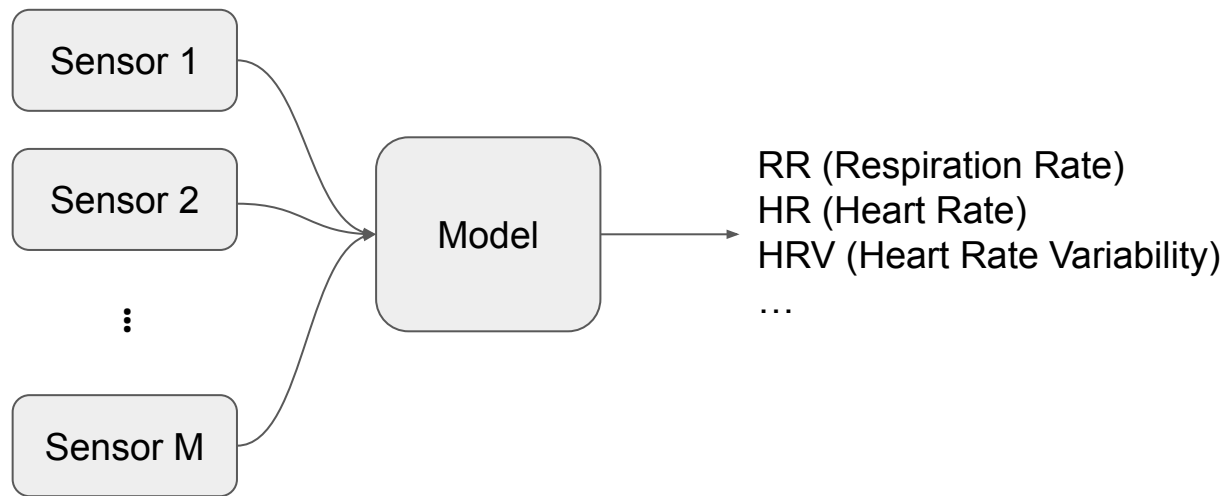


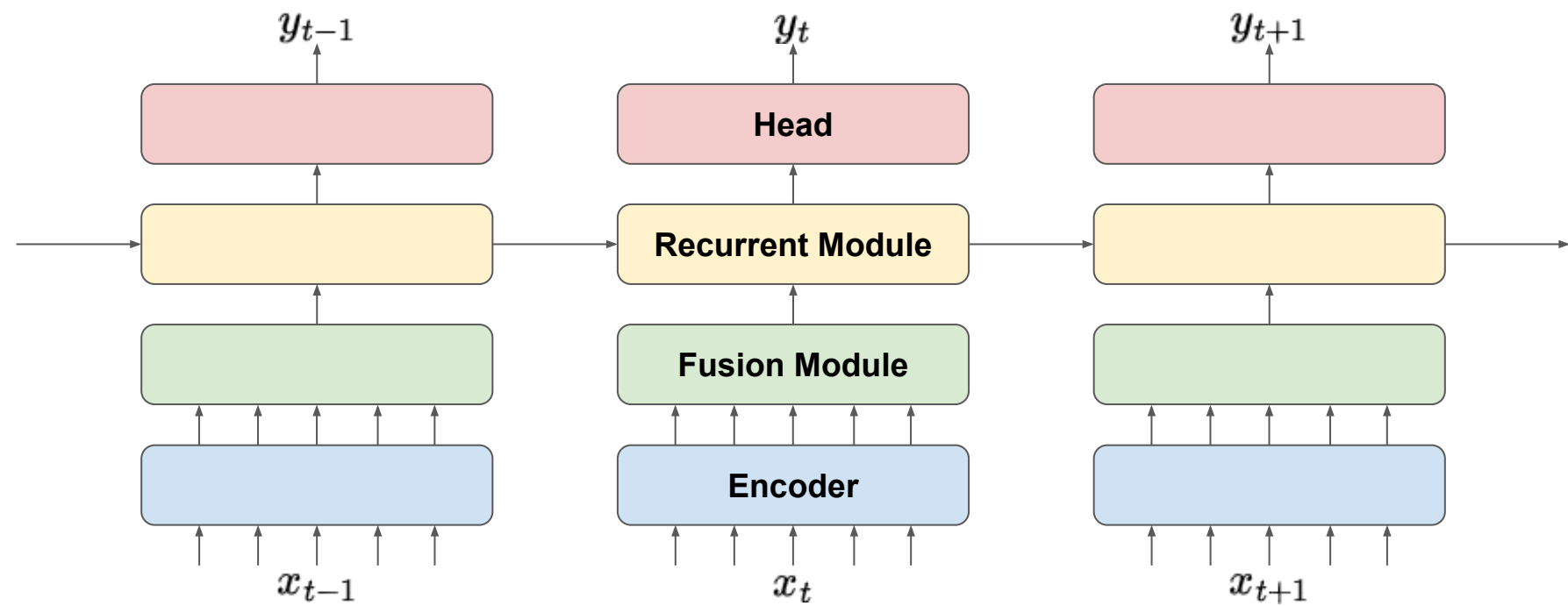
Multi-Sensor Biosignal Processing

주파수 추정과 신호 재구성을 위한
다중 센서 신호처리모델 및 학습방법

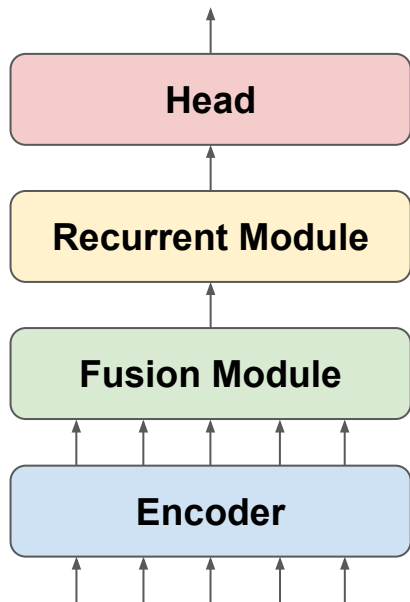
Target System



Overall Architecture



Overall Architecture — Modules



특징 벡터로부터 원하는 형태의 출력을 생성하는 모듈

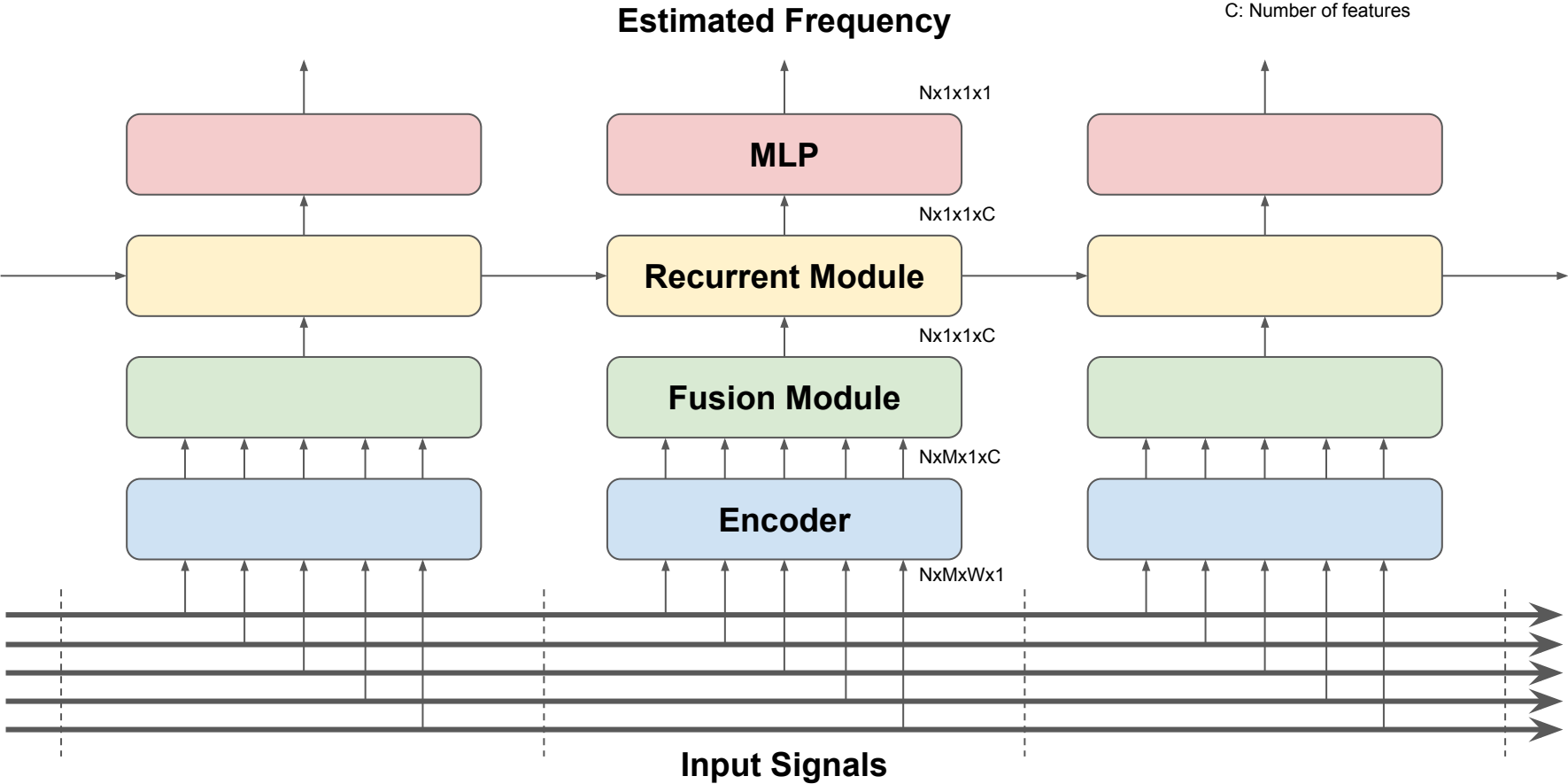
상대적으로 긴 시간에 대한 정보 전달 및 패턴 분석 모듈

각 센서의 특징 벡터들을 하나로 합하는 모듈

짧은 신호의 패턴을 분석하여 특징 벡터를 생성하는 모듈

Architecture — Frequency Estimation

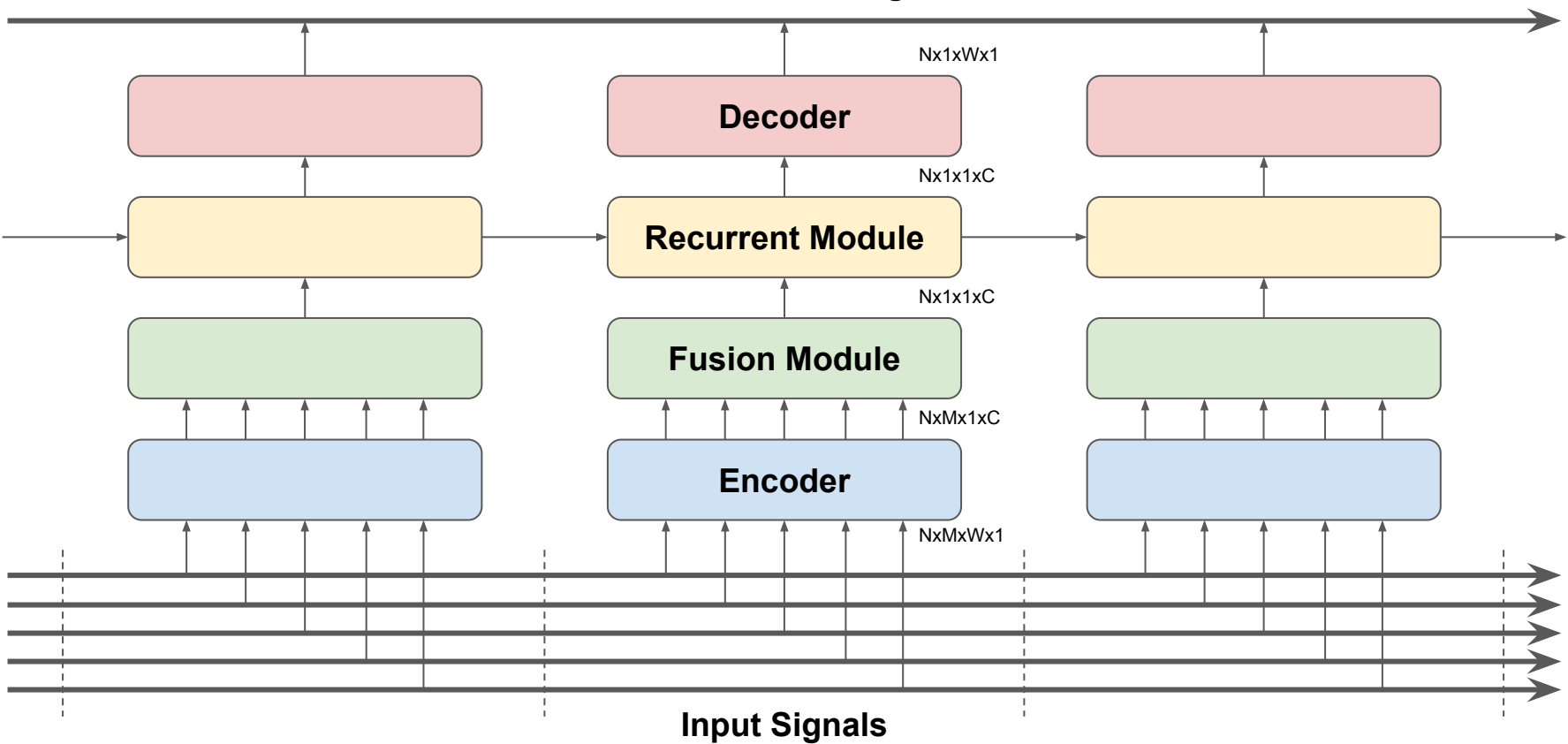
N: Batch size
M: Number of sensors
W: Window size
C: Number of features



