Современные дистрибутивно-семантические модели и их применение в лингвистических исследованиях
День 2

Андрей Кутузов
Университет Осло
Группа лингвистических технологий

Школа компьютерной лингвистики
Тюмень, февраль 2018

Содержание

Векторные представления документов

- Дистрибутивный подход позволяет извлекать семантику из неразмеченных данных на уровне слов.
- Но хотелось бы работать и с текстами разной длины!
  - для классификации,
  - для кластеризации,
  - для информационного поиска (включая веб-поиск).

Векторные представления документов

- Можно ли найти семантические похожие тексты так же, как мы находим похожие слова?
- Можно!
- Мы точно так же можем представить предложения, абзацы или целые документы в виде сжатых векторов.
- После того, как документы стали векторами, классификация, кластеризация и т.п. уже не представляют проблемы.

Можно ли обойтись без сложных моделей?

Мешок слов (bag-of-words) с TF-IDF

Очень эффективный базовый подход (baseline), часто не уступающий современным методам:

1. Извлекаем словарь V всех слов (термов) в обучающем корпусе, состоящем из N документов;
2. Для каждого терма вычисляем его документную частоту (document frequency): в скольки документах он встречается (df);
3. Представляем каждый документ в виде разреженного вектора частот всех термов из V в нём (tf);
4. Для каждого значения вычисляем взвешенную частоту wf используя term frequency / inverted document frequency (TF-IDF):
   - $wf = (1 + log_{10} tf) \times log_{10}(\frac{N}{df})$
5. Используем эти взвешенные вектора документов в любых практических задачах.

Векторные представления документов

Проблемы мешка слов

К сожалению, такой подход не учитывает семантические отношения между лингвистическими сущностями.
Он не позволяет найти семантическую близость между документами, у которых нет общих слов:

- ‘После выборов в Калифорнии прошли массовые протесты.’
- ‘Многие американцы были недовольны избранным президентом.’

Следовательно, нам нужны более продвинутые семантические методы, например дистрибутивные эмбеддинги.

Содержание

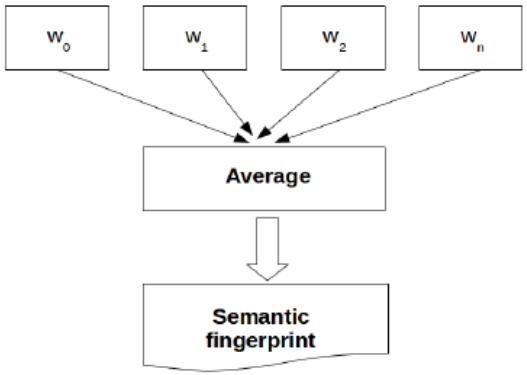
Дистрибутивные модели: композиция векторов

- Значение документа является композицией значений индивидуальных слов.
- Нам нужно как-то слить непрерывные вектора слов в непрерывные вектора документов.
- Способы это сделать называются функциями композиции.

Семантические отпечатки (semantic fingerprints)

- Одна из самых простых функций композиции: усредненный вектор $\vec{S}$ по всем словам $w_0 ... n$ в документе.
- Порядок слов и синтаксис при этом не учитывается.
- Если уже есть хорошая модель для векторов слов, такой подход очень эффективен и обычно работает лучше, чем мешок слов.
- Назовем такой средний вектор 'семантическим отпечатком' документа.
- Важно перед этим удалить стоп-слов (предлоги, союзы, прочие служебные части речи)!

Дистрибутивные модели: композиция векторов



$$\vec{S} = \frac{1}{n} \times \sum_{i=0}^{n} \vec{w_n} \tag{1}$$

- Не обязательно даже вычислять среднее. Можно просто сложить вектора: косинусная близость учитывает только углы, а не длины векторов.
- Но в случае, если мы используем другие метрики близости (например, Евклидово расстояние и т.п.), усреднение будет изменять результат.
- Кроме того, после усреднения документные вектора перестают зависеть от размера самих документов.

