



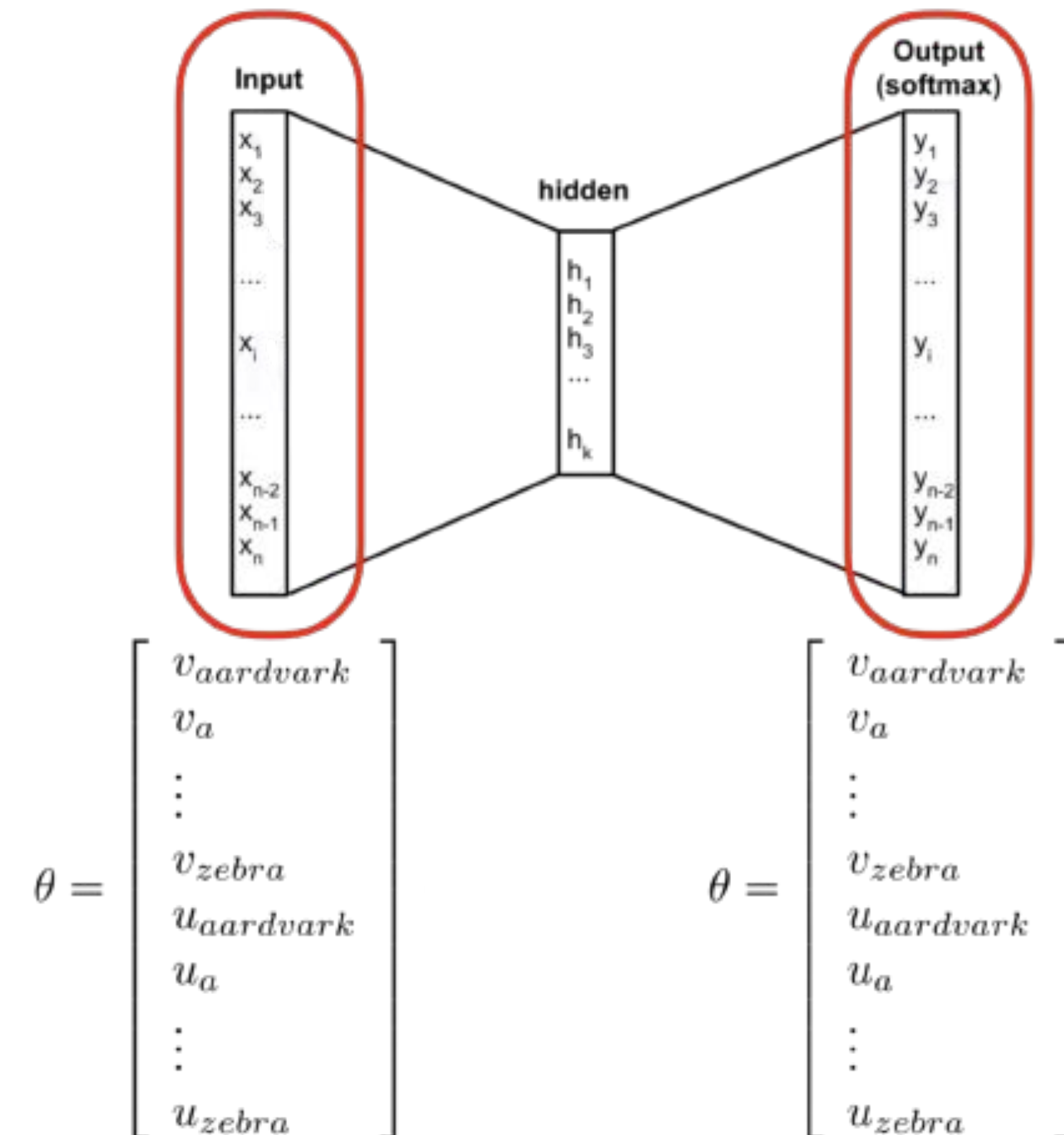
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Компьютерная лингвистика и информационные технологии

Неделя 9: Векторные представления, часть 2
(на основе лекций М. Апишева и Е. Артемовой, ФКН ВШЭ)