Architecture — Signal Reconstruction

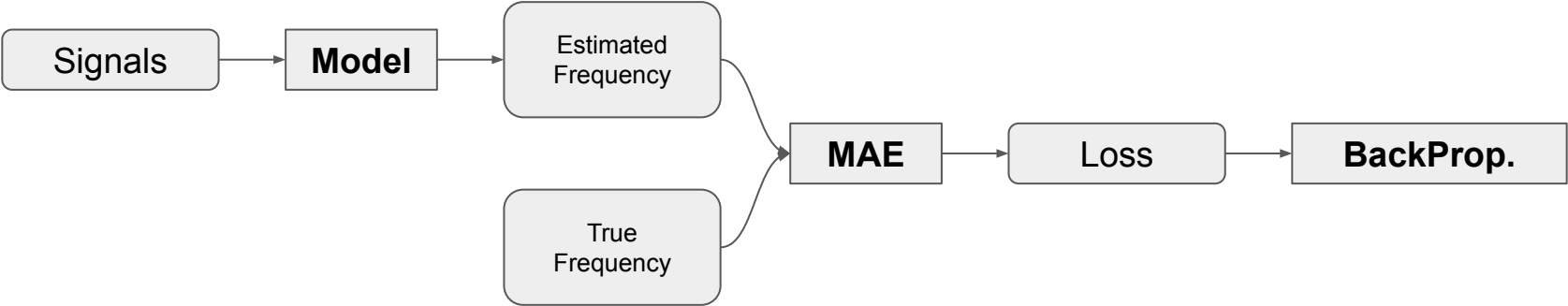
N: Batch size
M: Number of sensors
W: Window size
C: Number of features

