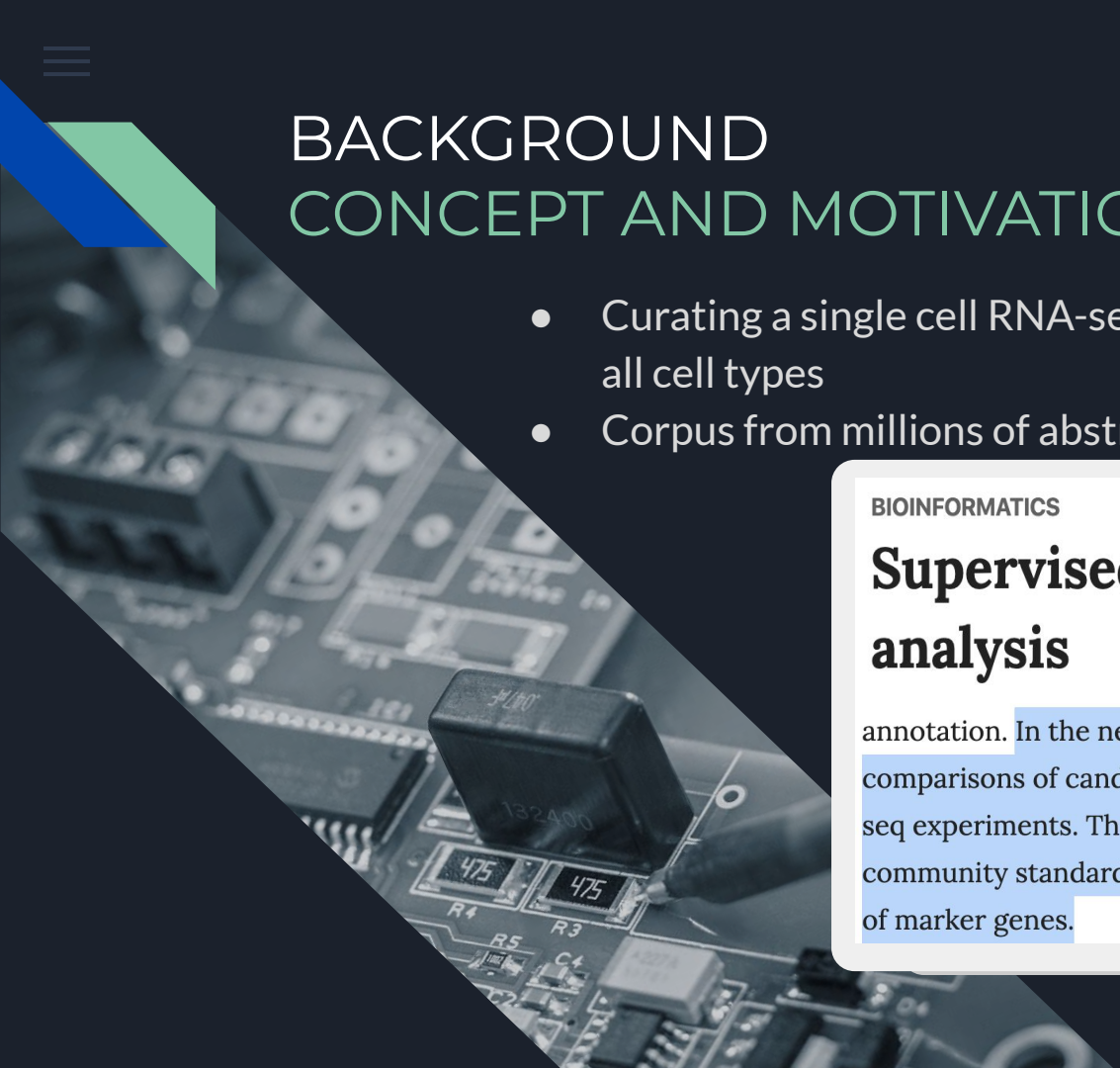
TYPER: Presentation

Steven Wu
Nicolas Cardozo
Samantha Borje

BACKGROUND
CONCEPT AND MOTIVATION

- Curating a single cell RNA-sequencing markers from across all cell types
- Corpus from millions of abstracts from pubmeds API



