딥러닝 모델을 이용한 욕설 필터링 시스템 개선

순천향대학교 빅데이터공학과
우지영

온라인에서의 폭력성

- 사이버 세계에서의 욕설의 심각성
  - 사용자 이탈 야기
  - 전염성

- 욕설 필터링 시스템
  - 게임 내 욕설을 방지하기 위해 게임사들은 게임 내 채팅방에서 특정 단어를 입력하면 그 단어가 별표(*)등으로 가려져서 나오게 하는 시스템을 갖추고 있음
  - ‘개xx’, ‘x새끼’등의 내용을 채팅 칸에 쓰면 그 단어가 ***식으로 나타남

- 문제점
  - 특수문자를 섞거나 이상한 신조어 등을 만들어 우회
  - 띄어쓰기를 하거나 하지 않는 방식으로도 우회

기계학습

- 기계학습
  - 데이터로부터 패턴을 찾아 분류를 할 수 있음
  - 전통적인 기계학습은 데이터의 특질이 정의되어있어야 함
  - 딥러닝은 기계 스스로가 데이터의 특질을 찾고 패턴화 할 수 있음

쓸모없는 특질: 머리 하나, 팔 두개, 다리 두개
유용한 특질: 머리 길이, 얼굴의 곡선 여부, 키

https://www.google.com/url?sa=i&url=https%3A%2F%2Fjmagazine.joins.com%2Feconomist%2Fview%2F320410&psig=AOvVaw0NyKt9RvIDmh9dWRlUVtdQ&ust=1594443064027000&source=images&cd=vfe&ved=2ahUKEwiJmuu08cHqAhWISJQKHbxAAsYQr4kDegUIARCWAQ

픽셀단위로 수치화된 값에서 기계 스스로 패턴을 찾도록 함

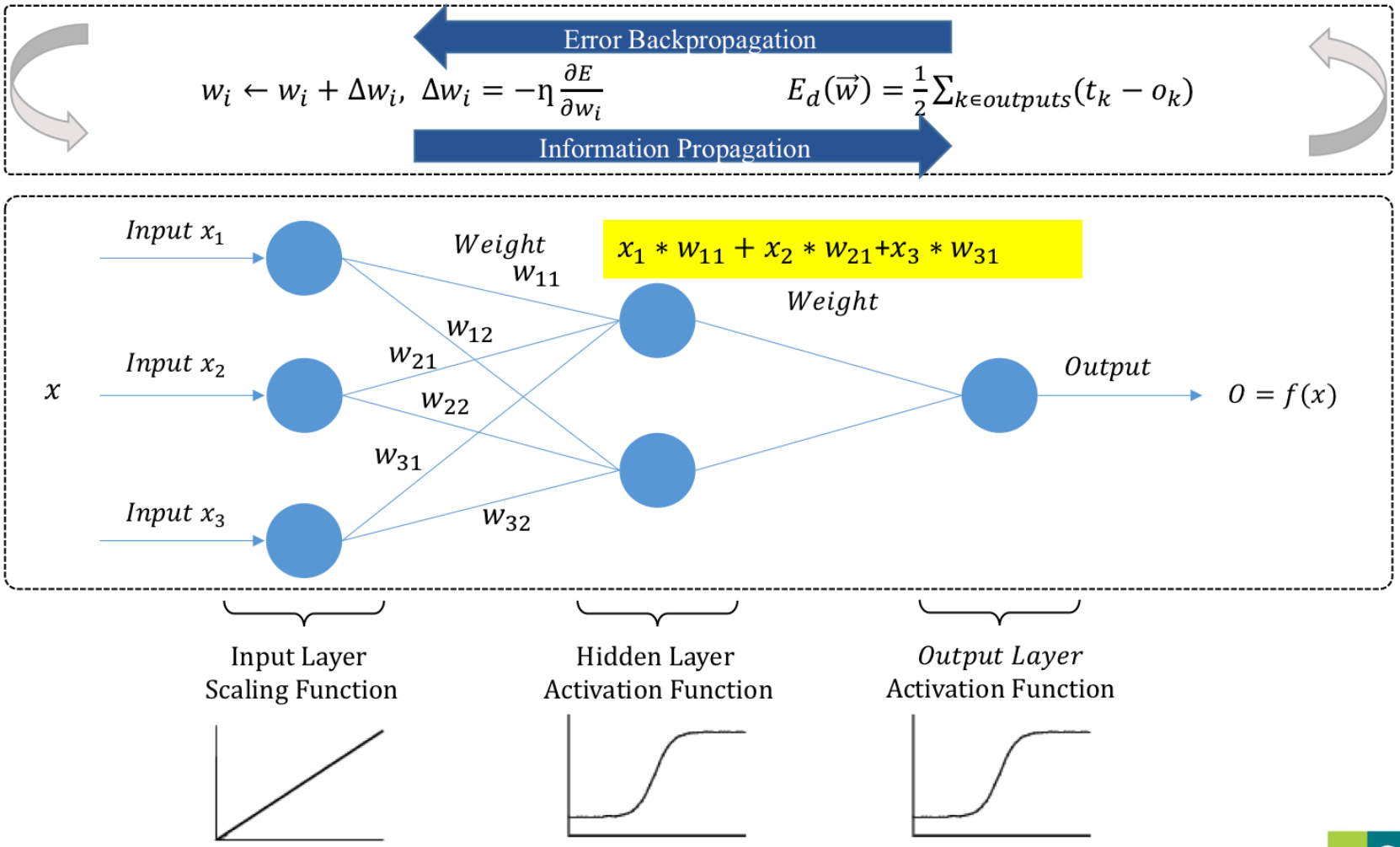
https://www.google.com/url?sa=i&url=https%3A%2F%2Fjmagazine.joins.com%2Feconomist%2Fview%2F320410&psig=AOvVaw0NyKt9RvIDmh9dWRlUVtdQ&ust=1594443064027000&source=images&cd=vfe&ved=2ahUKEwiJmuu08cHqAhWISJQKHbxAAsYQr4kDegUIARCWAQ