Reconstructed Signal

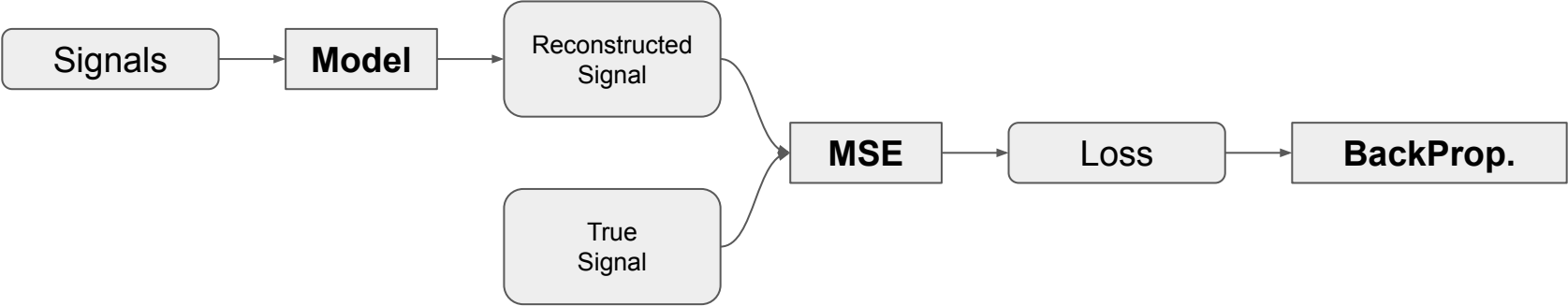


Training Flow

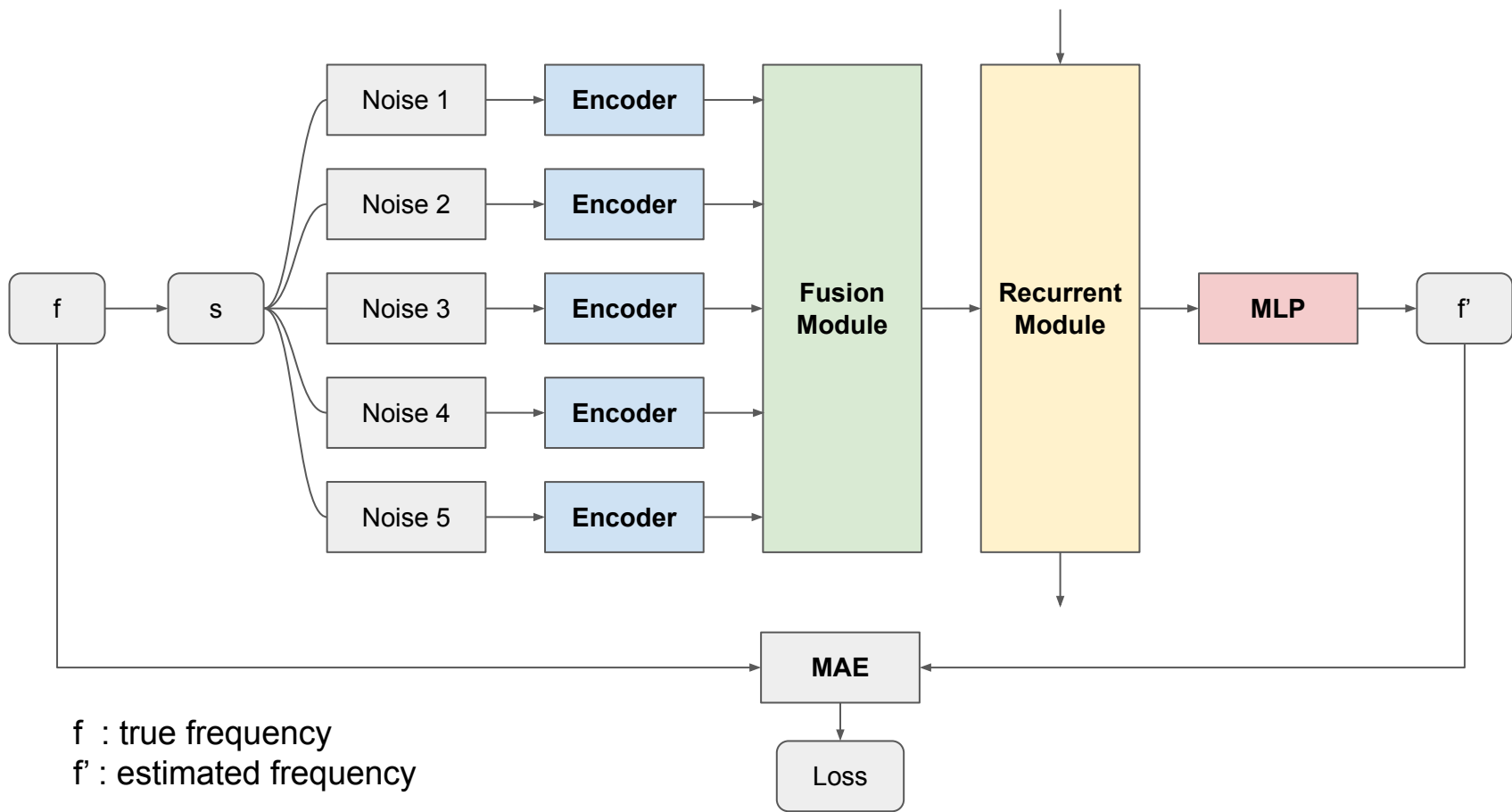
Frequency Estimation



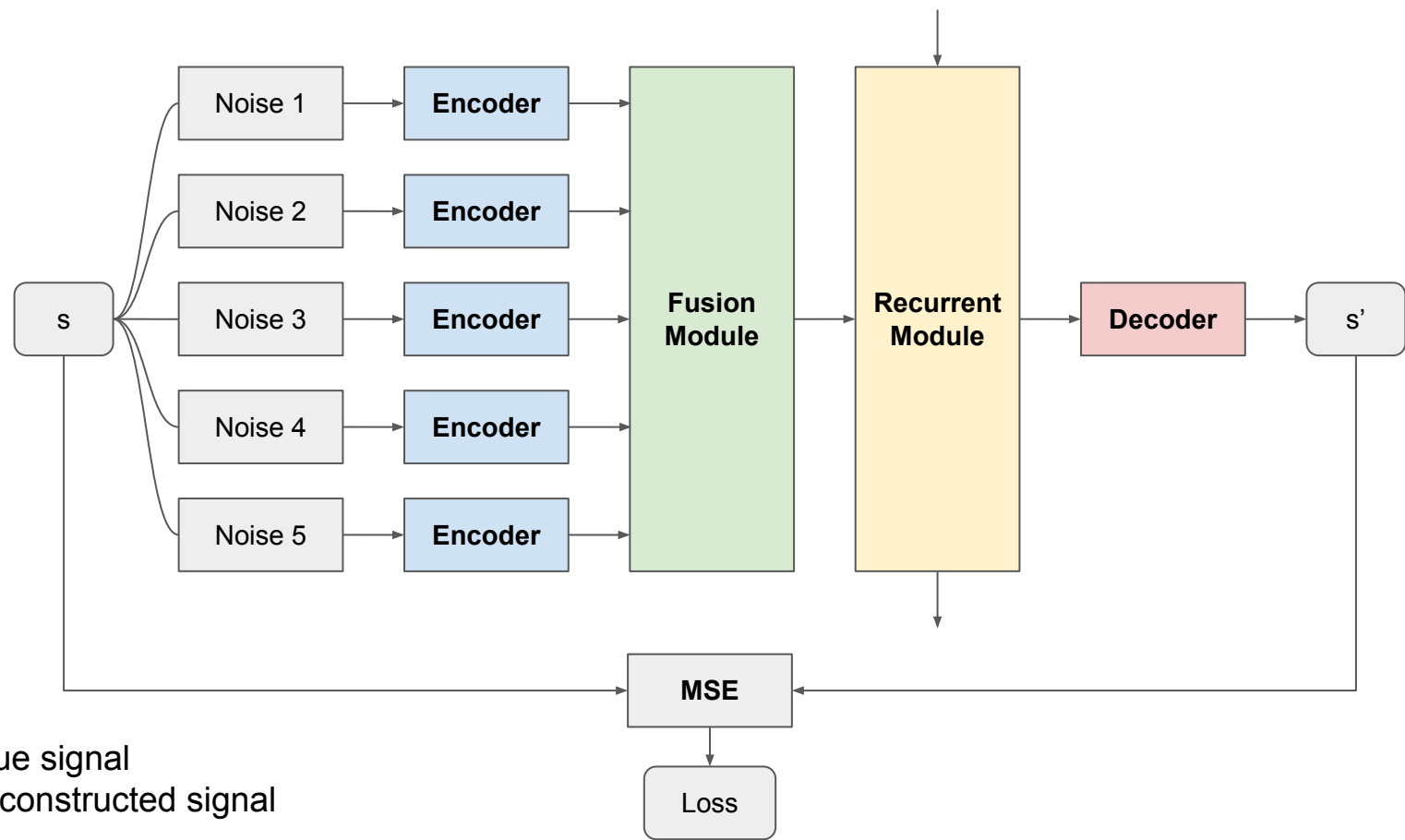
Signal Reconstruction



Training Flow — Frequency Estimation



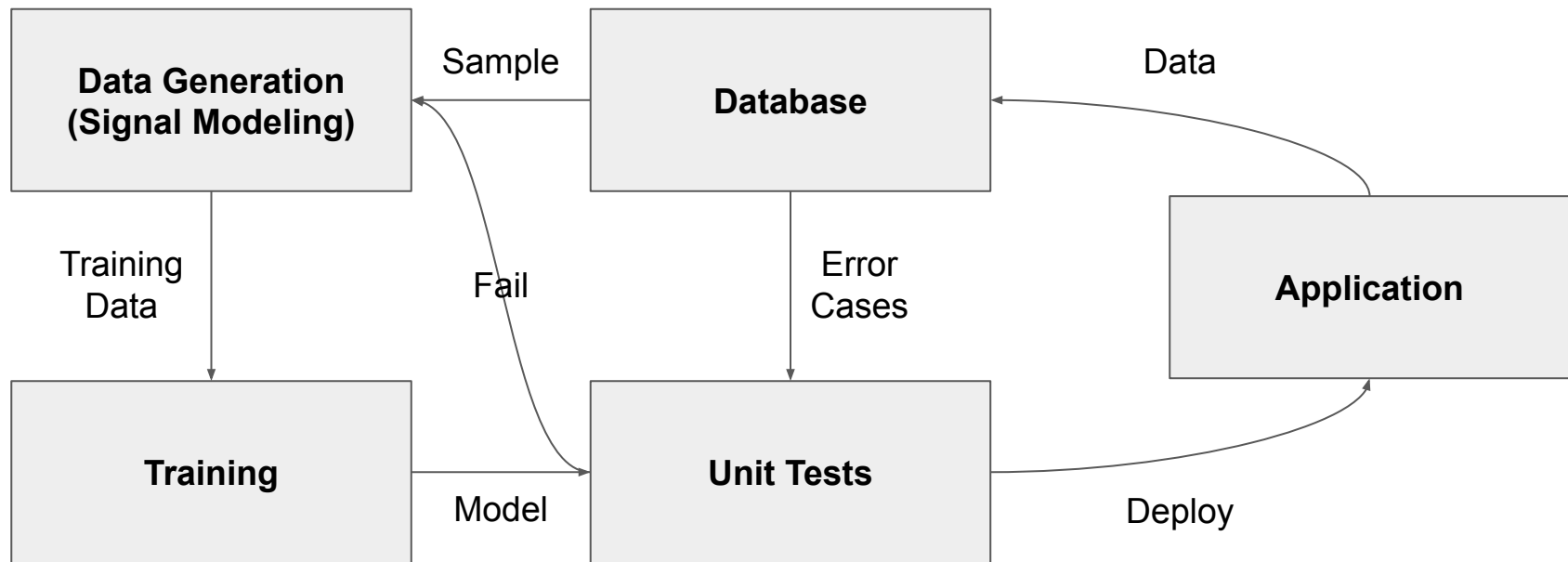
Training Flow — Signal Reconstruction



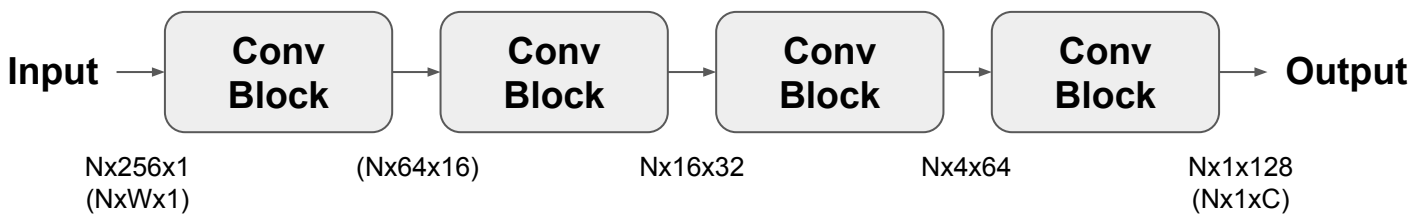
Training Pipeline

Training → Virtual Data

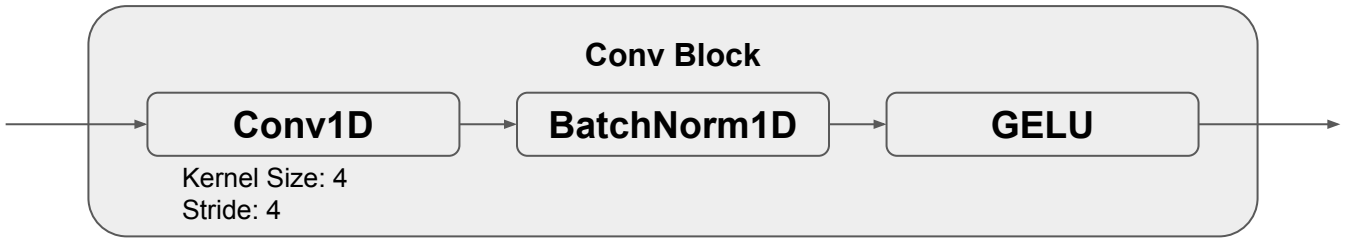
Unit Tests → Real + Virtual Data



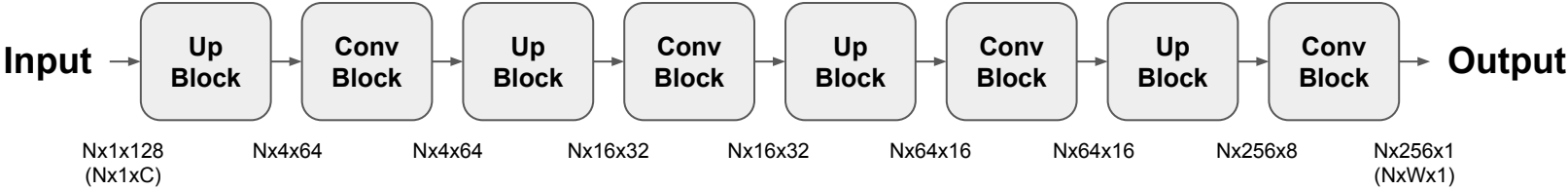
Encoder



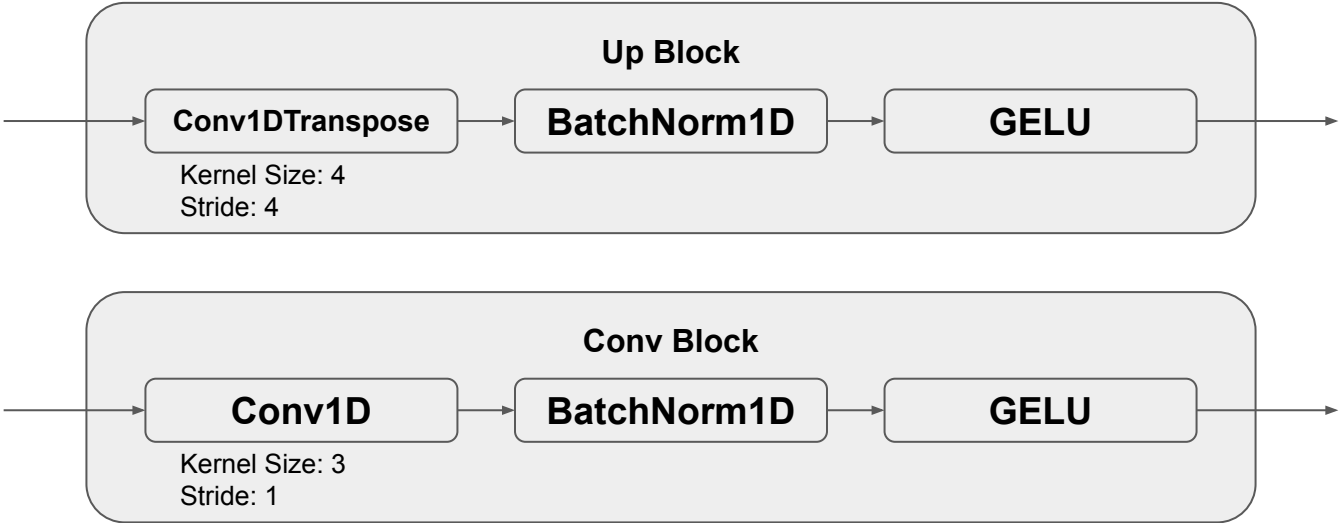
N: Batch size
W: Window size
C: Number of features



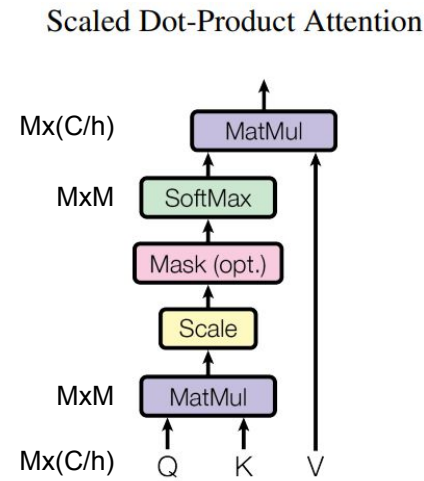
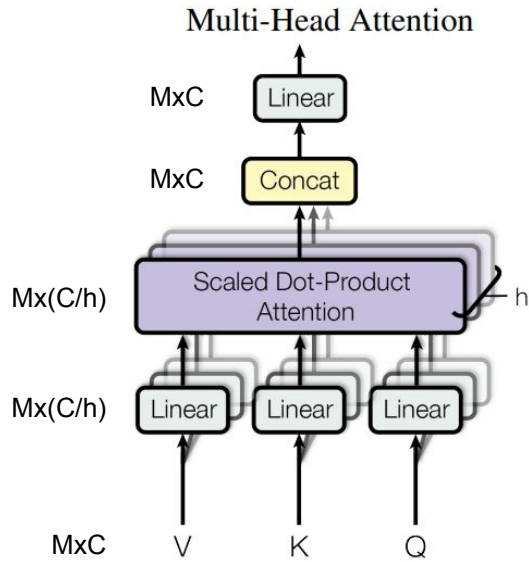
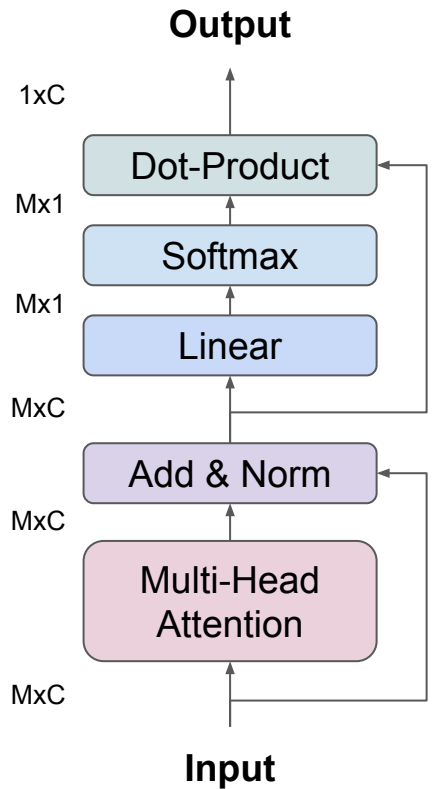
Decoder



