Utility of General and Specific Word Embeddings for Classifying Translational Stages of Research.

S39: Oral Presentation - Information Extraction and Classification

Vincent Major, Alisa Surkis,
Yindalon Aphinyanaphongs

NYU Langone Health

Disclosure

I and my spouse/partner have no relevant relationships with commercial interests to disclose.

Links and things

This project:

github.com/vincentmajor/ctsa_prediction (code)
arxiv.org/abs/1705.06262 (preprint)
zenodo.org/record/802965 (data and embeddings)

Me:

@vincentjmajor
vincent.major@nyulangone.org
github.com/vincentmajor
linkedin.com/in/vincentmajor/

Overview

- Two open questions about word embeddings for practical tasks:
  - Which documents, with what properties, lead to broadly functional embeddings?
  - Are ‘good’ embeddings broadly useful for specific classification tasks?
- We cannot answer these questions conclusively but, we
  - present one use-case classification task where
    - custom word embeddings are feasible and
    - do **not** drastically improve performance over either
      - generic embeddings or
      - benchmark models.

Background — Bag-of-words

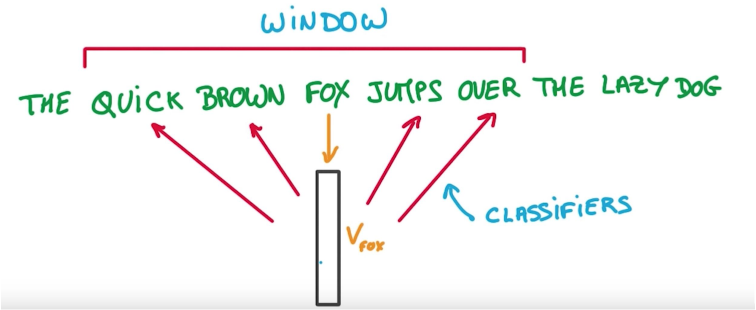
- Conventional text classification models make a ‘bag-of-words’ assumption that reduces text (fundamentally, a sequence of words) with a one-hot encoding into a binary vector.

- For small datasets, the vocabulary is going to be missing a lot of words.
    - If a word is not observed, it simply cannot be used to predict.
- But we know that words have meanings and have synonyms.
    - How can we know which words are semantically similar?
    - Can we use other data to learn semantics?
        - Spoiler: yes.

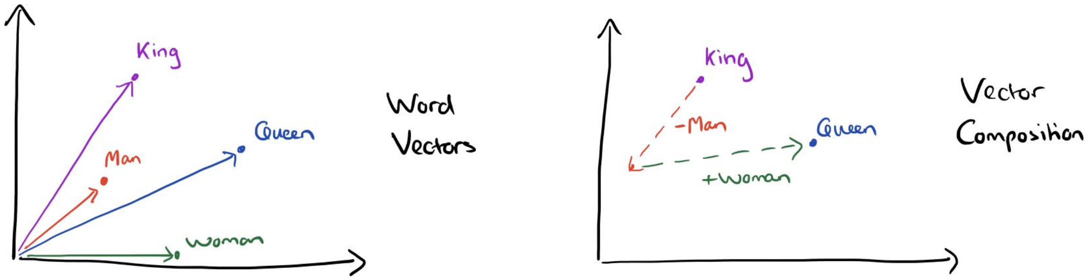
Udacity, https://www.youtube.com/watch?v=xMwx2A_o5r4

Background — word2vec

A recent algorithm, word2vec[1,2], learns a vector representation of words using a contextual window.



Udacity, https://www.youtube.com/watch?v=xMwx2A_o5r4



Adrian Colyer, https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/

- In this vector space, semantically similar words are close together.
  - Arithmetic: king - man + woman = queen
  - Capitals of countries, products of companies etc.
- Possible in any ‘large’ corpus
  - News articles and Wikipedia have been used successfully

1. Mikolov T et al. Efficient estimation of word representations in vector space. ICLR, 2013.
2. Mikolov T et al. Distributed representations of words and phrases and their compositionality. NIPS, 2013.

