수기 답안지 자동 채점 시스템

전현민, 이재영, 이원정, 이웅희, 김영훈

한양대학교 ERICA, 소프트웨어학부

gusalsrhkals@naver.com, wayexists02@gmail.com, itzmewj97@gmail.com,
woongheelee@hanyang.ac.kr, nongaussian@gmail.com

Auto Grading System for Hand-Written Answer Sheet

Hyunmin Jeon, Jaeyoung Lee, Wonjung Lee, Woonghee Lee, Younghoon Kim
Hanyang University ERICA, Division of Software

요 약

온라인에서 작성된 과제 답안을 채점하는 연구는 많으나, 오프라인에서 수기로 작성된 답안을 채점하기는 쉽지 않다. 이는 답안의 작성 패턴이 다양하며 각 답안 마다 개별적인 채점 모델을 구축하는 것이 쉽지 않기 때문이다. 본 연구는 단답형 문제의 프로토타입 답지로부터 유사한 답안을 검색하여 빠르게 채점을 돕는 도구를 개발하고자 한다. 특히 오프라인에서 작성된 답안을 샴 합성곱 신경망을 이용해 유사도를 측정하는 방법에 초점을 두었다. 이를 이용해 채점자가 점진적으로 정답/오답 답안 이미지를 판단해가며 수집된 답안과의 유사도 정렬을 통해 남은 답안의 채점을 빠르게 수행할 수 있다. 그리고 실제 답안지 데이터를 이용하여 유사도의 정확도를 실험적으로 확인하였다.

1. 서 론

온라인 강의 플랫폼 수업에서 제출된 과제나 시험지를 자동으로 채점하는 연구가 다수 진행된 바 있다[1-5]. 그러나 이 연구들은 온라인 강의 플랫폼이라는 제한된 시스템에만 적용 가능하며, 수기로 작성된 답안을 자동으로 채점하기는 쉽지 않다. 이는 수기로 작성된 답안의 작성 패턴이 다양하고, 각 답안마다 개별적인 채점 모델을 구축하기가 쉽지 않기 때문이다.

본 연구에서는 단답형 문제의 프로토타입 답지로부터 유사한 답안을 검색하여 빠르게 채점을 돕는 도구를 개발하고자 한다. 적은 데이터에도 학습이 용이한 샴 합성곱 신경망을 이용하였고, 답안지의 유사도를 측정하는 데 초점을 두었다. 이를 이용해 채점자가 점진적으로 정답/오답 답안 이미지를 판단해가며 수집된 답안과의 유사도 정렬을 통해 남은 답안의 채점을 빠르게 수행하고자 했다.

2. 관련 연구

[1]에서는 다중 오픈 온라인 코스(MOOC) 시스템에서 출제되는 프로그래밍 과제를 문제 단위로 나누어서 채점하는 시스템을 제안했다. 또한, [2]에서는 MOOC 시스템에서 출제되는 수학식 채점 문제를 해결하는 방법을 제안했다. [3]에서도 MOOC 시스템에서 채점하는 시스템으로, 단답형 답안을 클러스터링을 이용해서 채점하는 것을 제안했다. [4-5]에서는 학생들이 프로그래밍한 과제를 채점하고 피드백을 해 주는 기계 학습 방법을 제안했다. 이와 같은 연구는 수업에서 학생들에게 모두 같은 과제를 주는 것을 가정하였다. 또한, 오답의 패턴이 거의 비슷하다는 것에 주목했다. 이로부터 그래서 기계 학습 방법을 이용한 자동 채점이 가능하다는 점을 소개했다. 앞서 연구들은 모두 컴퓨터상에서 작성된 답안을 자동 채점하는 시스템으로, 수기로 작성한 답안을 채점하는데 적용할 수 없다는 한계가 있다.

