PROBABILISTIC FASTTEXT FOR MULTI-SENSE WORD EMBEDDINGS

BEN ATHIWARATKUN, ANDREW GORDON WILSON,
ANIMA ANANDKUMAR

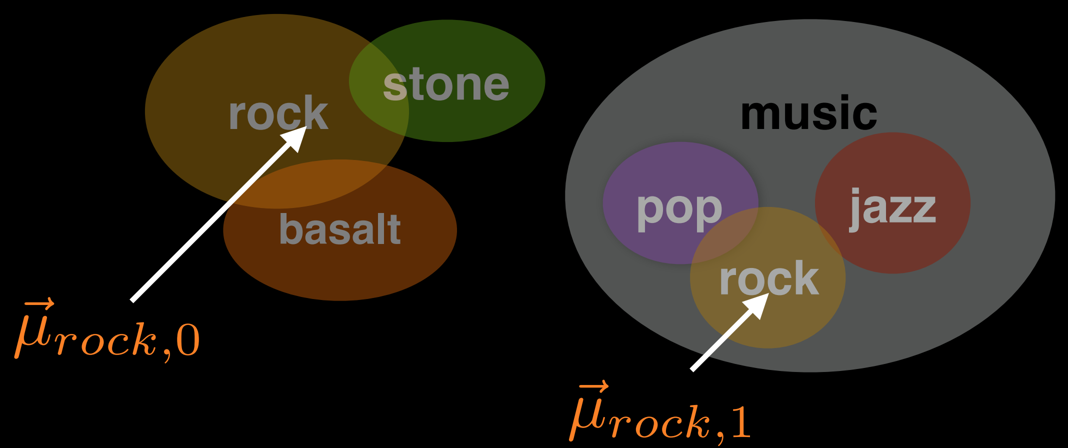
Cornell University

amazon web services

2-MIN SUMMARY

Probabilistic FastText = FastText + Gaussian Mixture Embeddings

Gaussian Mixture Embeddings

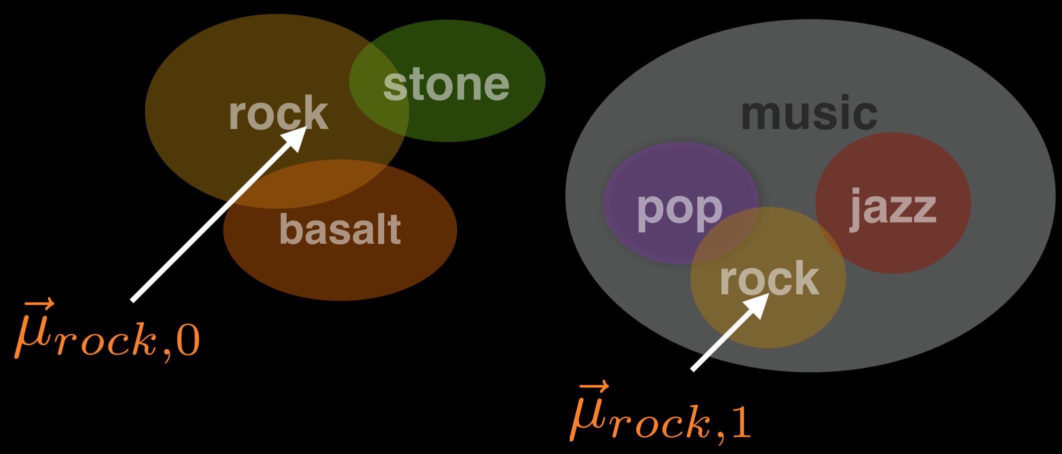


- Words as probability densities
- Each word = Gaussian Mixture density
- Disentangled meanings

