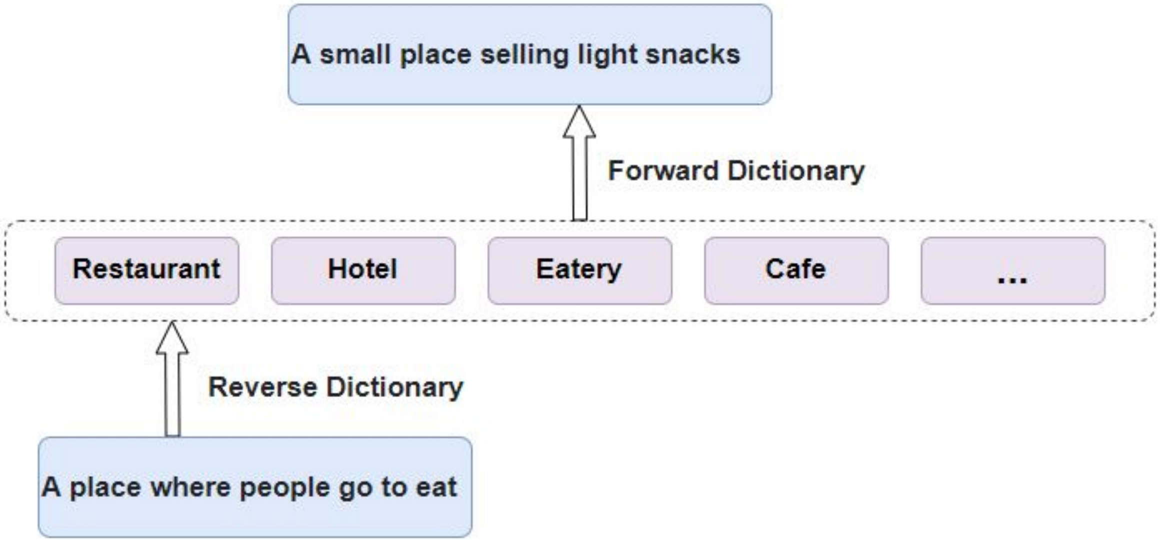
Reverse Dictionary

Aarti Pol

Under Guidance of
Dr. Gaurav Bhatnagar

Reverse Dictionary

- A regular (forward) dictionary maps words to their definitions.
- While a reverse dictionary takes the description of a word as input and outputs the target word together with other semantically similar words.



Motivation

- It can help with “tip-of-the-tongue” problem.
- It can also be helpful in case of some neurological disorders where people are able to identify and describe an object but struggle to name it.
- Provides assistance to new language learners with limited vocabulary.
- In NLP system:
    - can be used to evaluate the quality of sentence representations
    - beneficial to the tasks involving text-to-entity mapping
        - question answering
        - information retrieval

Existing Systems

OneLook.com

a road where cars go quickly without stopping

highway: a major road for any form of motor transport; more definitions... usage examples...

*Showing words related to **a road where cars go quickly without stopping**, ranked by relevance.*

