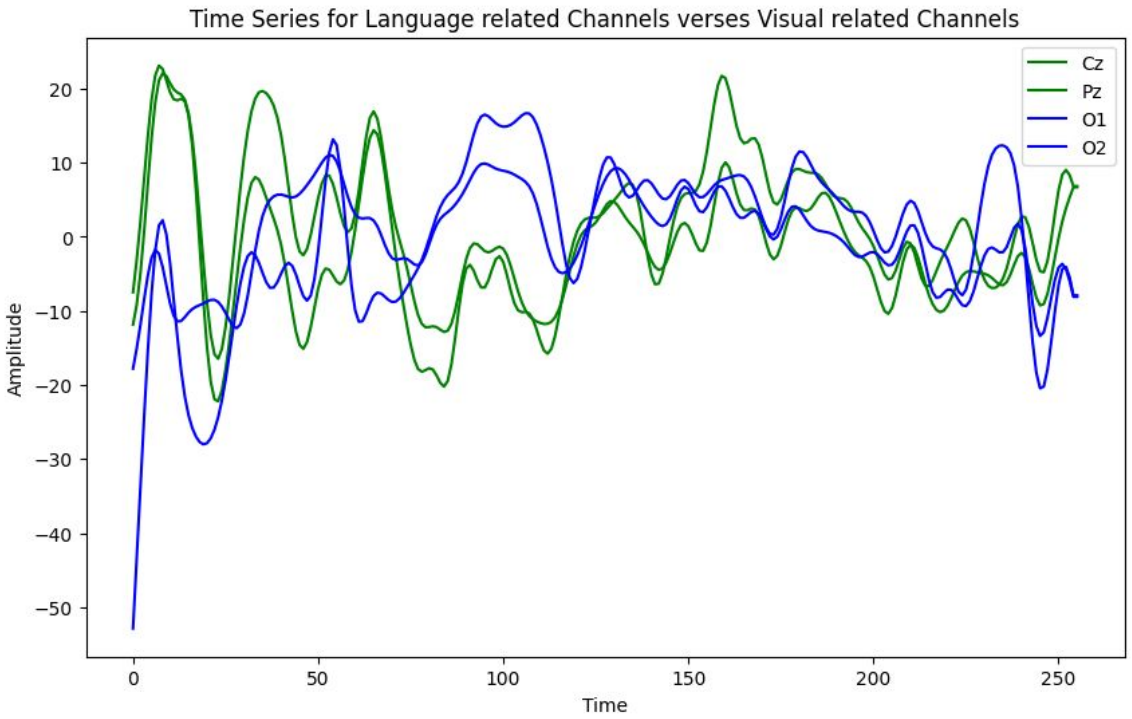
Bilingual Detection

Data - Participants Category

Translated	Bilingual	N400 effect exists
	Not Bilingual	N400 effect doesn't exists
Unrelated	Bilingual	N400 effect doesn't exists
	Not Bilingual	N400 effect doesn't exists

Data – EEG Channel Selection



- **Green line**: EEG channel associated with language processing
- **Blue line**: EEG channel not associated with language

The EEG signal from language-related channels shows a clear distinction.
Cz channel is chosen as the input data.

Data - Cleaning

Data Information

Word	Prime	Target	Relationship	Participant
neck	spanish	english	translation	26
chair	spanish	english	translation	30
neighbor	spanish	spanish	translation	36
city	english	spanish	translation	4
lawyer	spanish	english	translation	37

EEG

	0	1	2	3	4	...	246	247	248	249	250	251	252	253	254	255
neck	10.948781	11.949781	12.760781	13.441781	14.122781	...	3.166781	6.033781	7.861781	8.295781	7.320781	5.157781	2.125781	-1.461219	-5.301219	-5.301219
chair	1.806137	-1.544863	-3.813863	-5.139863	-6.124863	...	-7.915863	-8.830863	-9.105863	-8.831863	-8.214863	-7.414863	-6.466863	-5.345863	-4.092863	-4.092863
neighbor	-6.105105	-5.350105	-4.471105	-3.177105	-1.428105	...	13.504895	18.143895	25.016895	32.111895	35.276895	28.744895	7.447895	-29.489105	-76.644105	-76.644105
city	14.064312	-20.806313	-25.663312	-28.102313	-28.058312	...	0.576688	-1.767313	-4.140313	-5.845313	-6.616313	-6.651313	-6.400313	-6.247312	-6.310313	-6.310313
lawyer	12.118641	11.404641	9.846641	7.626641	5.313641	...	-13.093359	-13.245359	-11.933359	-9.454359	-6.469359	-3.775359	-1.973359	-1.207359	-1.107359	-1.107359

```
spanish_df = final_df[(final_df['Prime'] == 'spanish') | (final_df['Target'] == 'spanish')]
spanish_df = spanish_df[spanish_df['Relationship'] == 'translation']
```

Y label

	0
0	N
1	N
2	N
3	N
4	N
...	...

```
spanish_participants = [str(p) for p in info.loc[info['spanish'] == 1, 'participant'].tolist()]
'Y' if participant in spanish_participants else 'N'
```

```
Bilingual distribution: {'N': 2483, 'Y': 794}
```

