Experiments on Rebate Data

We have rebate data where we have cleaned description of the product as an input and 134 vendor categories.

Firstly, we picked top 10 vendor categories where we have data with highest count occurrences. We sampled 15000 points from each class and thus from a dataset of 150000 points we used 89865 points for training and 60135 points were used during validation/testing time.

Thus, we aim to design a 10-class classification neural network-based model which takes as an input the text which is a product description and classifies it to 10 categories.

Initially we use the RoBERTa-base tokenizer with maximum token length of 50. Then we use the RoBERTa base model to get the output weights from it and added on top of it two fully connected layers. Finally from the last layer we attach two parallel heads (of dimension equal to number of class) that gave us respectively the mean and log-variance of the logits. These two heads give us the estimate of mean and standard deviation values of the target distribution, through which we can sample some values across which uncertainty values can be estimated.

For training we used the batch size of 512 and a dropout probability of 0.2.

Following are the three different architecture/ways to calculate uncertainty.

1. **BASIC NEURAL NETWORK.**

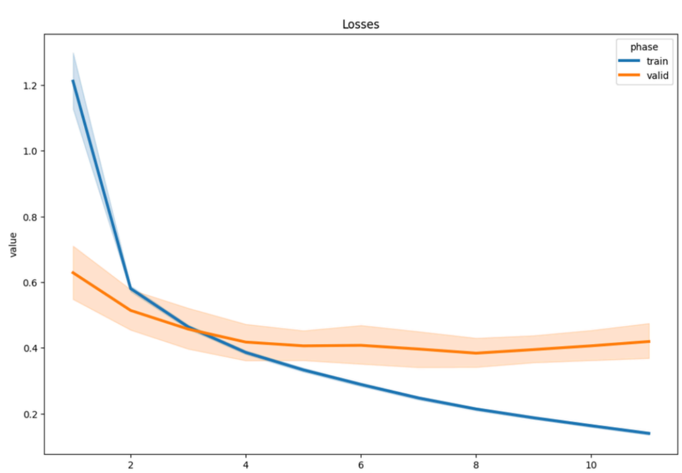
Here we used the neural network architecture described above where during training we learn the point estimates of the fully connected layer weights.

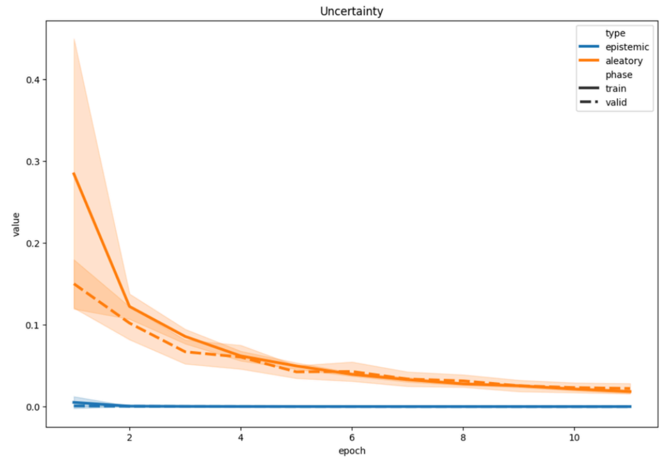
For calculating data uncertainty, we directly use the final output of the head which gives us log-variance values and the mean values of the variance in the classes will give us the data uncertainty (aleatoric uncertainty).

