SYSTEMS ENGINEERING RESEARCH CENTER

STEVENS INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®

WRT-1017: Keyphrase Extraction using Language Embeddings

Bingyang Wen, Stevens Institute of Technology

Sponsor: CCDC Armaments Center

AI Enabled Keyphrase Extraction

- Keyphrase extraction is the first step for a lot of downstream NLP tasks needed for competitive intelligence
  - Select relevant articles for SME attention
  - Rank ideas according to importance
  - Extract trends
  - Discover areas of interest
  - Create summaries of documents

Goal of work

- Propose new keyphrase extraction/generation algorithms that meets or exceeds state-of-the-art (SOTA) methods
- Keyphrase extraction (Phase I)
  - This method select keyphrases from the source document.
- Keyphrase generation (Phase II)
  - A generative model that can generate the corresponding keyphrases after analyzing the source document.
  - The generative model is usually deep-based and trained to output keyphrases for the input document.

Phase I: Keyphrase Extraction using Language Embeddings

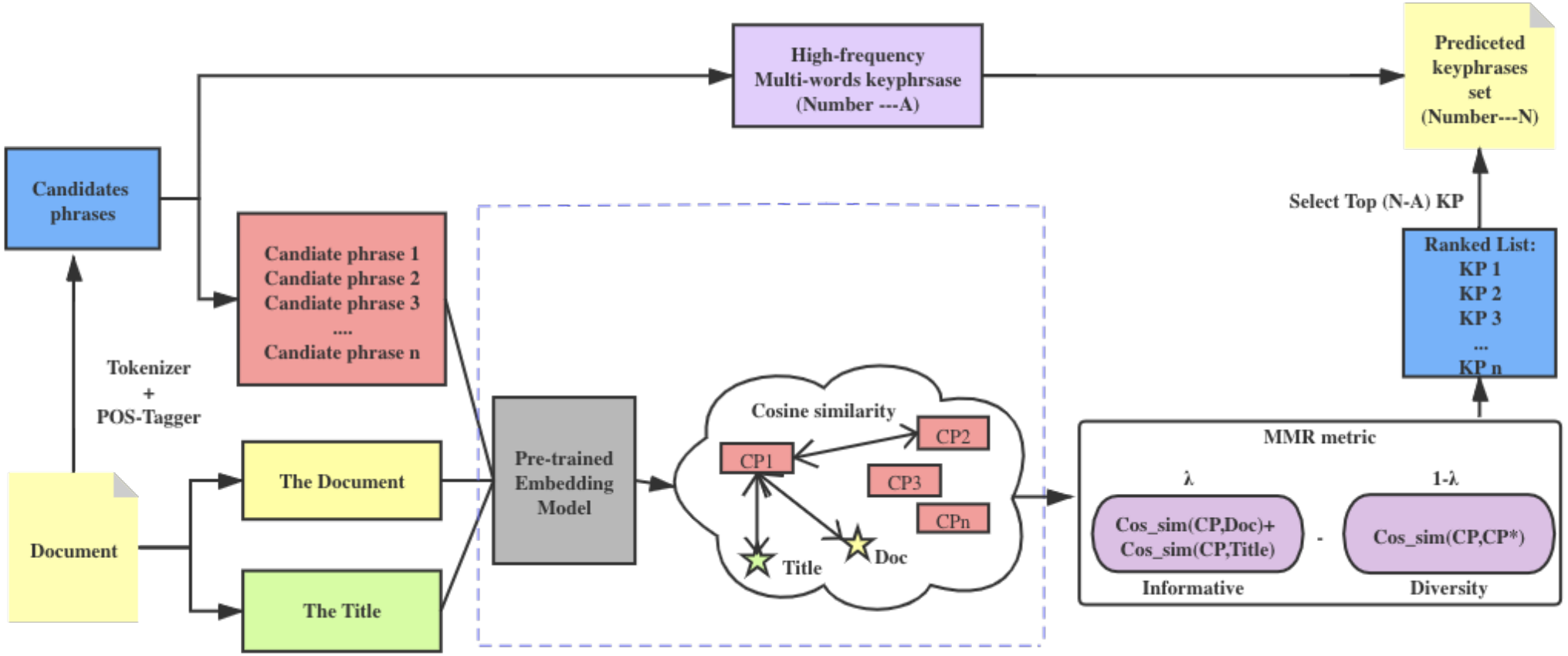
Overview

- Keyphrase extraction
- Unsupervised method
  - Low cost for adaption
    - Training free
    - Without human annotation

