Named Entity Disambiguation Boosted With Knowledge Graph

Brian Lin
Shane Ong
Cory Williams
Yun Bin (Matteo) Zhang

Overview

Knowledge Base:

- Jaguar Cars (car)
- Jaguar Land Rover (company)
- Jaguar (animal)

I want to drive my Jaguar around the city.

Overview

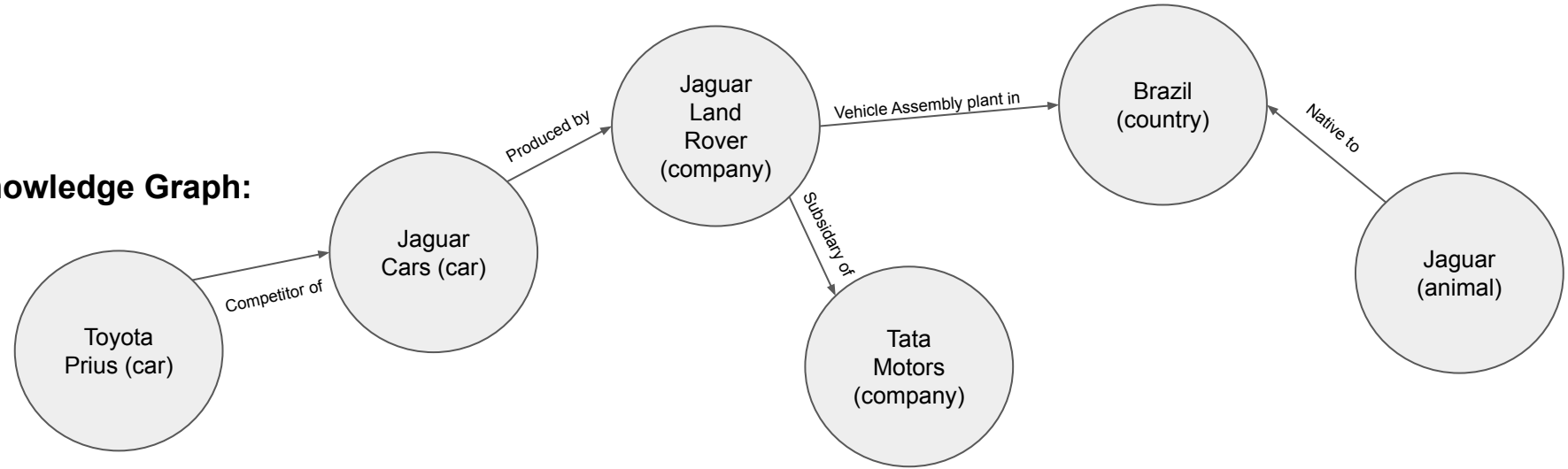
Knowledge Base:

- Jaguar Cars (car)
- Jaguar Land Rover (company)
- Jaguar (animal)

I want to drive my Jaguar around the city.

Overview

Knowledge Graph:

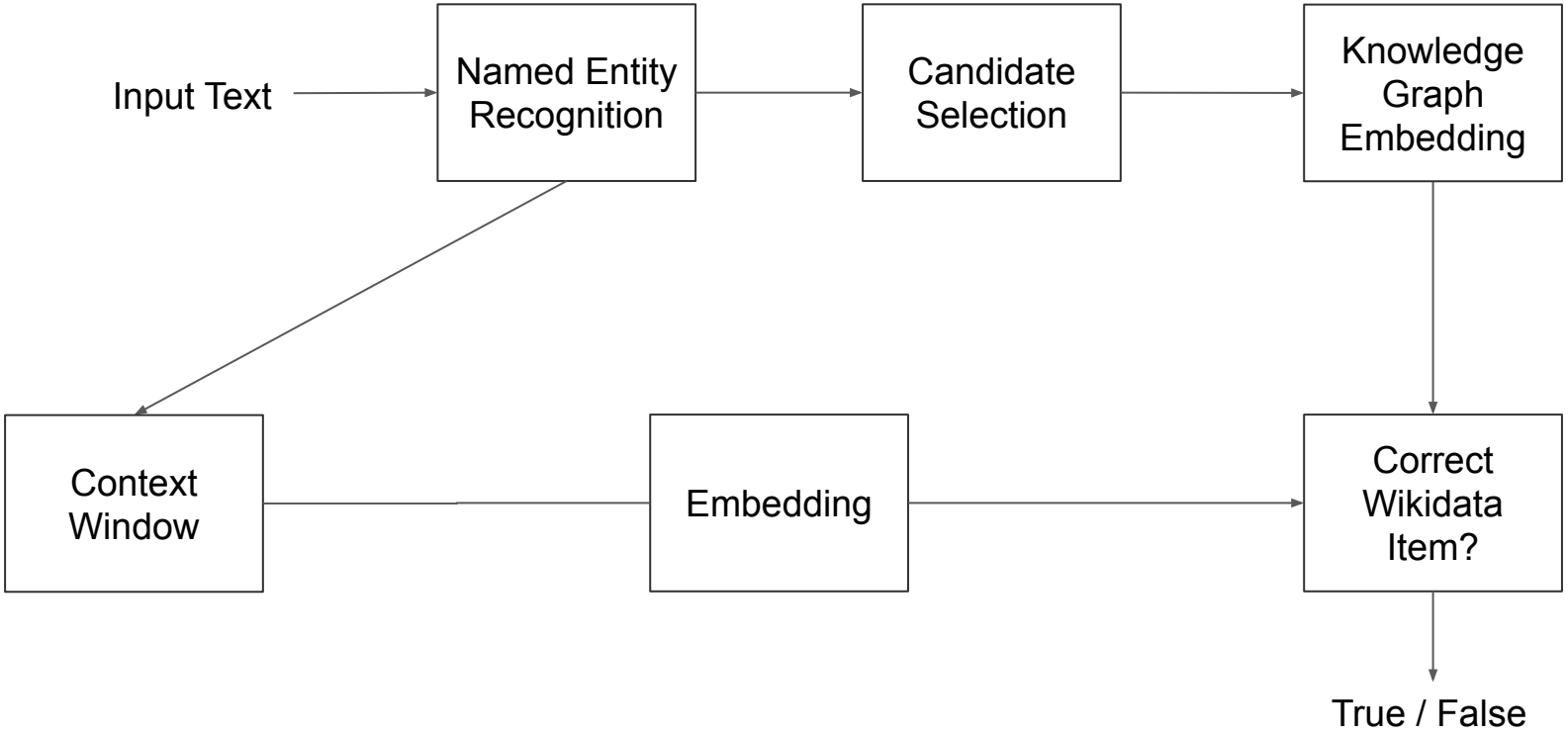


… I want to drive my Jaguar around the city. … It is so much better than my previous Toyota Prius