BIOINFORMATICS

Supervised clustering for single-cell analysis

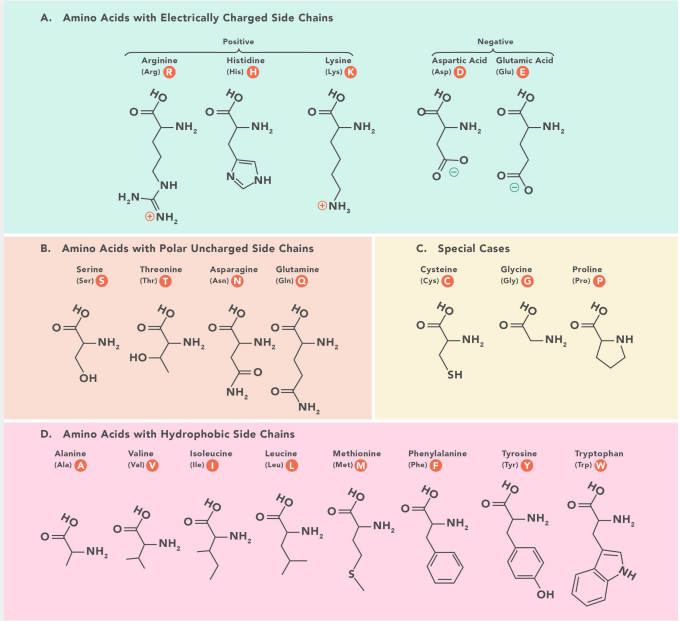
annotation. In the near future, we are likely to see large scale comparisons of candidate lists derived from the literature and scRNA-seq experiments. This will, in turn, drive the emergence of consistent community standards regarding the selection, evaluation and reporting of marker genes.

RESULTS
USE CASE

- The problem we are trying to solve is curating a list of single cell RNA-sequencing markers from across all cell types.

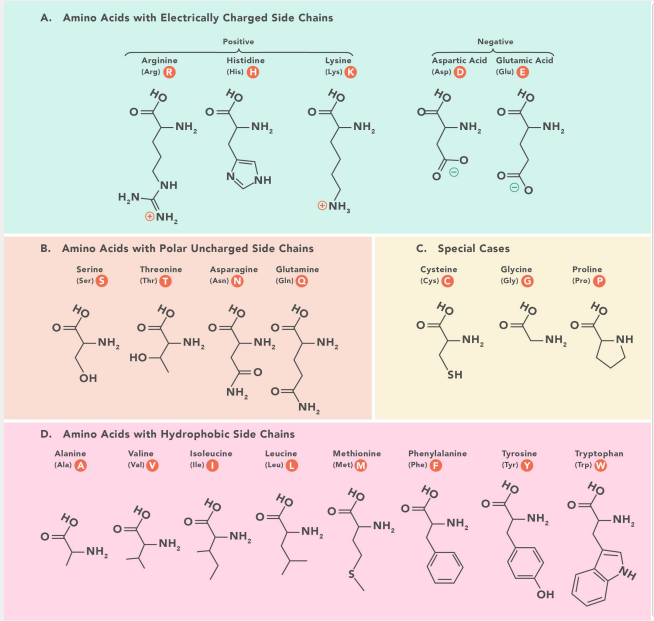
- We are going curate the corpus from millions of abstracts from pubmeds API.

RESULTS
PROOF OF CONCEPT



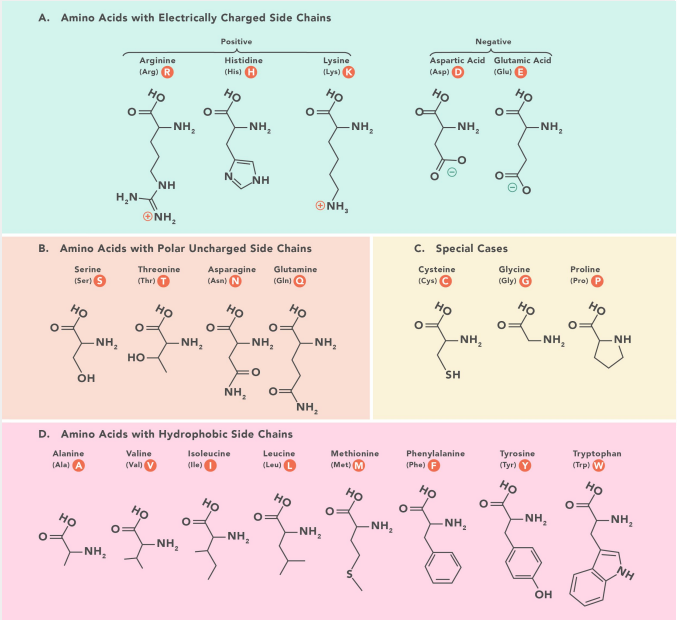

```
# load libraries
from gensim.models import
KeyedVectors
import numpy as np
import matplotlib.pyplot as plt
import umap
import pandas as pd
import seaborn as sns

# load word embeddings
word_vectors =
KeyedVectors.load_word2vec_format('pu
bmed2018_w2v_200D/pubmed2018_w2v_200D
.bin', binary=True)
```