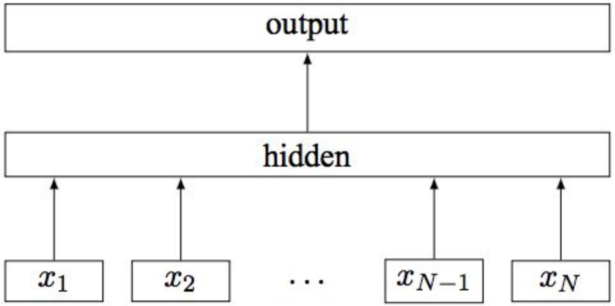
Background — is all text equal?

- We can learn text semantics in any ‘large’ corpus.
  - News articles and Wikipedia have been used and shared online.
  - But, does the source and style of these corpora influence the utility of their embeddings?
    - Can news translate to biomedical articles?

- word2vec is becoming very popular in all domains but few biomedical studies have compared methods to construct useful embeddings.
  - We decided to compare word2vec embeddings in an external text classification task.
    - To do so, we use another tool fastText to train classifiers using pretrained word2vec embeddings

Background — fastText

- To train a model on word2vec embeddings, we can use fastText[3].
  - Input is text, uses word embeddings to vectorize text, then averages words to form a vector text representation, which is then used in a linear classifier.
  - Output is a probability distribution over the predefined classes.



3. Joulin A et al. Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759, 2016.

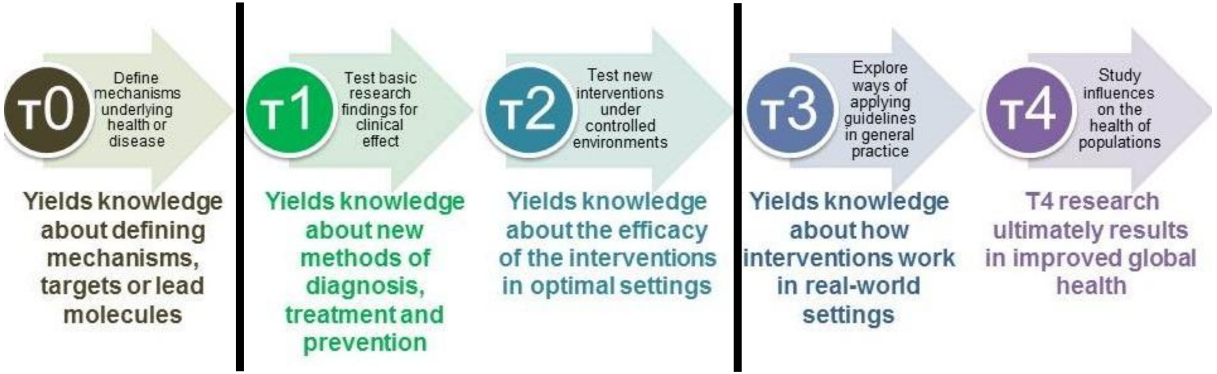
Methods

- Models:
  - Benchmark models
  - Word2vec
    - Download some off-the-shelf embeddings
    - Custom embeddings
      - (Aside: parameter optimization)

- Train and test in 5 fold cross-validation
  - With identical folds across models
  - Compare performance (mean AUCs for each class)

Use-case and labeled data

- Our use-case expands on previous work[4] using manually labeled biomedical research abstracts.
    - Fundamental biosciences, through preclinical and human studies, and into population level studies.
    - N = 542 broken down into T0 (n = 281), T1/T2 (n = 109), and T3/T4 (n = 152).



4. Surkis A et al. Classifying publications from the clinical and translational science award program along the translational research spectrum: a machine learning approach. J Transl Med 2016; 14: 235.