Innovation of our algorithm

- Title Similarity:
  - Take advantage of relationship between the article title and keyphrases
- Frequency:
  - Keyphrases are likelier to appear more often that other types of words
  - Select these phrases as candidates
- Candidate keyphrases:
  - Suggest new structure for keyphrases

Method Overview



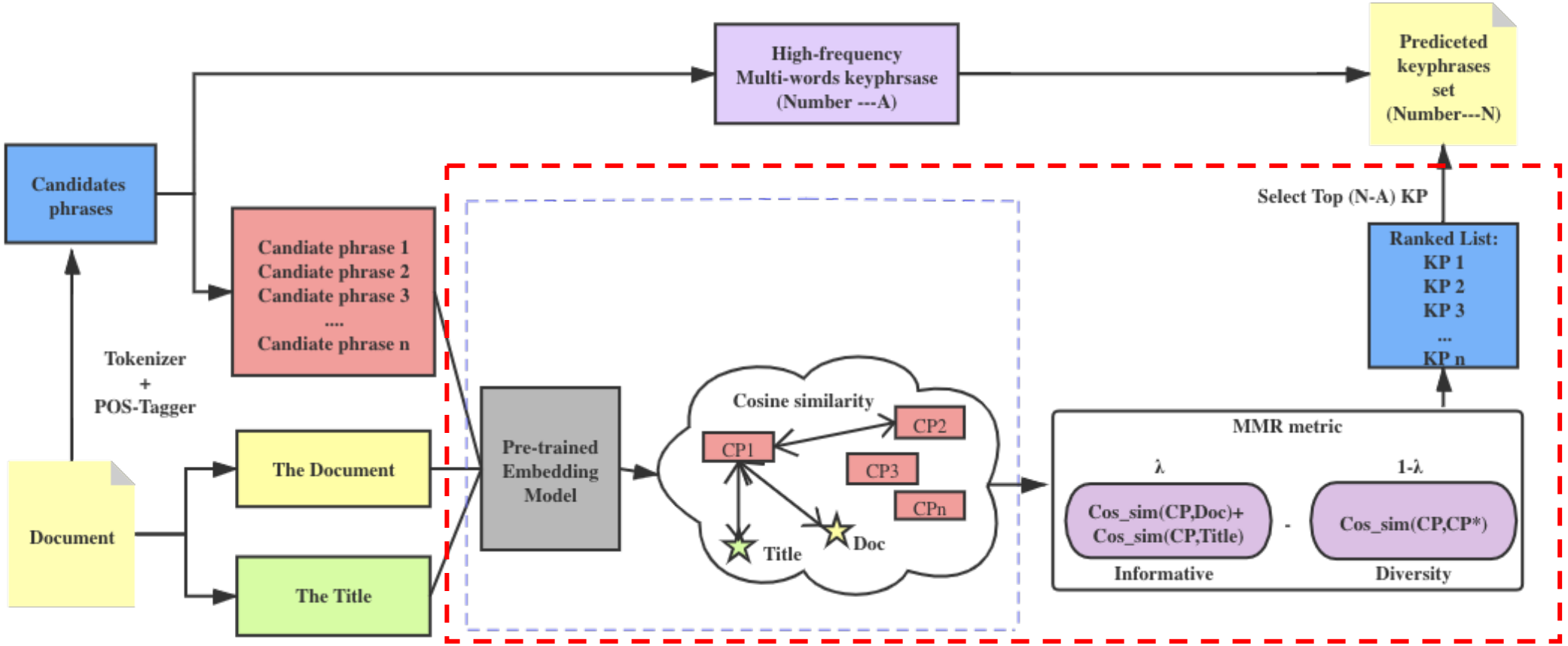
Proposed Method: Extracting candidate phrases

- Extract candidate phrases from the text
  - Keyphrases, (as opposed to keywords) always consist of two or more words and have some fixed part-of-speech patterns.
- Proposed patterns of candidate keyphrases
  - Adj(one or more) + Nouns (One or more)
  - Nouns (one or more) + Nouns (one or more)
  - VBN (verb, past participle) + Nouns
    - example: ``distributed system"
  - Adj + VBG (verb, present participle) + Nouns
    - example: ``minimal generating sets"
  - Adj + VBN + Nouns
    - example: "out-of-print print music"