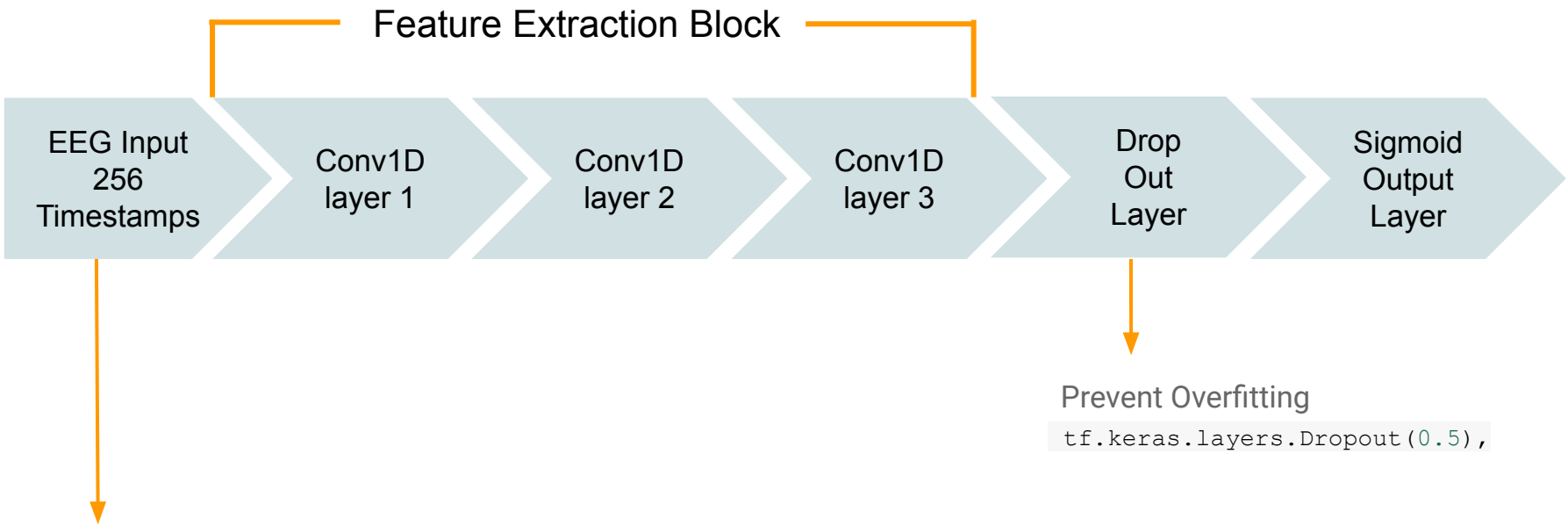
Model Selection

- **Time Series Clustering**
  - High data variability led to poor performance.
  - Could be effective with larger windows and reduced noise.

- **RNN**
  - Best for sequential dependencies, not local pattern recognition.
  - Less suited for detecting N400 effect patterns.

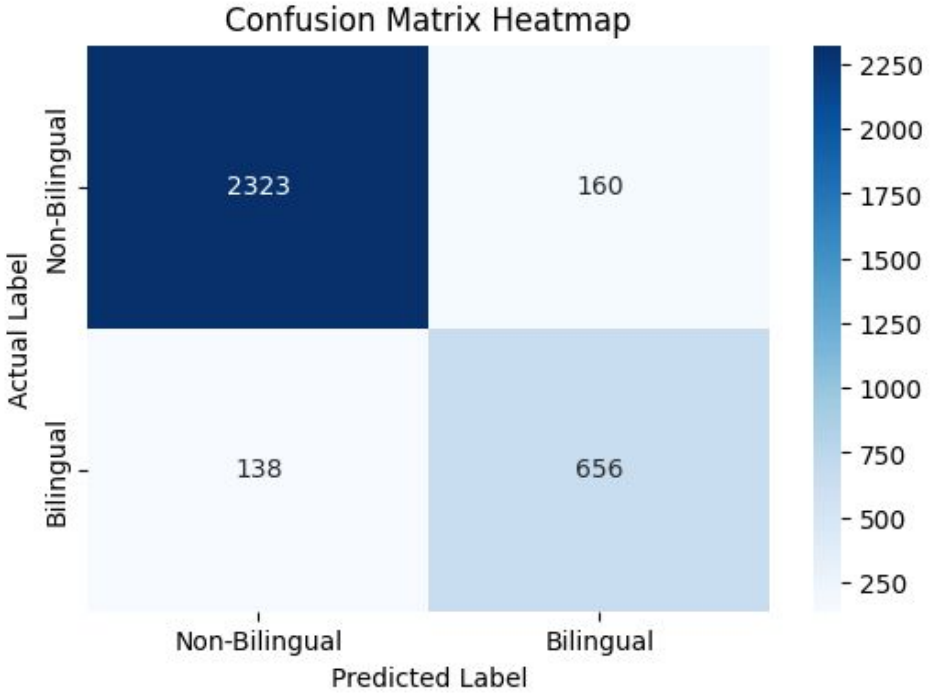
- **1D CNN**
  - Ideal for capturing localized patterns.
  - Well-suited for identifying N400-related patterns in EEG data.
  - Achieved the best performance among models tested.