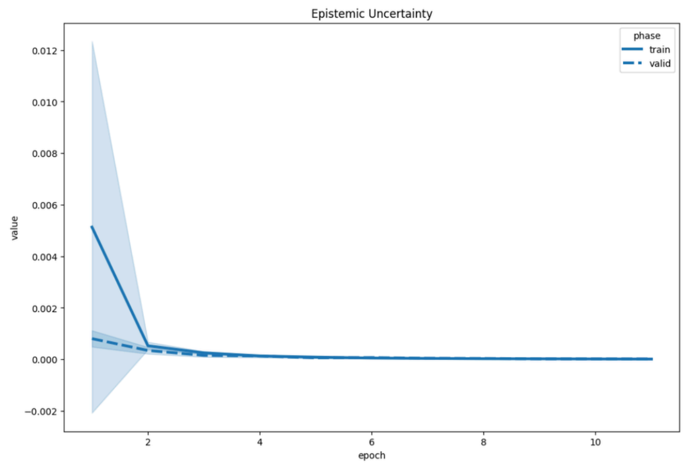
For calculating model (epistemic) uncertainty, we draw **n_samp** samples from the multivariate normal distribution which was constructed using the mean_logits and variance_logits values of the final head. The value of the variance across the samples will give us the estimate of the model uncertainty. The loss value of the network is the average of the cross entropy loss across each sample prediction and target value.

On increasing the value of n_samp to 20, we don't observe much change in the values range of data and model uncertainty.

Following are the losses and uncertainty values that we got during training of the neural network :







Following is the classification report for this model predictions:

class	precision	recall	f1-score	support
0	0.712401455	0.768941253	0.739592351	6111
1	0.97170288	0.963748747	0.967709469	5986
2	0.985528942	0.984709988	0.985119295	6017
3	0.870093293	0.830337645	0.849750731	5953
4	0.972250084	0.95815486	0.965151012	6070
5	0.913057743	0.925661068	0.919316211	6013
6	0.86089511	0.834086724	0.847278912	5973
7	0.926953381	0.94022644	0.933542734	6006
8	0.849050261	0.847625441	0.848337252	5959
9	0.949025487	0.94212006	0.945560166	6047
accuracy	0.899559325	0.899559325	0.899559325	0.899559325
macro avg	0.901095864	0.899561223	0.900135813	60135
weighted avg	0.900958212	0.899559325	0.900064859	60135

Following are the percentile values corresponding to the data uncertainty and the model uncertainty for all the validation datapoints:

Percentile	Data Uncertainty Value	Model Uncertainty Value
0.1	4.49E-03	2.46E-07
1	4.89E-03	3.46E-07
5	6.63E-03	6.34E-07
10	7.76E-03	1.01E-06
20	1.27E-02	2.31E-06
80	3.24E-02	2.20E-05
90	4.41E-02	3.89E-05
95	5.68E-02	7.01E-05
99	8.71E-02	2.12E-04
99.7	1.12E-01	3.91E-04
99.9	1.34E-01	6.09E-04
99.99	1.74E-01	1.20E-03

Here is the decile report giving information about number of correct predictions made for each decile, where deciling was done independently based on prediction probability, data uncertainty and model uncertainty respectively:

DECILE	PRED PROBABILITY (Correct Predictions)	DATA UNCERTAINTY (Correct Predictions)	MODEL UNCERTAINTY (Correct Predictions)
0	5993	5969	5975
1	6007	5847	5806
2	5993	5679	5731
3	5985	5665	5707
4	5955	5657	5631
5	5902	5732	5473
6	5797	5806	5472
7	5381	5414	5470
8	4330	4707	5042
9	2752	3619	3788

The class specific distribution of correct predictions made while deciling independently based on prediction probability, data uncertainty and model uncertainty can be found here: NN_DecileReport_Uncertainty_Values

2. **Dropout During Inference**

It uses the same neural network model as trained above to get an uncertainty estimate based on the predicted mean and log-variance values of the final logit layer. It's just that during inference as well we drop the neurons with a predicted probability of 0.2. Thus, each time the same input is passed to the network, different neurons are dropped and thus we get multiple different prediction values for each time. The variance of the mean predicted logit across the samples will give an estimate of model (epistemic) uncertainty and the mean values of the log-variance head will give an estimate of data (aleatoric) uncertainty.