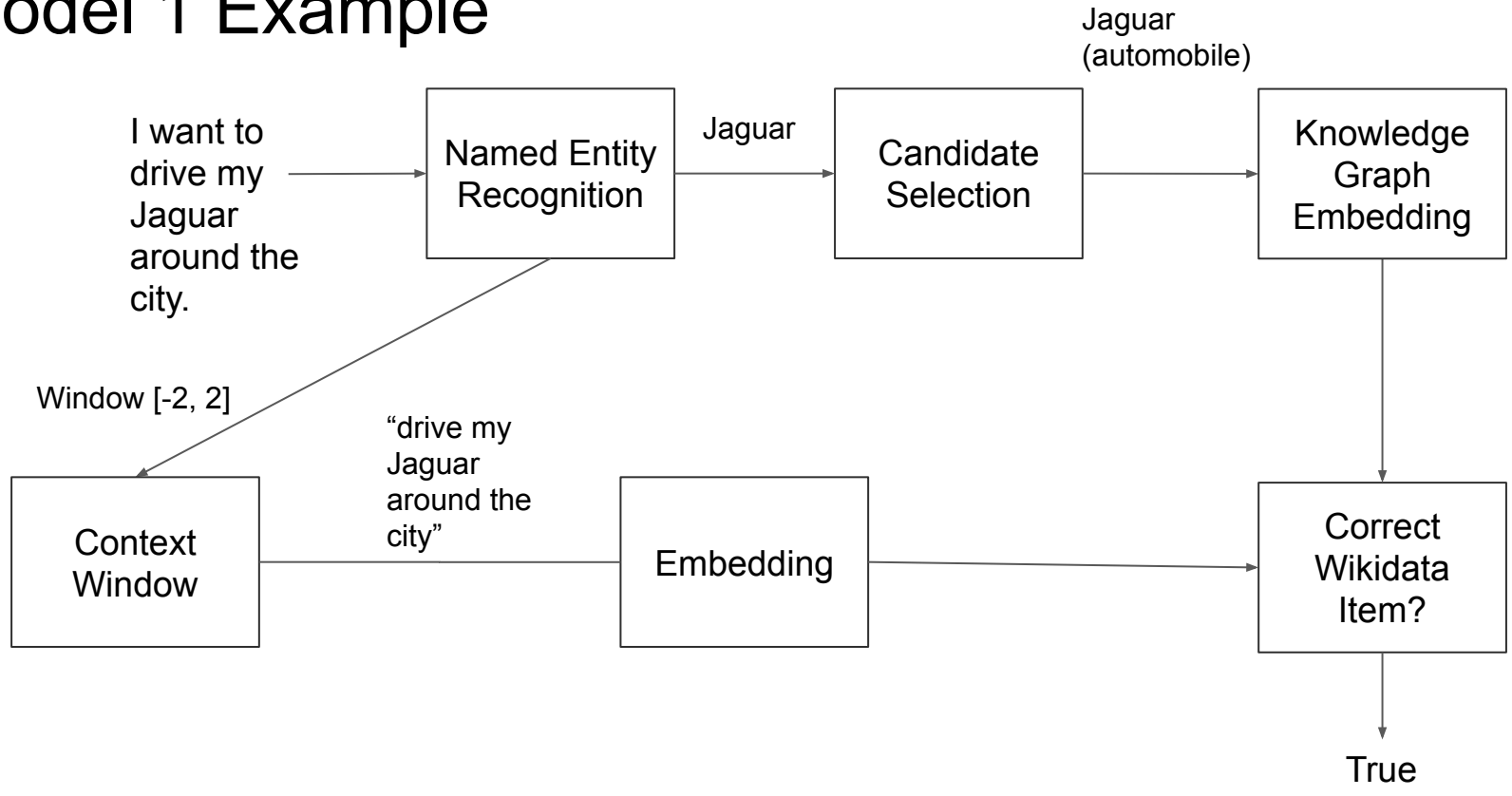
Data

- Moving from 5000 to 385000 Wikipedia article introductions (~1% of articles)
- Computational constraints to further expand
- Currently training on actual Wikipedia article hyperlinks rather than NER identified entities
  - Focuses on NED rather than NER and Candidate Selection
  - Test metrics are not ‘contaminated’ by previous stages
  - Final pipeline will include generic NER and Candidate Selection components

Model 1


Input Text
Named Entity Recognition
Candidate Selection
Knowledge Graph Embedding
Context Window
Embedding
Correct Wikidata Item?
True / False

Model 1 Example

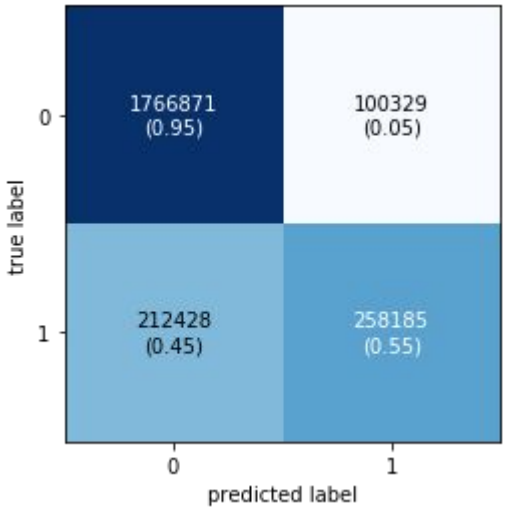


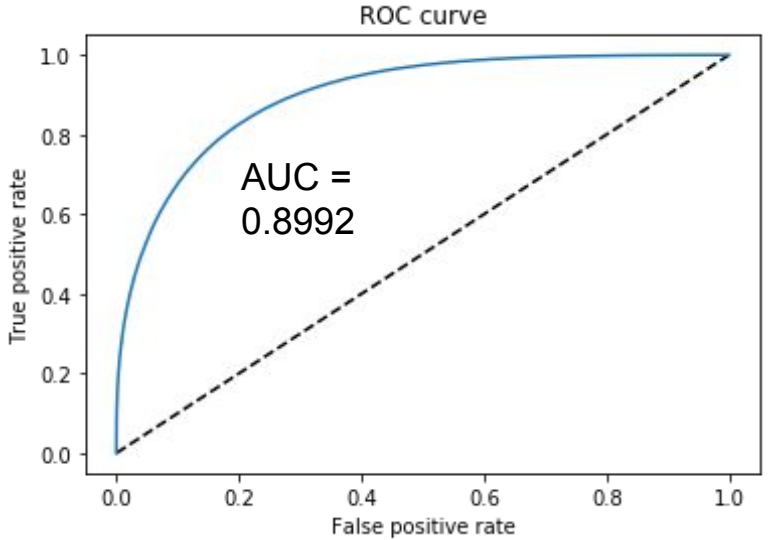
Model 1 Architecture

Layer (type)	Output Shape	Param #	Connected to
input_3 (InputLayer)	(None, 20)	0	
embedding_1 (Embedding)	(None, 20, 100)	44588100	input_3[0][0]
lstm_1 (LSTM)	(None, 128)	117248	embedding_1[0][0]
input_4 (InputLayer)	(None, 250)	0	
concatenate_1 (Concatenate)	(None, 378)	0	lstm_1[0][0] input_4[0][0]
dense_1 (Dense)	(None, 256)	97024	concatenate_1[0][0]
dropout_1 (Dropout)	(None, 256)	0	dense_1[0][0]
dense_2 (Dense)	(None, 1)	257	dropout_1[0][0]

Model 1 Results

- Training Set: 9363481 rows, Test Set: 2337813
  - 20.11% positive, 79.89% negative labels
  - Window size: 20
  - Test Accuracy: 0.8662





Model 1 Candidate List Results

I want to drive my Jaguar around the city.

