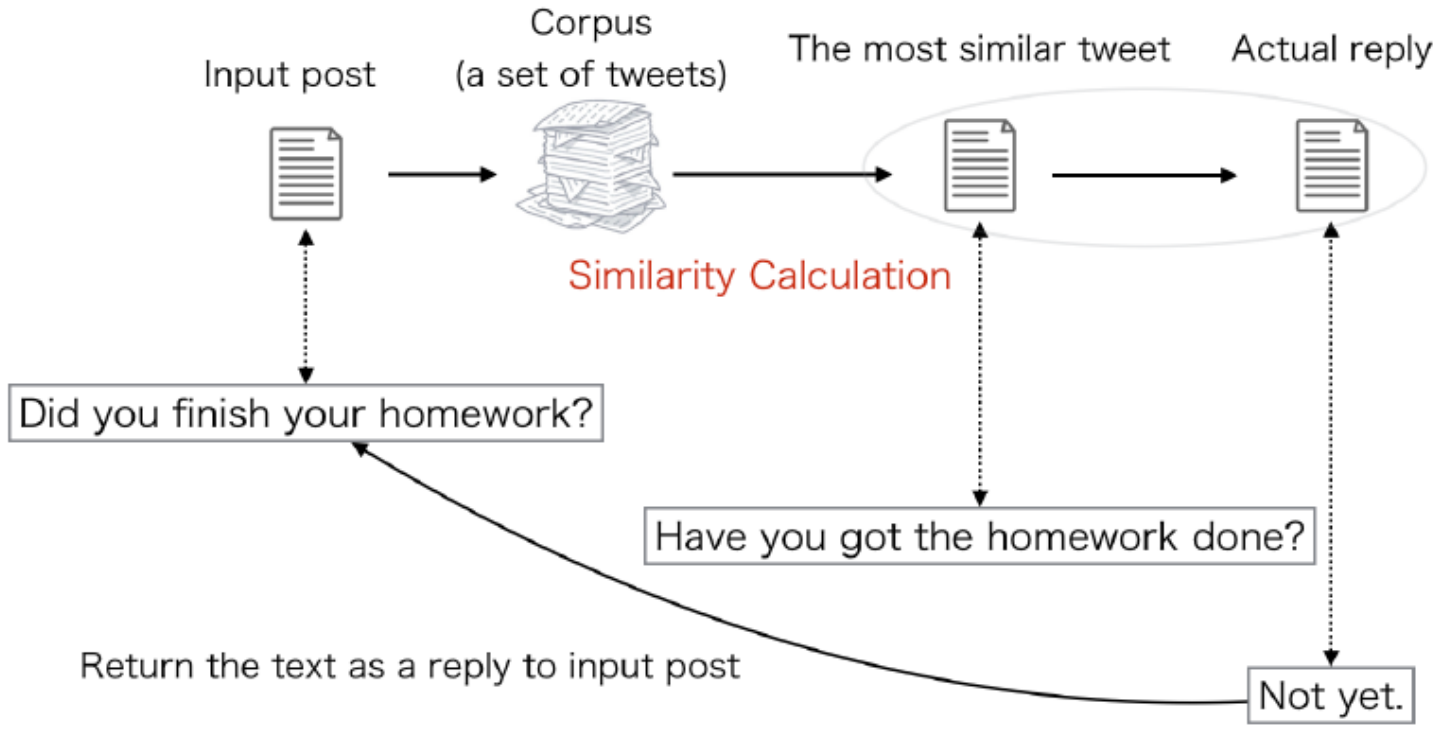
INTRODUCTION

- Short Text Conversation (STC) Task

  ✓Retrieve suitable replies from a pool of tweets

  → System's output = Ranking of replies

  ✓System design : Input → Text (similar to input) → Output (reply)