2-MIN SUMMARY

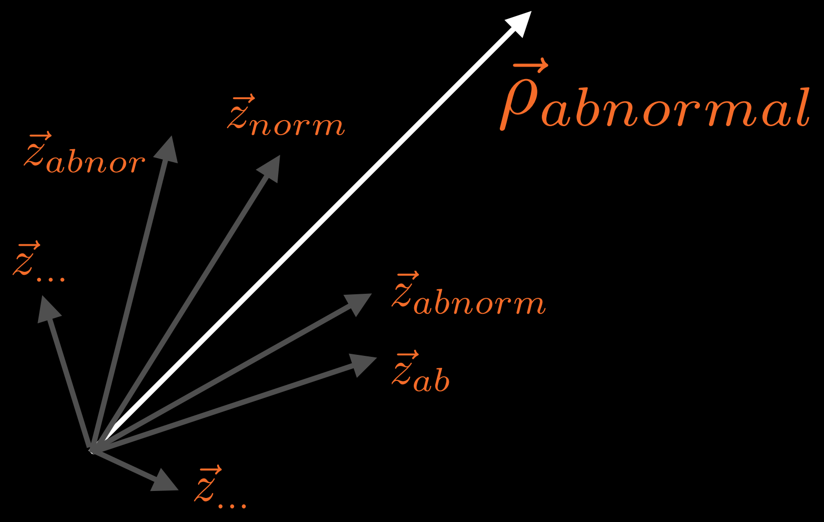
Probabilistic FastText = FastText + Gaussian Mixture Embeddings

Gaussian Mixture Embeddings



- Words as probability densities
- Each word = Gaussian Mixture density
- Disentangled meanings

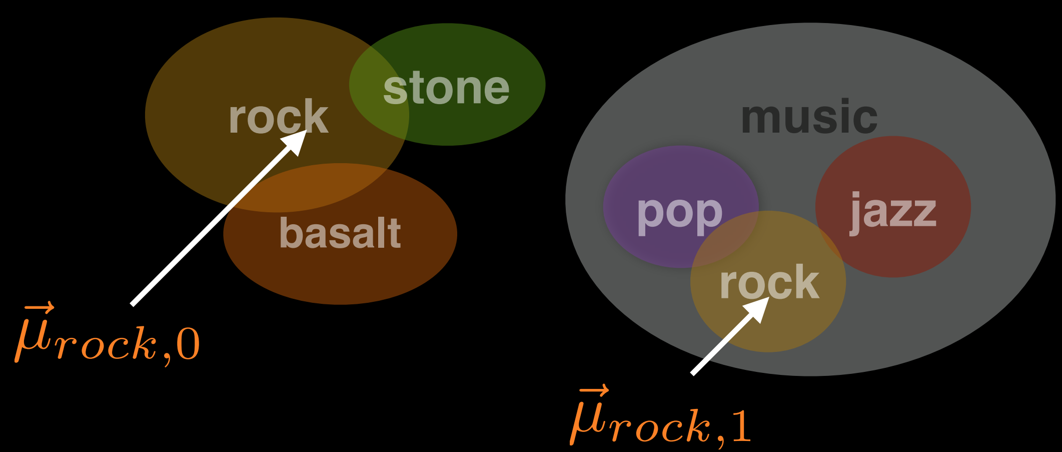
FastText



- Word embeddings: word vectors are derived from subword vectors
- SoA on many benchmarks especially RareWord
- Character based models allow for estimating vectors of unseen words and enhancing

