Workshop 8

COMP90051 Machine Learning

Semester 2, 2020

Learning Outcomes

By the end of this workshop you should be able to:

1. explain how to vectorise text data for input into neural nets
2. design a neural net for text classification with a recurrent architecture
3. implement recurrent neural nets in Keras

Sentiment analysis

- Goal: predict whether a movie review is positive or negative based on raw text
- Use the Large Movie Review Dataset (from IMDb.com)

Cats (2019)

Staring at a wall is a far better use of your time.
25 December 2019

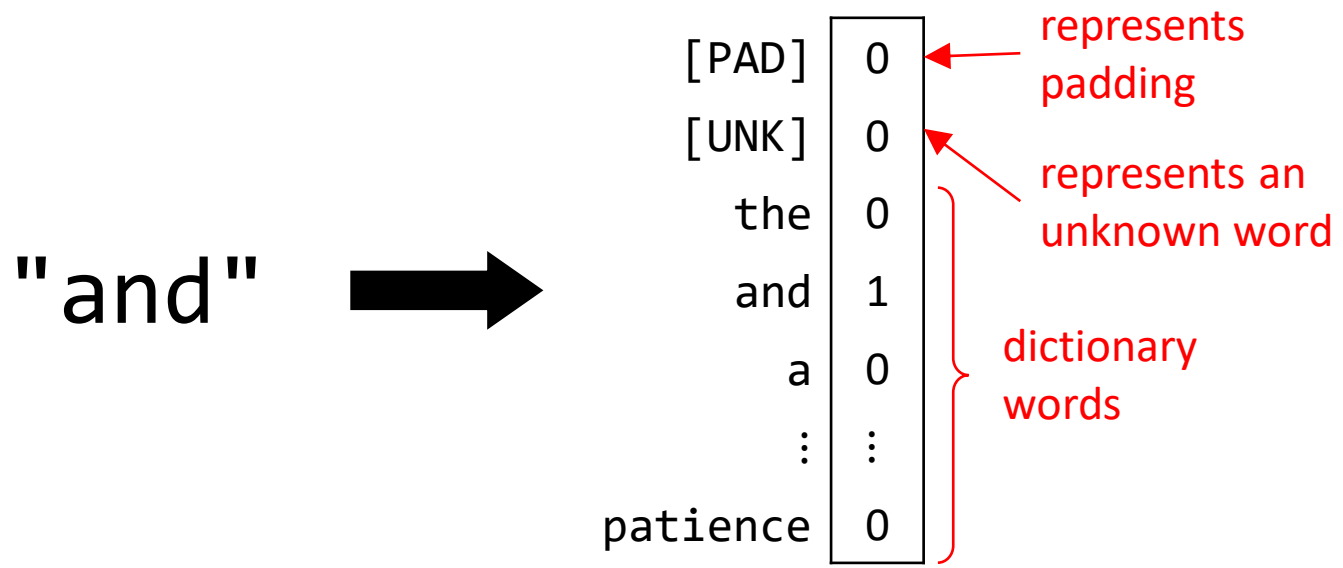
There is a wall in my house. The wall is a simple wall, eight feet tall, and fifteen feet wide, it sits in a corner in the basement. It has been painted beige, with a few coats to cover up the bumps and scratches from coats before. It's got it's idiosyncrasies, the way the light jumps off the paint, the not so even drywalling underneath. Stare at it long enough, and it begins to speak to you - or at least allow you to speak to it. Maybe there's something to the wall - or maybe it's just my romantic illusions of inspiration and contemplation that bring life to it. My point is this - staring at that blank wall for two hours is far more inspiring, interesting and enlightening than watching even just two minutes of the cinematic colonoscopy that is "Cats." A film so awful, so ridiculous and so void of substance its very existence is an affront to musicals, film and theatre. Unless you're a fan of 'so bad it's good' cinema, stay away. This is a watershed moment in bad movie history.

