COMPUTER SCIENCE
& ENGINEERING
TEXAS A&M UNIVERSITY

File Type Recognition and Error Correction for NVMs with Deep Learning

Pulakesh Upadhyaya

Anxiao (Andrew) Jiang

Motivation

- **Increase in volume of data** in storage systems.
- Strong need for substantially **improved error correction capabilities.**
- **New techniques** are needed to **assist ECCs** and **improve performance.**

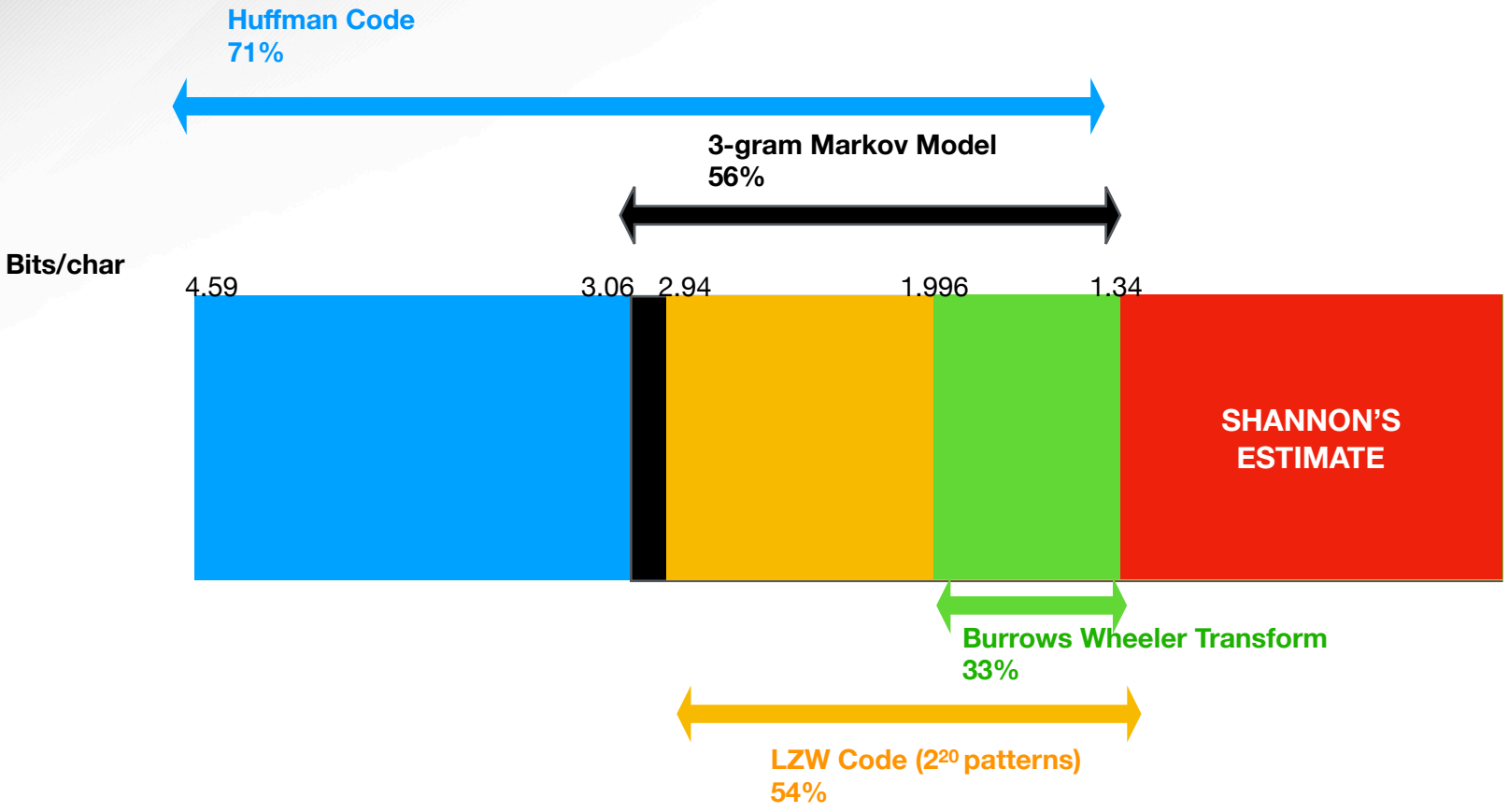
Natural Redundancy (NR)

- Natural redundancy (NR) is redundancy in data **even after compression**.

The Statue of Liberty is in the state of ……..