3. 제안 방법

3.1 문제 정의

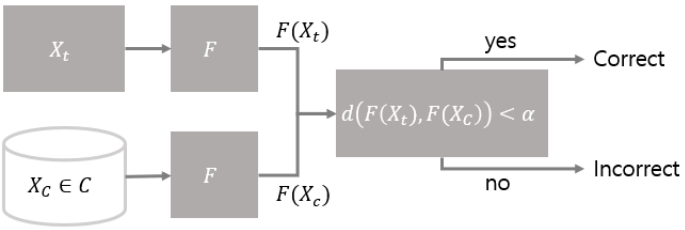


그림 1 답안 채점 프로세스

수기로 작성된 N개의 미채점 답지 X_t 의 집합 $D(t = 1, \dots, N)$ 와 정답으로 채점 된 프로토타입 답지 X_c 의 집합 C 가 있다고 하자. 본 연구에서는 미채점 답지 X_t 와 프로토타입 답지 X_c 의 유사도를 검색하여 유사도 임계값 α 보다 적을 경우 정답으로 판별하고자 하였다. 즉, 본 연구의 문제는 아래와 같이 정의할 수 있다.

문제정의 미채점 답지 X_t, 정답 프로토타입 답지 X_c가 주어졌을 때, $d(X_t, X_c) < \alpha$를 만족하는 거리 측정 함수 d를 찾는다.

위와 같은 문제 정의에 따라 그림 1 은 미채점 답안 X_t를 채점하는 과정이다. 기존의 정답 답안의 집합 C의 모든 답안 X_C 와 X_t 의 평균 거리 D 와 임계값 α 를 비교함으로써 정답인지 오답인지 판별한다.

3.2 데이터 전처리

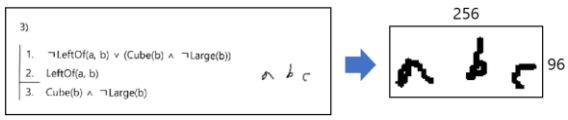


그림 2 시험지 이미지에서 답안만 추출한 결과

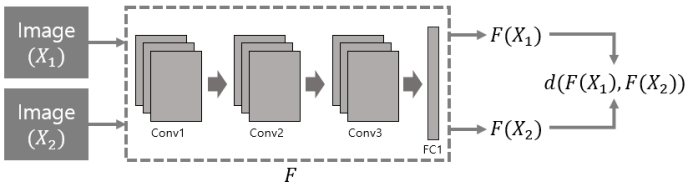


그림 3 모델 구조

시험지 이미지를 흑백으로 변환하고 이진화를 적용해서 모든 픽셀의 값을 0 또는 1 로 변환했다. 같은 문제의 모든 답안은 문제 부분이 일치하므로 답안만의 거리 계산을 위해 문제 부분을 제거할 필요가 있다. 답안을 적지 않은 시험지에 OCR[6]을 적용해서 문제 부분의 좌표를 구하고 답안이 적힌 시험지에서 문제 부분을 제거했다. 문제 부분을 제거한 이미지의 불필요한 여백은 OCR 을 한 번 더 적용해서 손글씨 부분만 인식한 후 모두 제거했다. 그림 2 는 전처리 과정을 거친 답지의 예시를 보이며 96×256 의 크기로 조정되어 모델의 입력으로 쓰이게 된다.

3.3 모델 정의

X_1 과 X_2 는 전처리 과정을 거친 학습데이터셋의 두 이미지라고 하자. 그림 3 은 주어진 두 이미지 X_1, X_2에 대해 거리를 계산하는 샴 합성곱 신경망[8]의 구조를 보여준다. 이를 F라 부를 때 네트워크 F에 따라 계산된 출력 $F(X_1)$ 과 $F(X_2)$ 의 거리는 $d(F(X_1), F(X_2))$ 로 나타낸다. 이때, [7]에 따라 0 과 1 사이 값으로 정규화된 유클리드 거리를 이용했다.

$$d(F(X_1), F(X_2)) = \frac{\|F(X_1) - F(X_2)\|_2}{\|F(X_1)\|_2 + \|F(X_2)\|_2}$$

