Compare DA

$$Compare \begin{pmatrix} name[Atlas\ 89] & name[Eurus\ 93] \\ isForBusiness[yes] & isForBusiness[no] \\ weight[2.3] & weight[1.12] \end{pmatrix}$$

InformCount DA

$$InformCount \begin{pmatrix} count[40] \\ family[don't\ care] \\ batteryRating[excellent] \end{pmatrix}$$

	Compare Input	InformCount Input
$x_1 =$	compare	inform_count
$x_2 =$	name	count
$x_3 =$	is_for_biz_yes	family_dont_care
$x_4 =$	weight	batteryrating
$x_5 =$	name	—
$x_6 =$	is_for_biz_no	—
$x_7 =$	weight	—

Compare Utterance

weighing $WEIGHT_1$ kg for business computing the $NAME_1$ is compared to the $NAME_2$ which weighs $WEIGHT_2$ kg and is not for business computing . which one do you like

InformCount Utterance

there are COUNT laptop -s with an BATTERYRATING battery rating if you do not care about the product family.

Figure 2: Example MR, input representations, and utterances for the Laptops and TVs datasets.

Inform DA

$$Inform \begin{pmatrix} name[The\ Mill] \\ near[Avalon] \\ food[Italian] \end{pmatrix}$$

	Input Lexicalized	Input Delexicalized
x_1	= eat_type_n/a	eat_type_n/a
x_2	= near_avalon	near_present
x_3	= area_n/a	area_n/a
x_4	= fam._friend._n/a	fam._friend._n/a
x_5	= cust._rating_n/a	cust._rating_n/a
x_6	= price_range_n/a	price_range_n/a
x_7	= food_Italian	food_Italian
x_8	= name_the_mill	—

Lexicalized Utterance

the mill serves up italian food near avalon .

Delexicalized Utterance

NAME serves up italian food near NEAR .

Figure 3: Example MR, input representation, and utterance for the E2E dataset.

A Additional Preprocessing Details

MR attributes can be one of two types, dictionary attributes, e.g. *Name* or *CustomerRating*, where the value for the attribute comes from a closed set of valid slot fillers, and binary attributes, e.g. *familyFriendly* or *hasUsbPort*, which can have values *yes* or *no*. Additionally, on the Laptops and TVs datasets, attributes can also have a distinguished *don't care* value, e.g. the MR *InformCount*(*count[40], priceRange[don't care]*) could be realized as "There are 40 laptops available if you do not care about the price range."

When using the delexicalied model, we do not represent the *Name* attribute in the input x, since every valid E2E MR contains this attribute, i.e. the model learns on its own to always generate the *NAME* token to be replaced at test time.

Certain attributes were inconsistently realized in the reference utterances. E.g., utterances would frequently refer to locations as restaurants even when the *eatType* attribute was not present so we ammended the MR to have the attribute *eatType[restaurant]* in those cases. Additionally, *priceRange[cheap]* and *priceRange[less than £20]* would be interchanged; *cheap* utterances that mentioned numerical amounts were remapped to *less than £20* and visa-versa. Similar corrections were made for the *customerRating* attribute. Note, these changes were only done on the training and validation set. We do not modify the test set at all.

B Model Input Representation

We show an example input sequence x for the E2E dataset in Figure 3 and for the Laptops and TVs datasets in Figure 2.

C CNN Classifier Details

We use a separate CNN classifier for each attribute to predict the corresponding value (or *n/a*) from an utterance y. We first look up the tokens in y in an embedding matrix E to obtain a matrix $W \in \mathbb{R}^{N \times D}$ where $D = 50$ is the embedding dimension.

We then apply a series of unigram, bigram, and trigram convolutional filters each with 50 output features. After concatenating and max-pooling over the sequence dimension, and applying a ReLU activation, we obtain a hidden layer in $\mathbb{R}^{150}$. We then apply another fully-connected layer with ReLU activation which down projects the hidden layer to $\mathbf{R}^{50}$. Finaly we apply the final softmax layer to predict the class label.

During training we apply dropout (with drop rate 0.25) to the embedding layer, convolutional filter outputs, and hidden layers. We train for 30 epochs with gradient descent using a learning rate of 0.25 and weight decay penalty of 0.0001, using validation set F1 as our model selection criterion.

We treat the utterance as a

D Augmented Dataset Samples

Samples and their parsed MR for the E2E datsets are shown in Table 9, Table 10, and Table 11.

Inform(*eatType*[pub] *food*[Italian] *name*[NAME] *priceRange*[high])
NAME is a pub that serves italian food in the high price range .

Inform(*area*[city centre] *customerRating*[5 out of 5] *eatType*[coffee shop] *familyFriendly*[no] *food*[Indian] *name*[NAME])
the NAME is a coffee shop that serves indian food . it is located in the city centre and has a customer rating of 5 out of 5 . it is not kid friendly .

Inform(*customerRating*[3 out of 5] *eatType*[pub] *food*[English] *name*[NAME] *near*[NEAR])
the NAME is a pub that serves english food and is located near the NEAR . it has a customer rating of 3 out of 5 .

Inform(*area*[city centre] *customerRating*[3 out of 5] *eatType*[pub] *familyFriendly*[yes] *food*[Fast food] *name*[NAME] *priceRange*[less than £20])
the NAME is a pub in the city centre that serves fast food for less than £20 . it has a customer rating of 3 out of 5 and is family - friendly .