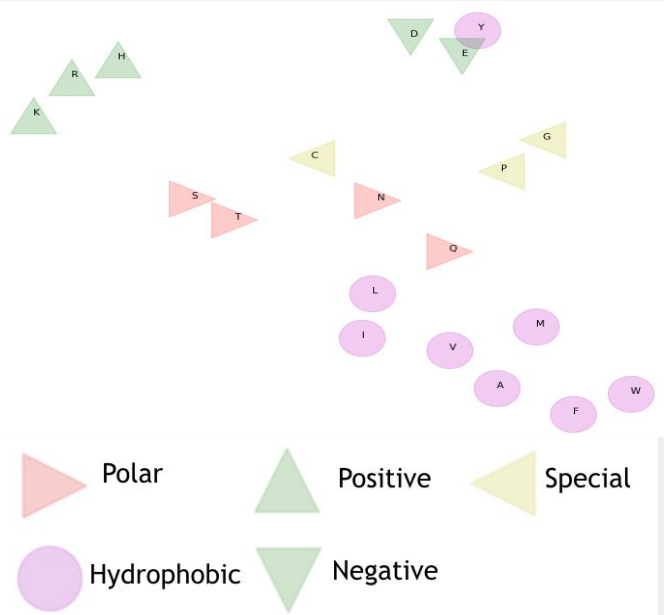
RESULTS
PROOF OF CONCEPT




```
# 20 AMINO ACIDS to query and their properties
# define marker/shapes for each amino acid based off of proprty
aa = ['alanine', 'isoleucine', 'leucine', 'methionine',
      'valine', 'phenylalanine', 'tryptophan', 'tyrosine',
      'asparagine', 'cysteine', 'glutamine', 'serine',
      'threonine', 'aspartic-acid', 'glutamic-acid', 'arginine',
      'histidine', 'lysine', 'glycine', 'proline']

aa_properties = ['Hydrophobic', 'Hydrophobic', 'Hydrophobic', 'Hydrophobi
c',
                 'Hydrophobic', 'Hydrophobic', 'Hydrophobic', 'Hydrophobic',
                 'Polar', 'Special', 'Polar', 'Polar',
                 'Polar', 'Negative', 'Negative', 'Positive',
                 'Positive', 'Positive', 'Special', 'Special']

aa_short = ['A', 'I', 'L', 'M', 'V', 'F', 'W', 'Y',
            'N', 'C', 'Q', 'S', 'T', 'D', 'E', 'R',
            'H', 'K', 'G', 'P']

m = ['>', '<', '^', 'v', 'o']

cmap_dict = {'Polar': 'r',
             'Special': 'y',
             'Positive': 'g',
             'Negative': 'g',
             'Hydrophobic': 'm'}

m_dict = {'Positive': '^',
          'Negative': 'v',
          'Polar': '>',
          'Special': '<',
          'Hydrophobic': 'o'}
```