- NR has been used to **help ECCs correct errors.**

NR in English Language



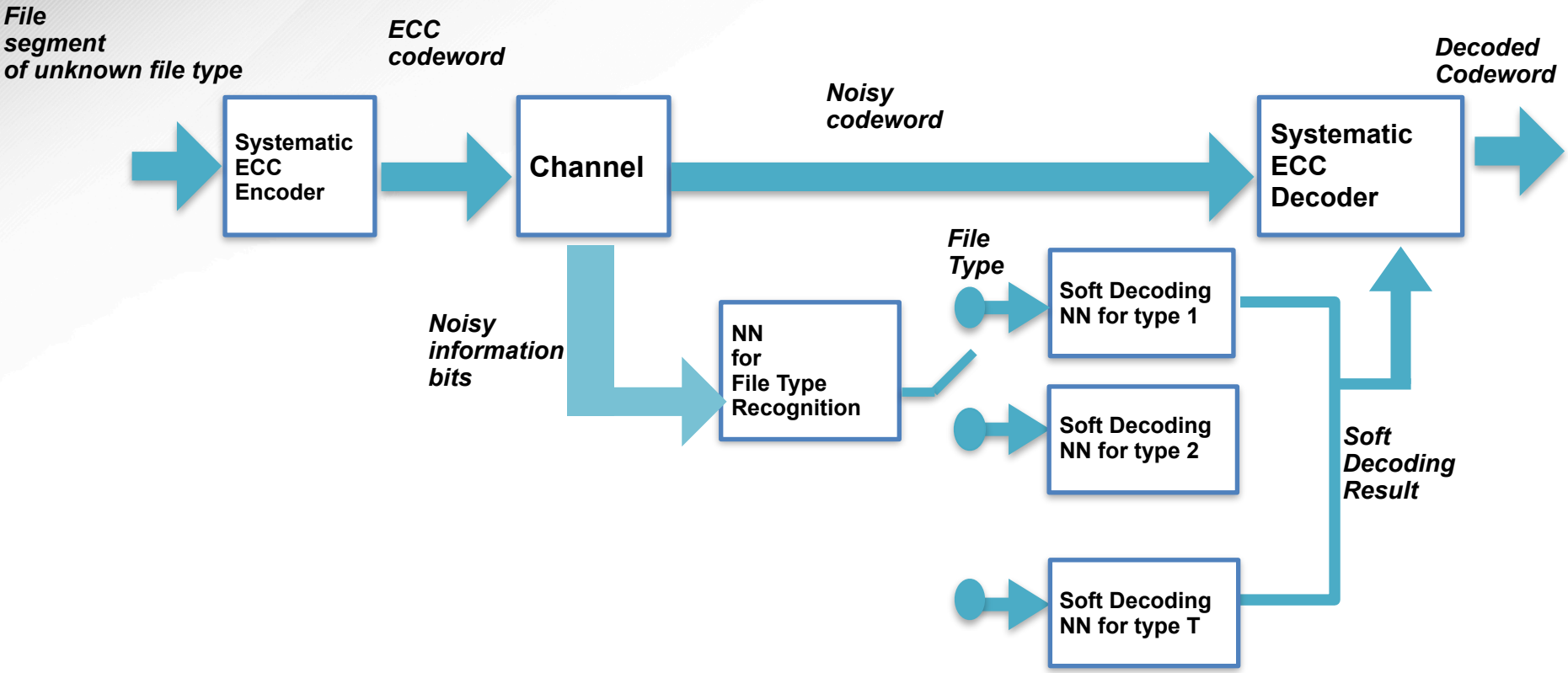
Representation Oblivious Scheme

- Previous schemes which used NR for error correction were not representation oblivious.
- Use **NR** for error correction in **a more practical setting**.

Why representation oblivious?

- **Publicly unrevealed** proprietary compression algorithms / file formats.
- Error correction is a **low-layer function** in storage architecture.
- Controllers do not always have access **to file systems.**
- We explore a **widely usable/practical** error correction based on NR.