→ negative sentiment

From https://www.imdb.com/review/rw5353770/?ref_=tt_urv

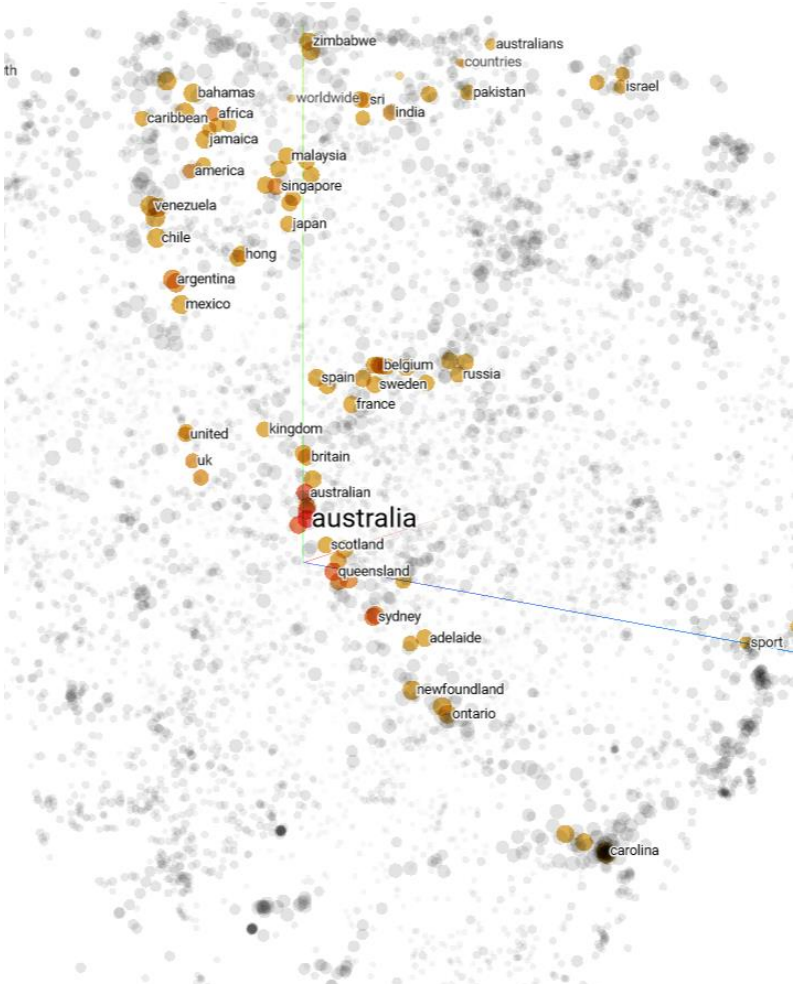
Word embeddings

- Need to represent words as vectors for compatibility with differentiable neural networks
- A familiar solution is one-hot encoding
- However it's inefficient for large dictionaries (very sparse)