AGENDA

① Introduction

➢Overview of STC task and Goal of our study

② Methods & System Design

- WTMF model
- Word2vec
- Random Forest and its learning method

③ Experiment

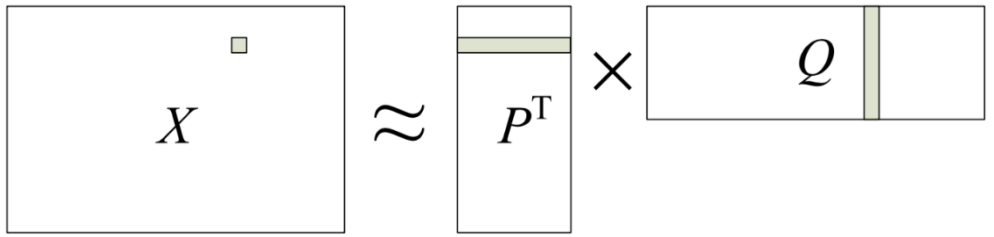
④ Result & Discussion

METHODS

- Weighted Text Matrix Factorization (WTMF)
  - ✓Simply models the text-to-word information without leveraging the correlation between short text.
  - ✓X: term-document matrix (row: document vector, cell: TF-IDF value)
    → Factorized into two matrix such that $X \approx P^T Q$ (where $P \in \mathbb{R}^{K \times M}$, $\mathrm{Q} \in \mathbb{R}^{K \times N}$)

$$\sum_{i,j} W_{i,j}(P_{\cdot,i} \cdot Q_{\cdot,j} - X_{i,j}) + \lambda||P||_2^2 + \lambda||Q||_2^2$$

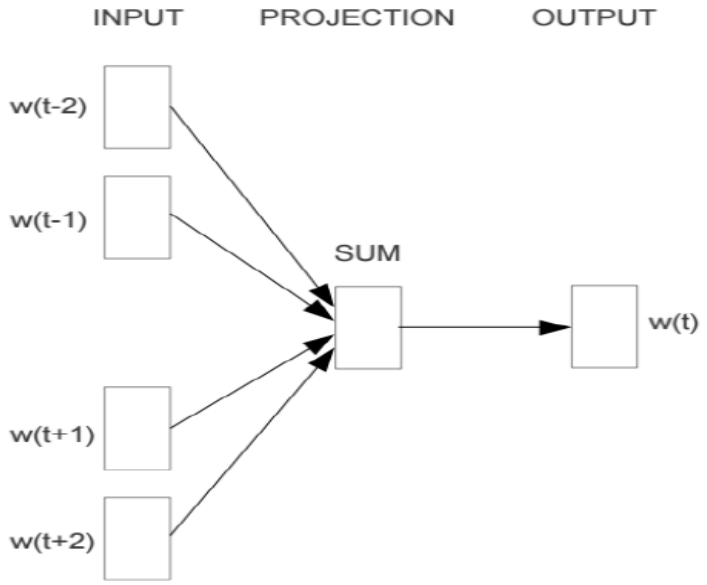
$$W_{i,j} = \begin{cases} 1 & (\text{if } X_{i,j} \neq 0) \\ w_m & (\text{if } X_{i,j} = 0) \end{cases}$$