Window-based Vector Models

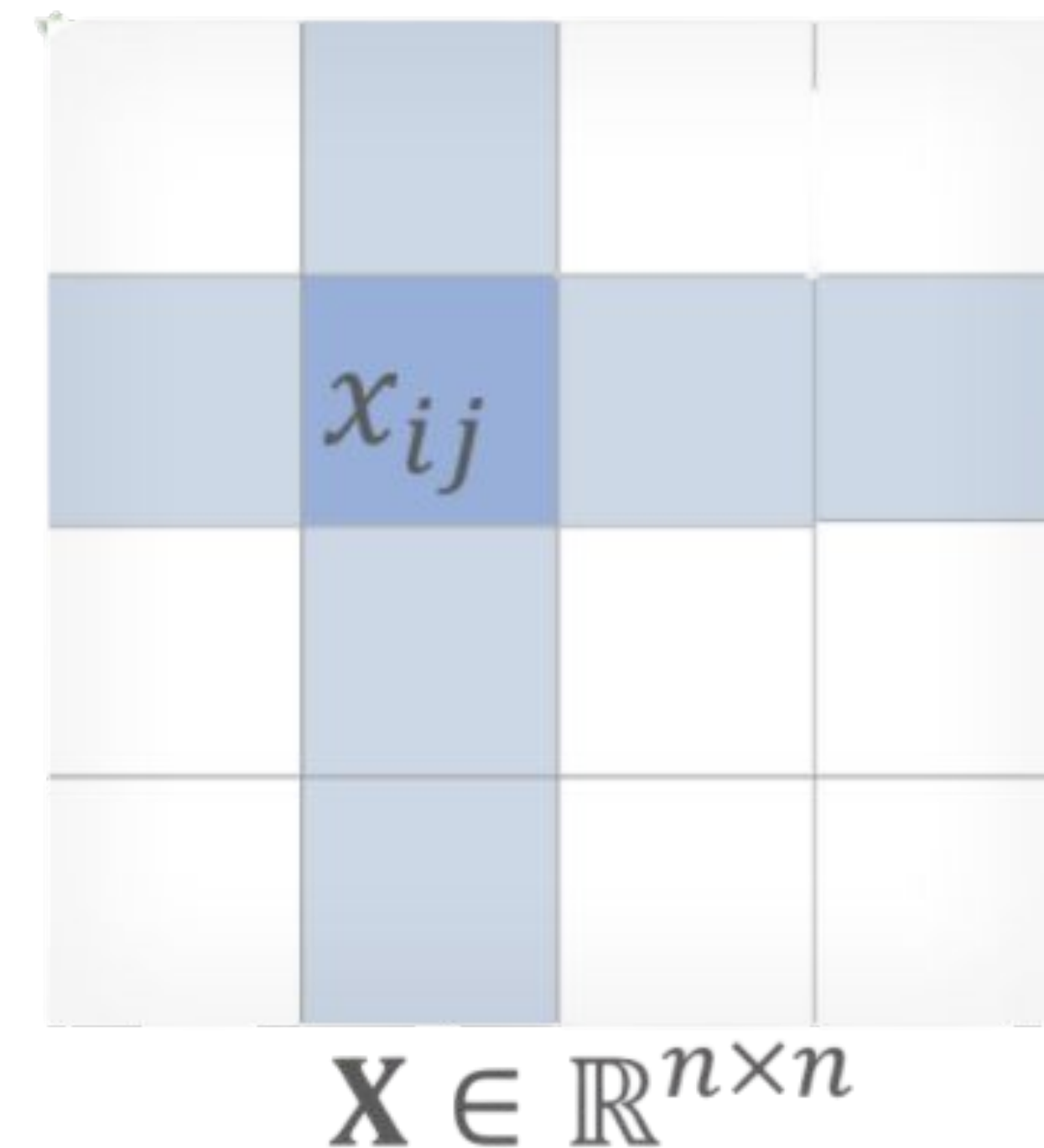
- Out-of-vocabulary слова (OOV)
- Scale with the pre-trained corpus size
- Только локальные контексты => не учитываются глобальные повторения в обучающих данных
- В явном виде не используется корпусная статистика
- Не кодируются морфологические особенности