Following is the classification report for this model predictions:

	precision	**recall**	**f1-score**	**support**
0	0.651147007	0.784977909	0.711826681	6111
1	0.975102319	0.955228867	0.965063291	5986
2	0.987895091	0.976566395	0.982198078	6017
3	0.866800658	0.796909121	0.830386837	5953
4	0.974754321	0.947775947	0.961075844	6070
5	0.920468035	0.902710793	0.911502939	6013
6	0.816733068	0.82370668	0.820205051	5973
7	0.929824561	0.926573427	0.928196147	6006
8	0.851397998	0.82782346	0.839445248	5959
9	0.951028737	0.924921449	0.937793427	6047
accuracy	0.88678806	0.88678806	0.88678806	0.88678806
macro avg	0.892515179	0.886719405	0.888769354	60135
weighted avg	0.892310773	0.88678806	0.888692772	60135

Following are the percentile values corresponding to the data uncertainty and the model uncertainty for all the validation datapoints:

Percentile	**Data Uncertainty Value**	**Model Uncertainty Value**
0.1	5.84E-03	2.99E-03
1	7.05E-03	4.00E-03
5	9.60E-03	5.07E-03
10	1.19E-02	5.68E-03
20	1.86E-02	6.50E-03
80	4.50E-02	1.10E-02
90	5.76E-02	1.33E-02
95	7.00E-02	1.61E-02
99	9.68E-02	2.41E-02
99.7	1.16E-01	3.07E-02
99.9	1.33E-01	3.60E-02

99.99	1.65E-01	4.93E-02

Here is the decile report giving information about number of correct predictions made for each decile, where deciling was done independently based on prediction probability, data uncertainty and model uncertainty respectively:

DECILE	PREDICTED PROBABILITY (Correct Predictions)	DATA UNCERTAINTY (Correct Predictions)	MODEL UNCERTAINTY (Correct Predictions)
0	6001	5988	5335
1	5990	5883	5596
2	5992	5728	5643
3	5969	5687	5629
4	5939	5699	5602
5	5900	5660	5602
6	5729	5520	5546
7	5173	5080	5455
8	3978	4514	5117
9	2656	3568	3802

The class specific distribution of correct predictions made while deciling independently based on prediction probability, data uncertainty and model uncertainty can be found here: DropoutExpt_Full_DecileReport.xlsx

3. Bayesian Neural Network (samples drawn from target distribution for uncertainty calculation)

Here uncertainty estimation happens while training a Bayesian Neural Network (BNN). From the regular Neural Network, BNN differs in the sense that instead of learning point estimates of weights of the Neural Networks, we learn the normal distribution parameters (μ and σ) for weights of the neural network. Thus, each time when the input is passed to the BNN, the weight values are sampled from the corresponding distribution, and we get an estimate of mean value of logits and log-variance of logits as final output of the layer. The uncertainty values can be calculated by two ways:

Type-1: Like we did in the normal Neural Network technique, we sample n_samp=10 from the target distribution and the variance across the mean predicted values of the logits will give us the estimate of model uncertainty and the mean values of the variance calculated from the logit log-variance values will give us an estimate of the data uncertainty.

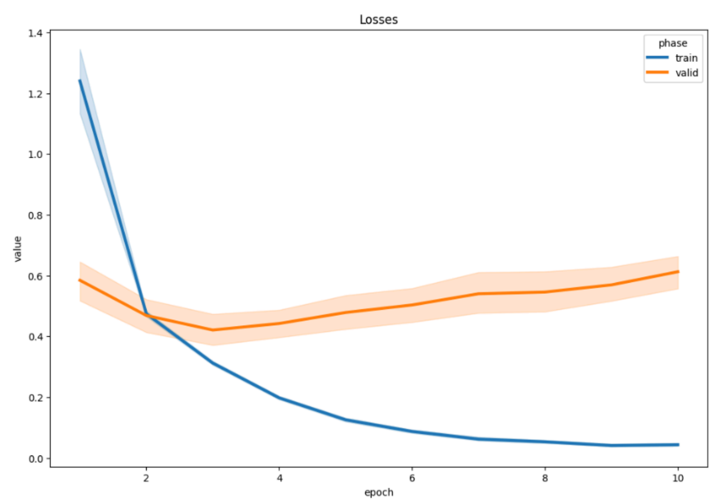
Type-2: Here the same input is passed **n_samp**=10 times and as each time the weight values are sampled from their corresponding weight distribution, we get multiple different predicted values of mean and log-variance values of logits. These values are used for estimating epistemic and aleatory uncertainty.