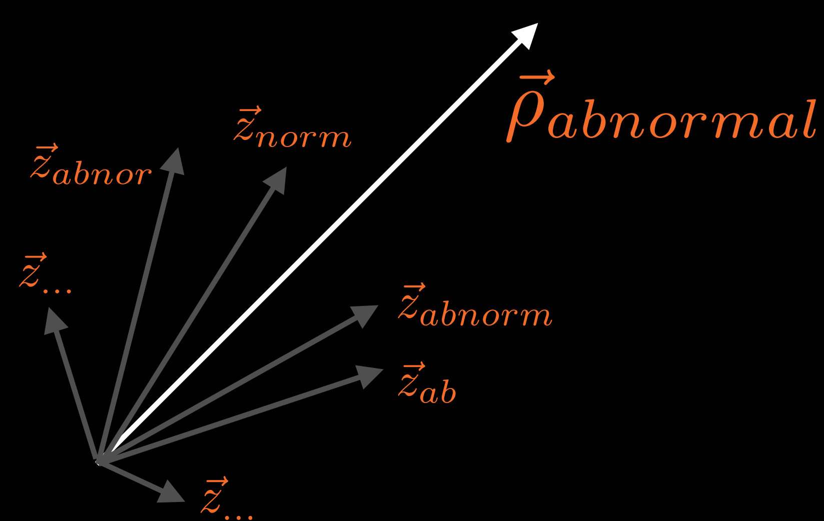
2-MIN SUMMARY

Gaussian Mixture Embeddings

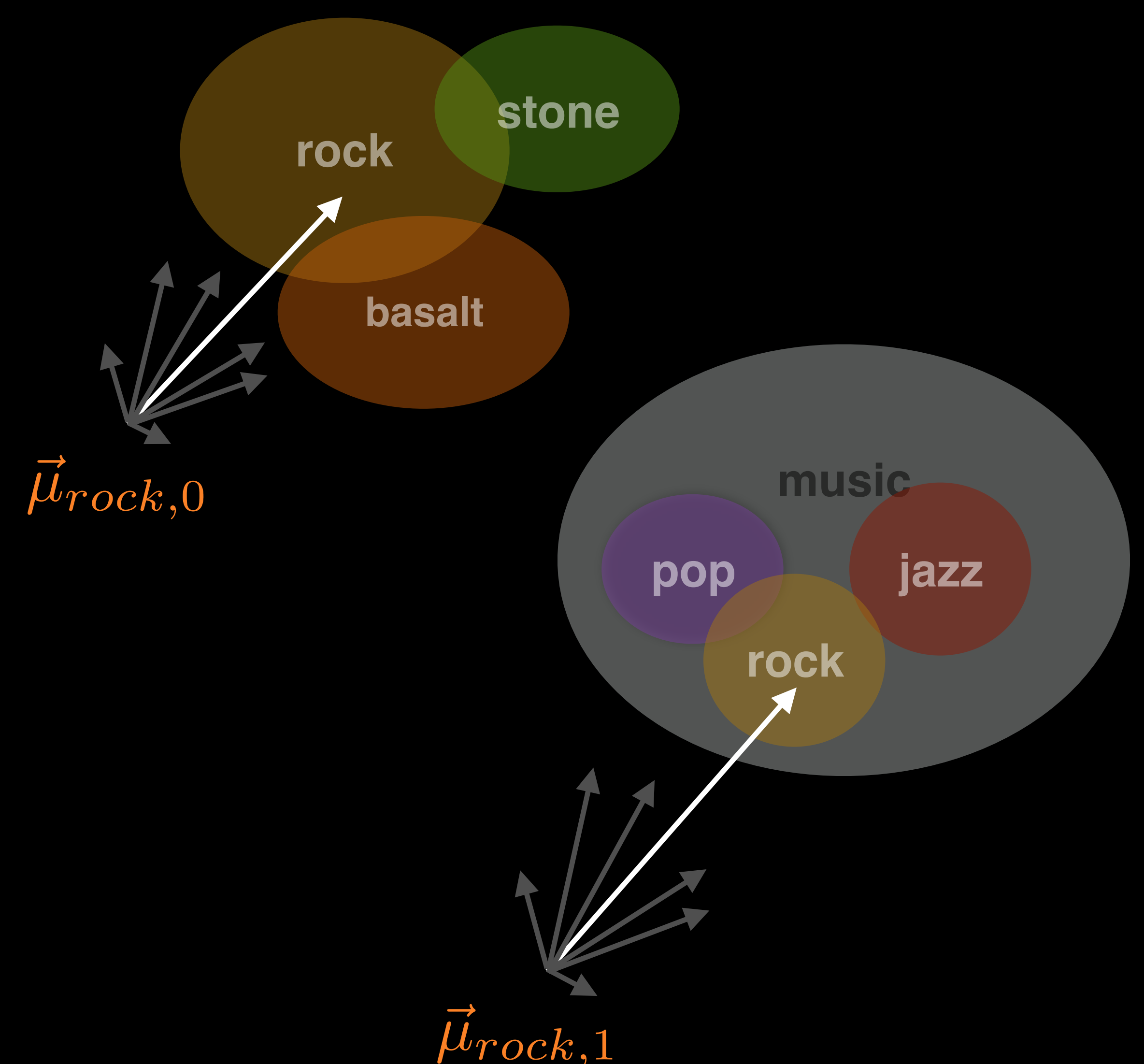