GloVe

Global Vectors for Word Representation

- Можем ли мы закодировать значение слова с помощью счетчиков?
- Векторы кодируют как локальную, так и глобальную информацию
- Матрица совместной встречаемости по обучающему корпусу
- Решаем задачу регрессии: предсказываем счетчики


$$\mathbf{X} \in \mathbb{R}^{n \times n}$$



GloVe

- X : матрица совместной встречаемости слово-слово
- X_{ij} : счетчик, показывающий сколько раз слово j встретилось в контексте слова i
- $X_i = \sum_k X_{ik}$: сумма счетчиков, или сколько раз любое слово k встретилось в контексте слова i
- $P_{ij} = P(w_j|w_i) = X_{ij} / X_i$: вероятность встретить слово j в контексте слова i

	кошка	собака	единорог
кошка	0	3	4
собака	3	0	2
единорог	4	2	0

GloVe

- Пусть w_i – векторное представление слова i
- Имеем две матрицы: W и W'
- Зададим функцию, показывающую, какое из слов (i или j) вероятнее встретить в контексте слова k :

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}, \quad F((w_i - w_j)^T \tilde{w}_k) = \frac{P_{ik}}{P_{jk}},$$

- $w_i = \text{sum}(W_i, W'_i)$

	кошка	собака	единорог
кошка	0	3	0
собака	3	0	2
единорог	0	2	0



GloVe

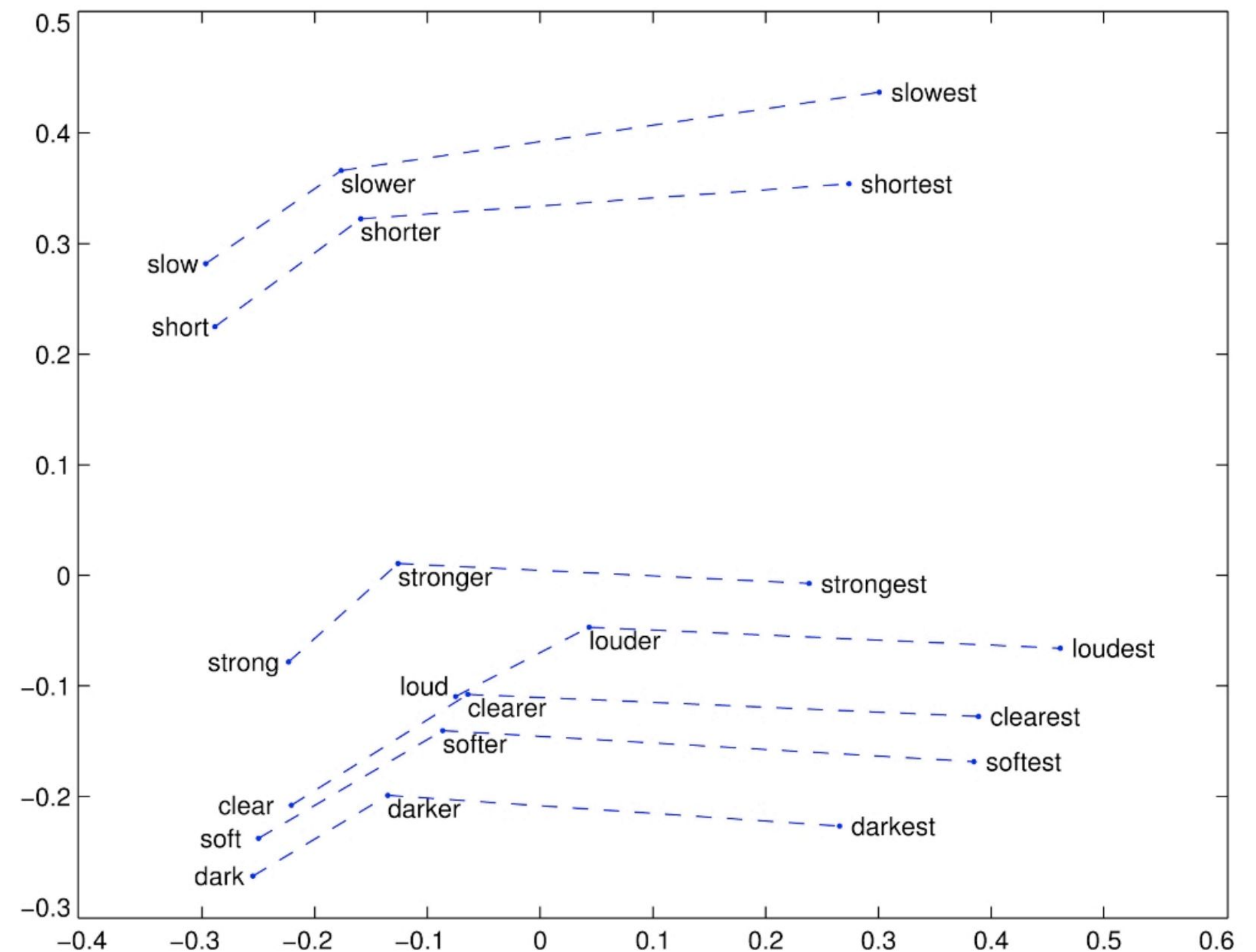
Target word: *frogs*

word2vec

amphibians
salamanders
tailed
birds
gulls
grieve

GloVe

frogs
toad
litoria
leptodactyl
lidaefrogs
toad





fastText

Enriching Word Vectors with Subword Information

- Слово как мешок посимвольных N-грамм
- Используем символы начала (<) и конца (>). Зачем?
- Обучим векторы для каждой N-граммы с помощью CBOW / Skip-gram
- Вектор слова есть среднее векторов N-грамм

N = 2: <к, ку, ук, кл, ла, а>

N = 3: <ку, кук, укл, кла, ла>

N = 4: <кук, кукл, укла, кла>

fastText

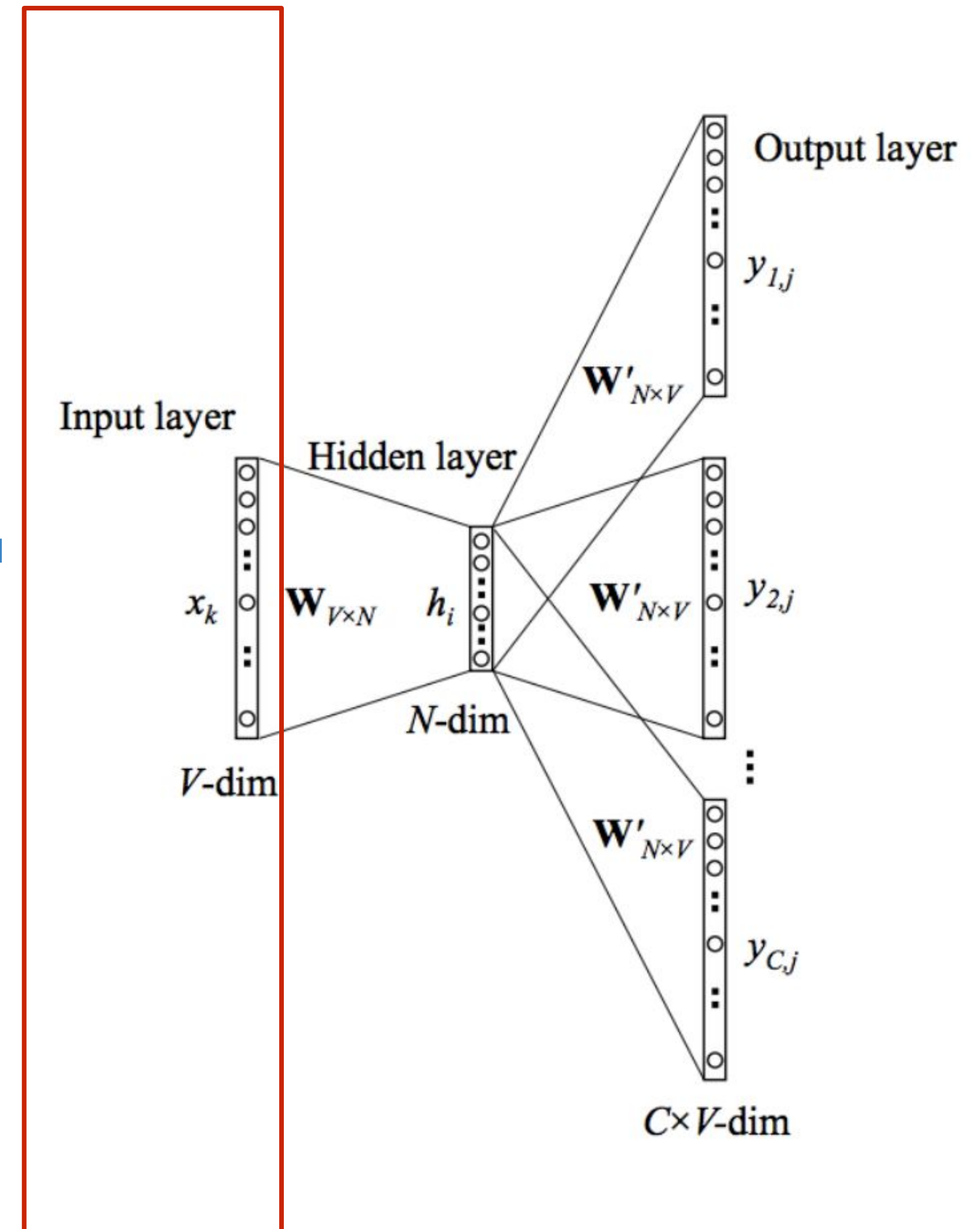
$$s(w, c) = \sum_{g \in \mathcal{G}_w} \mathbf{z}_g^\top \mathbf{v}_c.$$