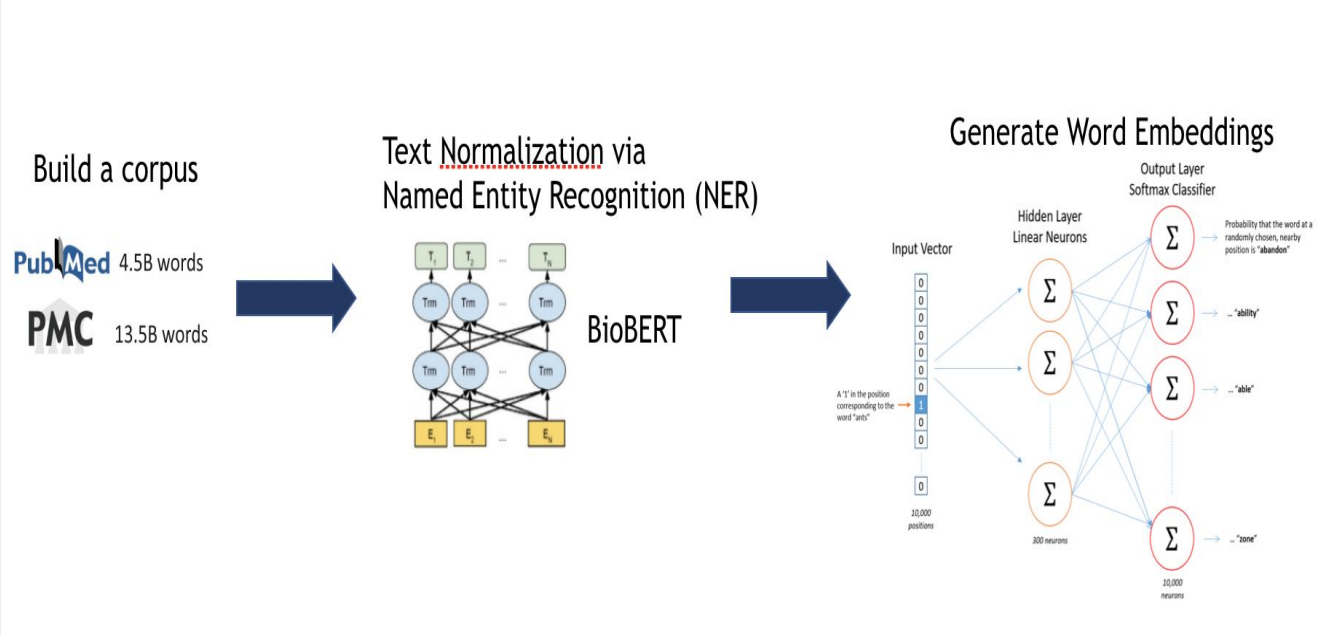
RESULTS
PROOF OF CONCEPT





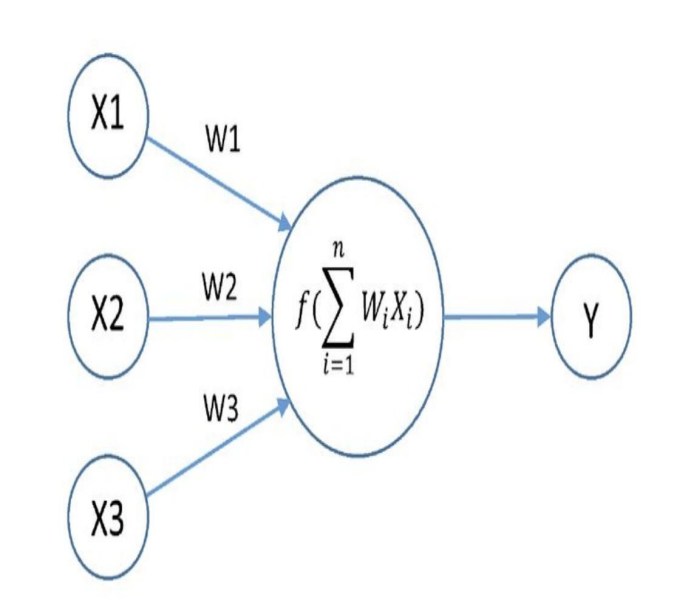
RESULTS
DEMONSTRATION

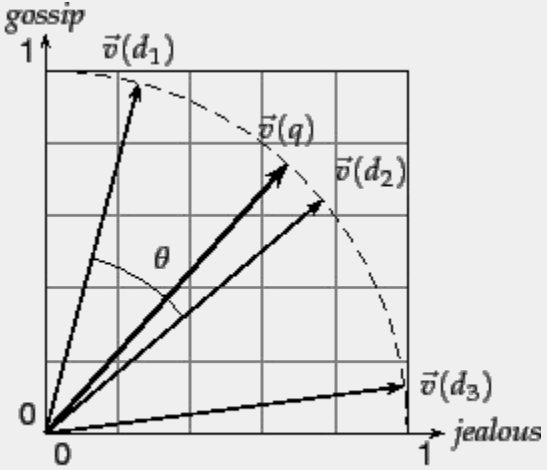
LOADING
Libraries
scRNA-seq data
gene2vec scores