Results — *Benchmarks*

- Labeled data only with all of the typical limitations
- Average [min, max] AUC over 5-folds:

Model	**AUC T0** (n = 281)	**AUC T1T2** (n = 109)	**AUC T3T4** (n = 152)
Naive Bayes	95.0% [93.2, 96.8]	81.6% [72.4, 86.6]	90.6% [84.4, 93.9]
Support Vector Machines	94.8% [93.1, 96.9]	73.2% [67.8, 82.4]	87.3% [78.8, 91.7]
Random Forest	94.8% [92.4, 96.7]	**86.1%** [79.1, 91.0]	88.5% [84.1, 90.9]
Bayesian Logistic Regression	**96.1%** [95.6, 96.9]	85.7% [81.0, 91.0]	**92.6%** [85.5, 96.5]

- Definitely room for improvement in T1T2 and T3T4.

(SVM was consistently worse in the second group.)

Results — *Off-the-shelf*

- We found five sets of word2vec embeddings.
  - Three general text corpora utilizing news articles (2) or Wikipedia (1)
  - One combining Pubmed and PubMed Central.
  - Another adding Wikipedia to PubMed and PubMed Central.
- The exact details of the corpora, their preprocessing and construction of vectors are often glossed over.
  - But, they have all been validated in some internal task and released supporting a paper.

Results — *Off-the-shelf*

Model	AUC T0	AUC T1/T2	AUC T3/T$
Bayesian Logistic Regression	96.1% [95.6, 96.9]	85.7% [81.0, 91.0]	92.6% [85.5, 96.5]

- Average [min, max] AUC over 5-folds:

Name	Data source(s)	Creator	Unique tokens	Model	Optimization	Dimensions	AUC T0 (n = 281)	AUC T1T2 (n = 109)	AUC T3T4 (n = 152)
Freebase	Freebase	*word2vec* 5	1.4 M	skip-gram		1000	94.1% [93.4, 96.0]	81.2% [78.2, 86.3]	86.3% [83.2, 89.7]
Google News	Google news	*word2vec* 6	3.0 M	CBOW	negative sampling	300	94.7% [94.0 ,96.1]	85.9% [83.9, 87.2]	91.3% [87.4, 95.3]
PubMed	PubMed+PMC	BioNLP 5,14,15	4.1 M	skip-gram	hierarchical softmax	200	94.6% [93.7, 96.0]	86.0% [83.9, 87.2]	91.1% [88.2, 95.1]
PubMed+ Wiki	PubMed+PMC + Wikipedia	BioNLP 5,14,15	5.4 M	skip-gram	hierarchical softmax	200	94.6% [93.7, 96.2]	86.4% [83.5, 88.1]	91.1% [88.2, 94.2]
Wiki	English Wikipedia	*fastText* *14*	2.5 M	*fastText* skip-	negative sampling	300	**95.5%** [94.1, 96.4]	**88.1%** [85.3, 91.1]	**92.2%** [90.0, 94.7]

1. Mikolov T, Chen K, Corrado G, Dean J. *Efficient estimation of word representations in vector space*. ICLR, 2013.
2. Pyysalo S, Ginter F, Moen H, Salakoski T, Ananiadou S. *Distributional Semantics Resources for Biomedical Text Processing*. LBM 2013.
3. Bojanowski P, Grave E, Joulin A and Mikolov T. *Enriching Word Vectors with Subword Information*. arXiv preprint arXiv:1607.04606, 2016.

Methods — Unlabeled Data

- To learn embeddings, you need a lot of data.
  - We used Pubmed articles instead:
    - download all of Medline/PubMed with complete title and abstracts,
    - select articles published Jan 2000 - Dec 2016 (10.5M articles),
    - concatenate titles and abstracts, remove all punctuation (1.67B words, 822k unique words), and
    - use word2vec to learn several sets of embeddings (with different parameters).

Results — Comparison

- Comparing the best benchmark, with the best off-the-shelf and the custom models.