N: Batch size
W: Window size
C: Number of features



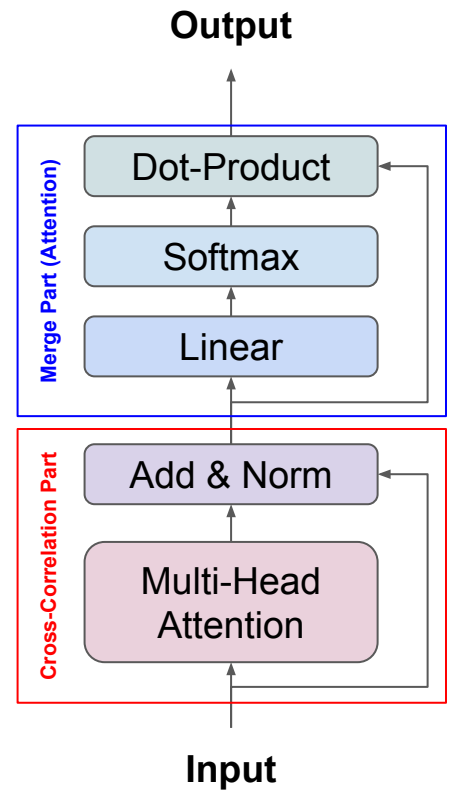
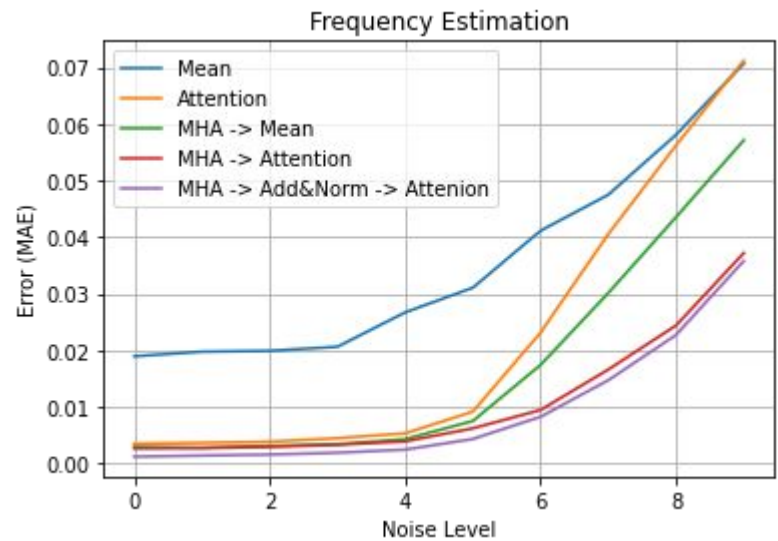
Fusion Module



M: Number of sensors
C: Number of features
h: Number of heads

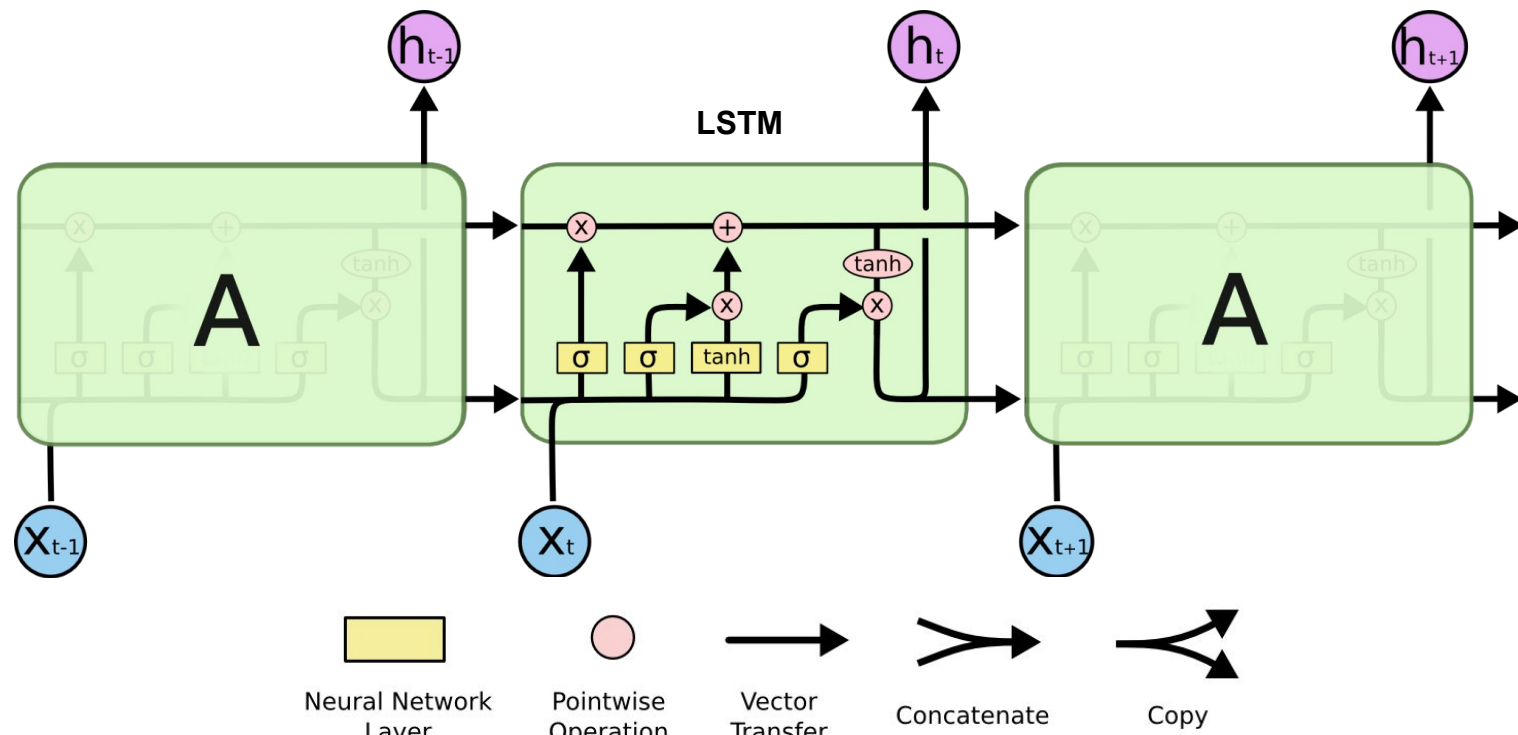
Fusion Module — Comparison

- 5가지 **Fusion Module** 구조의 성능을 비교해본 결과 MHA(Multi-Head Attention)과 Attention을 결합한 구조가 가장 좋은 성능을 보였습니다.

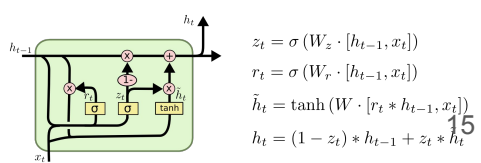


→ Fusion Module에 대한 상세 설명은 Appendix에 있습니다.

Recurrent Module

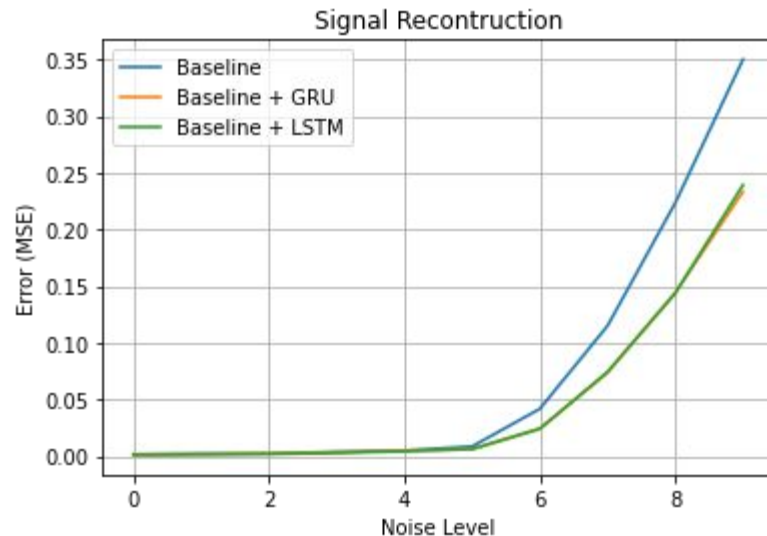
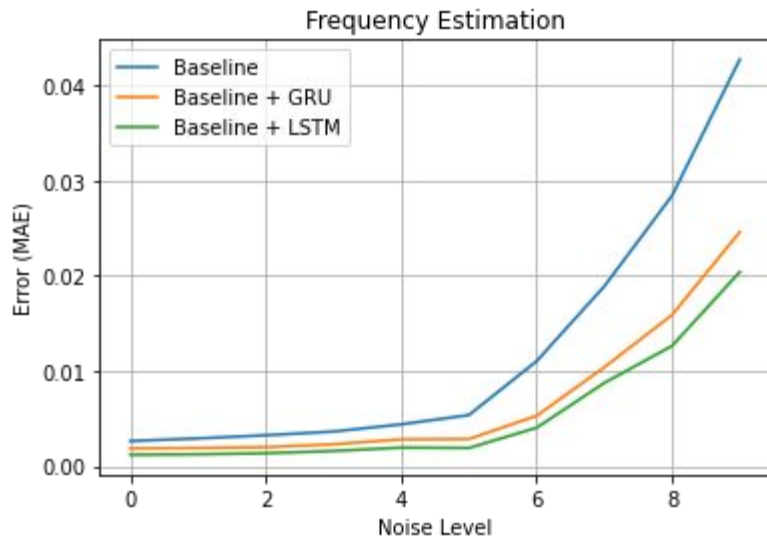


GRU

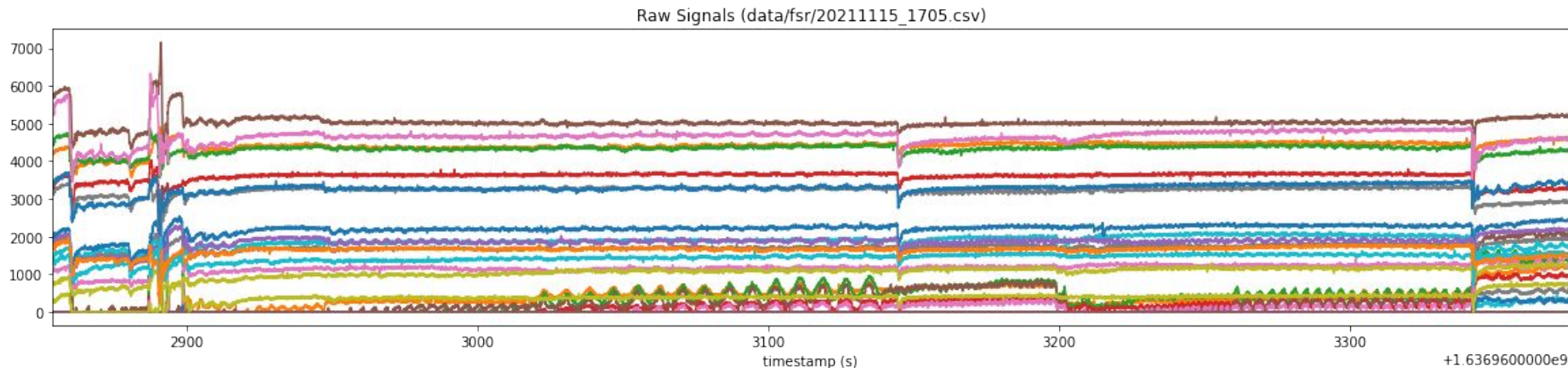


Recurrent Module — Comparison

- **Recurrent Module** (LSTM or GRU)를 추가하였을 때 유의미한 성능 향상을 확인하였습니다.
- (Baseline은 Encoder + Fusion Module + Head 로 이루어진 모델입니다.)