Source：Modeling Sentences in the Latent Space

METHODS

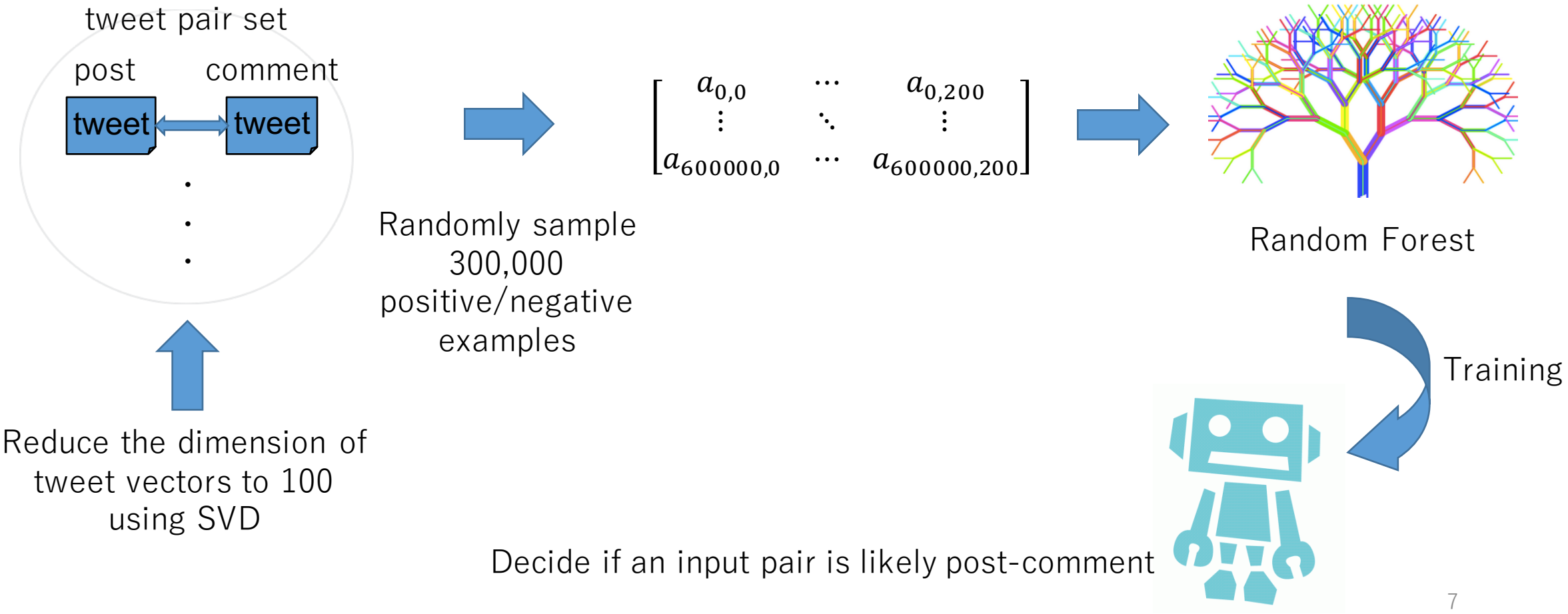
- Word Embedding
  - ✓Generate word vectors using a neural network
  - ✓Liner operation between words is possible
    (ex: ”king”-”man”+”woman” = “queen”)
  - ✓Adopted the implementation known as Word2vec tool
    → Output: low-dimensional vector representing words