Coding Scheme

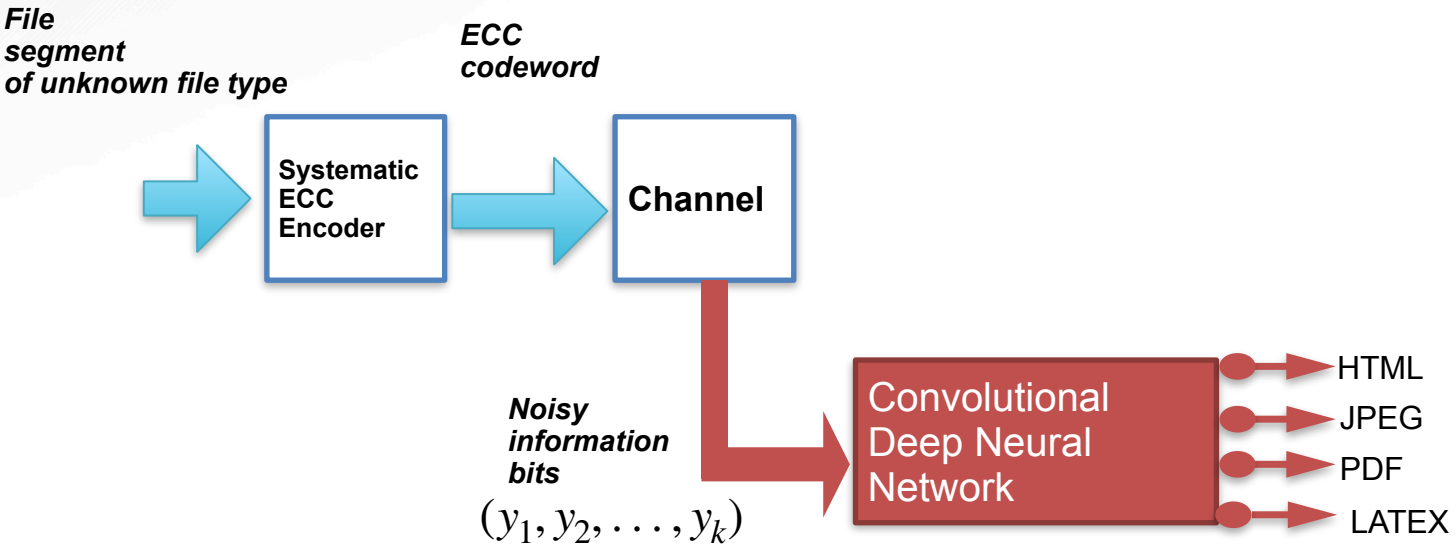


Contributions

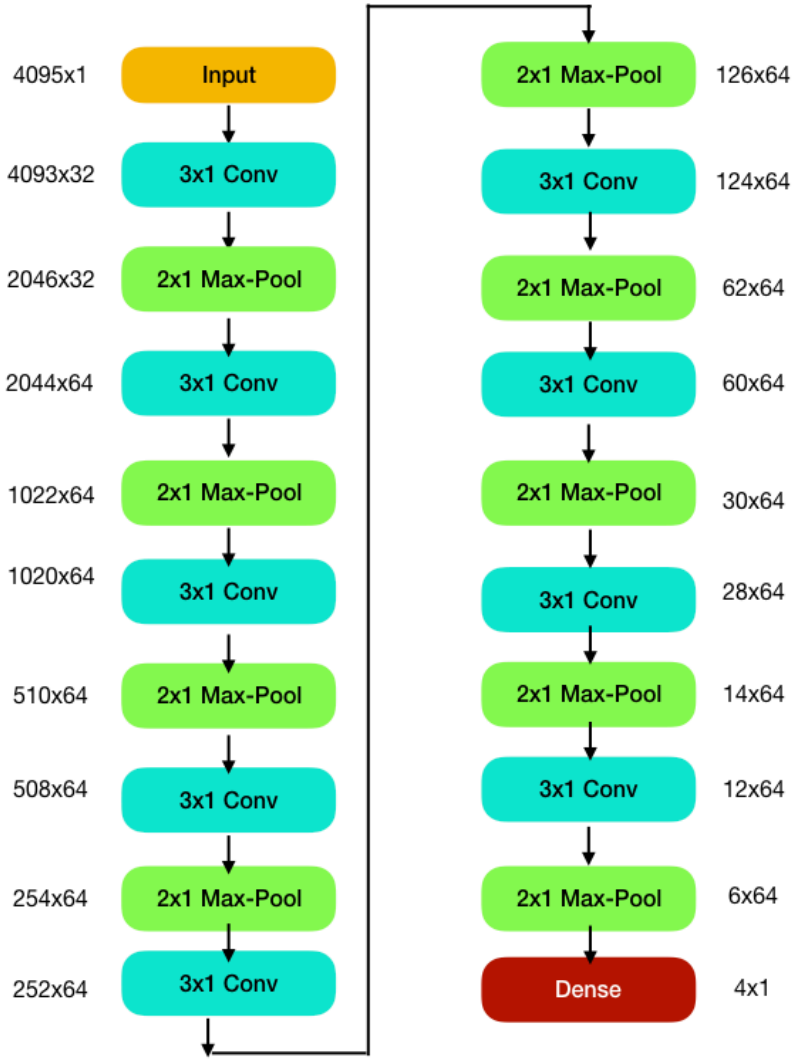
We show that

- The **file types** of bit sequences can be **recognized with high accuracy** by **deep learning**.
- Deep Learning can **perform effective soft decoding** based on natural redundancy.
- Deep Learning decoder can be **combined effectively with ECC decoder.**

File Type Recognition



DNN for File Type Recognition



- **Activation Function:** Relu for convolutional layers, sigmoid for output layer.
- **Loss function :** Cross entropy.
- **Optimizer :** Ada Delta Optimizer, whose parameters are: learning rate = 1.0, $\rho = 0.95$, $\epsilon =$ *none* and decay = 0.
- **Data : 24,000** for training data, **4,000** for validation data**, and 4,800** for test data.

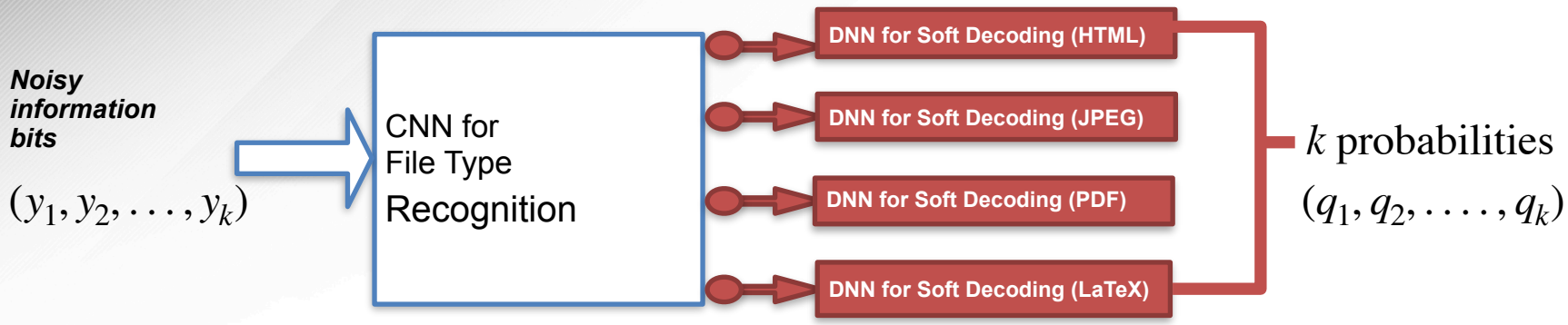
DNN for FTR : Results

Bit Error Rate (BER)	Overall Test Accuracy	HTML Test Accuracy	JPEG Test Accuracy	PDF Test Accuracy	LaTeX Test Accuracy
0.2%	99.61%	99.98%	99.52%	99.17%	99.77%
0.4%	99.69%	99.96%	99.60%	99.25%	99.96%
0.6%	99.60%	99.94%	99.48%	99.06%	99.90%
0.8%	99.69%	99.98%	99.50%	99.35%	99.92%
1.2%	99.66%	99.96%	99.23%	99.48%	99.96%
1.6%	99.58%	99.96%	99.60%	98.83%	99.92%

DNN for FTR : Results

File Type Recognition accuracy is high for all file types for **bit error rates as high as 1.6%.**