욕설 탐지 모델

- 기계학습 기반
  - 텍스트 특질 추출 후 기계학습
  - N-grams(단어의 조합)
  - linguistic features: 주어, 목적어, 동사 등등
  - syntactic features: 문장의 길이, 사용한 단어의 개수, 특수 기호 사용 횟수
  - lexicons: 긍정/부정(sentiment), 감정(affect)
  - Chen's et al.[4], Nobata at el.[5], Lee et al. [7], Martens et al. [8], 박교현 and 이지형[6]
- 딥러닝 기반
  - 텍스트에서 기계 스스로 특질 추출 후 학습
  - Founta et al. [10], Yenala1,et al. [11], 한글 연구 [13, 14]

4. Chen, Y., Y. Zhou, S. Zhu, and H. Xu. Detecting offensive language in social media to protect adolescent online safety. in 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing. 2012. IEEE.
5. Nobata, C., J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang. Abusive language detection in online user content. in Proceedings of the 25th international conference on world wide web. 2016.
6. 박교현 and 이지형, SVM 을 이용한 온라인게임 비속어 필터링 시스템. 한국정보과학회 학술발표논문집, 2006. 33(2B): p. 260-263.
7. Lee, H.-S., H.-R. Lee, J.-U. Park, and Y.-S. Han, An abusive text detection system based on enhanced abusive and non-abusive word lists. Decision Support Systems, 2018. 113: p. 22-31.
8. Märtens, M., S. Shen, A. Iosup, and F. Kuipers. Toxicity detection in multiplayer online games. in 2015 International Workshop on Network and Systems Support for Games (NetGames). 2015. IEEE.
10. Founta, A.M., D. Chatzakou, N. Kourtellis, J. Blackburn, A. Vakali, and I. Leontiadis. *A unified deep learning architecture for abuse detection*. in *Proceedings of the 10th ACM Conference on Web Science*. 2019.
11. Yenala, H., A. Jhanwar, M.K. Chinnakotla, and J. Goyal, *Deep learning for detecting inappropriate content in text*. International Journal of Data Science and Analytics, 2018. **6**(4): p. 273-286.
13. Yoon, T.-J. and H.-G. Cho, *The Online Game Coined Profanity Filtering System by using Semi-Global Alignment.* The Journal of the Korea Contents Association, 2009. **9**(12): p. 113-120.
14. Kim, S.Y. and J.Y. Lee, *Fact finding surveying adolescents's language and culture in online games and a countermeasure strategy.* The Journal of Korean Association of Computer Education, 2013. **16**(1): p. 33-42.