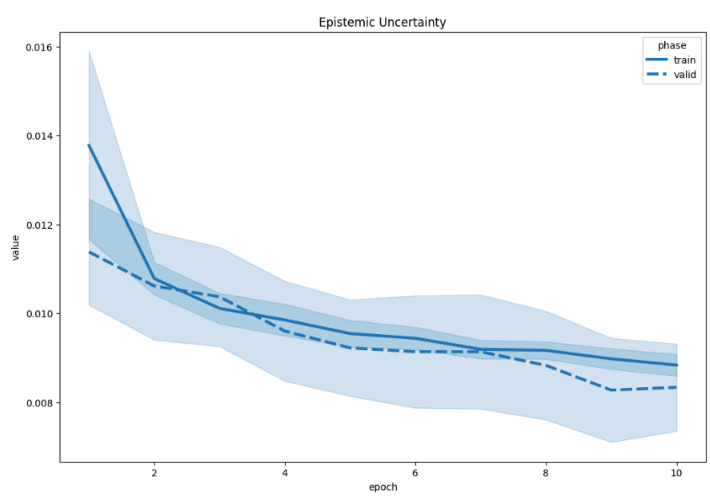
The loss value of the network for both the types are the average of the cross-entropy loss across each sample prediction and target value.

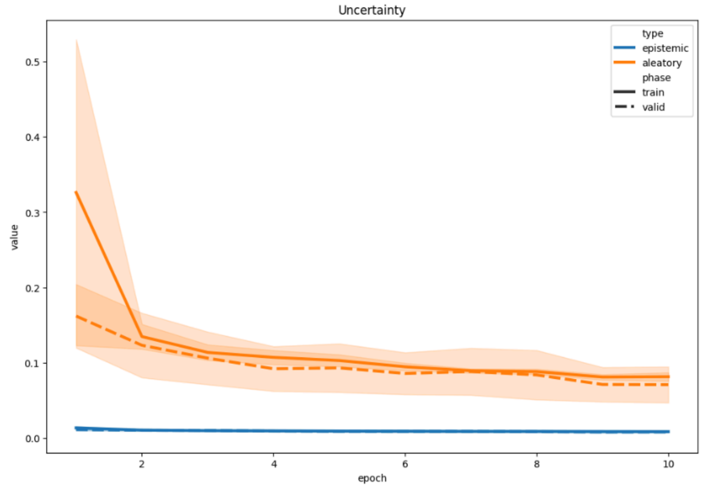
On observing the total correct predictions for each decile ranges for the two types, we observe that for the type-2, as we go to bottom decile values (higher uncertain results), the number of correct predictions increases continuously which seems to be counter-intuitive. So we used the results of type-1 as our estimates of uncertainty.

Decile Based on Epistemic (Model) Uncertainty	Type-1 Correct Predictions	Type-2 Correct Predictions
0	5935	4369
1	5730	5021
2	5607	5304
3	5587	5450
4	5589	5550
5	5543	5625
6	5435	5696
7	5282	5742
8	5037	5739
9	4417	5811

Following are the losses and uncertainty values that we got during training of the BNN:







Following is the classification report for this model predictions:

	precision	recall	f1-score	support
0	0.712395437	0.766486663	0.738451837	6111
1	0.973666442	0.963581691	0.968597817	5986
2	0.973834832	0.989695862	0.981701286	6017
3	0.847155014	0.852847304	0.849991629	5953
4	0.963182117	0.965403624	0.964291591	6070
5	0.911413132	0.925661068	0.918481848	6013
6	0.887017925	0.820190859	0.852296451	5973
7	0.93029402	0.937728938	0.933996683	6006
8	0.870427881	0.84661856	0.858358145	5959
9	0.951710262	0.938647263	0.945133628	6047
accuracy	0.900673485	0.900673485	0.900673485	0.900673485
macro avg	0.902109706	0.900686183	0.901130092	60135
weighted avg	0.901949149	0.900673485	0.901042884	60135