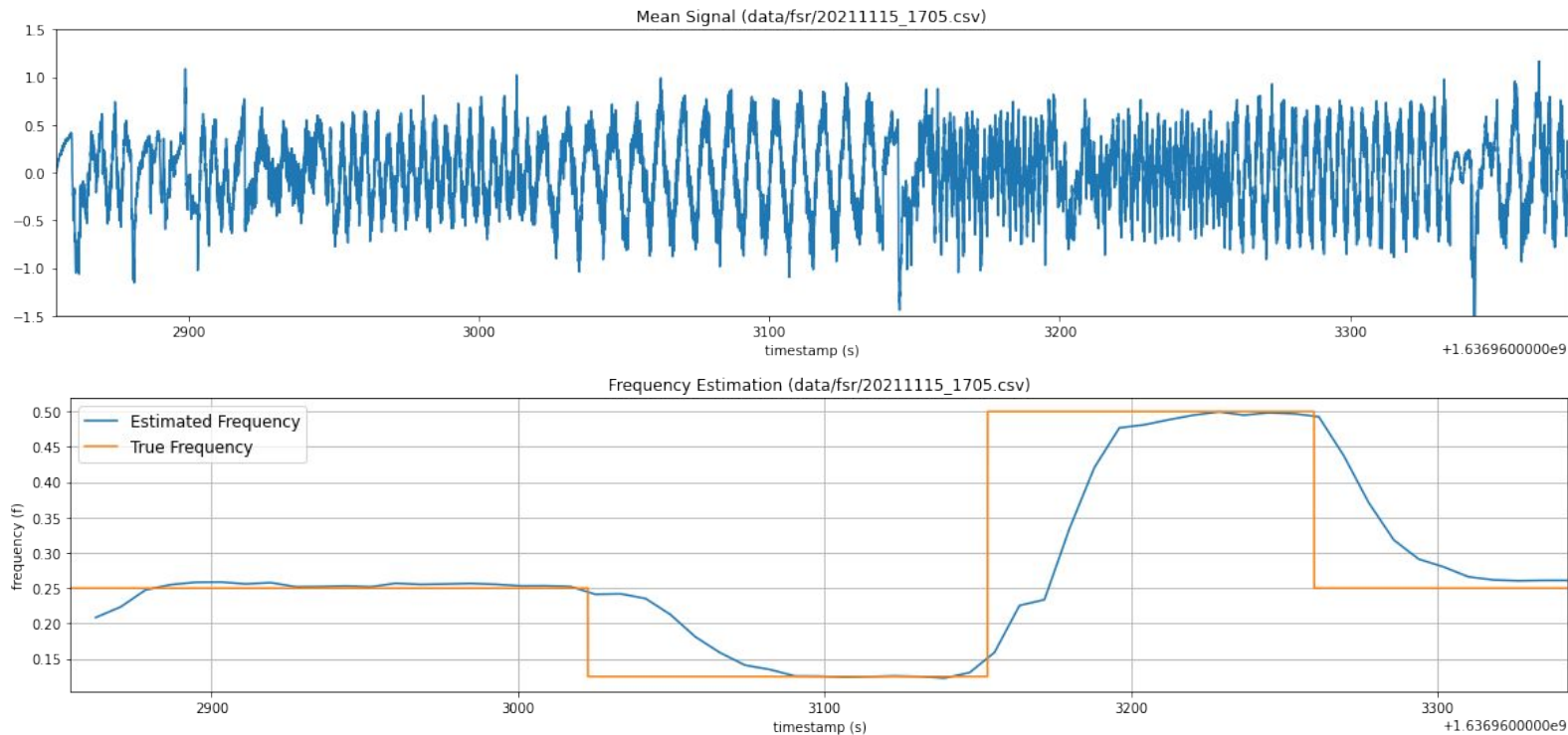


Experiment Results — FSR Sensor Signals



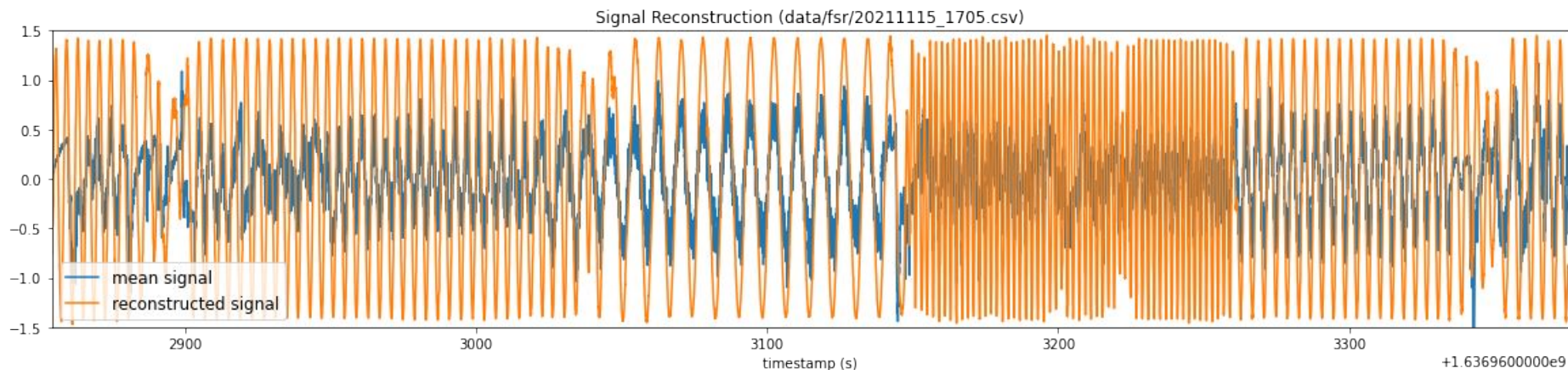
- 시트에 설치된 **32개의 FSR** 센서들의 신호 그래프입니다.
- 센서 데이터마다 호흡 신호의 세기와 **SNR(Signal-to-Noise Ratio)**이 다르게 나타납니다.
- 위 그래프를 통해 각 신호들을 적절히 융합할 수 있는 알고리즘의 필요성을 확인할 수 있습니다.
- 실제로 여러 사람의 데이터를 확인해보면 사람마다 또는 자세마다 센서들의 **SNR**이 달라집니다.

Experiment Results — Frequency Estimation



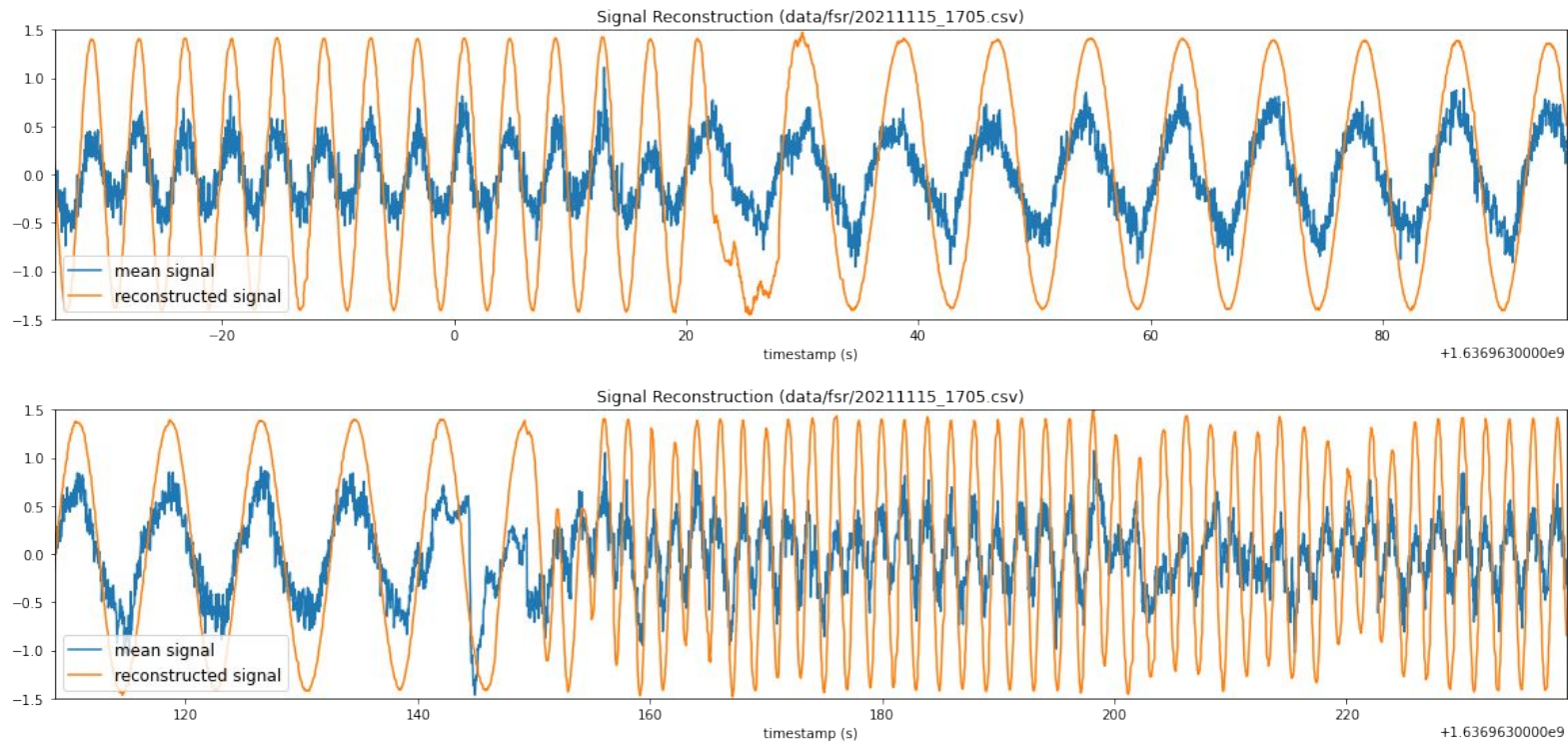
- 호흡수를 의도적으로 바꾸면서 측정한 데이터입니다. (0.25 → 0.125 → 0.5 → 0.25 Hz)
- 위 그림은 모든 센서의 평균 신호를 그린 그래프입니다.
- 신호처리모델로 주파수 추정한 결과 **Transition**이 일어나는 부분 외에는 안정적으로 추정하는 것을 확인할 수 있습니다.

Experiment Results — Signal Reconstruction & Frequency Estimation



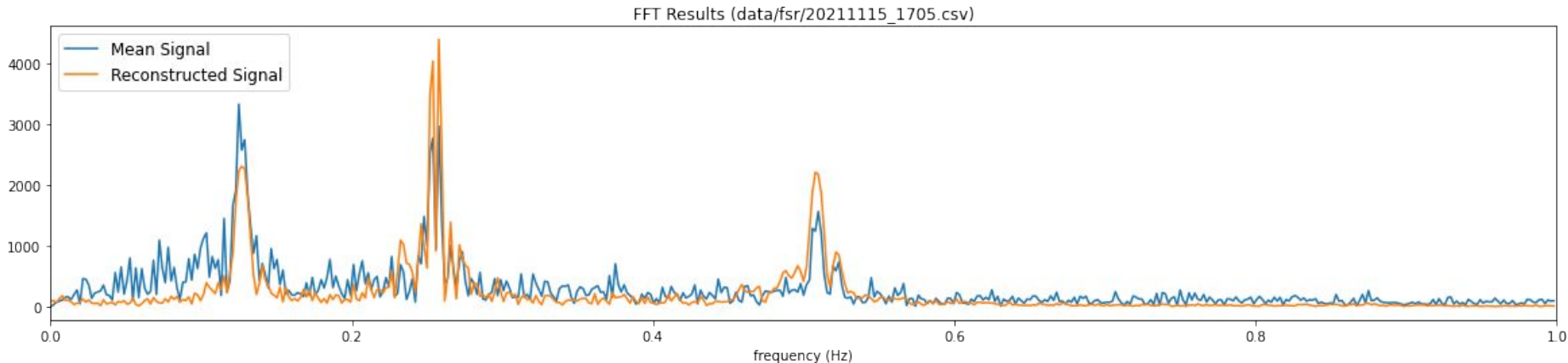
- 파란색은 모든 센서의 평균 신호이고, 주황색은 **Signal Reconstruction** 결과입니다.
- 신호 재구성 결과 **노이즈가 줄어들고 크기가 일정**한 신호가 생성됩니다.
- **Raw Data**의 처리 과정과 비교했을 때 추가 후처리 과정을 생략할 수 있습니다.