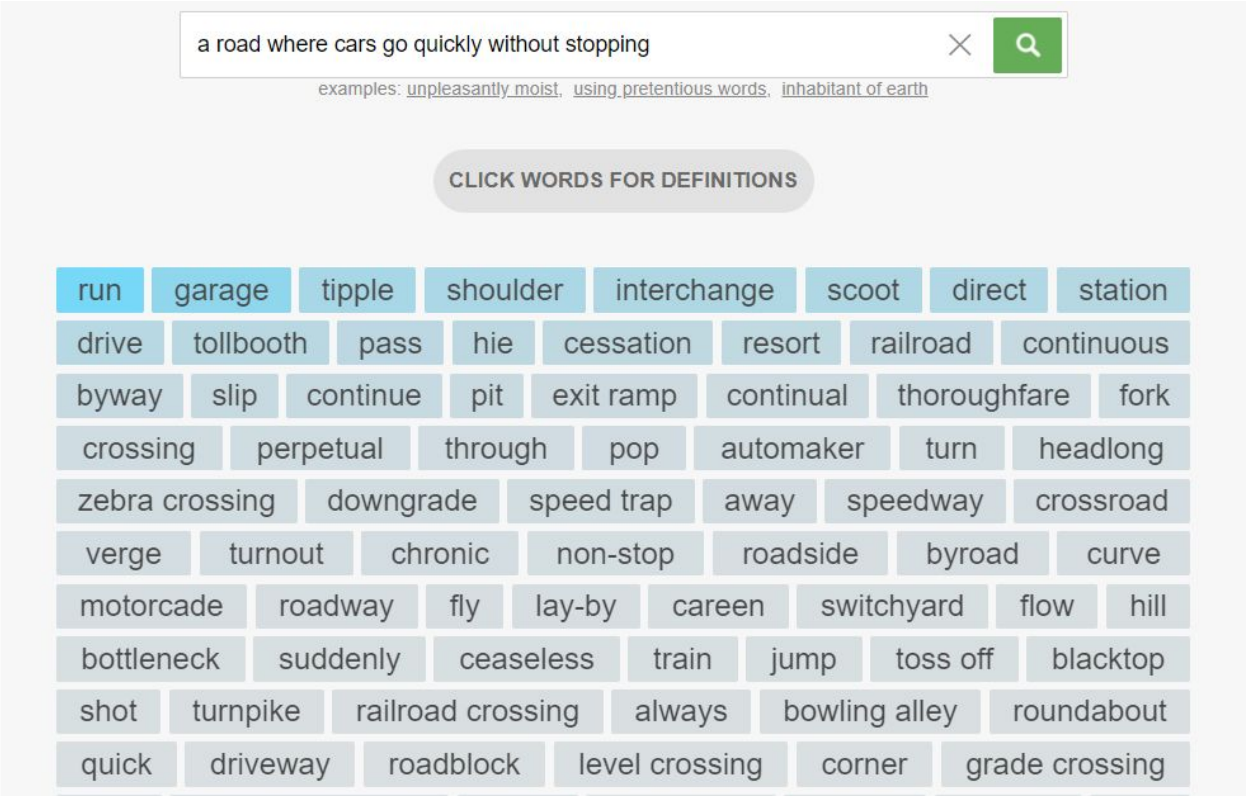
Filter by... Alphabetize

All | Nouns | Adjectives | Verbs | Adverbs

1. highway
2. freeway
3. expressway
4. pass
5. passed
6. station
7. stationing
8. stations
9. shoulder
10. stump
11. stumped
12. stumping
13. stumps
14. turnout
15. turn
16. turned
17. hotel
18. stop
19. stops
20. drag
21. layby
22. lay
23. bridge
24. bridged
25. bridging
26. through
27. draw
28. skid
29. take
30. frog
31. down
32. away
33. sight
34. sights
35. part
36. straight
37. rope
38. roping
39. stage
40. flash
41. lay-by
42. round
43. buzz
44. pull-off
45. rest area
46. rest stop
47. park
48. order
49. home
50. separation
51. run
52. sledge
53. sledging
54. point
55. push
56. jump
57. hack
58. door
59. spin
60. pole
61. travel
62. walk
63. scat
64. leaving
65. hacking
66. leave
67. speed trap
68. short
69. dead
70. depart
71. shorts
72. wau
73. hydroplane
74. slide
75. pit
76. speed
77. clearway
78. paddock
79. balking
80. tipple
81. end
82. rollercoaster
83. cease
84. ended
85. landing
86. continuous
87. triathlon
88. hell
89. interchange
90. consecutive
91. obstacle
92. transition
93. terminal
94. riot
95. cruise
96. control
97. transfer
98. pause
99. stall
100. discretion