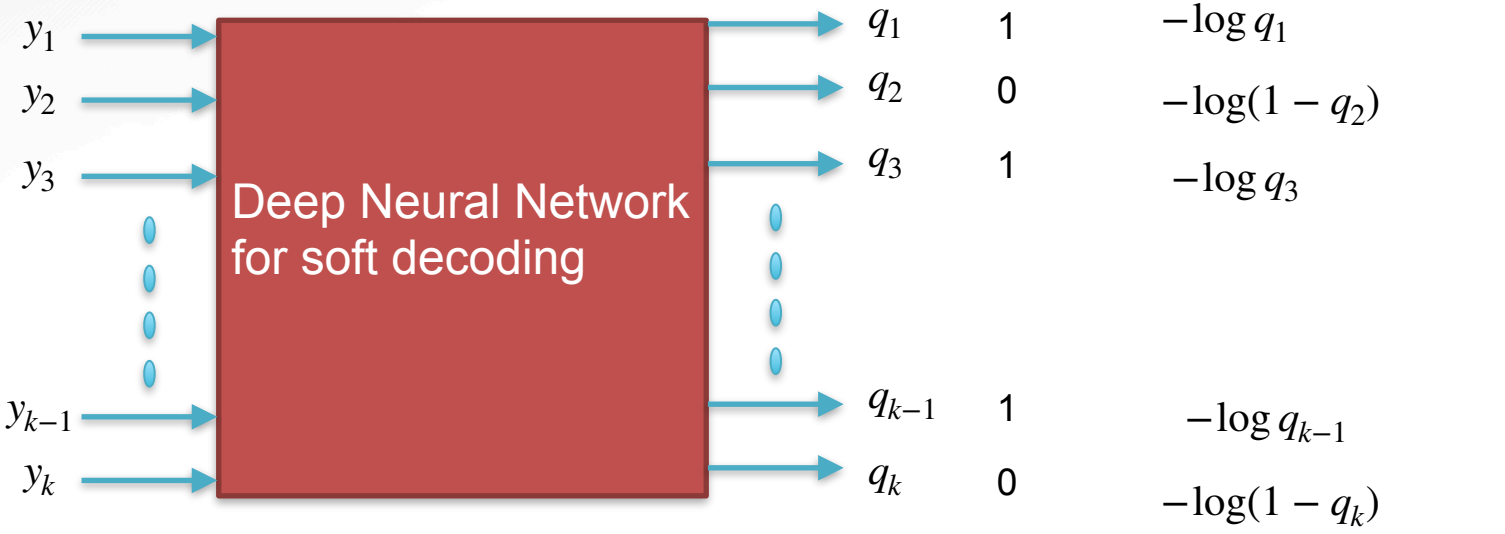
DNN for Soft Decoding



q_i : the DNN's belief that the correct value of the i-th bit of the file segment should be 1.

In experiments, we choose $k = 4095$ noisy information bits, and T = 4 file

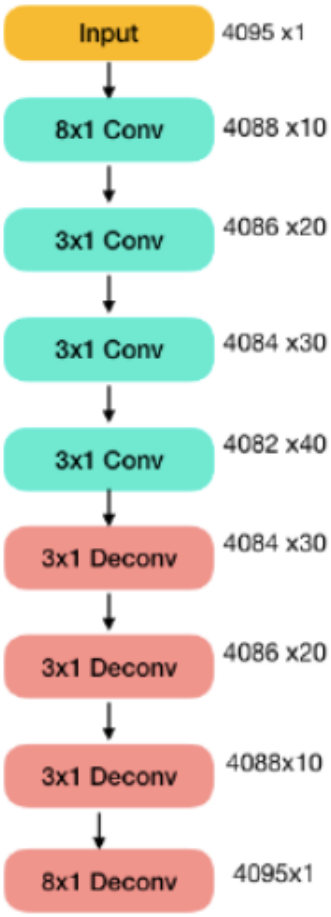
DNN for Soft Decoding



Average Loss = $-\frac{1}{k}(\log q_1 + \log(1-q_2) + \ldots + \log(1-q_k))$

- Estimation of probabilities is essentially a **regression task.**
- The above loss function is minimized when estimates are correct.
- The loss function is the same as cross entropy, therefore it can also be considered as a **classification task.**

DNN architecture



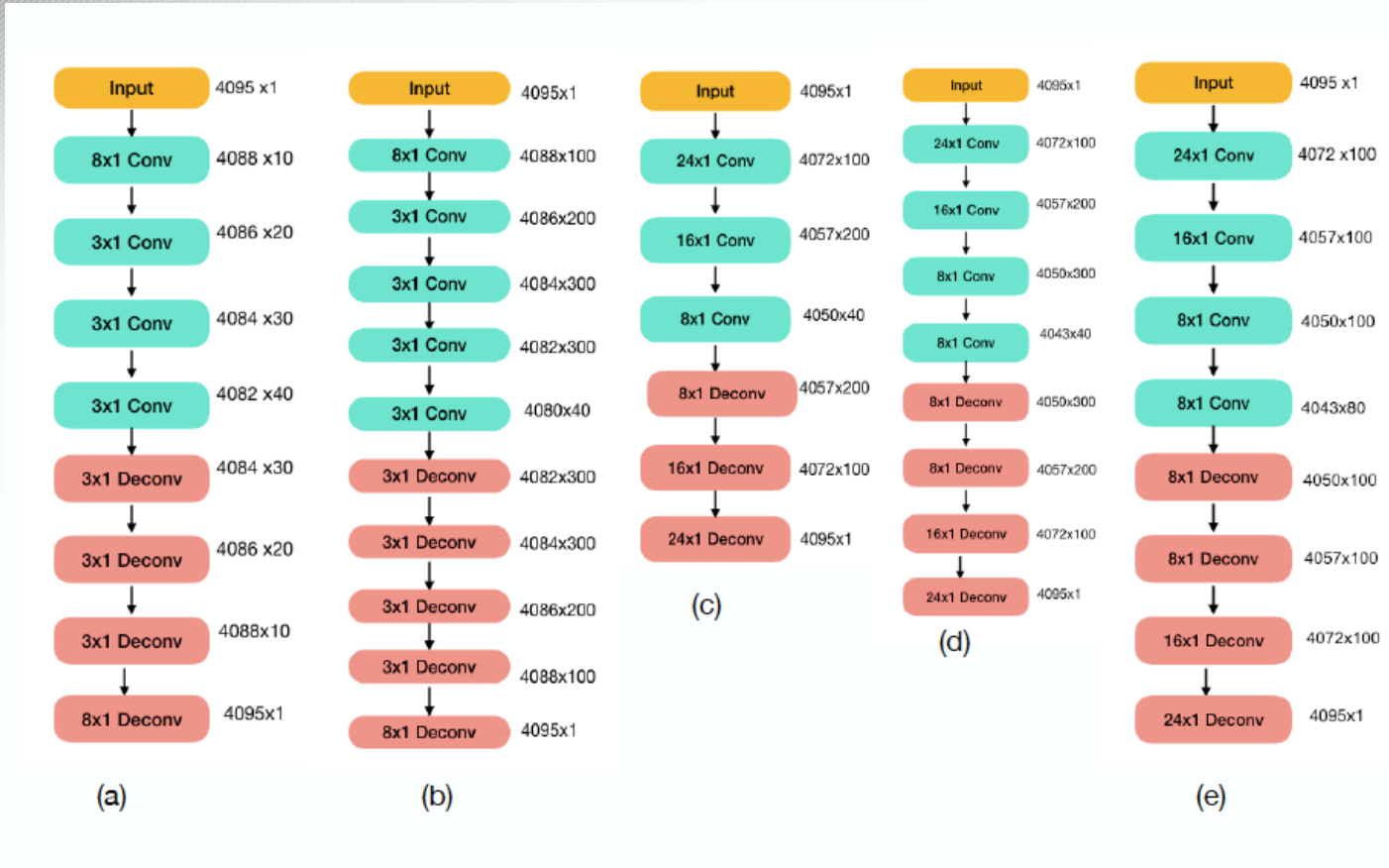
(a)