Source：Efficient estimation of word representations in vector space

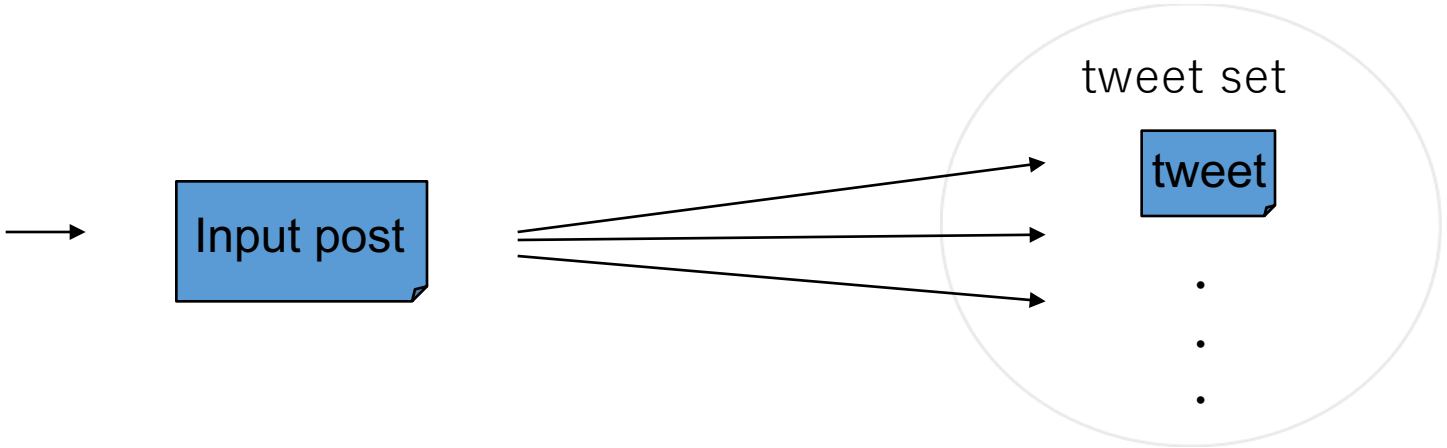
METHODS

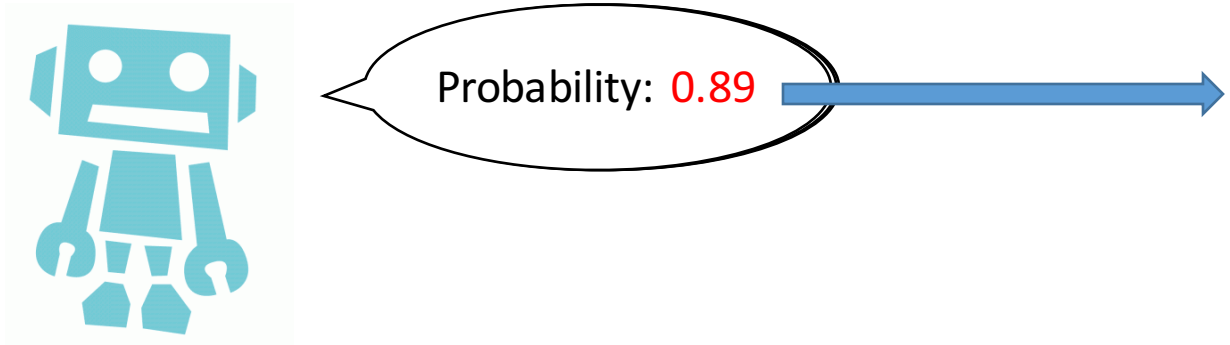
- Learning Post-Comment Pairs using Random Forest



METHODS

- Combination of Random Forest and TF-IDF





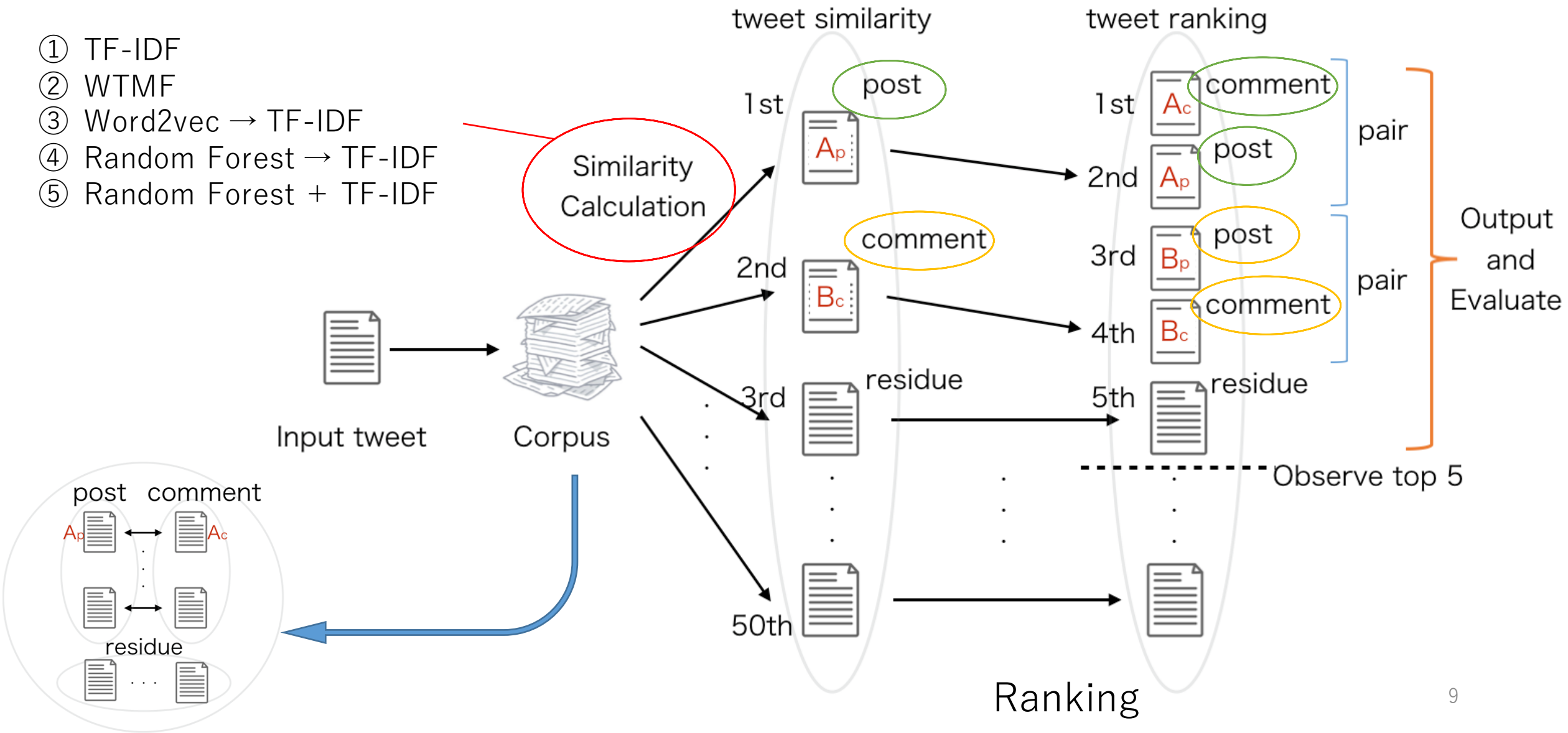
Combine probability output by Random Forest and the value of TF-IDF