Дистрибутивные модели: композиция векторов

Преимущества семантических отпечатков

- Семантические отпечатки работают быстро и используют уже обученные модели.
- Обобщенные репрезентации документов не зависят от отдельных слов.
- Опираются на ‘семантические признаки’, которые модель нашла при обучении.
- Тематически связанные слова совместно усиливают или ослабляют выраженность соответствующих семантических компонентов.
- Таким образом, тематические слова автоматически становятся более важными, чем случайные, мусорные слова.

Подробности в [Kutuzov et al., 2016].

Дистрибутивные модели: композиция векторов

Но...

Однако, в некоторых случаях композиционных подходов недостаточно и нам нужно обучать настоящие вектора документов.

Но как?

Содержание

Дистрибутивные модели: обучение векторов

Paragraph Vector

- [Le and Mikolov, 2014] предложили подход под названием Paragraph Vector;
- обычно применяется для векторов предложений;
- алгоритм принимает на вход предложения/документы помеченные (возможно, уникальными) идентификаторами;
- для каждого идентификатора выучивается дистрибутивная репрезентация, такая, что у похожих предложений оказываются похожие вектора;
- итак, каждое предложение представлено идентификатором и вектором, так же, как мы представляли слова;
- эти вектора служат чем-то вроде памяти о документе или темы документа.

Подход сравнительно новый, но его уже детально проанализировали и оценили в [Hill et al., 2016] и [Lau and Baldwin, 2016].

Дистрибутивные модели: обучение векторов

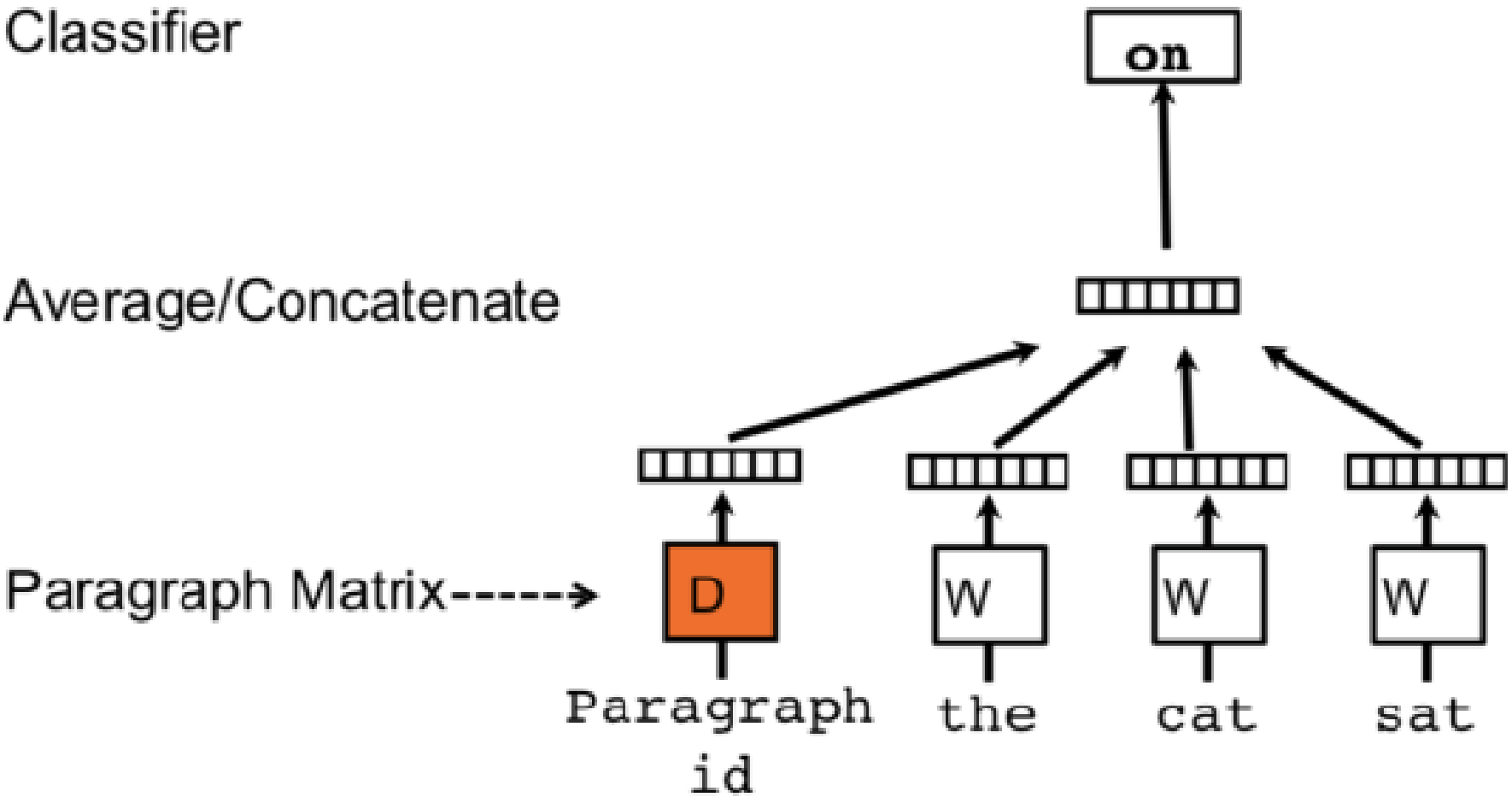
Paragraph Vector (он же doc2vec)