Experiment Results — Signal Reconstruction



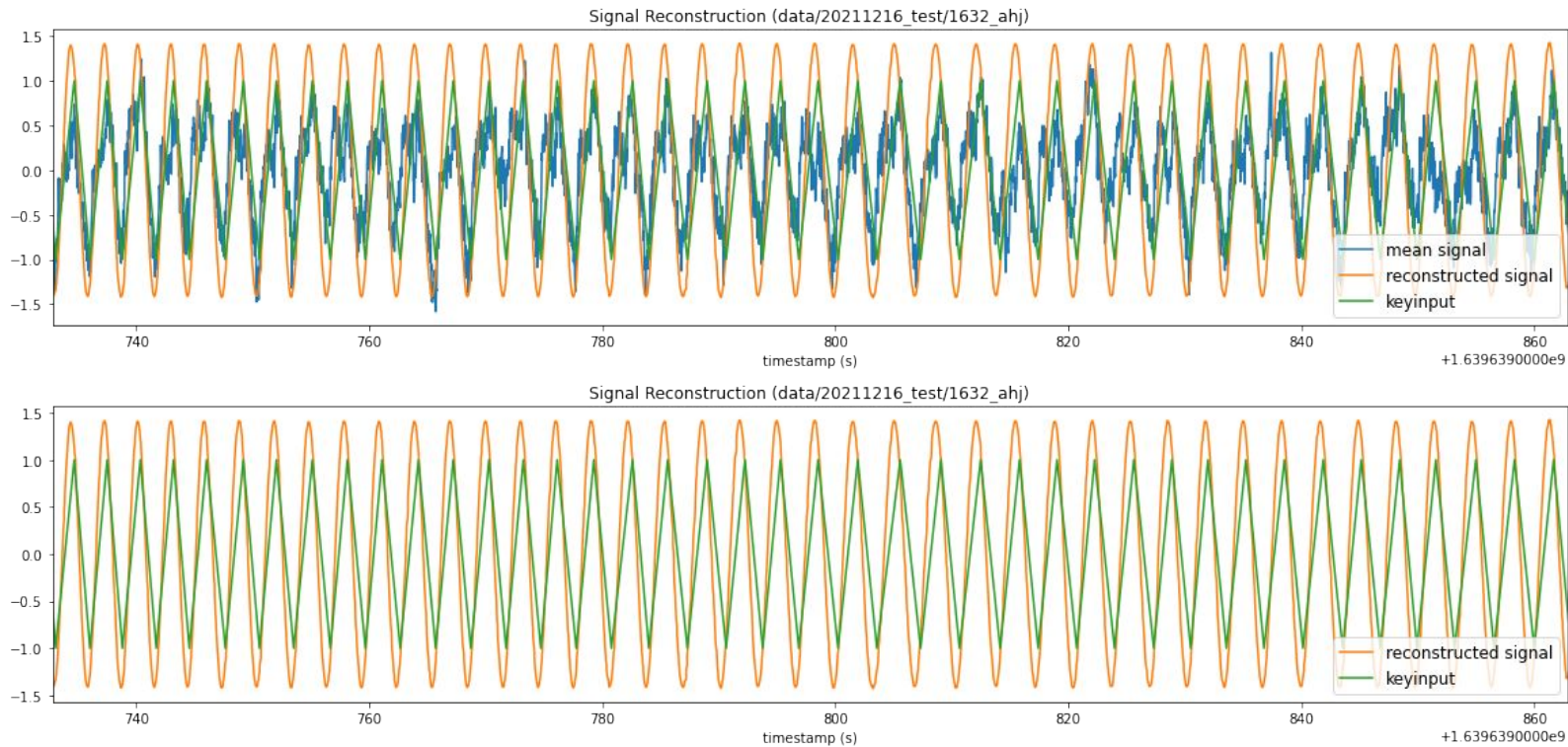
- 신호 재구성 결과를 부분 확대한 그래프입니다.
- 주파수가 변하는 부분을 제외하면 안정적인 신호 재구성 결과를 보여줍니다.

Experiment Results — Signal Reconstruction & FFT



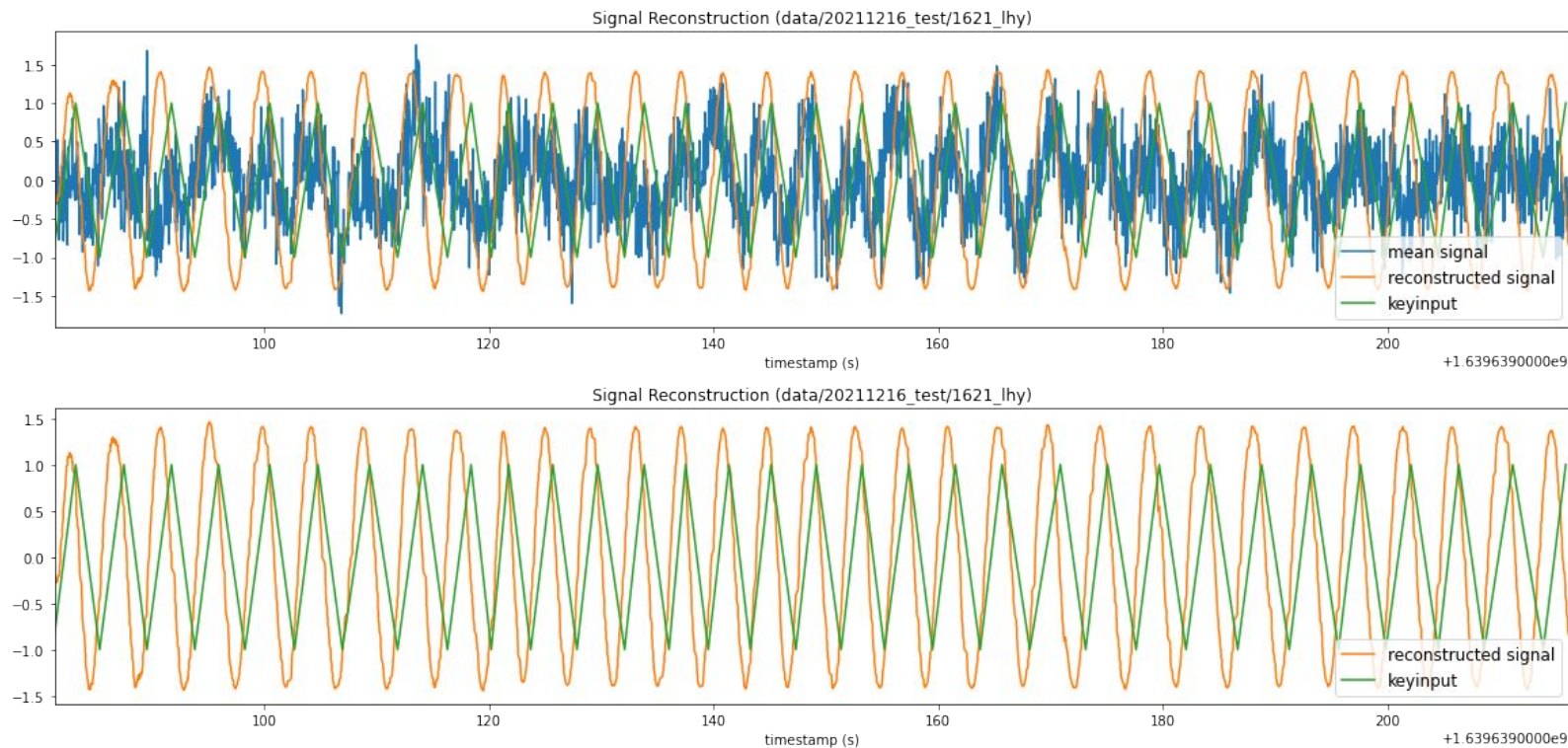
- 신호 평균과 **Reconstructed Signal**의 FFT 결과입니다.
- 실제 신호 주파수(0.125, 0.25, 0.5 Hz)를 제외한 나머지 주파수 성분(**Noise**)이 상대적으로 줄어든 것을 확인할 수 있습니다.

Experiment Results — Signal Reconstruction



- Signal Reconstruction 결과입니다.
- 측정 시 피실험자분이 호흡에 맞춰 키입력을 하였고 재구성 결과와 키입력 값이 일치하는 모습을 확인할 수 있습니다.

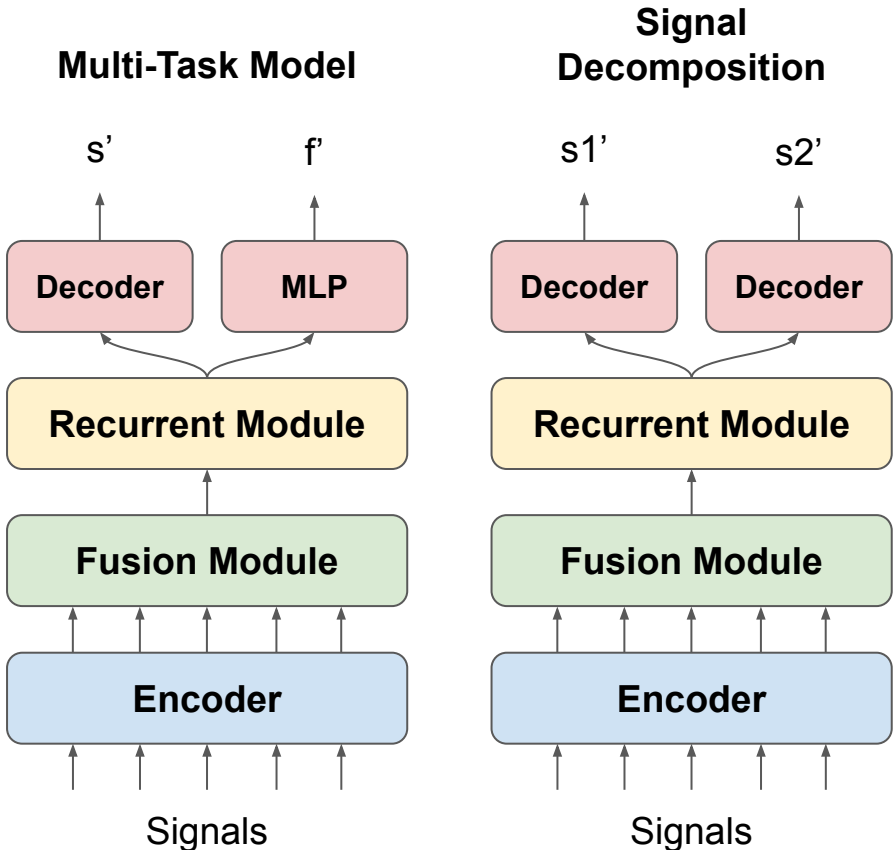
Experiment Results — Signal Reconstruction



- Signal Reconstruction 결과입니다.
- 측정 시 피실험자분이 호흡에 맞춰 키입력을 하였고 재구성 결과와 키입력 값이 일치하는 모습을 확인할 수 있습니다.

Future Works

- 0 Hz
- Time-Varying Frequency
- Phase Noise (Synchronization Error)
- Multi-Task Model
- Signal Decomposition



References

<https://arxiv.org/abs/1606.08415> (GELU)

<https://arxiv.org/abs/1706.03762> (Multi-Head Attention)

<https://colah.github.io/posts/2015-08-Understanding-LSTMs/> (LSTM)

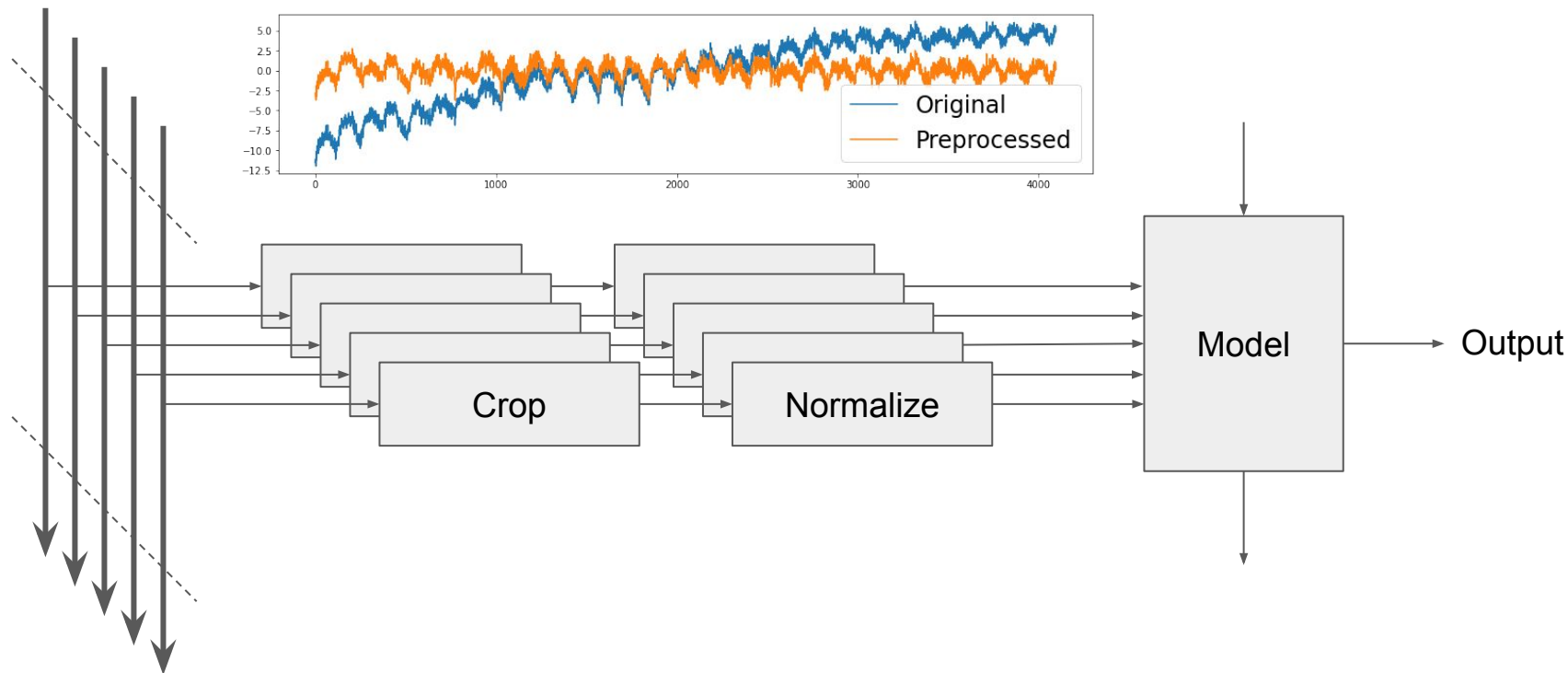
<https://pubmed.ncbi.nlm.nih.gov/29738456/> (HR/RR detection with a seat sensor)

<https://www.ijcai.org/proceedings/2019/0431.pdf> (attention-based sensor fusion)

Appendix

Signal Preprocessing

센서 신호 데이터의 **Drift** 현상을 제거하기 위해 모델 입력 전에 **Crop** → **Normalize** 처리합니다.