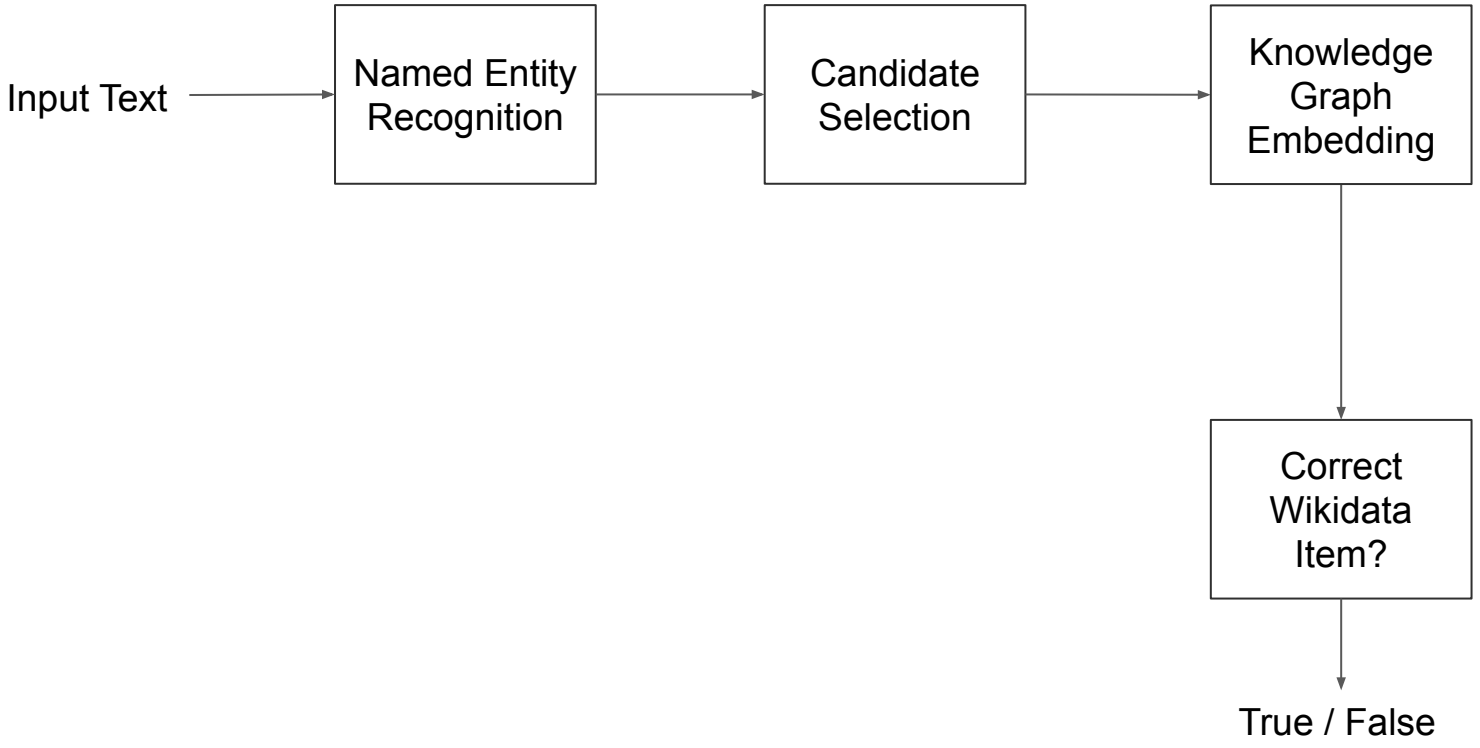
- From candidate selection:

  Candidate List = [Jaguar (animal), Jaguar (car), Jaguar (novel)]

- NED: Select the the highest predicted probability:
  - Jaguar (animal) : 0.9
  - Jaguar (car): 0.6
  - Jaguar (novel): 0.2
- Test set accuracy: 0.7497