- Реализован в Gensim под названием doc2vec;
- Distributed memory (DM) и Distributed Bag-of-words (DBOW);
- PV-DM:
  - обучает вектора слова как обычно (для всех документов сразу);
  - инициализирует случайные вектора для документов;
  - использует документные вектора и вектора слов чтобы предсказывать соседние слова в рамках заданного окна;
  - минимизирует ошибку, обучая таким образом вектора документов;
  - обученная модель может сгенерировать вектор для нового документа, который она не видела при обучении (сама модель при этом не меняется).
- PV-DBOW:
  - не использует скользящее окно;
  - при обучении просто пытается предсказать все слова в текущем документ, используя вектор документа.

Есть разные мнения о том, какой метод лучше.

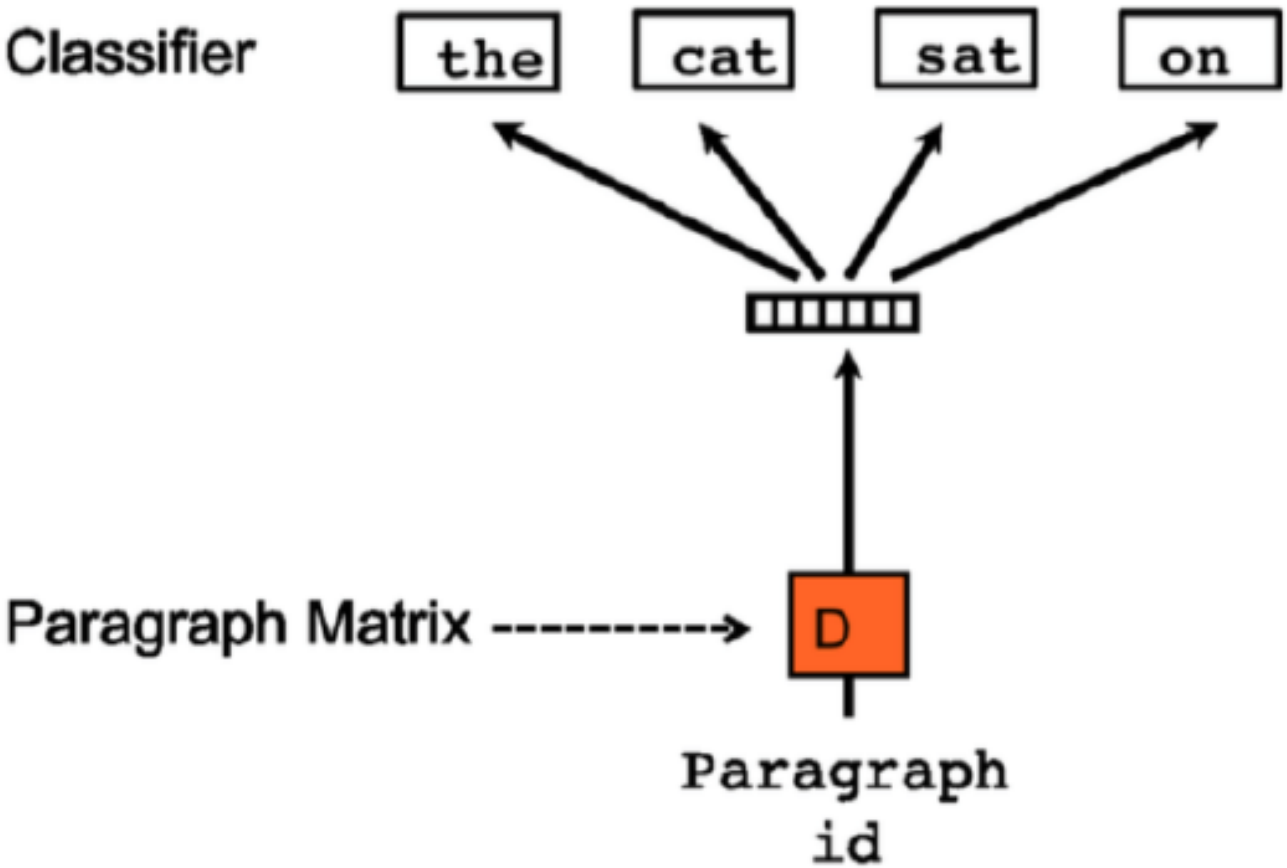
Дистрибутивные модели: обучение векторов

Paragraph Vector - Distributed memory (PV-DM)



Дистрибутивные модели: обучение векторов

Paragraph Vector - Distributed Bag-of-words (PV-DBOW)



Дистрибутивные модели: обучение векторов

Paragraph Vector (он же doc2vec)

- Обучаем модель, затем генерируем вектора для новых документов, которые мы хотим анализировать.
- Эти подходы показывают очень хорошие результаты в анализе тональности и в других задачах классификации документов, а также в информационном поиске.
- Очень требовательны к памяти (RAM): у каждого предложения свой вектор, а в реальных корпусах много миллионов предложений.
- Можно смягчить эти требования, если обучать поменьше векторов: например, группировать предложения в группы / классы и обучать вектора для классов.

Спасибо за внимание!
Вопросы?

Современные дистрибутивно-семантические модели
и их применение в лингвистических исследованиях
День 2

`http://rusvectores.org`

Андрей Кутузов (andreku@ifi.uio.no)
Language Technology Group
University of Oslo

References I

Hill, F., Cho, K., and Korhonen, A. (2016).
Learning distributed representations of sentences from unlabelled data.
In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1367–1377. Association for Computational Linguistics.

Kutuzov, A., Kopotev, M., Sviridenko, T., and Ivanova, L. (2016).
Clustering comparable corpora of Russian and Ukrainian academic texts: Word embeddings and semantic fingerprints.
In Ninth Workshop on Building and Using Comparable Corpora, page 3.

References II

Lau, J. H. and Baldwin, T. (2016).
An empirical evaluation of doc2vec with practical insights into document embedding generation.
In Proceedings of the 1st Workshop on Representation Learning for NLP, pages 78–86. Association for Computational Linguistics.

Le, Q. and Mikolov, T. (2014).
Distributed representations of sentences and documents.
In International Conference on Machine Learning, pages 1188–1196.