	Name	Model	AUC T0 (n = 281)	AUC T1/T2 (n = 109)	AUC T3/T4 (n = 152)	Cost	Runtime
The best benchmark	BLR	Bayesian Logistic Regression	**96.1%** [95.6, 96.9]	85.7% [81.0, 91.0]	92.6% [85.5, 96.5]	cheap	< 5 minutes
The best off-the-shelf	Wiki	*fastText* skip-gram	95.5% [94.1, 96.4]	88.1% [85.3, 91.1]	92.2% [90.0, 94.7]	cheap	< 5 minutes
custom	CBOW	CBOW	94.2% [91.8, 95.9]	87.6% [82.4, 91.9]	90.2% [87.7, 93.7]	expensive	~ 2 hours (28 cores)
	Skip	skip-gram	95.5% [94.0, 96.3]	**88.6%** [84.7, 91.9]	**92.8%** [89.6, 95.4]	very expensive	~ 9 hours (28 cores)
	fastText	skip-gram	95.4% [93.9, 96.5]	88.1% [84.1, 92.0]	92.7% [89.9, 95.6]	very expensive	~ 10 hours (28 cores)

- Is the extra effort worth it? *Probably not*

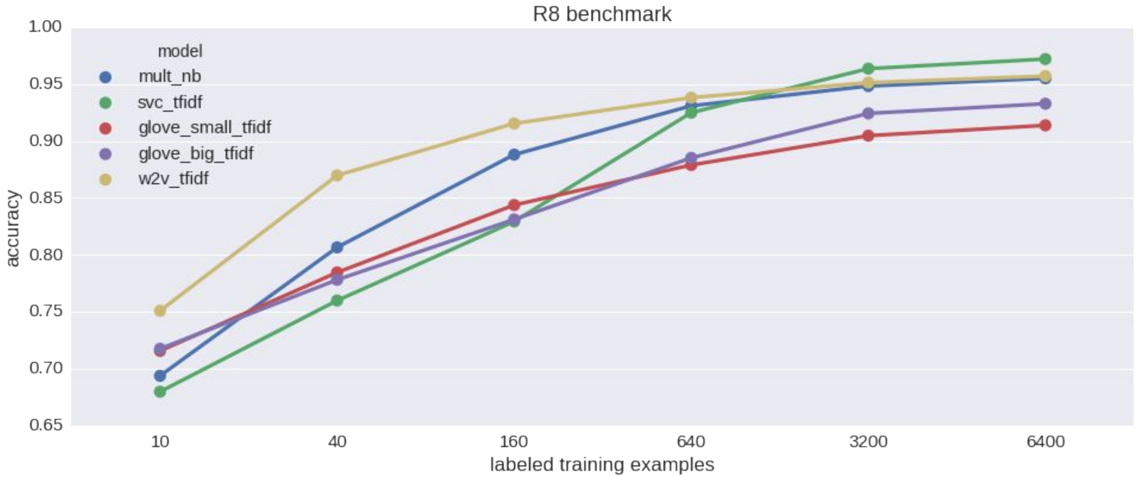
Discussion

	Model	AUC T0 (n = 281)	AUC T1/T2 (n = 109)	AUC T3/T4 (n = 152)
benchmark	Bayesian Logistic Regression	**96.1%** [95.6, 96.9]	85.7% [81.0, 91.0]	92.6% [85.5, 96.5]
off-the-shelf	Wiki	95.5% [94.1, 96.4]	88.1% [85.3, 91.1]	92.2% [90.0, 94.7]
custom	Word2vec Skip-gram	95.5% [94.0, 96.3]	**88.6%** [84.7, 91.9]	**92.8%** [89.6, 95.4]

- The bag-of-words model are good.
  - Worst in the small classes.
  - Optimization could improve.
- Off-the-shelf embeddings are surprisingly good.
  - Even those learnt on news data.
- Custom embeddings are expensive.
  - Parameter optimization even more so.
- But, the word2vec models aren't drastically better than the bag-of-words models.
  - Abstracts may be too terse for semantics to shine.
  - Interestingly, BLR > word2vec when the number of cases in the class is higher.