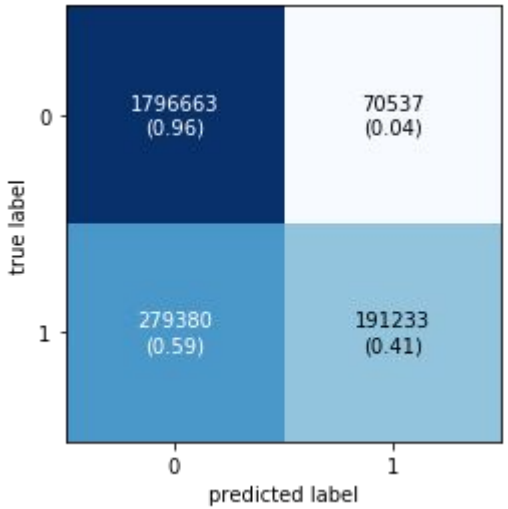
Model 1 Ablation Experiment 1

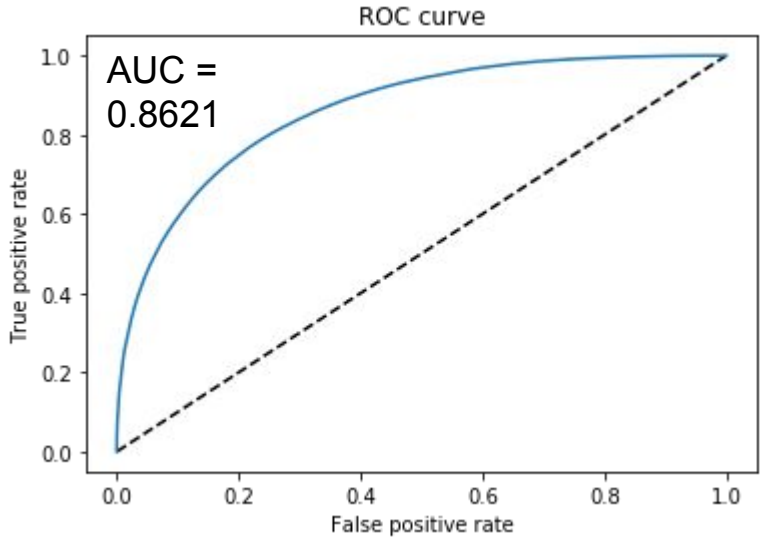
- Without word embeddings



Model 1 Ablation Experiment 1

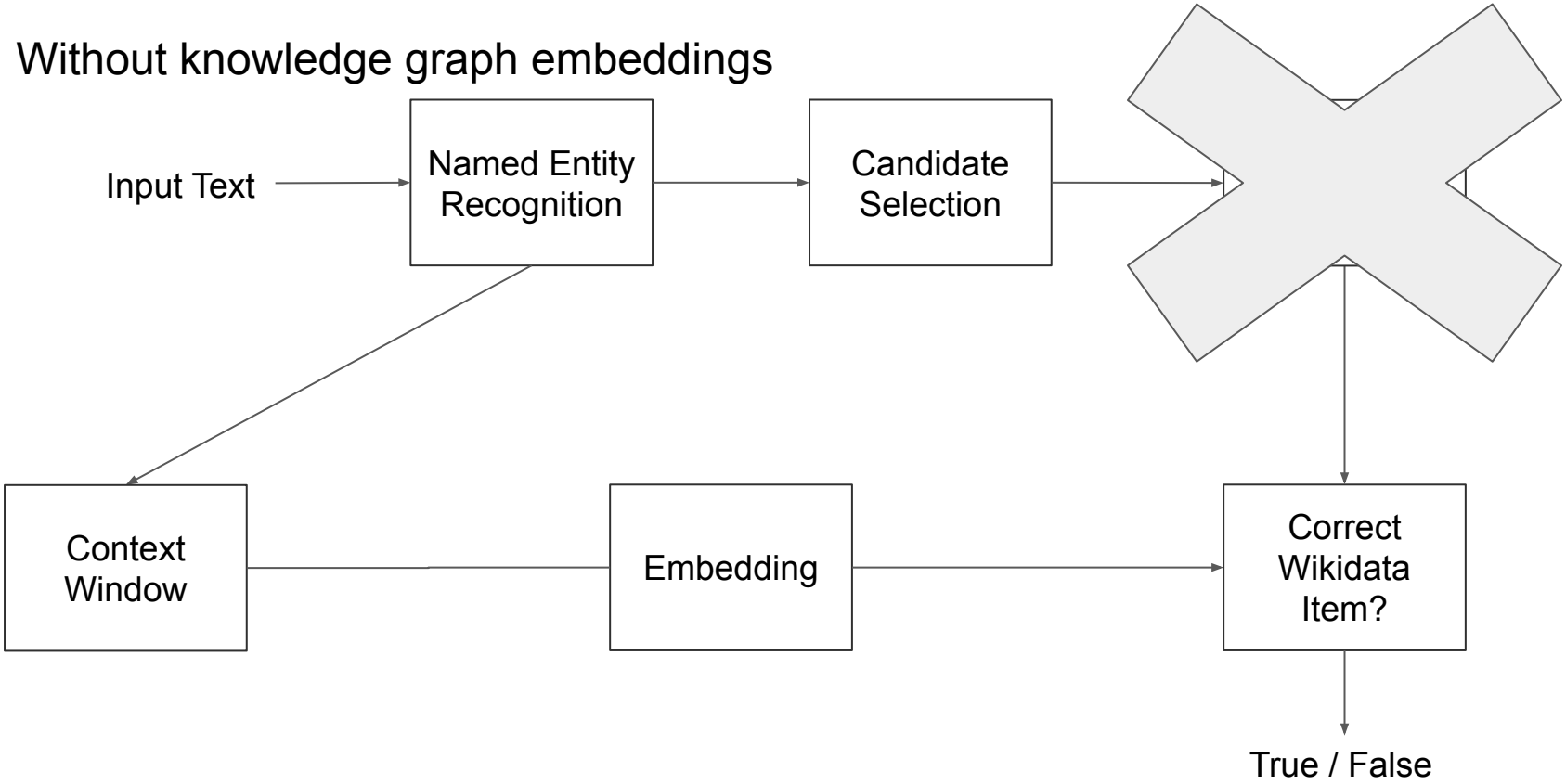
- Without word embeddings
- Test accuracy (individual): 0.8503
- Test accuracy (candidate list): 0.6894