딥러닝 학습 방식


Error Backpropagation
$w_i \leftarrow w_i + \Delta w_i,\ \Delta w_i = -\eta \frac{\partial E}{\partial w_i}$
$E_d(\vec{w}) = \frac{1}{2}\sum_{k \in outputs}(t_k - o_k)$
Information Propagation
Input x_1
Input x_2
Input x_3
x
Weight
w_{11}
w_{12}
w_{21}
w_{22}
w_{31}
w_{32}
$x_1 * w_{11} + x_2 * w_{21} + x_3 * w_{31}$
Weight
Output
$O = f(x)$
Input Layer
Scaling Function
Hidden Layer
Activation Function
Output Layer
Activation Function

금칙어 사전의 비효율성

- 다양한 변형에 대응하기 위해 금칙어 사전은 비효율적으로 구성

47	ㄱㅐ	개ㅅㅐ	개넌
48	ㄱㅐㅅㅐ	개새끼	개넘
49	ㄱㅐㅅㅐㄲㅣ	개새끼	개년
50	ㄱㅐ새	개새끼	개놈
51	ㄱㅐ	개새끼	개놈아
52	ㄱH	개`	개뇬
53	ㄱH넘	개같은	개눔
54	ㄱH년	개같은넘	개늠아
55	ㄱH놈	개같은년	개도라이
56	ㄱH뇬	개같은년	개때꺄
57	ㄱH뇬	개같은놈	개때끼
58	ㄱH늠	개같은놈	개또라이
59	갑ㅂ	개같은뇬	개련

기존 모델

- 금칙어 사전기반 모델의 한계점

사전	**채팅**
사정	**사정**을 여쭤보니
아갈	찾**아갈**게요, 날**아갈**듯
호구	발보**호구**
시키	**시키**지마세요
니미	못찾으**니미**치겠네
색골	저 보라**색골**랐는데
음부	처**음부**터
야한	도와줘**야한**다니, 있어**야한**대요
씨방, 시불	피**씨방**, 택**시불**러요
뒤져	마을 **뒤져**도 안보여요
죽어	**죽어**도 안되겠는데
보지	왜 옆에있는 날 **보지**못하고
제길	언**제길**러짐

데이터 소개

- 텍스트 태깅(TAGGING)

	Human-judged Profanity	Human-judged Normal Words	total
Dictionary-based Profanity	3,604	1,396	5,000

```
In [23]: p1 = pd.read_csv('./sampling_utf8.csv')

In [24]: p1
```

Out [24]:

	Unnamed: 0	NA.	chat	z_sum3	or
0	4215	148402	도대체 게임에서 현실드립을 왜쳐 새ㄱ끼?ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ	1	1.0
1	9973	381318	밑에미친놈있다	5	1.0
2	13092	555776	씨1발 ㅋㅋㅋㅋㅋ	2	1.0
3	19752	961806	시발	2	1.0
4	3681	131233	존만한년아	1	1.0
5	5939	221143	아 존1나 팅기네 것참	1	1.0
6	7967	288677	인터넷 뒤져보니깐	1	0.0
7	1576	46590	cafe.naver.com/a2f2	1	0.0
8	5986	222932	구라치지마	1	1.0
9	9698	368283	쌩양아치색희가	1	1.0

데이터 소개

- 채팅 텍스트

```
In [23]: p1 = pd.read_csv('./sampling_utf8.csv')
```

```
In [24]: p1
```