Discussion — Closing the gap

- Word embeddings bridge the gap for small datasets by collapsing similar words into similar vectors.
    - But by doing so, every word is minorly related to all others it introduces noise
    - For sufficiently large datasets, the benefits of vector representation diminish.
    - Example for a benchmark 8-class task:



1. Nadbor Drozd. nadbordrozd.github.io/blog/2016/05/20/text-classification-with-word2vec/

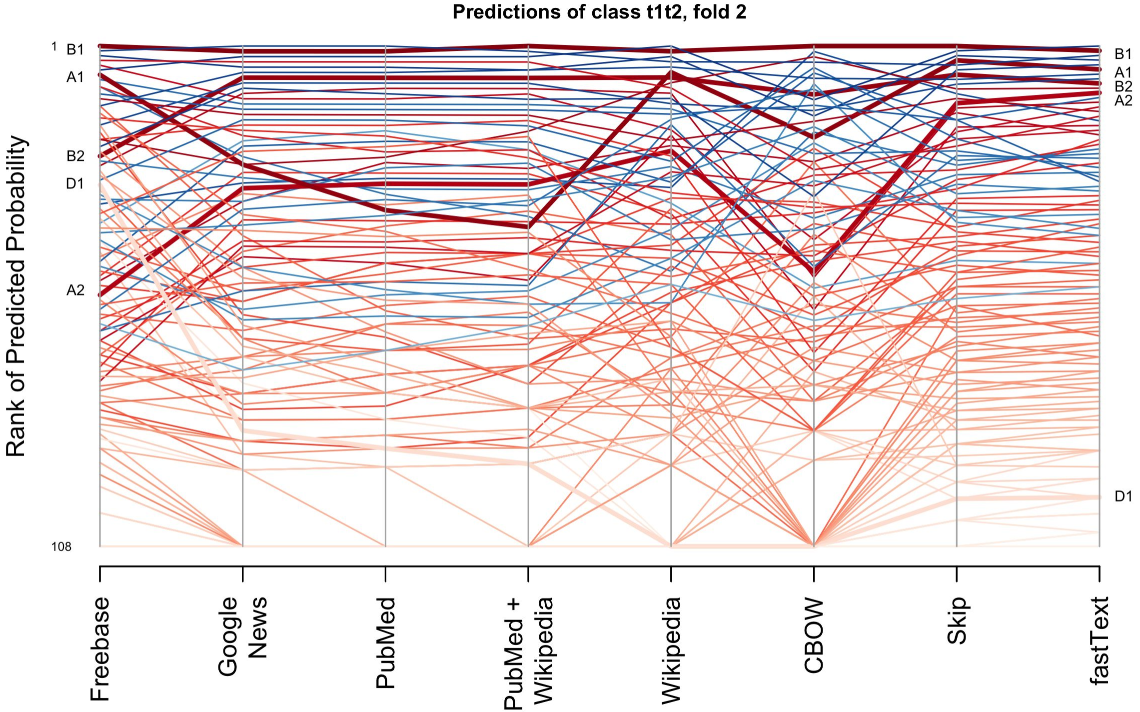
Error Analysis

- We have 8 different sets of word embeddings that:
  - ***should*** represent semantics of natural language, but
  - they are different.
- How do these hidden differences influence predictions for individual articles?
  - Within one experimental fold, rank all (n~108) articles and compare one articles rank across models.