Model Structure




```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0, stratify=y)
class_weights = class_weight.compute_class_weight('balanced', classes=np.unique(y_train), y=y_train)
```

Prediction Result - Spanish



Accuracy: 90.9%

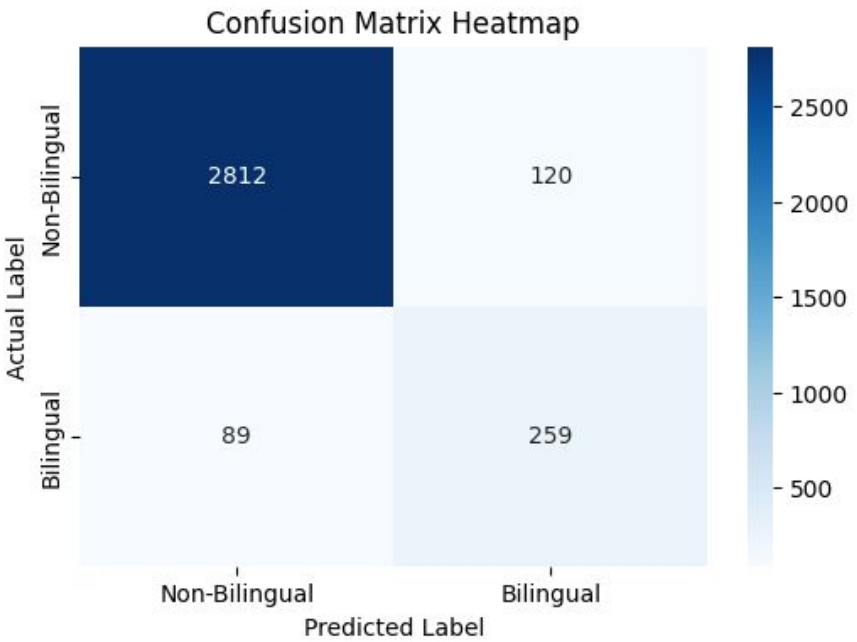
```
new_y_pred = (model.predict(new_X) > 0.5).astype("int32")
```

```
Random Sample Index: 162
Model Prediction: Non-Bilingual
Actual Label: Non-Bilingual
```

```
Random Sample Index: 9
Model Prediction: Bilingual
Actual Label: Bilingual
```

Prediction Result - German

```
german_df = final_df[(final_df['Prime'] == 'german') | (final_df['Target'] == 'german')]
german_df = german_df[german_df['Relationship'] == 'translation']
```

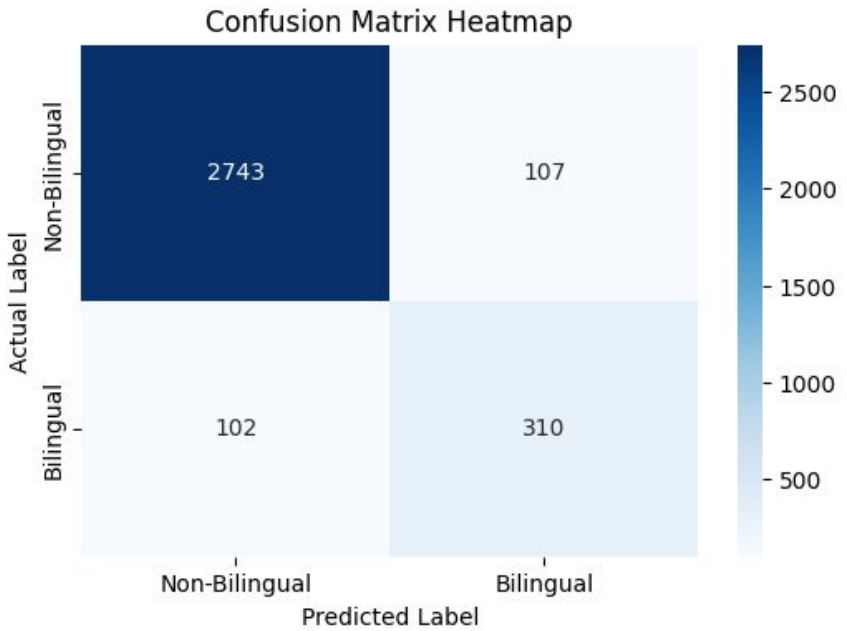


Accuracy : 93.7%

```
new_y_pred =
(model_german.predict(new_X) >
0.65).astype("int32")
```

Prediction Result - French

```
french_df = final_df[(final_df['Prime'] == 'french') | (final_df['Target'] == 'french')]
french_df = french_df[french_df['Relationship'] == 'translation']
```



Accuracy : 93.6%

```
new_y_pred =
(model_french.predict(new_X) >
0.5).astype("int32")
```

Thank You