Architecture of deep neural networks (DNNs) for soft decoding of noisy file segments.

p is the BSC error probability.

DNN architecture for **HTML files for trained for $p = 0.8\%, 1.2\%, 1.6\%$.**

DNN Architecture



Architectures of deep neural networks (DNNs) for soft decoding of noisy file segments. ***p is the BSC error probability.***

(a)**HTML files for $p = 0.8\%, 1.2\%, 1.6\%$.**

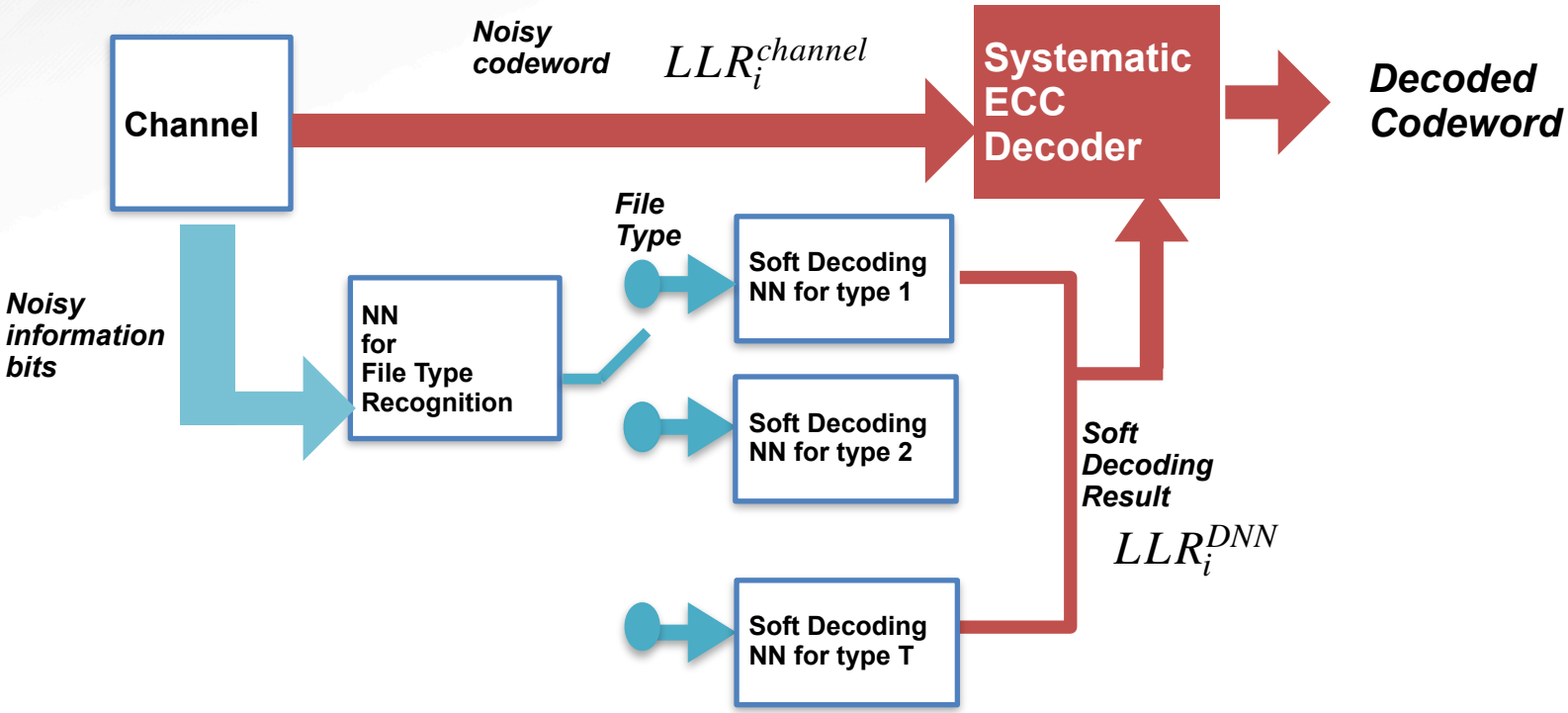
(b)**LaTex files for $p = 0.8\%, 1.2\%, 1.6\%$.**