First step *(contd.)* : Extract candidate phrases

- Delete unimportant words for candidate phrase
  - **e.g.:** `Such More Other Most Less Few Little Many Better`
  - Example of trimmed phrases:
    - ['such information architecture'] → ['information architecture']
- Compute high frequency keyphrases (that occur two or more times) and output them directly as predicted keyphrases

Candidates Distill by Using Embedding



Second step: Language Embedding

- Embedding can capture the semantic and syntactic meaning.
- Use sentence embedding to represent:
  - the candidate phrases
  - the title
  - the document in the same high-dimensional vector space
- We consider three sentence embedding schemes:
  - Universal Sentence Encoder (USE) [Cer et al. 18]
  - Smooth Inverse Frequency(SIF) [Arora et al. 16]
  - Sentence Bert [Reimers and Gurevych 19]

Third Step: Maximal marginal relevance (MMR)

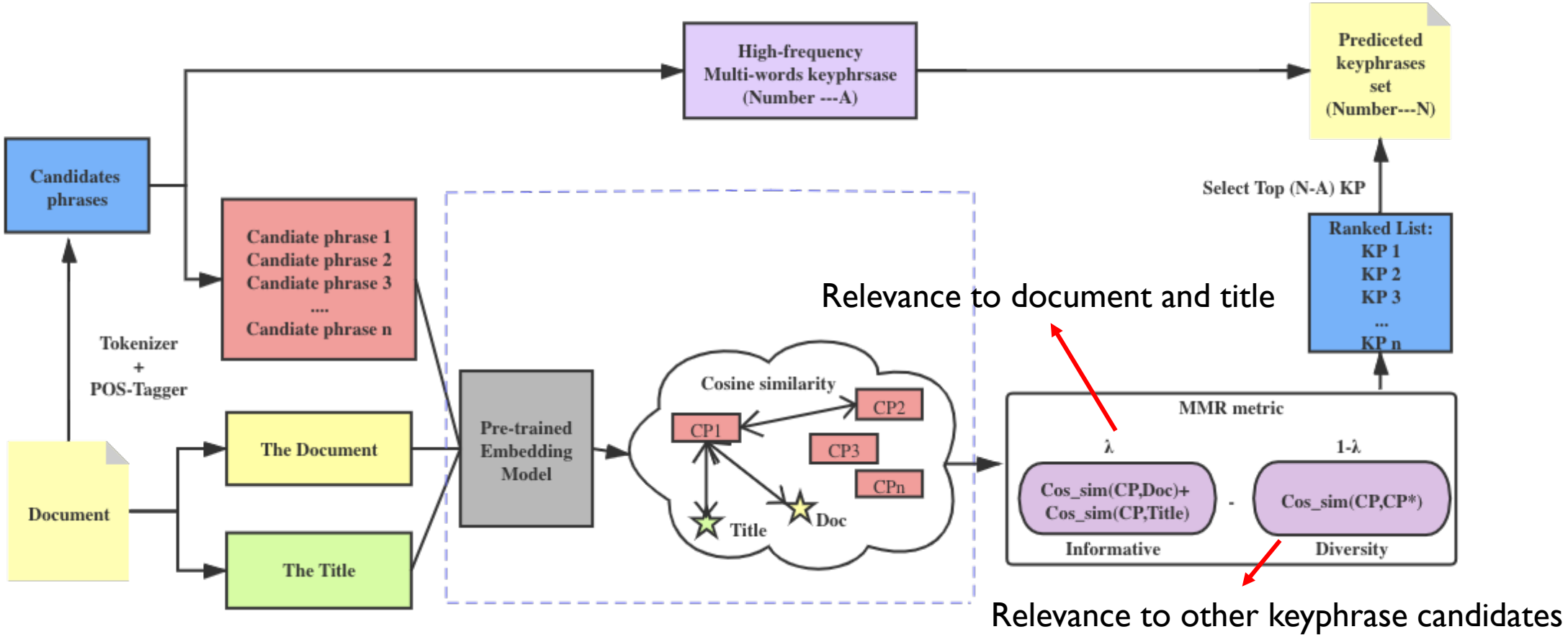
- MMR:
  - A measurement used to evaluate the relevance of the candidate keyphrase to the document and to trade-off relevance with the diversity of keyphrases.
  - Cosine Similarity:

$$\mathrm{sim}(c_i, \mathrm{doc}) = \frac{c_i \cdot \mathrm{doc}}{\|c_i\| \, \|\mathrm{doc}\|}$$

$$\mathrm{sim}(c_i, \mathrm{title}) = \frac{c_i \cdot \mathrm{title}}{\|c_i\| \, \|\mathrm{title}\|}$$

$$\mathrm{sim}(c_i, \mathrm{c_j}) = \frac{c_i \cdot \mathrm{c_j}}{\|c_i\| \, \|\mathrm{c_j}\|}$$

Third Step: Maximal marginal relevance (MMR)



Experiment: Datasets

- Scientific Papers
  - Inspec: 2000 documents
  - Semeval: 244 documents
  - Krapivin: 2340 documents
  - NUS: 211 documents

Experiment: Metrics

- Precision: # Matching predicted keyphrases / total # predicted keyphrases.
- Recall: # Matching predicted keyphrases / total # ground-truth keyphrases present
- F_1 score:

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

- $F_1@k$: F_1 score calculated with k predicted keyphrases.

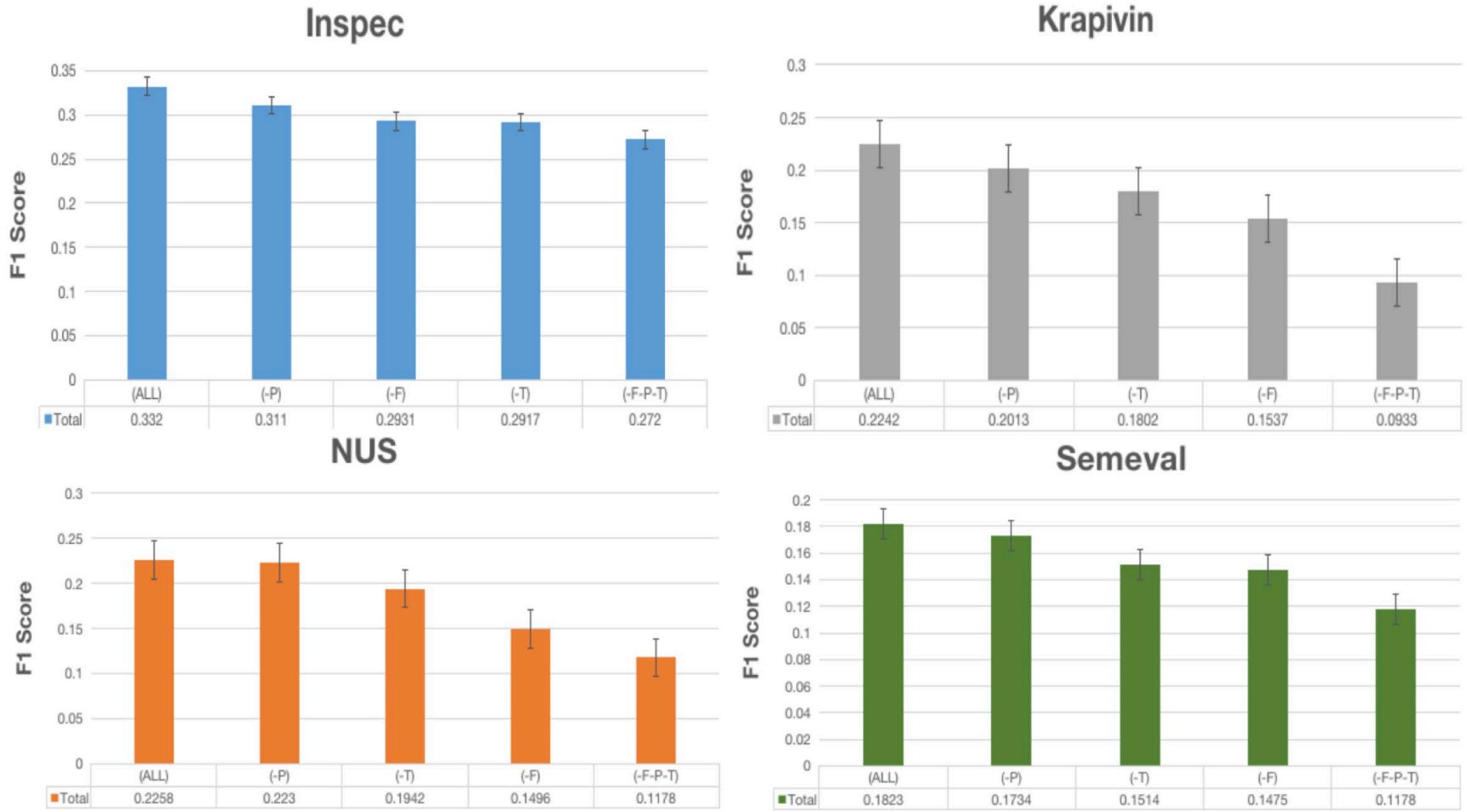
Experiment: Result

Model	NUS			Semeval			Krapivin			Inspec		
	$F_1@5$	$F_1@10$	$F_1@15$	$F_1@5$	$F_1@10$	$F_1@15$	$F_1@5$	$F_1@10$	$F_1@15$	$F_1@5$	$F_1@10$	$F_1@15$
Statistical model												
TFIDF	17.35	17.37	16.20	10.94	12.29	12.42	12.72	11.68	10.20	10.26	11.70	11.61
YAKE	12.66	16.98	16.78	11.39	14.17	15.18	10.00	10.46	10.17	13.48	14.97	15.49
Graph-based models												
TextRank	6.80	11.95	13.10	7.80	12.25	14.62	6.62	9.50	10.22	20.55	25.59	27.69
SingleRank	11.20	14.51	16.41	9.70	14.51	16.24	9.63	11.03	11.19	21.19	26.74	27.43
TopicRank	14.48	14.36	13.01	9.80	12.78	12.56	10.43	10.07	9.10	18.26	21.48	22.30
PositionRank	14.72	17.85	17.51	13.00	17.17	17.54	12.17	12.46	11.73	21.61	25.32	26.24
Multipartite	15.56	16.58	16.62	11.00	14.10	14.85	12.07	11.00	10.15	18.61	22.16	23.52
Embedding-based models												