Existing Systems

RverseDictionary.org

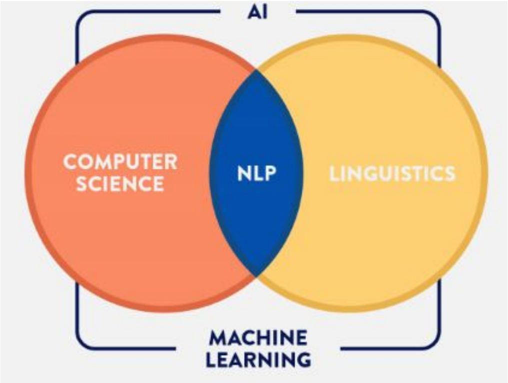


Limitations of existing systems

- Code is not open source
  - Unreliable
  - Paid
- Not multilingual
  - Work only on English language
- Not cross-lingual
  - Can have input in one language and output in another

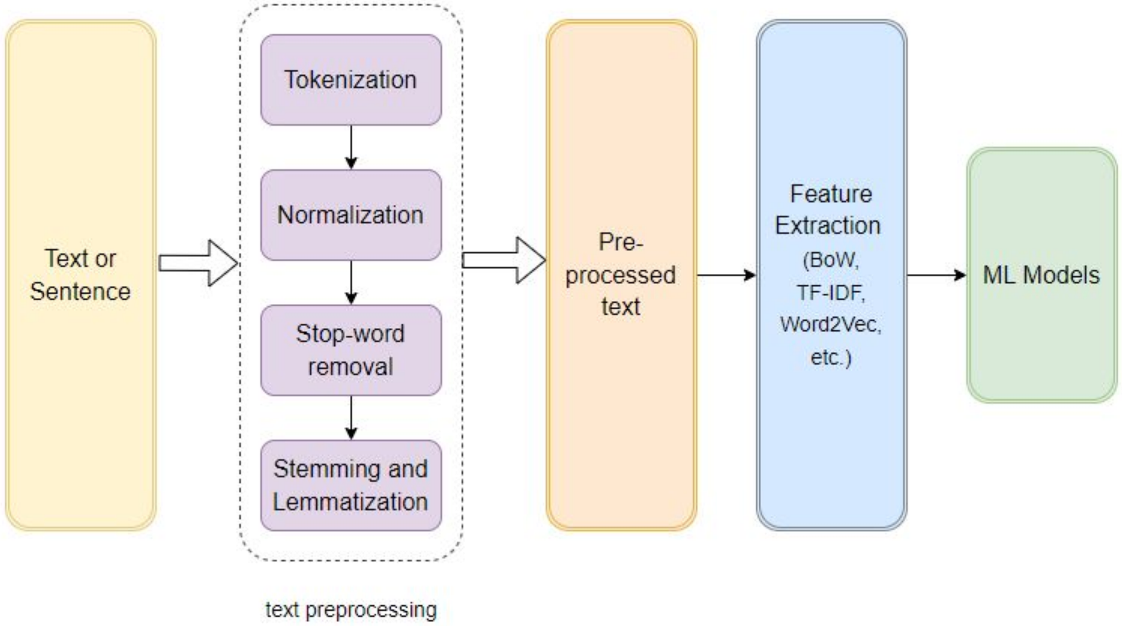
Natural Language Processing (NLP)

- Interdisciplinary field of computer science, linguistics and artificial intelligence
- Deals with understanding of human language and their interaction with computer systems.
- Many interesting problems
  - Classification
  - Translation
  - Information Extraction
  - Question answering
  - Dialogue systems
- Major progress in the last decade due to enhancement in deep learning models



Text Pre-processing

- Corpus
- Tokenization
- Normalization
- Stemming
- Lemmatization
- Stop Words
- Parts-of-speech (POS) Tagging