(c)**PDF and JPEG files when $p = 0.8\%$**

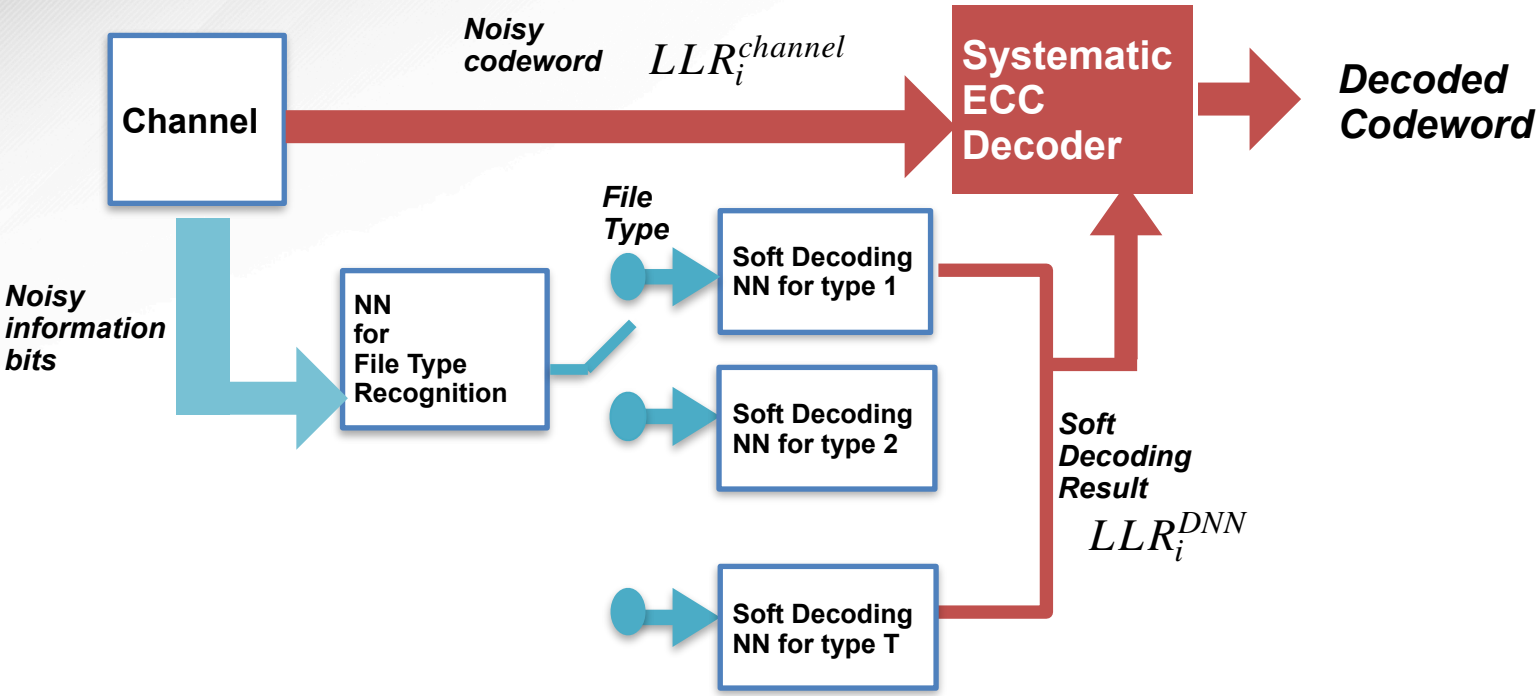
(d)**PDF files when $p = 1.2\%, 1.6\%$**

(e)**JPEG files when $p = 1.2\%, 1.6\%$.**

Combine DNN Soft Decoding & LDPC Decoding



Combine DNN Soft Decoding & LDPC Decoding

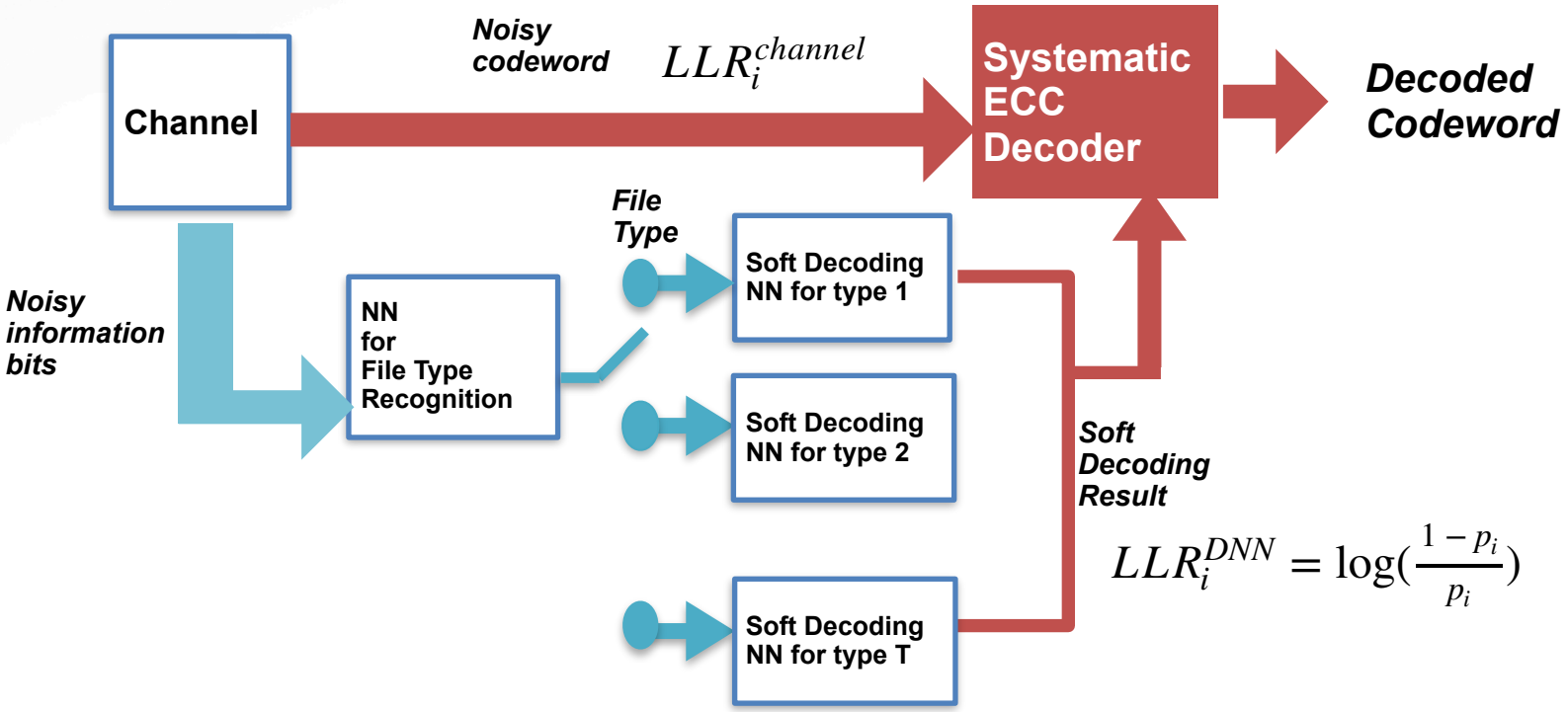


- The **output of the soft decoding DNN** is $(p_1, p_2, \ldots, p_k)$, where p_i is the estimated probability that the i-th information bit is 1.