A) **Random Forest → TF-IDF**
Tweets have the probability more than 0.5 → TF-IDF

B) **Random Forest + TF-IDF**
Sum the probability and TF-IDF (cos similarity) value

DESIGN OF CONVERSATION SYSTEM

① TF-IDF
② WTMF
③ Word2vec → TF-IDF
④ Random Forest → TF-IDF
⑤ Random Forest + TF-IDF



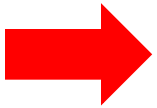
AGENDA

① Introduction

➢Overview of STC task and Goal of our study

② Methods & System Design

- WTMF model
- Word2vec
- Random Forest and its learning method



③ Experiment

④ Result & Discussion

EXPERIMENT

- Setting

  ✓Goal: Evaluate a ranked list of potential replies to an input tweet

  ✓10 annotators assigned a score to each reply (tweet)

  → score: +2, +1, and 0

  → The larger score the tweet has, the better reply it is!

  ✓Evaluation criteria: nDCG@1, nERR@5, and Accuracy

  → Each method has a value between 0.0 and 1.0 using these criteria

  → The larger score is better

Name of method	Technique
Oni-J-R1	④Random Forest → TF-IDF
Oni-J-R2	⑤Random Forest ＋ TF-IDF
Oni-J-R3	①TF-IDF
Oni-J-R4	③Word2Vec → TF-IDF
Oni-J-R5	②Weighted Text Matrix Factorization