EmbedRank	11.20	14.72	14.65	11.22	15.84	16.10	9.33	11.34	11.42	27.20	34.98	36.45
SIFRank	11.78	16.89	18.04	13.35	18.90	20.88	12.91	15.25	14.97	29.22	**39.04**	**39.80**
TaURank(Ours)	20.25	21.66	20.97	16.25	20.24	20.91	17.50	15.99	14.42	32.02	37.18	36.59
TaSBRank(Ours)	20.42	21.76	21.21	16.48	20.28	21.69	17.31	16.58	15.10	32.06	36.82	36.44
TaSIFRank(Ours)	**22.58**	**25.65**	**23.69**	**18.23**	**22.66**	**23.42**	**17.50**	**22.16**	**20.74**	**33.30**	38.33	38.25

Experiment: Ablation Study

- We consider the following methods:
  - (-F) remove frequency
  - (-P) use the less keyphrases patterns: *Adj (zero or more)* + *Nouns (One or more)*
  - (-T) no title information

Experiment: Ablation Study (Contd.)



Conclusion

- INSPEC, NUS, Krapivin and SemEval : Our method achieves the best or second best (on INSPEC) results compared with other unsupervised method.

Phase II: Transformer-based Adversarial Keyphrase Generation

Overview

- Keyphrase generation
  - Can generate absent keyphrases:
    - Keyphrases that do not appear in the source document

Dataset	# Keyphrase	% Present	% Absent
Inspec	19,275	55.69	44.31
Krapivin	2,461	44.74	52.26
NUS	2,834	67.75	32.25
SemEval	12,296	42.01	57.99

Title: A study on meaning processing of dialogue with an example of development of travel consultation system.

Abstract: This paper describes an approach to processing meaning instead of processing information in computing. Human intellectual activity is supported by linguistic activities in the brain. Therefore, processing the meaning of language
… …
user and retrieve information through dialogue. Through a simulation example of the system, we show that both information processing and language processing are integrated.