+

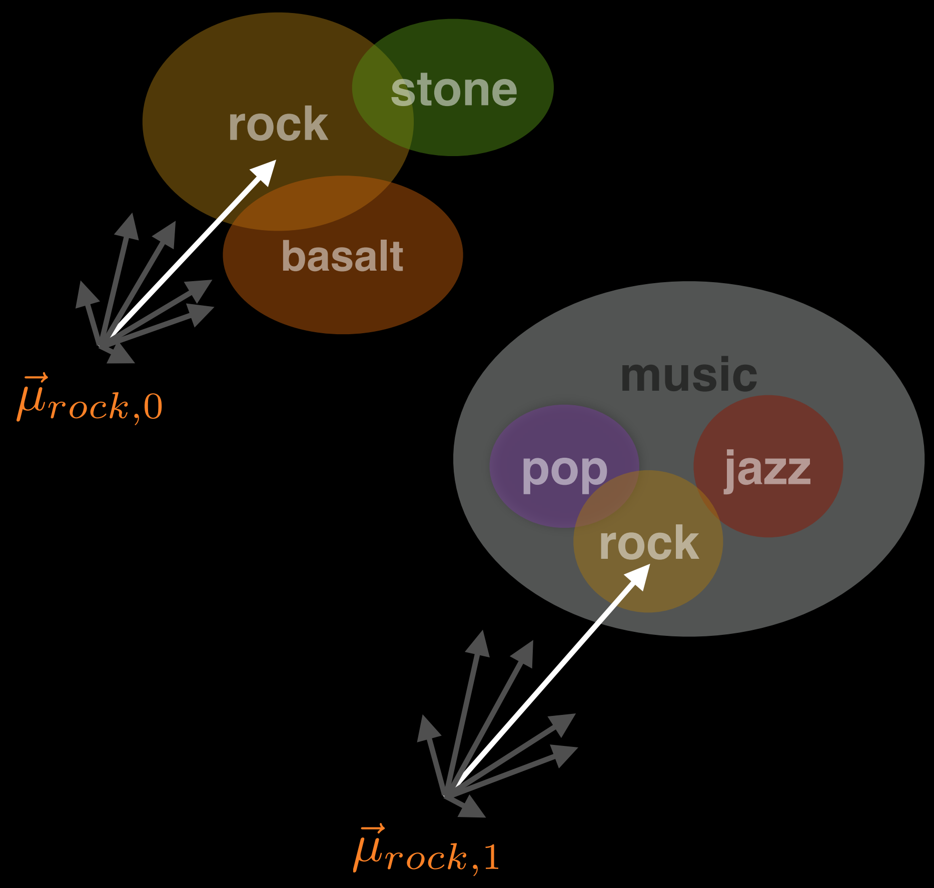
FastText



Probabilistic FastText (PFT)