OVERVIEW OF THIS PRESENTATION

① Introduction

➢Overview of STC task and Goal of our study

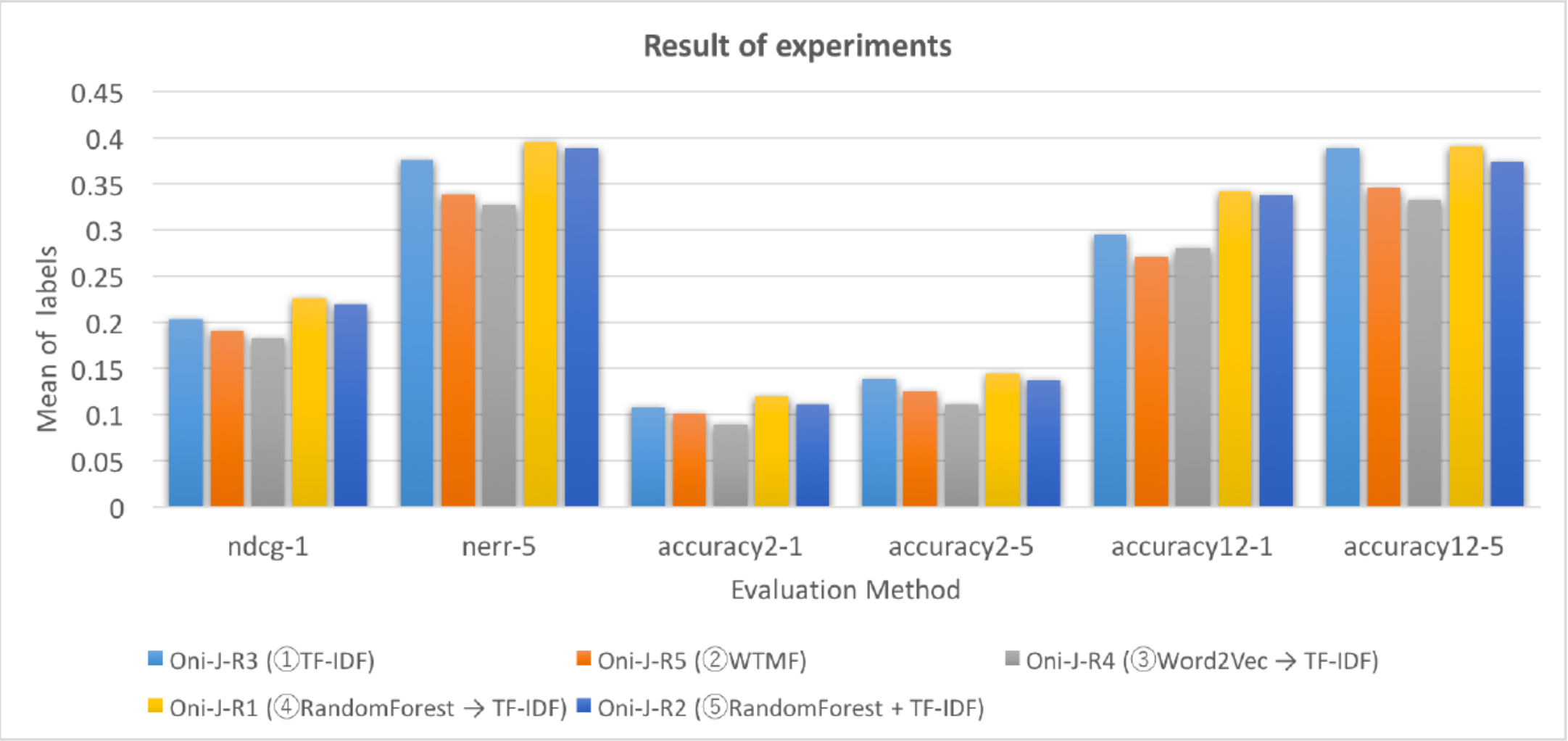
② Methods & System Design

- WTMF model
- Word2vec
- Random Forest and its learning method

③ Experiment

④ Result & Discussion

RESULT


Result of experiments
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
Mean of labels
ndcg-1
nerr-5
accuracy2-1
accuracy2-5
accuracy12-1
accuracy12-5
Evaluation Method
Oni-J-R3 (①TF-IDF)
Oni-J-R5 (②WTMF)
Oni-J-R4 (③Word2Vec → TF-IDF)
Oni-J-R1 (④RandomForest → TF-IDF)
Oni-J-R2 (⑤RandomForest + TF-IDF)