Ground-truth Keyphrases: linguistic activities; human intellectual activity; meaning processing; information processing; travel consultation dialogue system; language processing; information retrieval; user utterance understanding;

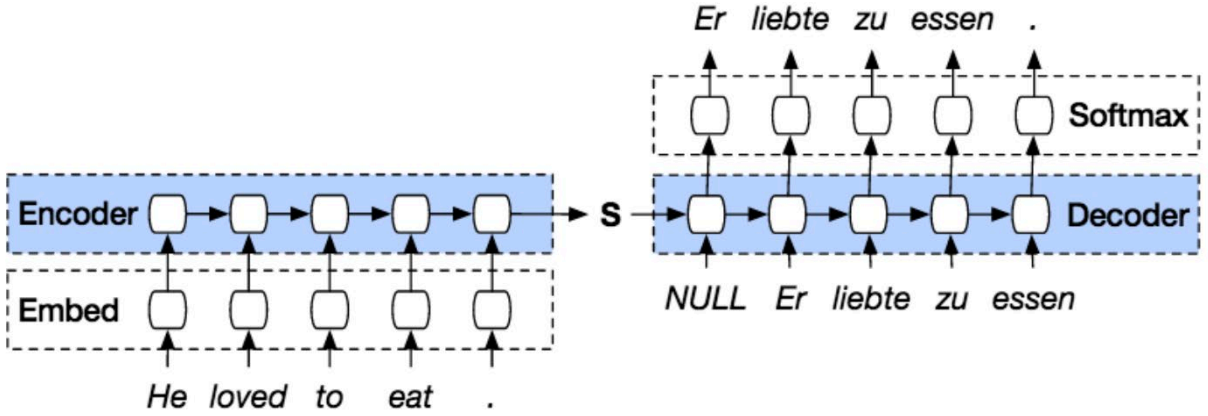
Present Keyphrase

Absent Keyphrase

- Supervised method
  - Overall better performance that unsupervised methods

Motivation

- Problems with MLE-based Keyphrase Generation:
  - Exposure bias
    - Mismatch of hidden states in train and test time:
      - Train time: using ground-truth keyphrases.
      - Test time: using output from previous time steps.



Motivation (Contd.)

- Generating "unreadable" phrases:
  - MLE objective can be viewed to improve 1-gram precision of a sequence.
  - 1-gram is not enough to represent a good distribution for "golden" keyphrases which are usually contains more than one word.

Innovation of our algorithm

- Applying adversarial learning to transformer-based keyphrase generation.
- Using reinforced F1-score objective to further improve the sequence-level performance.
- A new discriminator structure.

Preliminary: Adversarial Learning for Text Data

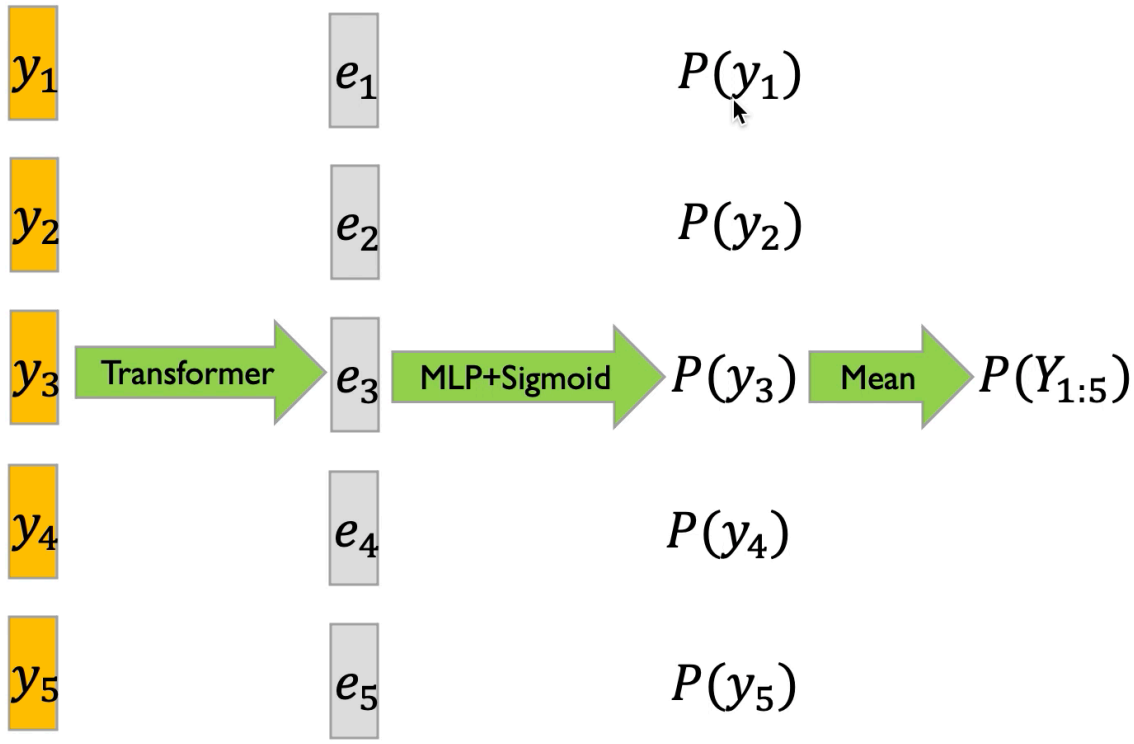
- Generative Adversarial Training:
  - Game between generator and discriminator.
  - Generator tries to generate samples that can fool discriminator.
- Challenge for text data:
  - The discrete nature of text data.
- Remedy
  - Policy gradient: using reinforcement learning
  - The feedback of discriminator can be seen as reward and the text generation can seen as choosing an action from the vocabulary
  - Generator: get higher reward; Discriminator: assign lower reward for generated samples