- The LLR available from the DNN for the i-th information bit: $LLR_i^{DNN} = \log(\frac{1-p_i}{p_i})$

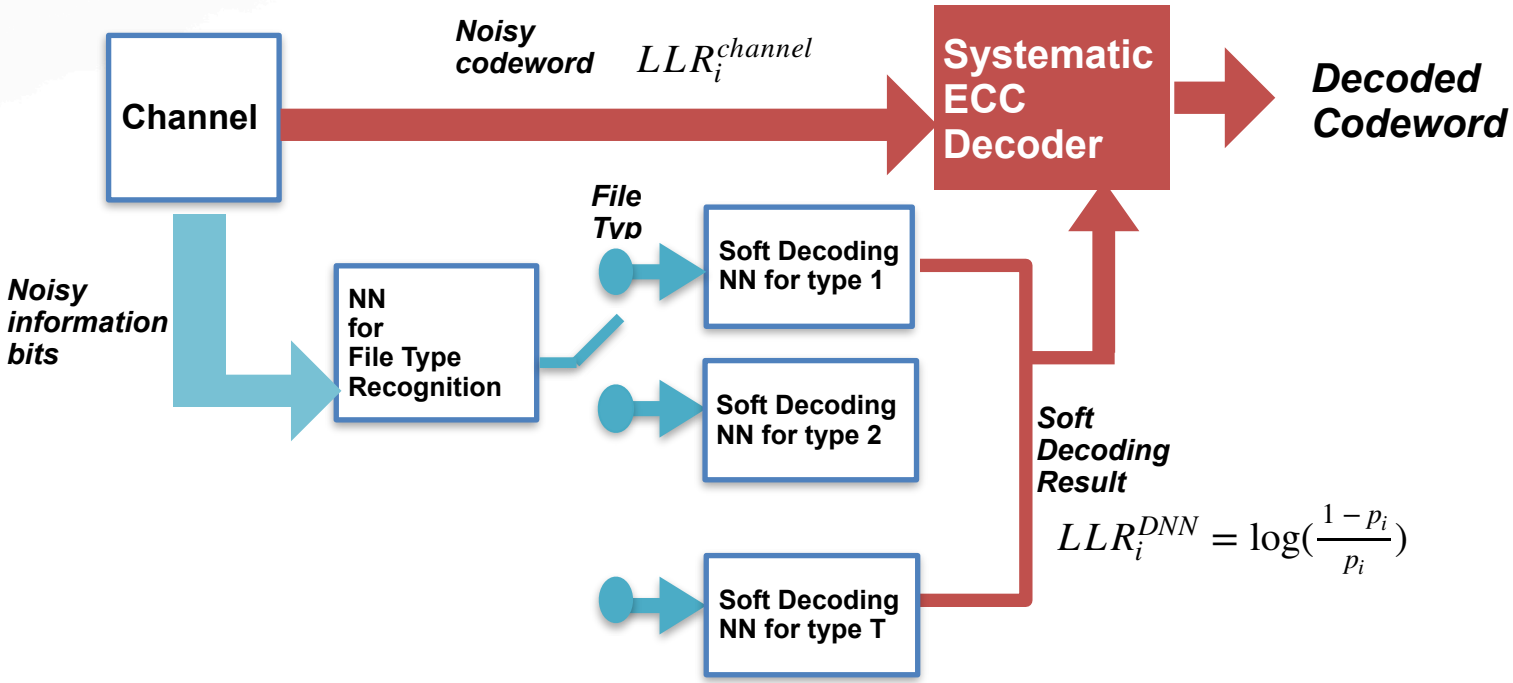
Combine DNN Soft Decoding & LDPC Decoding

- For $i = 1,2,\cdots,n$, let the **LLR for the *i*-th codeword bit** (both information/parity bits) **derived for the BSC** be : $LLR_i^{channel}$



Combine DNN Soft Decoding & LDPC Decoding

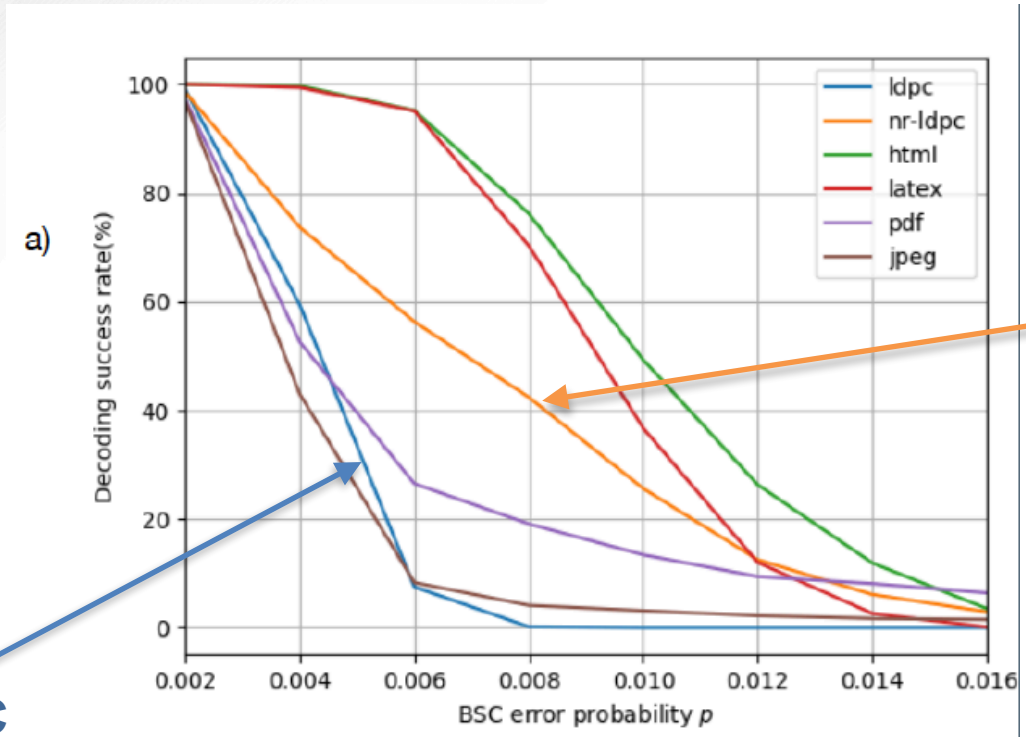
- Information bits are bits with index $1 \le i \le k$. $LLR_i^{int} = LLR_i^{channel} + LLR_i^{DNN}$
- Parity bits are the bits with index $k+1 \le i \le n$ $LLR_i^{int} = LLR_i^{channel}$