DISCUSSION

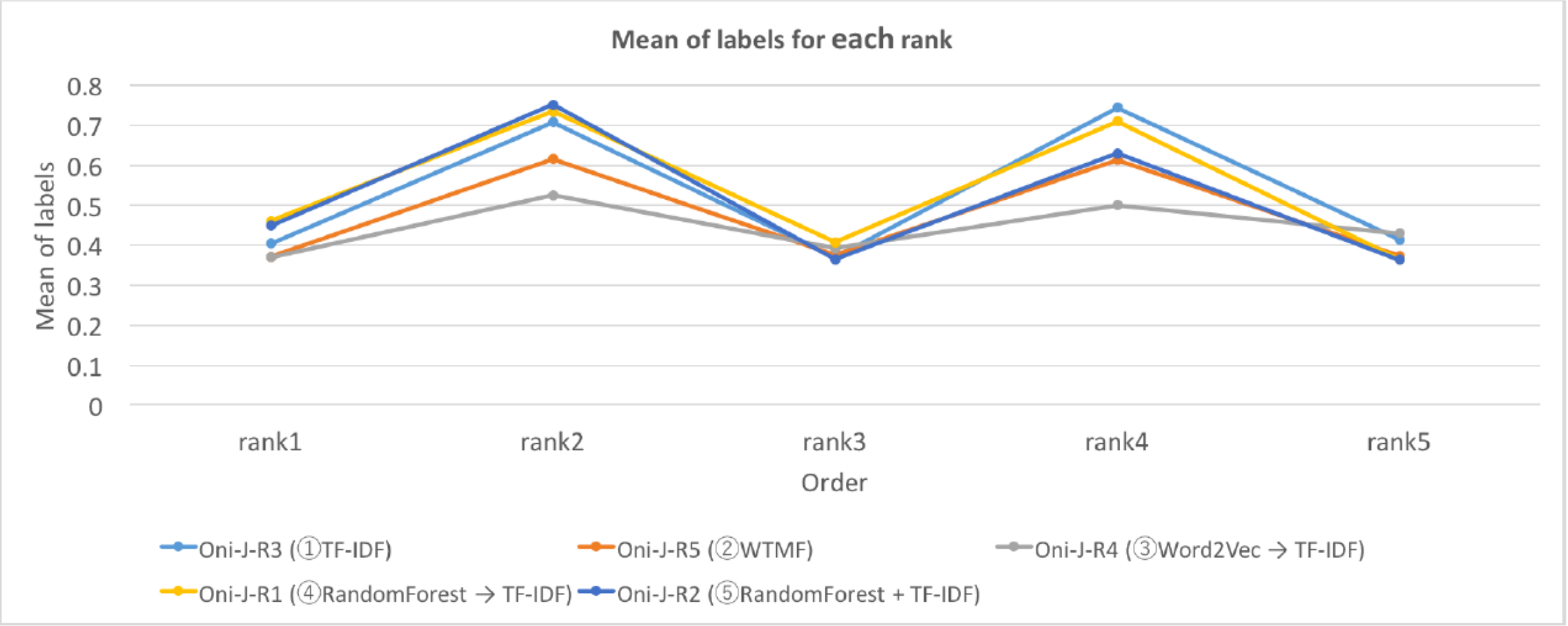


Figure 8: Mean of labels for each rank

DISCUSSION

Table 2: Effect of R1 and R3 due to the number of characters

input	$0 \leq$ # of characters ≤ 20		$20 <$ # of characters ≤ 40		$40 <$ # of characters ≤ 60		$60 <$ # of characters	
system	R1	R3	R1	R3	R1	R3	R1	R3
rank1	0.810	0.587	0.424	0.408	0.376	0.279	0.350	0.350
rank2	0.860	0.793	0.722	0.709	0.703	0.685	0.691	0.644
rank3	0.503	0.550	0.401	0.328	0.371	0.365	0.375	0.341
rank4	0.583	0.670	0.731	0.749	0.694	0.732	0.775	0.806
rank5	0.227	0.600	0.359	0.417	0.471	0.356	0.397	0.294

(Oni-J-R1: ‘Random Forest → TF-IDF’, Oni-J-R3’s method: ‘TF-IDF’)

DISCUSSION(2)

Table 3: Example of replies generated by Oni-J-R1

Input post	フォローありがとうございます！ Thank you for following!
rank1	こちらこそ、懐かしい画像をありがとうございました♪ よろしくお願いします！ Thank you too for sharing the nostalgic photos! Keep in touch.
rank2	RT ありがとです♪ フォローさせていただきました！ Thank you for retweeting. I had followed you!
rank3	お休みなさい♪ 今年も宜しくお願いします！ Good night. Wish we'll have fun together!
rank4	寝ます。おやすみなさい。 さっそく今日から仕事だ！ Going to bed. Good night. Gotta work from today!
rank5	んぅ いくぅ hmmm...

(Oni-J-R1's method is 'Random Forest → TF-IDF')

DISCUSSION(3)

Table 4: Example of output from Oni-J-R3 and Oni-J-R4

Input post	LINE 使えんので用のある人はこっちでお願いします！ Cannot use LINE now. Please contact me via Twitter!	
Method	R3	R4
rank1	新しいアカウント作ったから教えてー！ Let me know your account! I've just created one.	PC で扱うようなでかい解像度で色々したりするんでなければいよいよ十分な感じなんです？ It should be satisfactory unless you process high-resolution images that require PC-like performance.
rank2	LINE 使えるようになった？ Has LINE recovered?	んー、パソコンで使えた機能の大半はもう使えてるよね。 単純に使える、だけだけど。　次の 8 コアのスマートフォン向けの cpu が出るからそれで単純に使える機能から卒業して、もうスマートフォンで十分になる機能は出るだろう Most functions in PCs are already covered by smart phones, although they are just functioning... They should be actually usable with the next 8-core CPU for smart phones.
rank3	パソコンの LINE ツンデレすぎるよ！ もう！ 早く携帯よとどけ！ LINE app for PC is hard to use... Get me a cell phone!	単発でやる環境に長くいたせいで、最近キャンペ参加できるようになったのですがなかなか上手い感じにキャンペ用キャラが組めないんですよね…… I cannot figure out how to organize my party for online campaign as I've been playing alone...