- Окно может быть скользящим (дефолт 3-6)
- Огромная мощность словаря N-грамм
- Хэш-функция: сопоставление индексу слова множества посимвольных N-грамм
- Фиксированное количество частотных N-грамм

?
?
?
?

<М
мы
ыл
ла
>

?
?
?
?





Что мы знаем?

1	CBOW	<ul style="list-style-type: none">• вход: 2C one-hot векторов• одно слово – один вектор• out-of-vocabulary words• предсказываем слово по контексту
2	Skip-gram	<ul style="list-style-type: none">• вход: one-hot вектор центрального слова• одно слово – один вектор• out-of-vocabulary words• предсказываем контекст по слову
3	GloVe	<ul style="list-style-type: none">• вход: w_i, w_j, w_k• одно слово – один вектор• out-of-vocabulary words• предсказываем счетчики
4	fastText	<ul style="list-style-type: none">• вход: k векторов N-грамм• одно слово – агрегация векторов N-грамм• решается проблема out-of-vocabulary• конфигурации CBOW/Skip-gram



Оценка векторных представлений

Внутренняя (intrinsic)
Word Similarity
Syntactic analogies
Semantic analogies
Correlation between human evaluation and cosine similarity

Внешняя (extrinsic)
Downstream tasks
Предобученные эмбединги
Дообучение эмбедингов
NER, POS, Sentiment Analysis, etc.



Внутренняя оценка

- **Word analogies:** *Athens is to Greece as Berlin is to ???*
- **Word similarity:** WordSim-353, SCWS, RW, SimLex695
- **Syntactic similarity:** *bad -> worst – big -> ???*
- **Correlation:** $\text{corr}(\text{human_score}(x_i, y_i), \text{cos_score}(x_i, y_i))$

(hyper)parameters:

- Размерность;
- Объем корпуса;
- Домен корпуса;
- Специфика корпуса;
- Размер окна;
- Метод агрегации
промежуточных или
итоговых представлений