학습 절차: 주어진 학습 이미지 데이터 T_C와 T_W는 각각 답안 이미지와 오답 이미지의 집합이다. 샴 신경망 학습을 위해 T_C 에서 n_1 개의 이미지 쌍 (X_i, X_j) 을 무작위로 추출하여 동일 정답 쌍 학습 데이터 $((X_i, X_j), 1)$를 만들고, T_C와 T_W에서 각각 n_0개의 이미지 쌍을 추출하여 정/오답 쌍 학습 데이터 $((X_i, X_j), 0)$ 를 추출하였다 (즉, $X_i \in T_C$, $X_j \in T_W$). 이에 따라 최소화하고자 하는 손실 함수 L은 아래와 같다.

$$L = y * \delta^2 + (1 - y) * (1 - \delta^2),$$

$$\delta = d(F(X_1), F(X_2))$$

이때, X_1, X_2 는 학습 샘플의 두 이미지이고 y는 정답 쌍인지 정/오답 쌍인지를 나타내는 범주 값이다. (즉, $y \in \{0,1\}$)

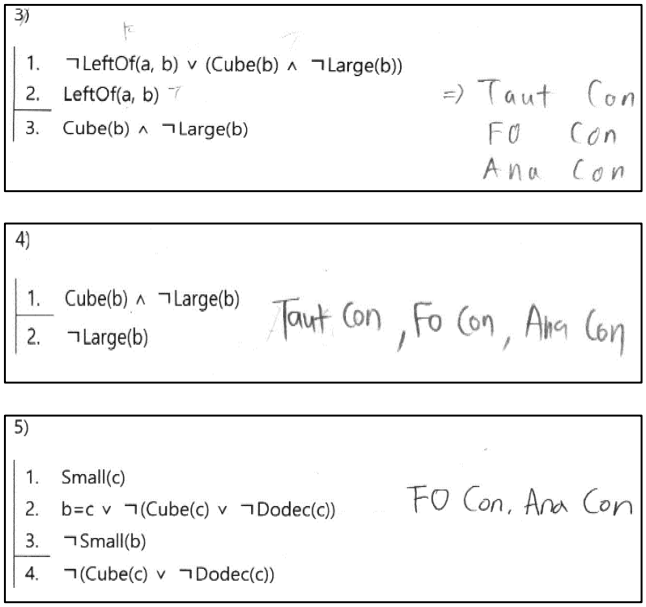


그림 4 데이터로 사용한 시험 문제의 종류

4. 실험

4.1 데이터 수집

본 연구에서는 그림 4 와 같은 세 종류의 단답형 시험 문제를 데이터로 사용했고 각각의 문제마다 각각의 신경망을 학습시켰다. 실험자는 수집된 답안지로부터 정답 답안과 오답 답안으로 분리했다. 또한 답안이 적힌 시험지로부터 문제 부분을 제거하기 위해, 백지 시험지를 준비했다.

데이터는 한 수업으로부터 한 학기 분량을 수집하였다. 따라서, 학습에 사용할 데이터가 부족하기 때문에 글자의 크기나 위치를 변경하는 등의 방법으로 데이터의 수를 늘렸다. 데이터 확장을 하면 데이터의 분포가 편향되는 것을 방지할 수 있고, 미래의 데이터에 대해 좀 더 일반화된 모델을 구축할 수 있다.

4.2 실험 방법

앞선 전 처리를 통해 시험지 채점 문제를 그림 2 와 같은 단순한 손글씨 문자 이미지를 비교하는 문제로 간소화했다.

정답 답안의 집합을 C, 채점하고 싶은 답안을 X_t라고 할 때, C의 모든 원소 X_C와 X_t의 평균 거리 D를 계산해 D 가 임계값 α 보다 작으면 정답, 아니면 오답으로 분류한다. D는 아래와 같이 계산한다.