Proposed Method: New Discriminator

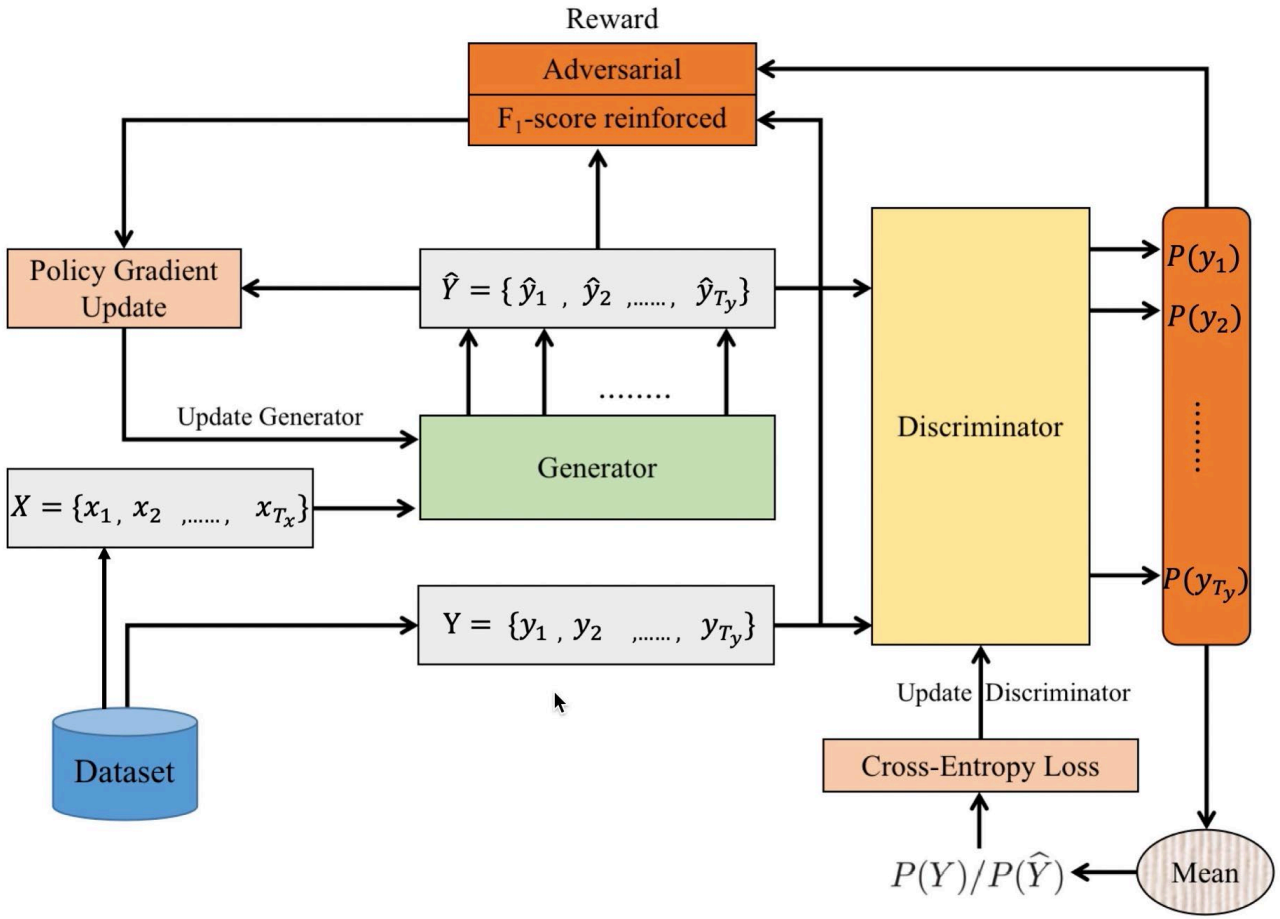
- Problems in the previous methods:
  - Discriminator is used in sequence-level
  - And using discriminator to discriminate