(Oni-J-R3's method: 'TF-IDF', Oni-J-R4: 'Word2vec → TF-IDF')

SUMMARY

- In this study, we compare the conventional methods to handle short text.
- We use unsupervised methods to generate vectors
- We also use supervised methods to learn if a pair of tweets can be a post-comment pair.
- As the result of the formal run in STC, ①Random Forest → TF-IDF outperformed other methods.
- The method using Word2vec shows interesting results in some context, however there is room for improvement in this method.

REFERENCES

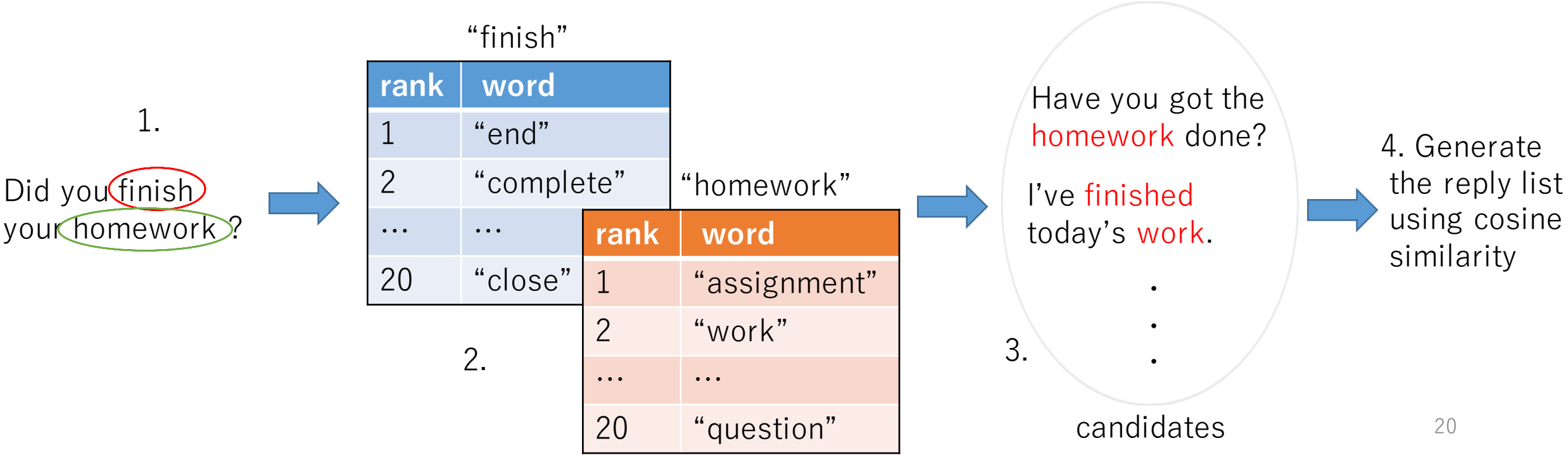
- Modeling Sentences in the Latent Space
  W. Guo, and M. Diab
  In Proceeding of ACL (2012)

- Distributed Representations of Sentences and Documents
  T. Mikolov, Q. V. Le
  In Proceedings of ICML(2014)

- Overview of the NTCIR-12 short text conversation task
  L. Shang, T. Sakai, Z. Lu, H. Li, R. Higashinaka, and Y. Miyao
  In Proceedings of the NTCIR Workshop Meeting on Evaluation of Information Access Technologies(2016)

APPENDIX (1)

- The way of Word2Vec → TF-IDF
  1. Extract nouns and verbs in an input post
  2. Extract 20 most similar words for each noun and verb using word2vec
  3. Search tweets containing these words in the corpus
  4. Generate the reply list from these tweets in the same manner with ①TF-IDF



APPENDIX (2)

- Method of Generating the reply list
  - ✓We use the relationship of a pair of post-comment in Twitter.
  - ✓Why is the similar tweet itself included in the reply list?
    - → Because the similar tweet can be useful as a reply in some context.

For example…

Post : It is pretty cool in Hokkaido today.
Comment : Summer is the best season in Hokkaido.

  - ✓In this case, the direction of post-comment can be reversed. Thus each tweet can be a reply for an input post.