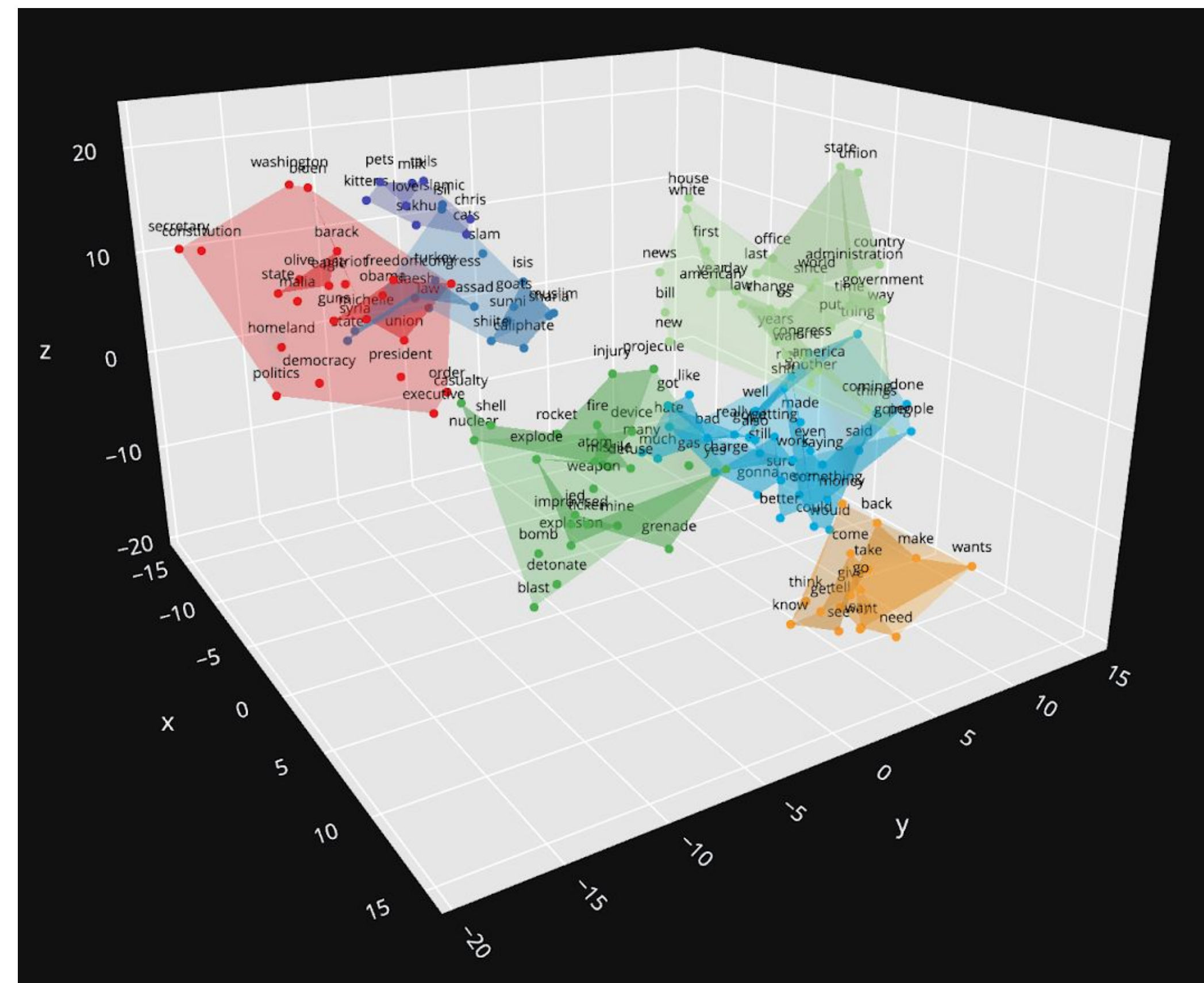
Внешняя оценка

- Имеем несколько предобученных векторных моделей
- Фиксируем параметры классификатора
- Качество классификатора косвенно отражает качество предобученных моделей
- Дообучение предобученных векторных моделей



Предобученные векторные модели

- [fastText](#)
- [RusVectors](#) (fastText, CBOW, Skip-gram)
- [DeepPavlov](#) (fastText)
- [navec](#) (compressed GloVe)





Bag of Tricks for Efficient Text Classification

(Joulin et al., 2016)