$$D(C, X_t) = \frac{\Sigma_C d(F(X_C), F(X_t))}{n}$$

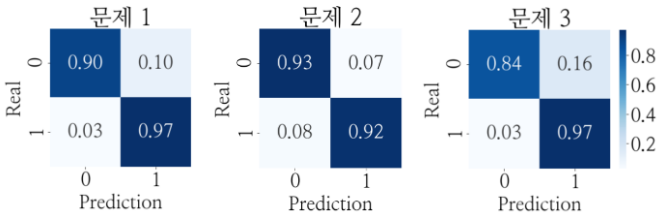
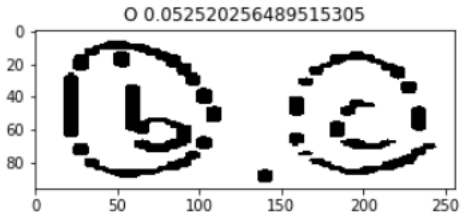
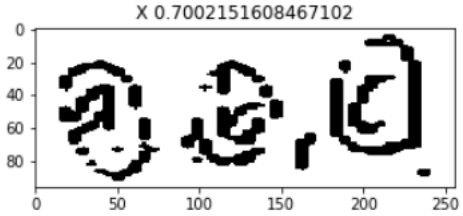


그림 5 각 문제 별 실험 결과의 혼동 행렬



(a)



(b)

그림 6 학습된 샴 합성곱 신경망을 통해 테스트 시험지와 기존의 정답 답안들과의 거리를 계산한 결과이다. (a)는 정답, (b)는 오답 답안이다.

일반적인 경우의 성능 검증을 위해 수집한 288 개의 이미지를 222 개의 학습 데이터와 66 개의 테스트 데이터로 나눴다. 학습 데이터는 118 개의 정답 데이터와 104 개의 오답 데이터로 이루어져 있고 이를 무작위로 3000 개의 쌍으로 만들어 학습에 사용했다. 이와 같은 방법으로 10-겹 교차 검증 방식을 써서 모델의 일반적 성능을 검증하고자 했다.

학습된 모델에 테스트 데이터를 입력으로 주고 학습 데이터로 사용했던 정답 이미지들과의 거리를 측정해 정답인지 오답인지 분류했다. 그림 5 에 세 종류의 문제별 테스트 결과를 혼동 행렬로 표현하였다. 각 문제별로 실제 정답 중 97%, 92%, 97%를 정답으로 분류했고, 실제 오답 중 90%, 93%, 84%를 오답으로 분류했다. 이때, 셋째 문제의 오답에 대한 정확도가 다른 경우보다 낮은 이유는 정답이 (b), (c)일 때, 오답인 (a), (c)인 경우를 정답으로 인식하는 문제가 있었기 때문이다. 더 다양한 글씨 데이터를 통해 모델의 성능을 향상시켜야 할 것으로 분석된다.

5. 결론

본 연구를 통해 수기로 작성된 답안을 자동으로 채점해 주는 시스템을 개발했다. 샴 합성곱 신경망과 다양한 전처리 방법을 사용하여 단답형 문제에 대해서 약 평균 92%의 정확성을 보였다. 자동채점 시스템을 이용하여 시험지나 과제를 채점할 때, 채점의 정확도는 가장 중요하고 민감한 문제이다. 향후 연구에서는 시스템의 정확도를 향상시키기 위해 더 효과적인 데이터 확장 방법을 적용하고, 서술형 문제와 같은 복잡하고 다양한 답안이 가능한 문제를 채점할 수 있는 시스템을 개발하고자 한다.

참고 문헌

[1] G Singh., S Srikant., V Aggarwal.:. Question Independent Grading using Machine Learning: The Case of Computer Program Grading. In KDD, 2016.

[2] A. S. Lan, D. Vats, A. E. Waters, and R. G. Baraniuk. Mathematical language processing: Automatic grading and feedback for open response mathematical questions. In Proceedings of the Second ACM Conference on Learning at Scale, 2015.

[3] Brooks, M., Basu, S., Jacobs, C., and Vanderwende, L. Divide and correct: Using clusters to grade short answers at scale. In Proc. 1st ACM Conf. on Learning at Scale, 2014

[4] Srikant, S., and Aggarwal, V. A system to grade computer programming skills using machine learning. In Proc. 20th ACM SIGKDD, 2014

[5] R. Singh, S. Gulwani, and A. Solar-Lezama. Automated feedback generation for introductory programming assignments. In ACM SIGPLAN Notices, 2013

[6] Břetislav H.: handwriting-ocr. https://bit.ly/2Wy5dkU

[7] Dhwaj R.: deep-siamese-text-similariy. https://bit.ly/2J5f5zi

[8] Koch, Gregory, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In ICML 2015