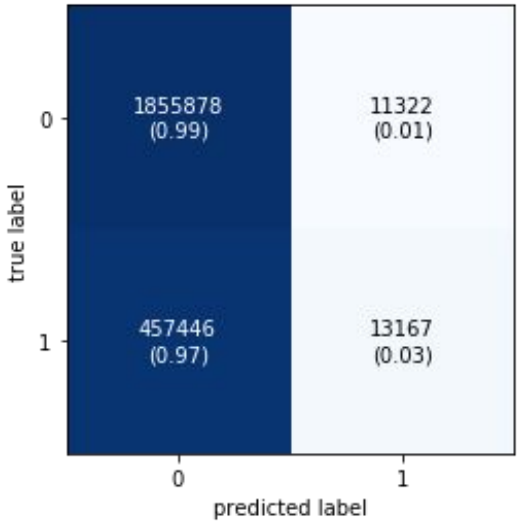
Model 1 Ablation Experiment 2

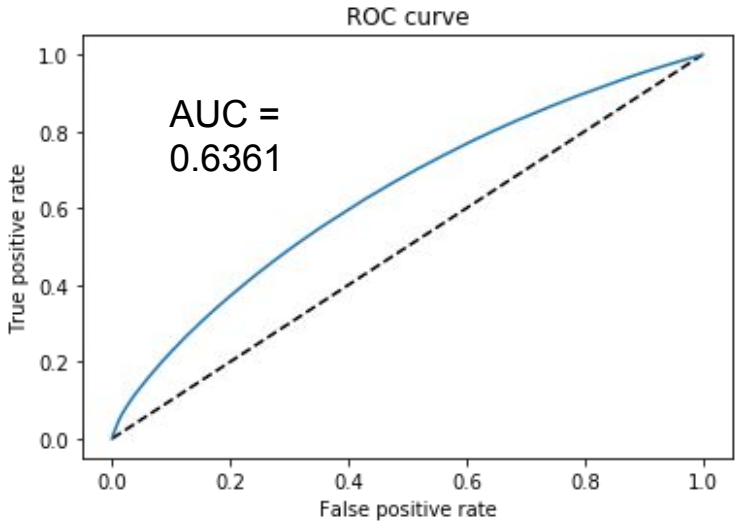
- Without knowledge graph embeddings



Model 1 Ablation Experiment 2

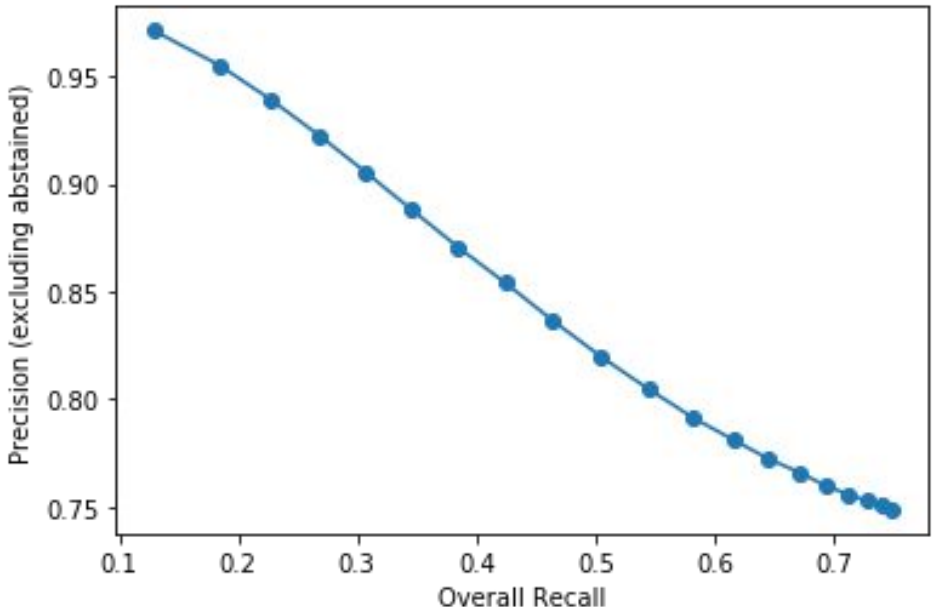
- Without knowledge graph embeddings
- Test accuracy (individual): 0.7995
- Test accuracy (candidate list): 0.3559