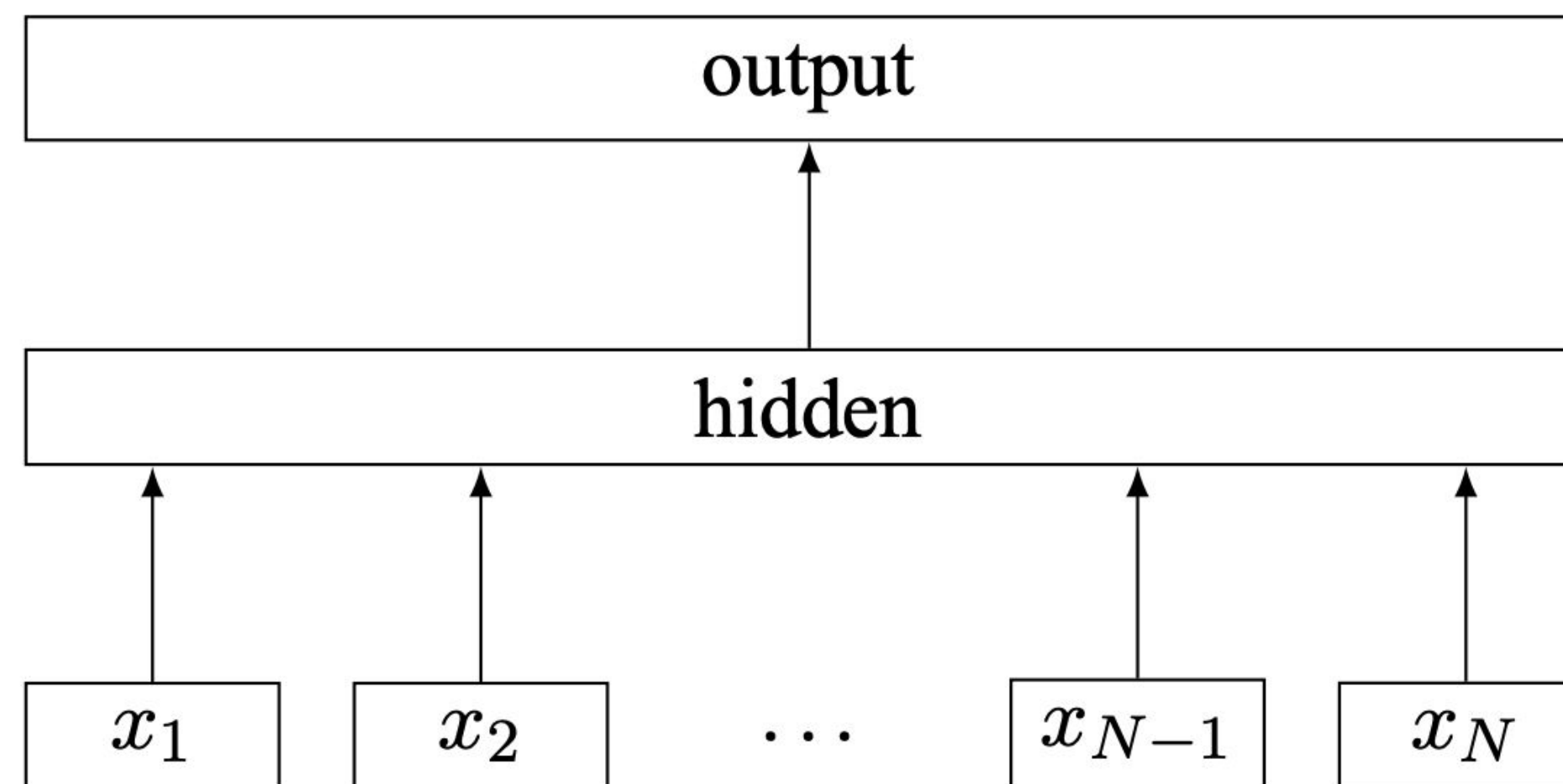


Figure 1: Model architecture of `fastText` for a sentence with N ngram features x_1, \dots, x_N . The features are embedded and averaged to form the hidden variable.



Полезные материалы

- [CS224N: GloVe](#)
- [Pennington et al., 2014: GloVe](#)
- [Bojanowski et al., 2017: fastText](#)
- [fastText-based text classification](#)
- [Joulin et al., 2017: Bag of Tricks for Efficient Text Classification](#)