Out [24]:

	Unnamed: 0	NA.	chat	z_sum3	or
0	4215	148402	도대체 게임에서 현실드립을 왜쳐 새□刻?ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ	1	1.0
1	9973	381318	밑에미친놈있다	5	1.0
2	13092	555776	씨1발 ㅋㅋㅋㅋㅋ	2	1.0
3	19752	961806	시발	2	1.0
4	3681	131233	존만한년아	1	1.0
5	5939	221143	아 존1나 팅기네 것참	1	1.0
6	7967	288677	인터넷 뒤져보니깐	1	0.0
7	1576	46590	cafe.naver.com/a2f2	1	0.0
8	5986	222932	구라치지마	1	1.0
9	9698	368283	쌩양아치색희가	1	1.0

데이터 전처리

- 텍스트 데이터 수치화

나는 오늘 밥을 먹었다

나는 오늘 밥을 먹을것이다

표준형 변환

나 오늘 밥 먹다

one-hot encoding

1000, 0100, 0010, 0001

Word2Vec

(0.8 0),(0.1 0.8), (0.2 0.3), (0 0.3)

이런식으로 숫자로
변환하고 싶음

잘못된 인코딩

1 2 3 4

그런데 이렇게
아무렇게나 숫자로
만들면 거리 개념이
임의로 만들어짐

데이터 전처리

- 한글 토근화의 문제점
  - 띄어쓰기가 잘 안지켜짐

나 오늘 밥먹다

나오늘밥먹다

✓ 자소 분리

시발 → ㅅ|ㅂ|ㅏ|ㄹ → ㅅ(초성)|(중성)ㅂ(초성)|(중성)ㅏ(중성)ㄹ(종성)