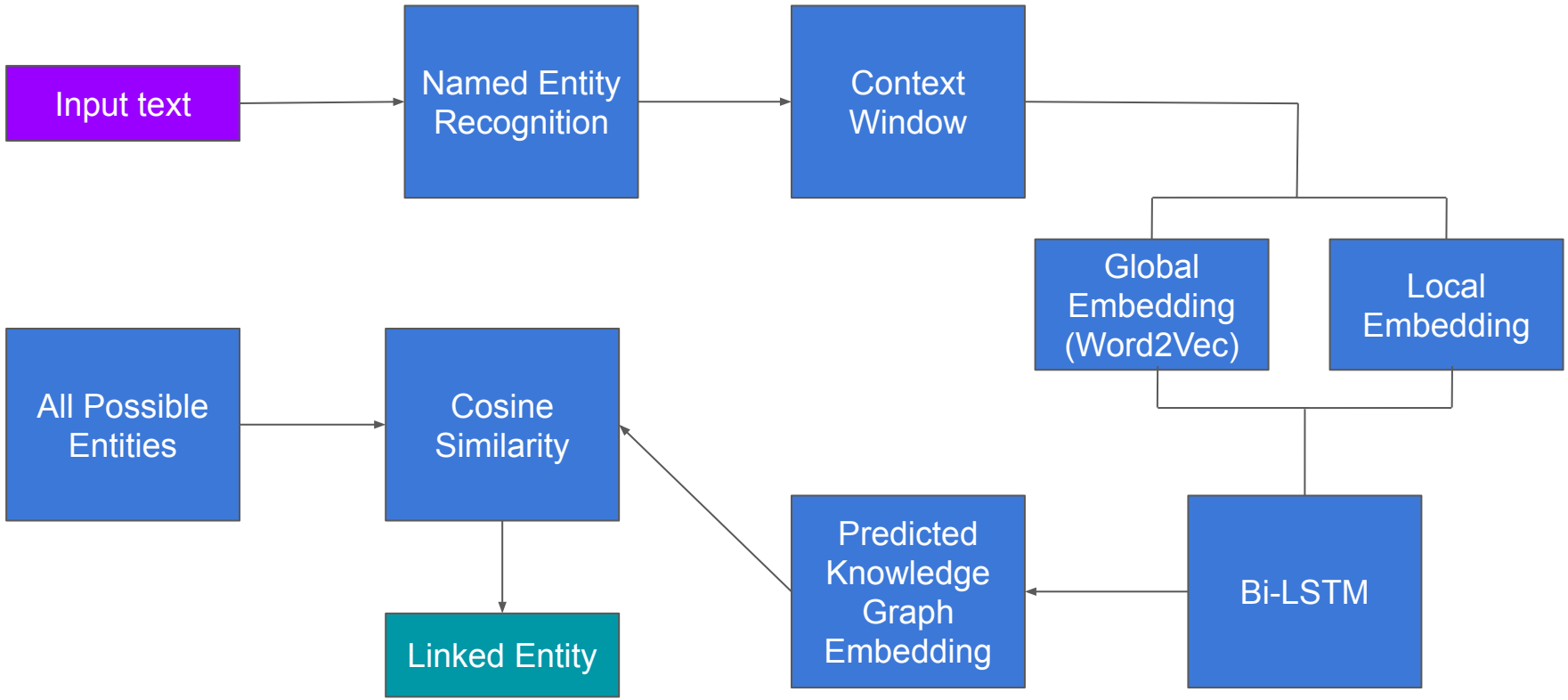


Model 1 Abstaining By Threshold

- Candidate list with missing candidates
- Add the option of abstaining when max probability below threshold
- Without abstaining (candidate list):
  - Test accuracy: 0.75
  - Test recall: 0.75
- Abstaining at threshold = 0.25
  - Test accuracy: 0.77
  - Test recall: 0.65
  - Abstaining rate: 16.18%