Signal Model

$$s(n) = \cos(2\pi \frac{f}{f_s} n + \phi)$$

$$x(n) = s(n) + w(n)$$

$$w \sim \mathcal{N}(0, \sigma)$$

$$f \sim \mathcal{U}(0.1, 0.5)$$

$$\phi \sim \mathcal{U}(-\pi, \pi)$$

$$\sigma \sim \mathcal{U}(\sigma_{min}, \sigma_{max})$$

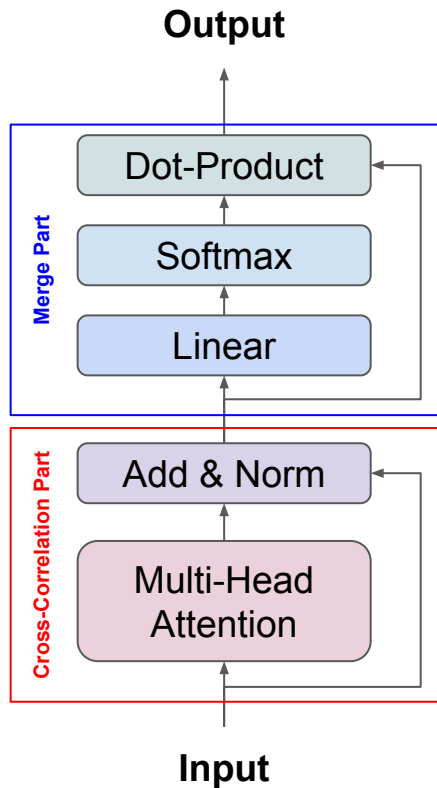
f : frequency

f_s : sampling frequency

ϕ : phase

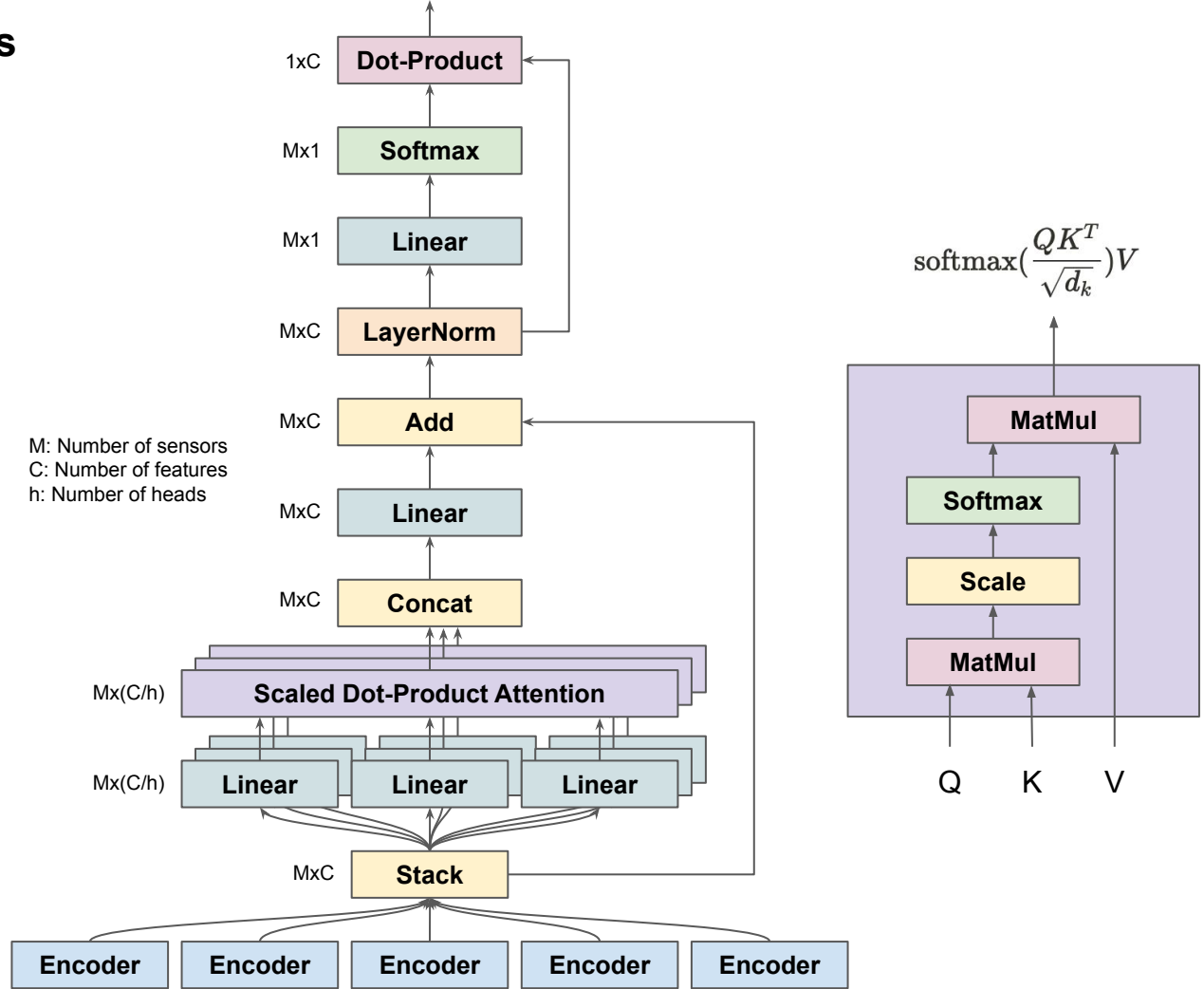
- $s(n)$ 은 타겟 신호로 호흡 또는 심박 신호가 될 수 있습니다.
- 대상과 센서에 따라 신호 파형이 다르기 때문에 **적합한 신호 파형**을 생성하는 것이 중요합니다.
- $w(n)$ 은 노이즈로 현재는 **white Gaussian noise**로 표현했지만 여러 환경에서 나타날 수 있는 노이즈를 고려하고 적용해야 합니다.
- $x(n)$ 은 센서에서 측정된 신호로 타겟 신호와 노이즈가 합쳐진 신호입니다.
- 다양한 신호를 생성하기 위해 주파수와 위상은 균일 분포 랜덤 변수로 생성합니다. 이 때 랜덤 변수의 범위는 목표로 하는 시스템 스펙이 기준이 됩니다. (ex. 측정 호흡수 범위: 6~30 bpm (0.1~0.5 Hz))
- 노이즈의 파라미터도 랜덤 변수로 설정하여 센서에 따라 노이즈 파워가 달라지도록 합니다.
- 학습 파이프라인이 진행되면 늘어나는 **Unit Tests**를 통과하기 위해 **Signal Model**이 지속적으로 업데이트됩니다.

Fusion Module — Details



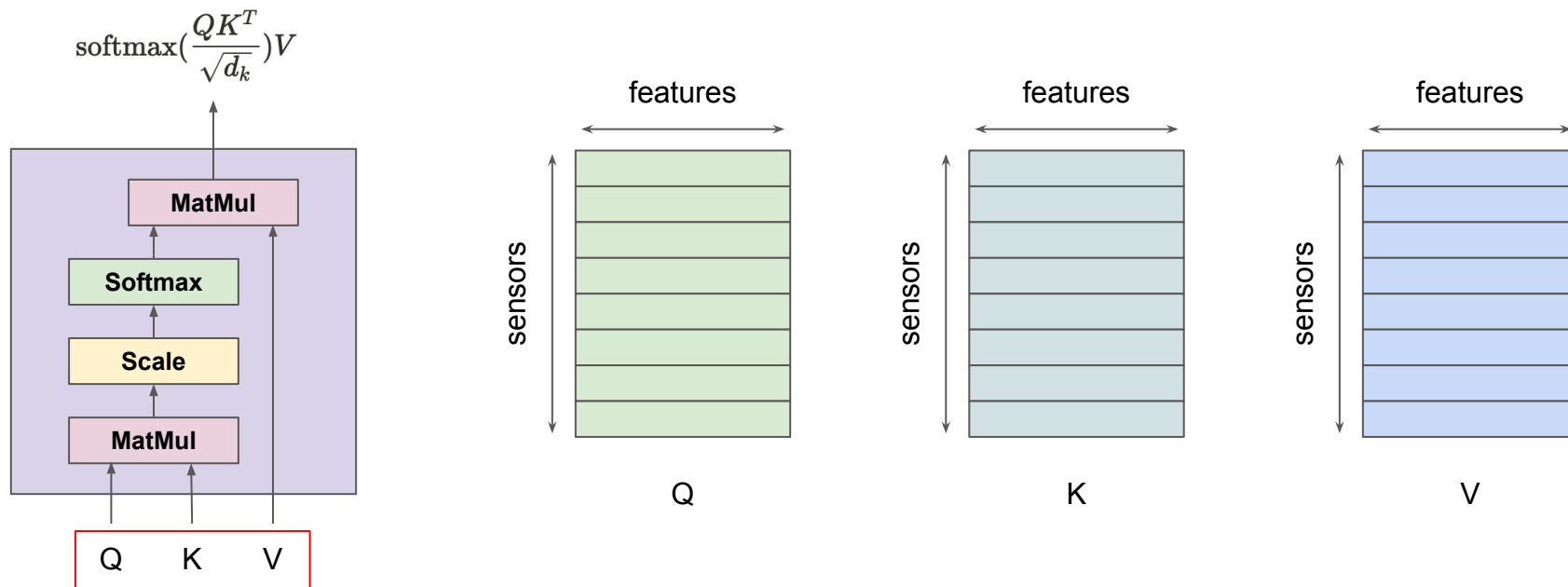
- 서로 다른 센서 데이터를 융합하는 Fusion Module은 두가지 부분으로 구성됩니다.
 - 센서 데이터 사이의 상관 관계 고려 → **Cross-Correlation Part**
 - 센서 데이터의 신뢰도 판단 및 선별 → **Merge Part**
- **Cross-Correlation Part**는 서로 다른 센서의 feature vector들의 **similarity**를 기반으로 새로운 feature vector를 만듭니다.
- 따라서 서로 다른 센서 사이의 **correlated components**를 추출할 수 있습니다.
- **Merge Part**는 **Cross-Correlation Part**에서 얻어낸 correlated components 중 신뢰할 수 있는 **components**를 선별합니다.
- **Merge Part**의 Linear 레이어 파라미터는 적합한 **components**를 선별하기 위한 기준이 됩니다.