Following are the percentile values corresponding to the data uncertainty and the model uncertainty for all the validation datapoints:

Percentile	Data Uncertainty Value	Model Uncertainty Value
0.1	4.08E-03	3.05E-07
1	5.24E-03	6.12E-07
5	7.15E-03	1.43E-06
10	9.24E-03	2.71E-06
20	1.57E-02	7.44E-06
80	4.55E-02	8.39E-05

90	6.22E-02	1.54E-04
95	8.03E-02	2.61E-04
99	1.26E-01	7.23E-04
99.7	1.62E-01	1.32E-03
99.9	1.96E-01	2.26E-03
99.99	2.86E-01	4.53E-03

Here is the decile report giving information about number of correct predictions made for each decile, where deciling was done independently based on prediction probability, data uncertainty and model uncertainty respectively:

DECILE	PREDICTED PROBABILITY (Correct Predictions)	DATA UNCERTAINTY (Correct Predictions)	MODEL UNCERTAINTY (Correct Predictions)
0	5994	5980	5935
1	6000	5840	5730
2	5990	5800	5607
3	5986	5766	5587
4	5942	5742	5589
5	5883	5695	5543
6	5785	5502	5435
7	5339	5215	5282
8	4412	4693	5037
9	2831	3929	4417

The uncertainty values for the validation/test data, their analysis based on the decile values (created based on predicted probability of the model, data uncertainty, model uncertainty) and also the classification report and confusion matrix can be found here: BNN_15000_10samp_target_DecileReport_full.xlsx.

EXPERIMENTS ON DATASET OF DIFFERENT SIZES

We further used the same dataset with 10 classes but of different total size like 75000, 112500, 130000 and 150000 for uncertainty estimation during training and validation using the three techniques as described above. We observed that the accuracy of the model decreases every time as we moved towards smaller dataset size. But there is not a monotonic trend observed for the effect of dataset size on uncertainty values although for majority of time we saw uncertainty values decreases as the dataset size increases.

Mean **Model Uncertainty** for different dataset sizes are below:

MEAN MODEL UNCERTAINTY	Normal NN	With Dropout in NN	BNN-type1 (n_samp=10)
(Dataset Size) 75000	7.36E-05	1.29E-02	1.70E-04

112500	1.49E-05	1.01E-02	9.82E-05
130000	1.67E-05	9.85E-03	6.72E-05
150000	1.96E-05	9.20E-03	7.14E-05

Following are the mean values of model uncertainty for the experiments with dataset size 75000 using BNN of different types and cases:

BNN Type1(n_samp=10)	BNN Type1(n_samp =100)	BNN Type2(n_samp =10)	BNN Type2 (n_samp =100)	BNN-Type2 (n_samp=10(during training), n_samp=100 (during inference))
1.70E-04	2.00E-04	8.30E-03	8.64E-03	8.28E-03

Here we observed that the values of epistemic uncertainty values for BNN-type1 is larger than that of BNN-type2. Thus, these are the two different approaches that can be used to calculate the uncertainty and the preference is given to the method which shows more consistency with model and data assumptions (more complex model or less dataset size should give high uncertain values). For our case we saw BNN-type1 to be more consistent after observing the number of corrections for each decile range.