Inform(*customerRating*[high] *eatType*[coffee shop] *food*[French] *name*[NAME] *near*[NEAR] *priceRange*[cheap])
NAME is a cheap coffee shop near NEAR that serves french food . it has a high customer rating .

Inform(*area*[city centre] *customerRating*[5 out of 5] *eatType*[pub] *familyFriendly*[yes] *food*[Chinese] *name*[NAME] *priceRange*[cheap])
the NAME is a family - friendly pub in the city centre . it serves cheap chinese food and has a customer rating of 5 out of 5 .

Inform(*customerRating*[high] *familyFriendly*[yes] *food*[Indian] *name*[NAME] *priceRange*[high])
the NAME serves high priced indian food . it has a high customer rating and is child friendly .

Inform(*customerRating*[3 out of 5] *food*[Japanese] *name*[NAME] *near*[NEAR] *priceRange*[less than £20])
NAME serves japanese food for less than £20 . it is located near NEAR and has a customer rating of 3 out of 5 .

Table 9: Example E2E samples obtained using noise injection sampling with delex. p_0 and $q_{\mathfrak{R}}$ to parse the MR.

Inform(*customerRating*[3 out of 5] *familyFriendly*[no] *food*[English] *name*[NAME] *priceRange*[£20-25]) the NAME serves english food in the £20 - £25 price range . it is not kid friendly and has a customer rating of 3 out of 5
Inform(*area*[riverside] *customerRating*[3 out of 5] *eatType*[restaurant] *familyFriendly*[yes] *food*[Chinese] *name*[NAME] *near*[PRESENT] *priceRange*[high]) NAME is a family friendly restaurant serving chinese food in the riverside area near NEAR . it has a high price range and a customer rating of 3 out of 5
Inform(*area*[city centre] *eatType*[restaurant] *food*[English] *name*[NAME] *priceRange*[less than £20]) NAME is a restaurant providing english food in the less than £20 price range . it is located in the city centre
Inform(*customerRating*[3 out of 5] *food*[Italian] *name*[NAME] *near*[PRESENT] *priceRange*[£20-25]) NAME has a price range of £20 - 25 . it has a customer rating of 3 out of 5 and serves italian food . it is near the NEAR
Inform(*area*[riverside] *customerRating*[3 out of 5] *food*[Chinese] *name*[NAME] *priceRange*[moderate]) the NAME serves chinese food in the riverside area . it has a moderate price range and a customer rating of 3 out of 5
Inform(*area*[riverside] *customerRating*[low] *familyFriendly*[yes] *food*[English] *name*[NAME]) the NAME serves english food in the riverside area . it has a low customer rating and is kid friendly
Inform(*customerRating*[low] *eatType*[coffee shop] *familyFriendly*[yes] *food*[Japanese] *name*[NAME]) the NAME is a family friendly japanese coffee shop with a low customer rating
Inform(*customerRating*[1 out of 5] *familyFriendly*[no] *food*[Japanese] *name*[NAME] *near*[PRESENT]) NAME serves japanese food near NEAR . it is not kid friendly and has a customer rating of 1 out of 5

Table 10: Example E2E samples obtained using noise injection sampling with delex. p_0 and q_ϕ to parse the MR.

Inform(*area*[riverside] *customerRating*[5 out of 5] *eatType*[restaurant] *food*[French] *name*[The Eagle] *near*[The Sorrento] *priceRange*[less than £20])
the eagle is a french restaurant with a 5 out of 5 customer rating and a price range of less than £20 . it is located in the riverside area near the sorrento .

Inform(*area*[city centre] *eatType*[coffee shop] *familyFriendly*[no] *food*[English] *name*[The Wrestlers] *near*[Raja Indian Cuisine])
the wrestlers is a coffee shop that serves english food . it is located in the city centre near raja indian cuisine . it is not child friendly .

Inform(*area*[riverside] *name*[Taste of Cambridge] *priceRange*[cheap])
taste of cambridge is located in the riverside area . it is cheap .

Inform(*customerRating*[3 out of 5] *eatType*[restaurant] *familyFriendly*[yes] *name*[Zizzi])
there is a kid friendly restaurant called zizzi . it has a customer rating of 3 out of 5 .

Inform(*area*[city centre] *eatType*[pub] *name*[The Cambridge Blue] *near*[Yippee Noodle Bar] *priceRange*[high])
the cambridge blue is a pub in the high price range . it is located in the city centre near the yippee noodle bar .

Inform(*area*[riverside] *customerRating*[average] *name*[The Phoenix])
the phoenix is located in the riverside area near the riverside . it has an average customer rating .

Inform(*customerRating*[average] *familyFriendly*[yes] *food*[Indian] *name*[Loch Fyne])
loch fyne provides indian food . it is family friendly and has an average customer rating .

Inform(*area*[riverside] *customerRating*[1 out of 5] *food*[Italian] *name*[The Phoenix] *near*[The Six Bells])
the phoenix is located in the riverside area near the city centre , near the six bells . it serves italian food and has a customer rating of 1 out of 5 .

Table 11: Example E2E samples obtained using noise injection sampling with lex. p_0 and $q_{\mathfrak{R}}$ to parse the MR.