Fusion Module — Details



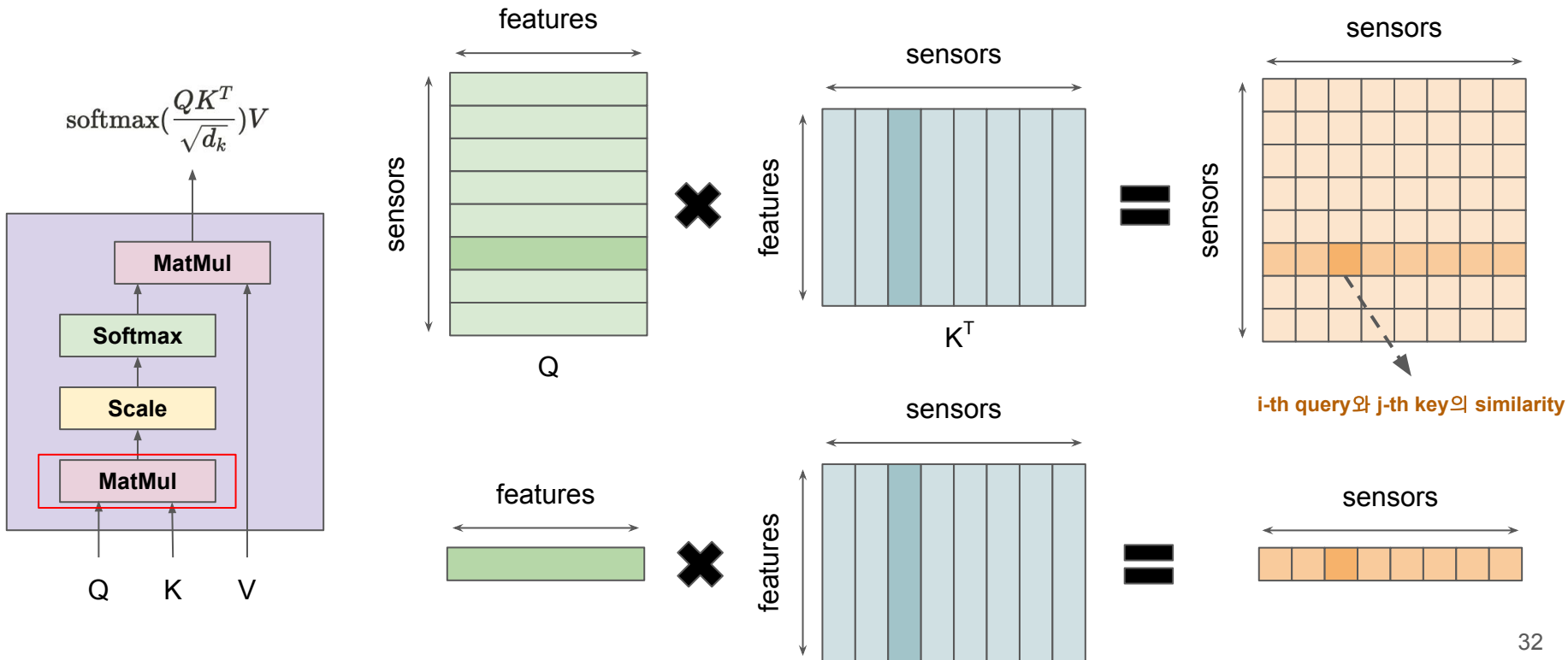
Fusion Module — Cross-Correlation Part (Scaled Dot-Product Attention)

- Scaled-Dot Product Attention의 입력은 Q(Query), K(Key), V(Vector)로 이루어집니다.
- Q, K, V는 feature vectors로부터 서로 다른 Linear Embedding을 통해 생성됩니다.
- Q와 K를 통해 얻은 센서 간의 상관 관계를 기반으로 V로부터 새로운 feature vectors를 만듭니다.



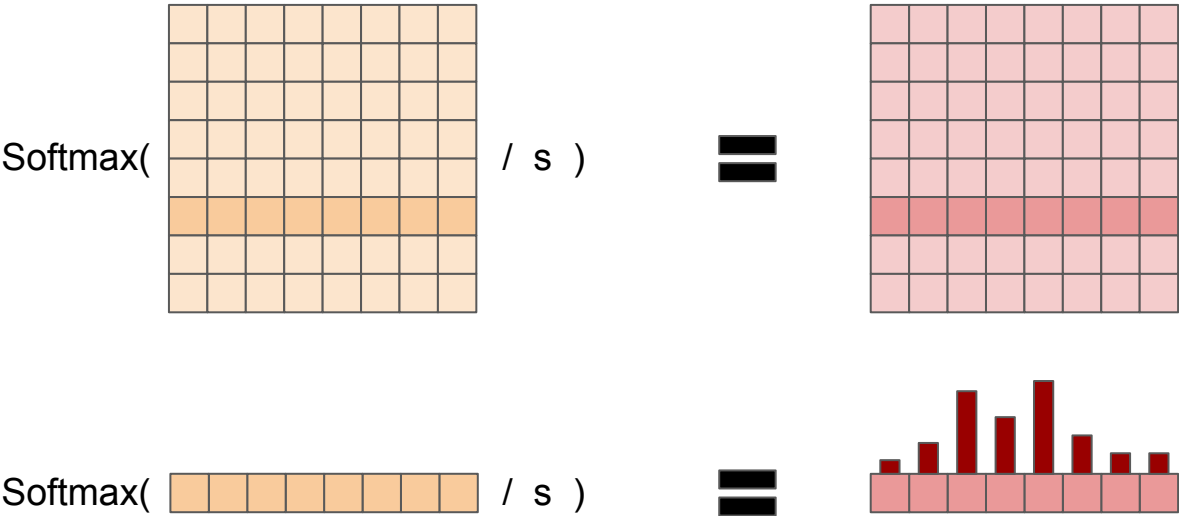
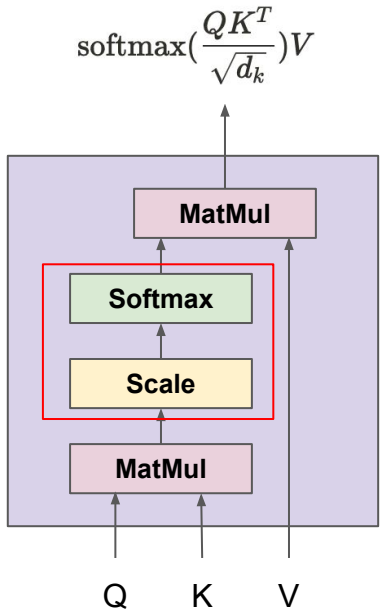
Fusion Module — Cross-Correlation Part (Scaled Dot-Product Attention)

- 새로운 feature vectors 생성은 Query가 기준이 되고 Key와의 Dot-Product 연산을 통해 센서 간의 **similarity**를 계산합니다.



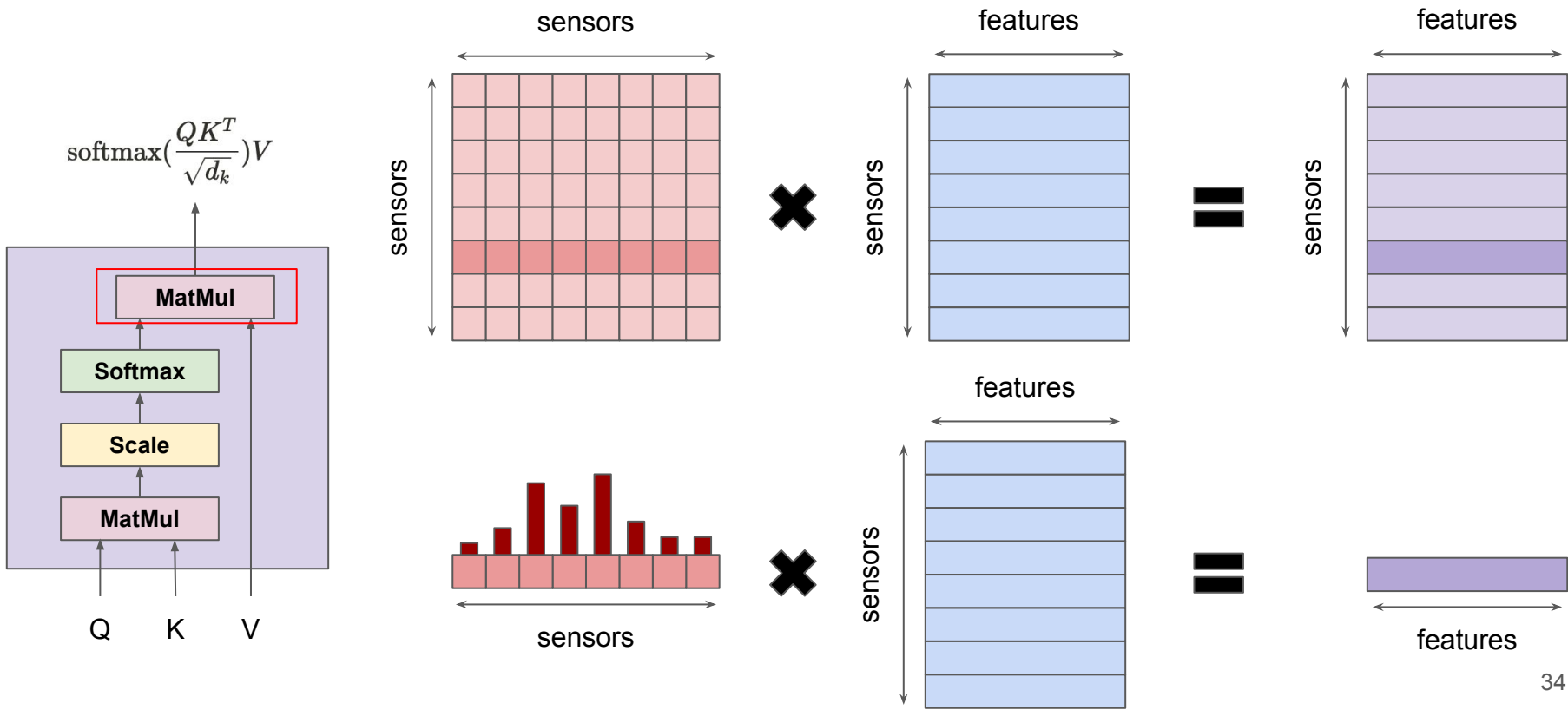
Fusion Module — Cross-Correlation Part (Scaled Dot-Product Attention)

- Query와 Key의 similarity는 Softmax 연산을 통해 **확률 분포**로 표현됩니다.
- (Softmax 연산 이전 **scaling**을 통해 분포가 치우치지 않도록 합니다.)



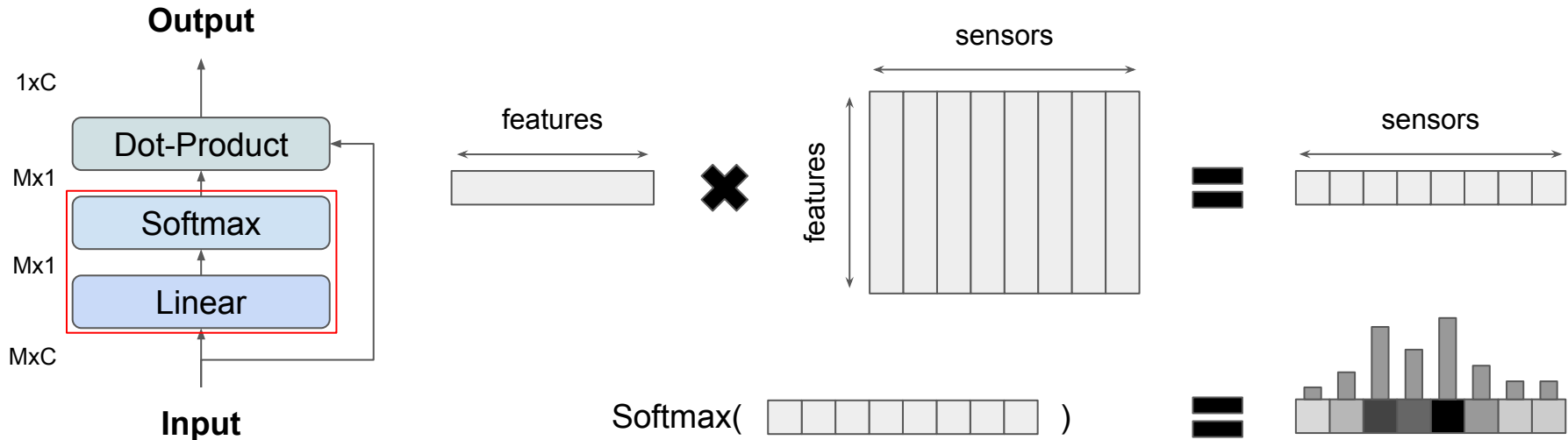
Fusion Module — Cross-Correlation Part (Scaled Dot-Product Attention)

- Key와 similarity 확률 분포의 Dot-Product를 통해 새로운 feature vectors를 생성합니다.



Fusion Module — Merge Part

- Merge Part에서는 Linear → Softmax 연산을 통해 센서에 대한 신뢰도를 판단합니다.
- Linear 레이어의 파라미터는 신뢰도 판단의 기준이 되며, 여기서 신뢰도는 기준과의 **similarity**라고 표현할 수도 있습니다.



Fusion Module — Merge Part

- 신호도와 feature vectors의 Dot-Product(weighted sum)을 통해 최종적으로 융합된 feature vector를 출력합니다.