Error Analysis (class t1t2)

blue = correct, red = wrong
Some highlighted and labeled

- A1+A2: are all very variable.
  - Specific language
- B1+B2: consistently high.
  - Secondary use of data.
- D1: high only in Freebase.
  - Adoption of EHRs and meaningful use



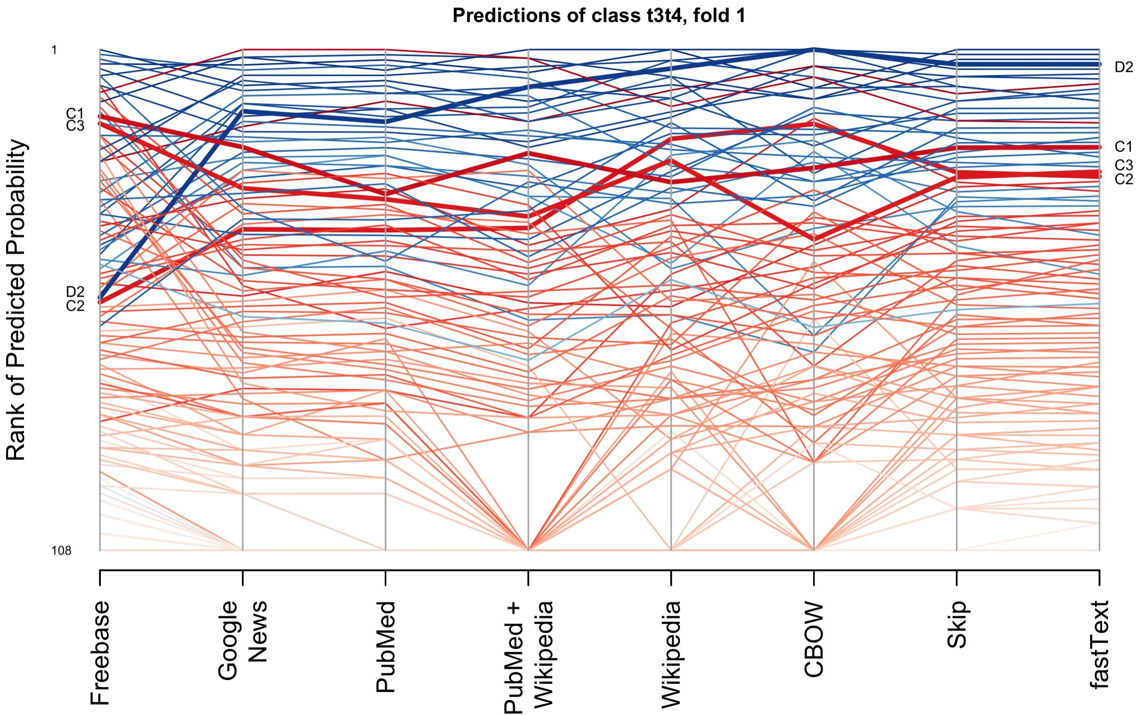
(More detail on these instances in the paper)

Error Analysis (class t3t4)

blue = correct, red = wrong
Some highlighted and labeled

- D2: mostly high except in Freebase.
    - Database of nurse surveys
- C1+C2+C3: quite consistent but wrong.
    - Small studies but describe specific populations, and conduct chart review.

(More detail on these instances in the paper)



Conclusion

- Text is discrete, but can be embedded into a continuous space.
  - Similar words should overlap in space and allow a model to learn meaning rather than individual words.
  - The embeddings are learnt unsupervised and may learn incorrect associations.
- In a 3-class prediction task using article titles and abstracts:
  - Bayesian logistic regression works well.
  - word2vec parameter optimization results are consistent with literature.
  - Off-the-shelf embeddings are comparable to custom embeddings and BLR.
    - Some article characteristics are consistently misclassified, others are more variable.
  - Improvement in classification performance over BLR is observed for the smallest class and the converse for the larger.
    - Abstracts may be too terse.
- word2vec has great potential when only a small amount of data is labeled
  - But it's improvements likely diminish with sample size increases.

Thank you!

Email me at: vincent.major@nyulangone.org

@vincentjmajor