RESULTS
DEMONSTRATION

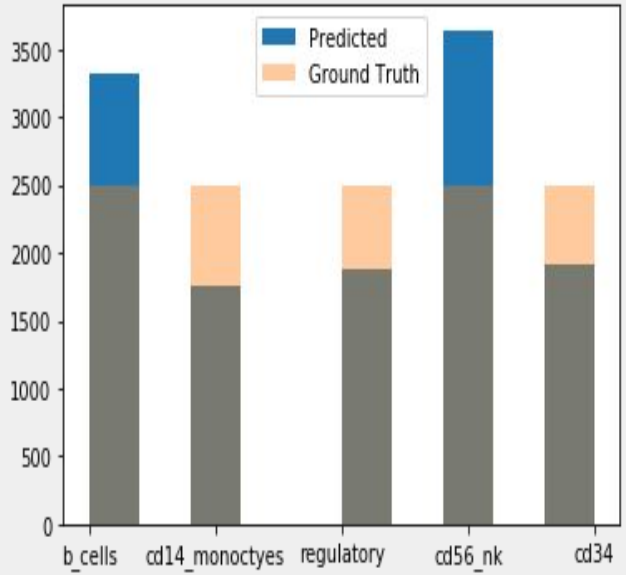
CLASSIFYING INDIVIDUAL CELLS

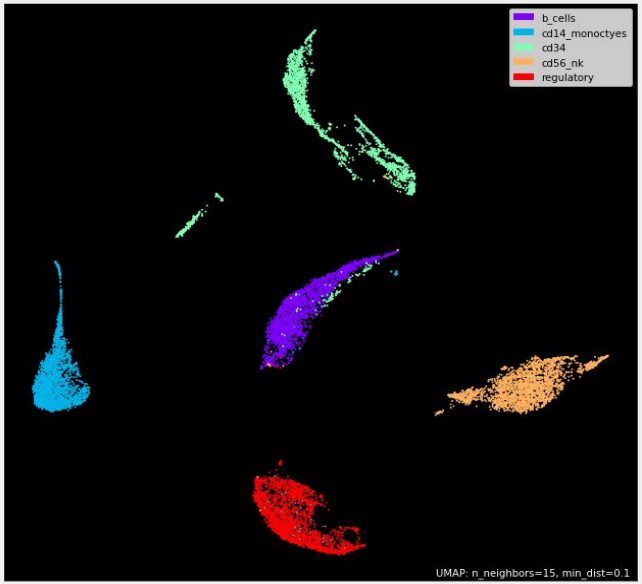




Cosine similarity illustrated. $\text{sim}(d_1, d_2) = \cos\theta$.

RESULTS
DEMONSTRATION

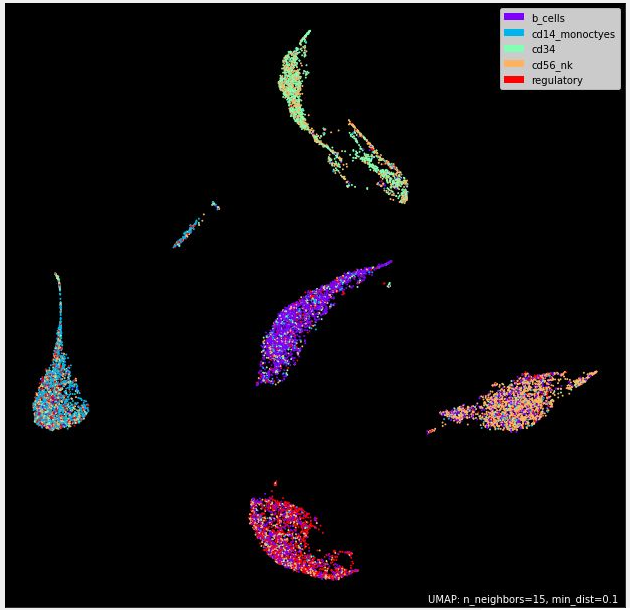


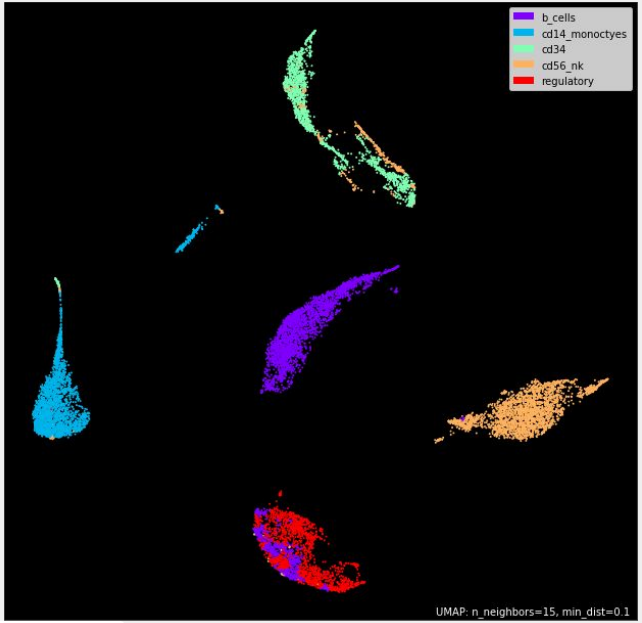


VISUALISATION

RESULTS
DEMONSTRATION

VISUALISATION





DISCUSSION
FUTURE DIRECTIONS

- Improved User Interface (Website? App?)
- Download data on SQL
- Implement the actual BERN algorithm

Thank you!