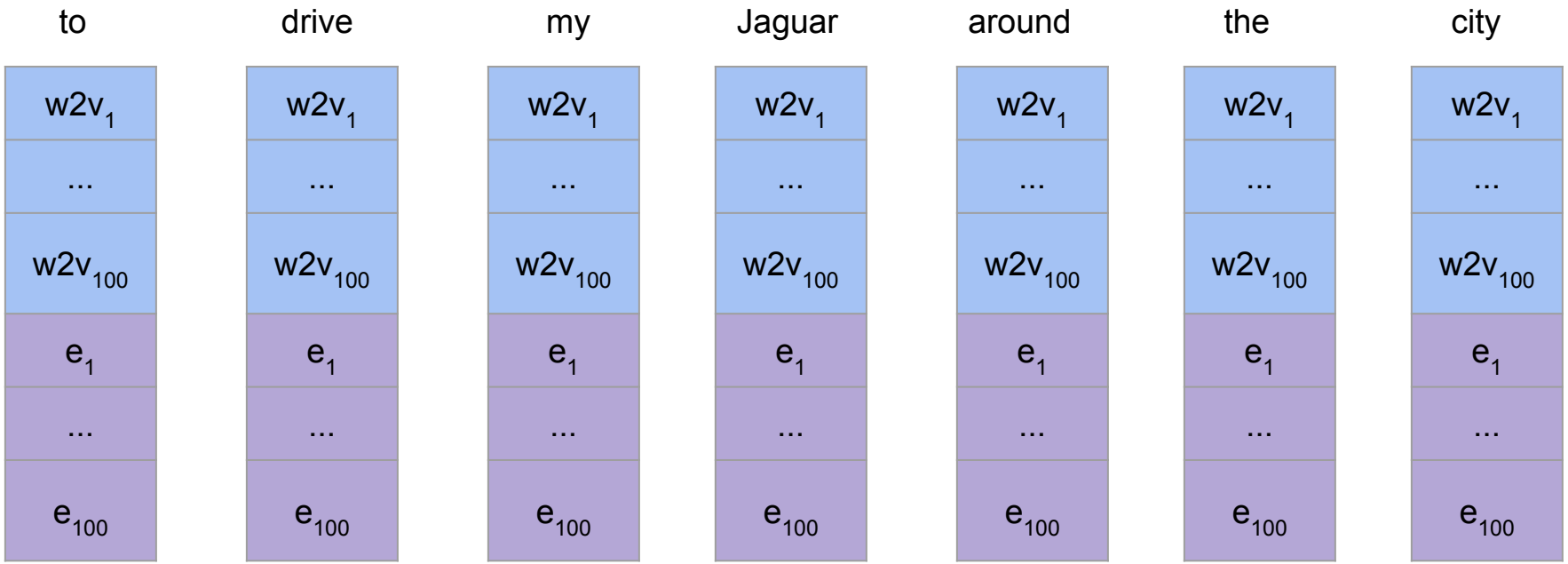


Model 2

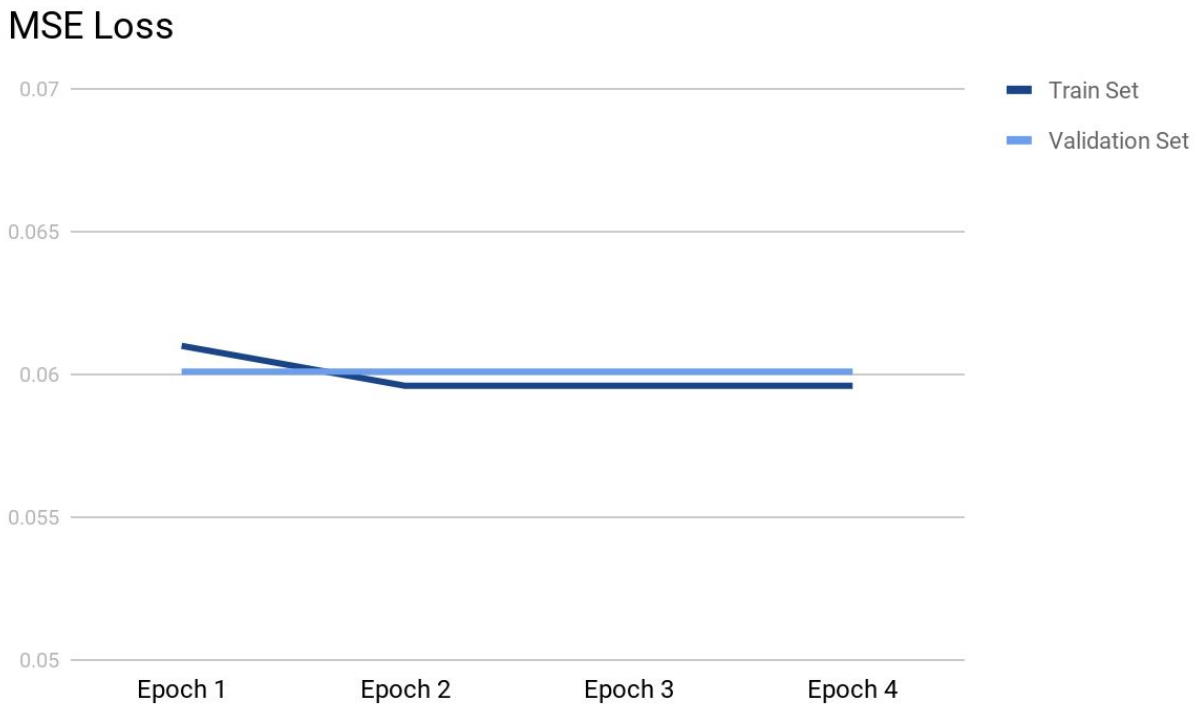

Input text
Named Entity Recognition
Context Window
Global Embedding (Word2Vec)
Local Embedding
All Possible Entities
Cosine Similarity
Linked Entity
Predicted Knowledge Graph Embedding
Bi-LSTM

Model 2 - Embedding

...I want to drive my **Jaguar** around the city...



Model 2 - Results



Model 2 - Results

Actual KG Embedding

Mean: 0.015
Std Dev: 0.322

-0.012
-0.823
0.473
0.094
0.637
...

Predicted Embedding

Mean: 0.0066
Std Dev: 0.098

0.003
-0.004
0.008
-0.009
0.012
...

Next Steps

- **Create overall pipeline using Model 1**
  - Input: Text file
  - Output: Annotated text file with linked entities

- **Get Model 2 working**
  - Change loss function?
  - Add candidate list avg KG embeddings as input to model?