  (assign reward) for single output is difficult.

- Our proposed discriminator
  - Assign reward for each output element



Method Overview



Experiment: Dataset

- Train Dataset
  - Train: 80% of KP20K (567K documents)
  - Validation: 10% of KP20K no overlap with train
- Test Dataset
  - Inspec: 2000 documents
  - Semeval: 244 documents
  - Krapivin: 2340 documents
  - NUS: 211 documents
  - KP20K: 10% of KP20K no overlap with train or validation

Experiment: Main Result

- Overall Performance
  - Transformer + F1 better on present keyphrase generation
  - Transformer + Hybrid is better on absent keyphrase generation

Methods	Inspec		Semeval		Karpivin		NUS		KP20k	
	F1@5	F1@M	F1@5	F1@M	F1@5	F1@M	F1@5	F1@M	F1@5	F1@M
Present Keyphrase										
Transformer	**0.238**	**0.291**	**0.255**	0.295	0.232	0.320	0.289	0.355	0.245	0.332
Transformer+F1	0.215	0.266	0.254	**0.299**	**0.263**	**0.328**	**0.307**	0.365	**0.267**	0.344
Transformer+Adv	0.218	0.253	0.217	0.250	0.234	0.292	0.279	0.327	0.247	0.336
Transformer+Hybrid	0.216	0.268	0.252	0.297	0.260	0.326	**0.307**	**0.367**	0.266	**0.346**
Absent Keyphrase										
Transformer	**0.006**	**0.010**	**0.027**	**0.034**	0.010	0.014	0.011	0.015	0.009	0.013
Transformer+F1	0.004	0.007	0.023	0.031	0.015	0.025	0.010	0.014	0.013	0.020
Transformer+Adv	0.004	0.005	0.019	0.022	0.003	0.004	0.007	0.013	0.010	0.015
Transformer+Hybrid	0.005	0.007	**0.027**	**0.034**	**0.017**	**0.029**	**0.013**	**0.018**	**0.014**	**0.023**

Table 1: **F1 Scores:** F1@5 and F1@M scores for both present and absent keyphrases on 4 datasets.

Experiment: Diversity

- α-nDCG
  - An extension of nDCG, which is a measure of ranking quality
  - It measures diversity by rewarding new keyphrases and penalizing the duplicated keyphrases.

Model	Inspec	Semeval	Karpivin	NUS	KP20K
Trans.	**0.894**	0.829	**0.709**	0.804	0.617
Trans.+F1	0.864	0.828	0.700	0.826	0.667
Trans.+Adv	0.849	0.737	0.658	0.779	0.679
Trans.+Hybrid	0.871	**0.839**	0.704	**0.831**	**0.690**

Table 3: α-nDCG@M metrics

Conclusion

- Generally, our proposed method (F1 + GAN) can improve generative diversity and performance of generating both present and absent keyphrases.
- By ablation study:
  - F1-score reinforced objective: good at improving present keyphrase performance.
  - GAN training: good at improving absent keyphrase performance.