Text to features

- One-hot encoding

mango	1	0	0	0	0
is	0	1	0	0	0
sweet	0	0	1	0	0
and	0	0	0	1	0
pulpy	0	0	0	0	1

- Bag of Words (BoW)
    - Unique words: ['and', 'is', 'mango', 'orange', 'puppy', 'sour', 'sweet'].
    - BoW is number of times that word has occurred in the given sentence

	and	is	mango	orange	puppy	sour	sweet
mango is sweet and puppy	1	1	1	0	1	0	1
orange is sweet and sour	1	1	0	1	0	1	1

Text to features

- TF-IDF Model

$$tf_{t,d} = \frac{n_{t,d}}{Number\ of\ terms\ in\ the\ document}$$

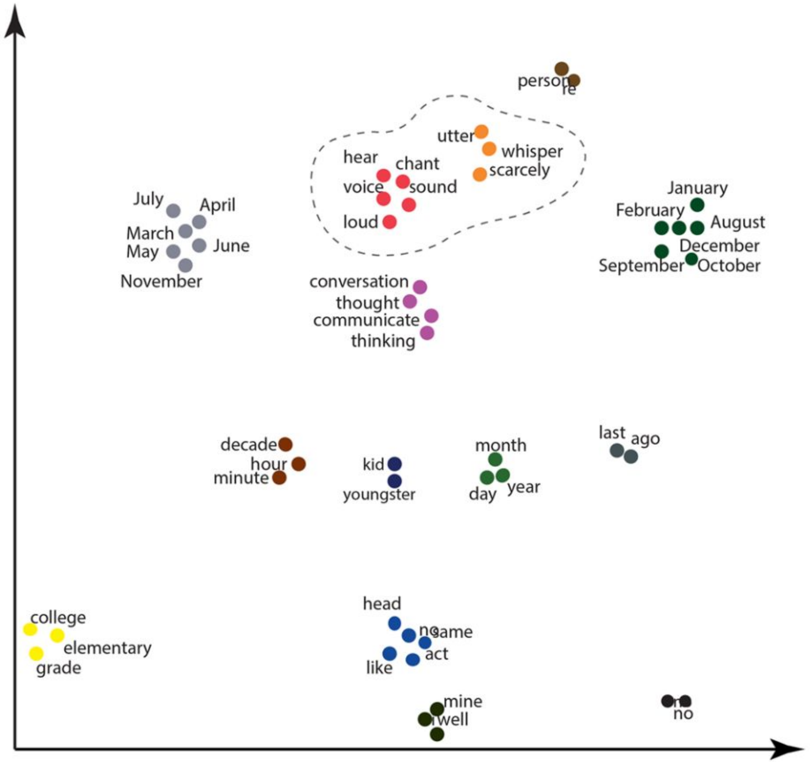
$$idf_t = log \frac{Number\ of\ documents}{Number\ of\ documents\ with\ term\ 't'}$$

$$(tf - idf)_{t,d} = tf_{t,d} * idf_t$$

- Advantage of TF-IDF over BoW:
  - Gives weightage to more and less frequently occurring words accordingly
- Drawbacks of BoW and TF-IDF:
  - don’t capture the meaning for words and sentences themselves.