✓ 음절 분리

시발 → 시 발

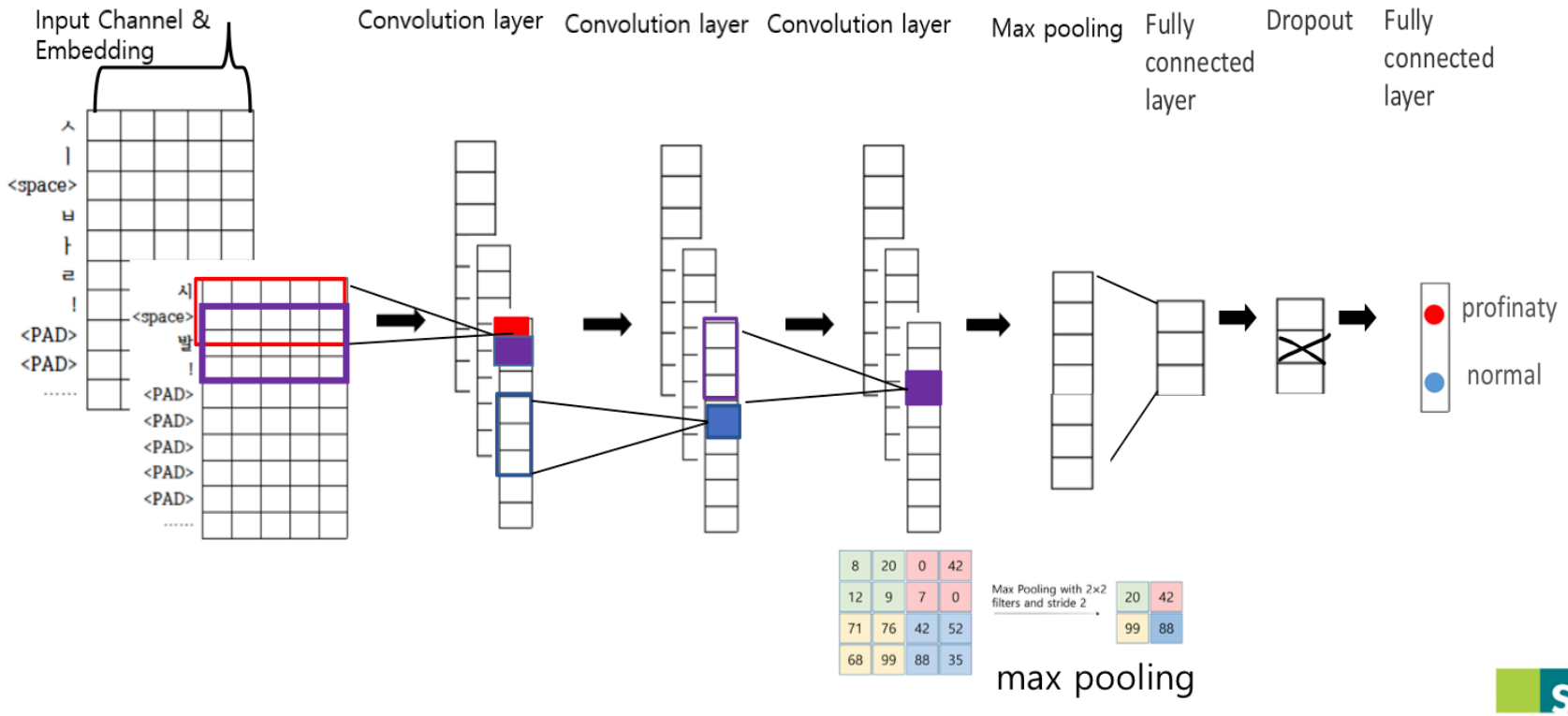
자소 기반 사전 구축

사전	빈도수	사전번호
'<PAD/>'	1226614	0
' '	7798	1
'ㅇ'	7716	2
'ㅏ'	6534	3
'ㄴ'	6191	4
'ㅣ'	5001	5
'ㅅ'	3936	6
'ㅈ'	3697	7
'ㄹ'	3506	8
…	…	…
'ㄶ'	1	124
'[UNK]'	0	125

욕설 판별을 위한 딥러닝 모델

- CNN

(0.8 0 0 0 0),(0.1 0 0 0 0.8) 이렇게 수치화했으면 5차원



딥러닝 모델

- 모델과 파라미터 수

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 469, 64)           12800
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 469, 16)           3088
_________________________________________________________________
batch_normalization_1 (Batch (None, 469, 16)           64
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 469, 16)           784
_________________________________________________________________
batch_normalization_2 (Batch (None, 469, 16)           64
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 469, 8)            392
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 234, 8)            0
_________________________________________________________________
flatten_1 (Flatten)          (None, 1872)              0
_________________________________________________________________
dense_1 (Dense)              (None, 8)                 14984
_________________________________________________________________
dropout_1 (Dropout)          (None, 8)                 0
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 18
=================================================================
```

실험결과

- 자소분리

	Profane class	Normal class
Precision	90.93%	83.39%
Recall	94.25%	75.43%
F1 measure	92.56%	79.21%
Accuracy	89.04%	

- 음절분리

	Profane class	Normal class
Precision	92.22%	81.60%
Recall	93.14%	79.48%
F1 measure	92.68%	80.53%
Accuracy	89.36%	

- 자소분리 + SPACE제거

	Profane class	Normal class
Precision	89.56%	81.79%
Recall	93.92%	71.39%
F1 measure	91.68%	76.23%
Accuracy	87.68%	

자소를 분리한 경우 총 사용된 사전의 수는 126개, 한 글자씩 분리한 경우 1065개 필요

모델 튜닝

- 모델에 따른 정확도 변화

CNN structure	Accuracy
Conv+Conv+Pool+Dropout+FC	86.88%
Conv+Pool+Dropout+FC	85.6%
Conv+Conv+Conv+Pool+FC+Dropout+FC	89.04%
Conv+globalpool+FC+Dropout+FC	90.4%
Conv+Conv+globalpool+FC+Dropout+FC	90.08%
Conv+Pool+Conv+Pool+FC	86.4%
Conv+Pool+Conv+Pool+Conv+Pool+FC	85.7%

질의응답

감사합니다.