Combine DNN Soft Decoding & LDPC Decoding

- We adopt a **robust** scheme here. All DNNs have been **trained with a constant BER** p_{DNN} .
- However, they are **used for a wide range of BERs** p for the BSC channel.
- **For example**, the DNNs may be trained just for p_{DNN}= 1.2%, but are used for any BER p from 0.2% to 1.6%.
- The threshold BER of the LDPC code we use is **0.2%.**

Combine DNN Soft Decoding & LDPC Decoding

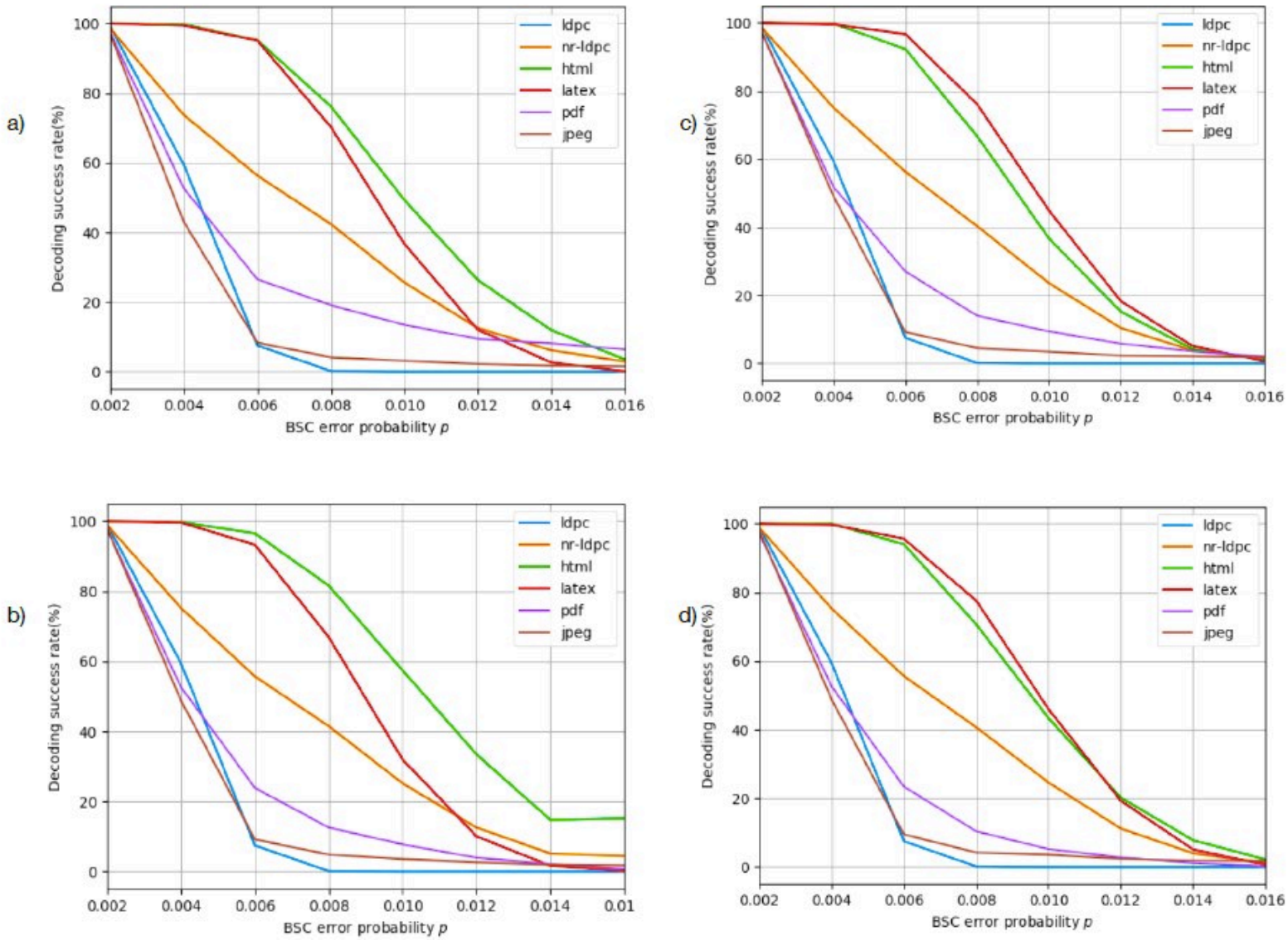


We consider BERs **greater than threshold.**

Decoding success rate vs bit error rate for p_{DNN} = **1.0%**

Results

Decoding success rate vs bit error rate for (a) $p_{DNN} = 1.0\%$, (b) $p_{DNN} = 1.2\%$, (c) $p_{DNN} = 1.4\%$; (d) $p_{DNN} = 1.6\%$



Conclusion & Future Work

- Better and more practical error correction using natural redundancy.

- There is no need to know the file types in advance.

Future Work

This scheme can be extended to

A. *More file types*.

B. *More DNN architectures*.

C. *Iterative combination of NR and LDPC decoders*.

THANK YOU