Word Embedding

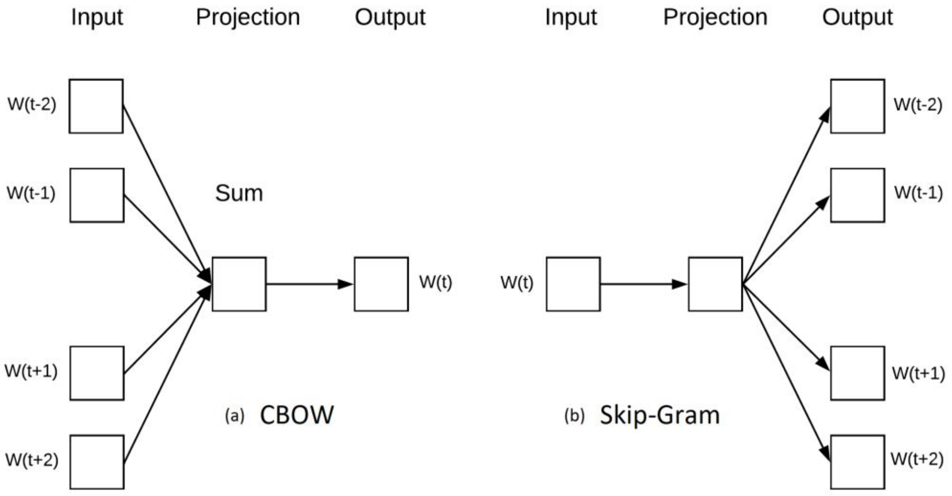
- Words or phrases are embedded (mapped) to real valued vectors such that their meaning is captured.
- Main idea:
    - “distributional hypothesis” or
    - “You shall know a word by its neighbours”
- We can apply mathematical operations on these vectors.
- Commonly used word vectors come from these embeddings like
    - Word2Vec, GloVe - word based
    - FastText - subword based

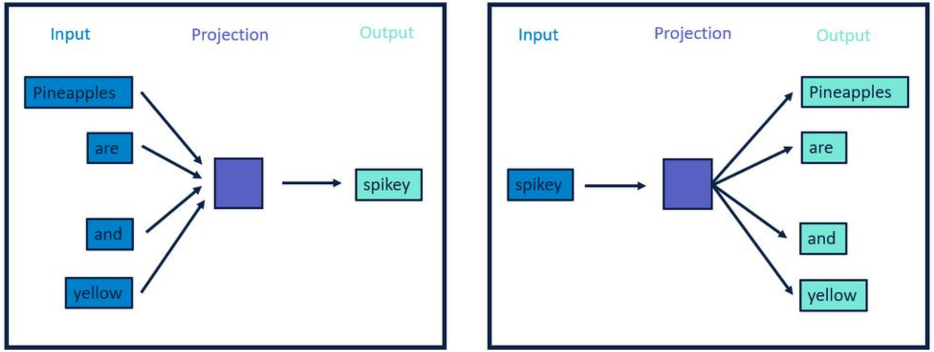


Word vectors in a 2d plane

Word Embedding methods

- Continuous Bag of Words (CBOW)
    - Context vector to word
    - Uses weighted sum
- Skip-Gram
    - Word vector to context vectors
- Argmax over the probability distribution to get the predicted word.





CBOW Skip-gram

Neural Language Model (NLM)

- Hill et al., TACL, 2016 (from Yoshua Bengio's lab) first introduced the problem of reverse dictionary in this paper.
- Input sentence is mapped to representation of a word which defines its meaning
- Used LSTM model M to get context vector *M(s)*
- Train the models *M* to map the input definition phrase s_c of word *c* to a location close to the the pre-trained embedding emb_w of target word.
    - reduce the cosine distance between *M(s)* and emb_w
- Finally, return words whose embeddings are closest to the input query representation.

Issues

- Many words are of low-frequency (Zipf's Law) and hence have a poor embeddings,
- As, the the objective is to bring the representation of $M(s_c)$ and emb_w closer, words that have poor embedding will affect the overall performance.

Observations

- RNN models do not significantly outperform the BOW models -> BOW models recover concepts from descriptions strikingly well, even when the words in the description are permuted.

- Pre-training input embeddings seems to help most on the concept descriptions, which are furthest from the training data in terms of linguistic style.

- OneLook performs better on seen words at the cost of a greater memory footprint, since the model requires access to its database of dictionaries at query time.

- While NLMs seem to generalise more consistently than OneLook on unseen dataset.