Mean **data uncertainty** for different dataset sizes are below:

MEAN DATA UNCERTAINTY	Normal NN	Dropout_NN	BNN type1(n_samp=10)
75000	4.21E-02	5.36E-02	5.56E-02
112500	2.29E-02	3.07E-02	4.10E-02
130000	2.22E-02	3.01E-02	3.38E-02
150000	2.54E-02	3.38E-02	3.32E-02

Following are the mean values of data uncertainty for the experiments with dataset size 75000 using BNN of different types and cases:

BNN Type1(n_samp=10)	BNN Type1(n_samp =100)	BNN Type2(n_samp =10)	BNN Type2 (n_samp =100)	BNN-Type2 (n_samp=10(during training), n_samp=100 (during inference))
5.56E-02	6.22E-02	5.00E-02	6.24E-02	5.01E-02

Following is the test accuracy for each of the methods and for the datasets of different sizes:

ACCURACY(%)	Normal NN	With Dropout in NN	BNN type1(n_samp=10)
75000	87.897	86.336	87.494
112500	89.38	87.943	89.1245
130000	89.633	88.133	89.549
150000	89.956	88.679	90.067

Following is the test accuracy for the experiments with dataset size 75000 using BNN of different types and cases:

BNN Type1(n_samp=10)	BNN Type1(n_samp =100)	BNN Type2(n_samp =10)	BNN Type2 (n_samp =100)	BNN-Type2 (n_samp=10(during training), n_samp=100 (during inference))
87.494	87.5478	87.495	87.611	87.538

The following table gives information about the range of data and model uncertainty values along with their mean values for each network type and dataset with different sizes

Network Type	Size	n_samp	Mean Data Uncertainty	Mean Model Uncertainty	Min Data Uncertainty	Max Data Uncertainty	Min Model Uncertainty	Max Model Uncertainty
Normal NN	75000	10	4.21E-02	7.36E-05	1.16E-02	2.17E-01	2.73E-06	1.96E-03
Dropout	75000	10	5.36E-02	1.29E-02	1.40E-02	2.00E-01	2.90E-03	6.45E-02
BayesianNN-Type1	75000	10	5.56E-02	1.70E-04	8.00E-04	3.72E-01	6.92E-07	7.10E-03
BayesianNN-Type1	75000	100	6.22E-02	2.00E-04	1.00E-02	3.27E-01	2.67E-06	5.70E-03
BayesianNN-Type2	75000	10	5.00E-02	8.30E-03	1.44E-02	2.23E-01	1.00E-03	1.78E-02
BayesianNN-Type2	75000	100	6.24E-02	8.64E-03	1.74E-02	2.60E-01	2.33E-03	1.40E-02
BayesianNN(Type-2, model trained using 10 samples)	75000	100	5.01E-02	8.28E-03	1.54E-02	2.17E-01	1.70E-03	1.35E-02
Normal NN	112500	10	2.29E-02	1.49E-05	3.60E-03	2.16E-01	1.23E-07	2.72E-03
Dropout	112500	10	3.07E-02	1.01E-02	3.69E-03	1.89E-01	1.74E-03	7.97E-02
BayesianNN-Type1(samples from target dist.)	112500	10	4.10E-02	9.82E-05				
Normal NN	130000	10	2.22E-02	1.67E-05	3.00E-03	2.53E-01	1.03E-07	2.49E-03
Dropout	130000	10	3.01E-02	9.85E-03	3.00E-03	1.91E-01	1.95E-03	6.43E-02
BayesianNN-Type1(samples from target dist.)	130000	10	3.38E-02	6.72E-05	3.35E-03	2.74E-01	2.89E-07	7.24E-03

Normal NN	150000	10	2.54E-02	1.96E-05	4.25E-03	2.11E-01	1.66E-07	1.70E-03
Normal NN	150000	20	2.54E-02	1.96E-05	4.26E-03	2.11E-01	2.11E-07	1.70E-03
Dropout	150000	10	3.38E-02	9.20E-03	4.90E-03	1.96E-01	1.79E-03	5.72E-02
BayesianNN-Type1(samples from target dist.)	150000	10	3.32E-02	7.14E-05	2.89E-03	3.93E-01	8.93E-08	9.00E-03
BayesianNN-Type2(same i/p multiple times)	150000	10	3.02E-02	7.00E-03	4.95E-03	2.65E-01	1.00E-03	1.68E-02

The percentile values and the decile report formed independently based on predicted probability, data uncertainty, model uncertainty for each of the cases can be found here:
different_size_data_inference_uncertainty_results.xlsx