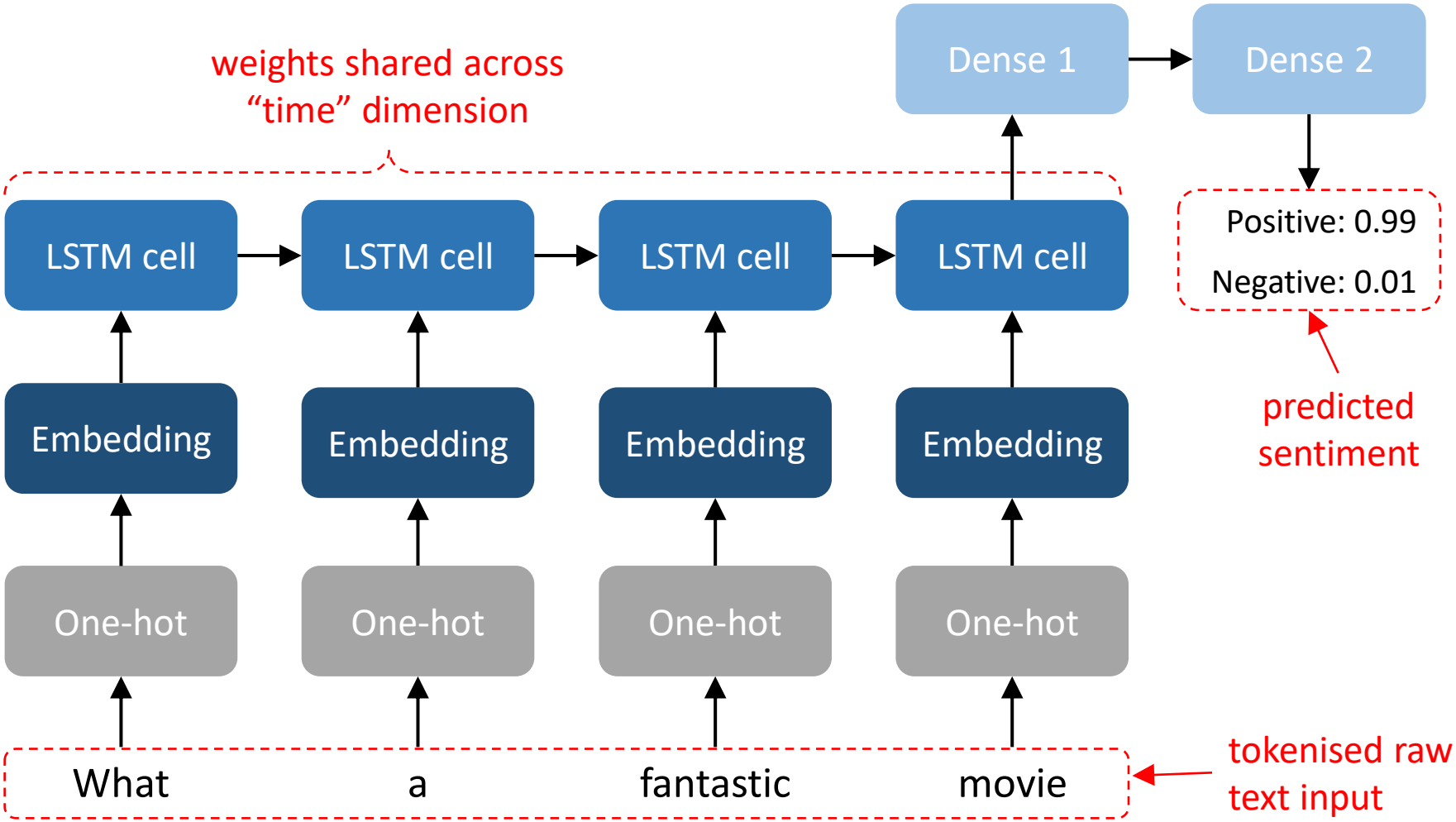
Word embeddings

- To improve efficiency, we can learn an embedding in a lower-dimensional space
- Embedding is a linear map from one-hot encoded word vectors $\mathbf{x}_{\text{o-h}} \in \{0,1\}^k$ to dense word vectors $\mathbf{x}_{\text{dense}} \in \mathbb{R}^d$, parameterised by a weight matrix $\mathbf{W} \in \mathbb{R}^{k \times d}$
- Common to use pre-trained embeddings



t-SNE plot of word2vec embedding from http://projector.tensorflow.org/

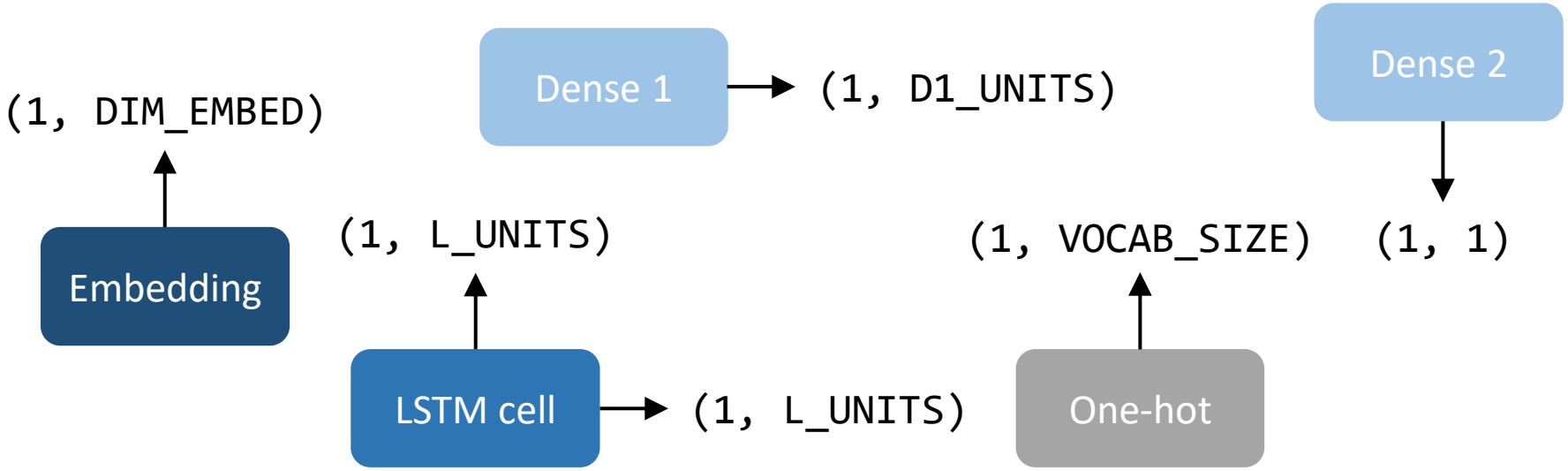
Sentiment classifier architecture



Exercise

Let `VOCAB_SIZE` be the size of the word dictionary, `DIM_EMBED` be the dimension of the embedding space, `L_UNITS` be the number of units in the LSTM cell, and `D1_UNITS` be the number of units in dense layer 1.

Write down the shape of the tensor output for each layer when a single sample is fed through the network.



Worksheet 8