Multi-channel Reverse Dictionary Model

- Zhang et al., AAAI 2020 proposed a model inspired by how humans tries to infer the word from a given description.
  - For e.g. we unknowingly consider the category, entity type and even pos tags.
- Having knowledge of these characteristics makes it easier to search the target word.
- Morpheme is smallest meaningful unit of word
  - “unbreakable” -> {”un”, ”break”, ”able”}
- Sememes represents meaning of a word at atomic/morpheme level
  - for e.g. morpheme ”un” means ”not”
- Apart from original task of predicting target word from phrase, they created 4 other tasks using 2 channels:
  - Internal channels - for predicting pos tag and morpheme of words in input phrase
  - External channel - for predicting word category and sememe of words input phrase

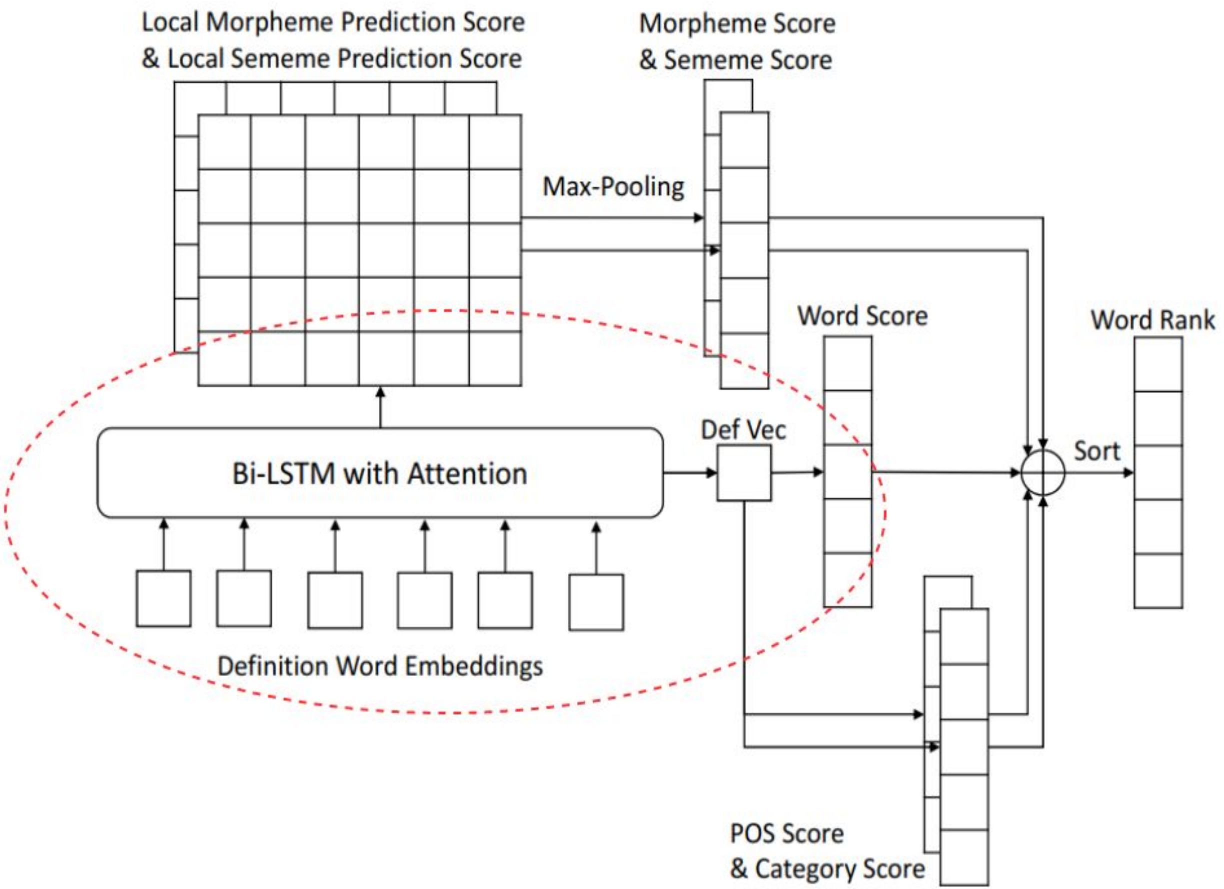
Multi-channel Reverse Dictionary Model

- Learning to predict these characteristics have several advantages:
    - POS tags helps to identify where the target word will be a verb, noun etc.
    - Category helps model the ability to remove semantically similar target words which are not from the same category as input.
        - “Voice” and “hear” are closely related in terms of semantic but belong to different categories.
    - Learning to predict morpheme and sememe helps the model to learn target word’s compositional information i.e. units that form the target word.

- Finally, combine confidence/probability scores from all predictors i.e. direct word prediction and 4 indirect characteristic prediction to make final prediction.

Multi-channel Reverse Dictionary Model

- Main task along with 4 additional task
  - Pos tag prediction
  - Category prediction
  - Morpheme prediction
  - Sememe prediction
- Dotted ellipse is NLM model with only main task.

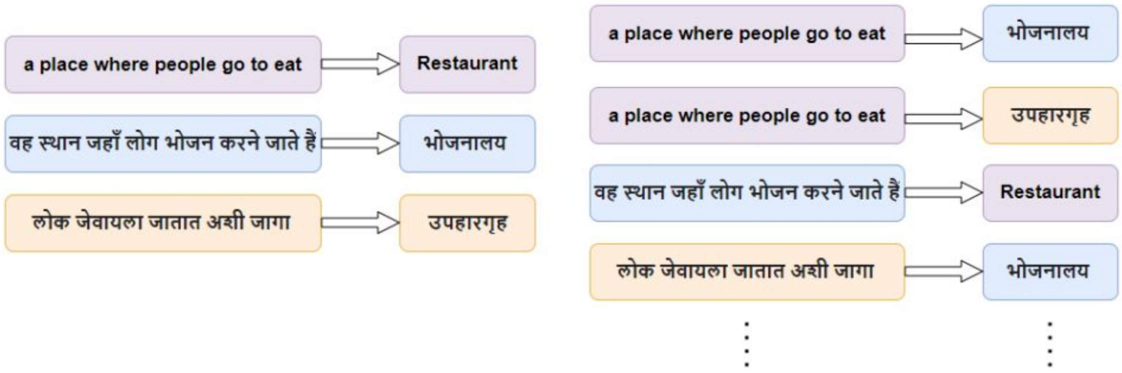


Our Problem Statement

- Humans are quite good at solving the reverse dictionary problem due to understanding of the world around them.

- For a model to solve reverse dictionary problem:
  - Learn representations of input phrases and words
  - Learn a model that can map these representations by bringing them closer in embedding space.

- Multilingual setting
- Cross-lingual setting



Future Work and Challenges

- Solving the problem in multilingual and cross-lingual setting, requires collection of parallel data - not currently available:
    - Multilingual: {phrase_L1, target_word_L1}, {phrase_L2, target_word_l2} ...
    - Cross-lingual: {phrasel1, target_wordl2}, {phrasel2, target_wordl3} …

- For *n* number of languages:
    - We have n*n models representing each pair
        - For e.g. [eng -> eng], [eng ->hi], [eng ->mr], [hi ->hi] [hi ->eng], [hi ->mr], [mr -> mr] …
    - Computationally expensive train these number of models and then save them
- To decrease number of sub-models, we need to look into multi-lingual word embeddings and transformer models.

Proposed Solution

- Hill et al. proposed a simple variation of this in a bilingual setting (English/French)
  - Replace the target monolingual embeddings with those from a bilingual embedding space (here English+French embeddings).
  - Train the model on Monolingual task i.e. English Phrase -> English Concept
  - At the inference time, return the the nearest French word neighbours for the given definition.
  - Results will depend on quality of bilingual embedding.

- Using Multilingual Embeddings like can be even more helpful
  - MUSE - 157 languages, fastText trained on Common Crawl and Wikipedia.
  - Multilingual BERT (mBERT) - 104 languages, BERT trained on Wikipedia content with a shared vocabulary

Thank you !!