PROBABILISTIC FASTTEXT



- Able to estimate distributions of unseen words

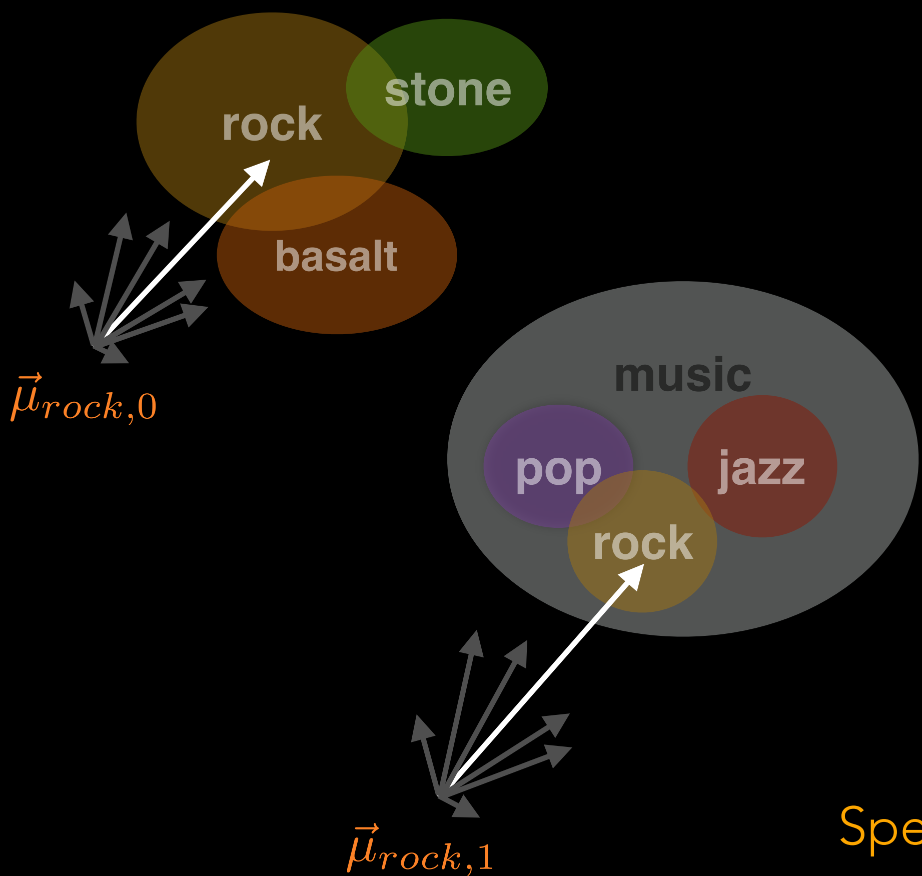
$$L[\text{“cool”}] = \text{COOL}$$
$$L[\text{“coolz”}] = \ ?$$
$$L[\text{“coolzz”}] = \ ?$$

dictionary-based embeddings

$$f(\text{“cool”}) = \text{COOL}$$
$$f(\text{“coolz”}) = \text{COOLZ}$$
$$f(\text{“coolzz”}) = \text{COOLZZ}$$

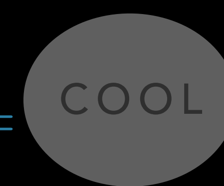
character-based probabilistic embeddings

PROBABILISTIC FASTTEXT



- Able to estimate distributions of unseen words

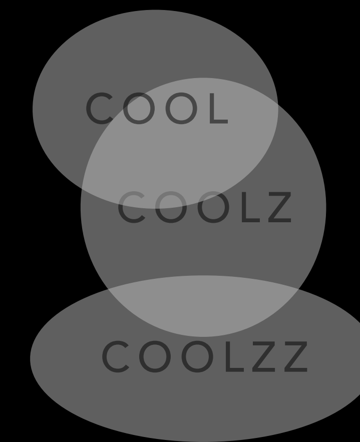
$$L[\text{“cool”}] =$$



$$L[\text{“coolz”}] = \ ?$$
$$L[\text{“coolzz”}] = \ ?$$

dictionary-based embeddings

$$f(\text{“cool”}) =$$
$$f(\text{“coolz”}) =$$
$$f(\text{“coolzz”}) =$$

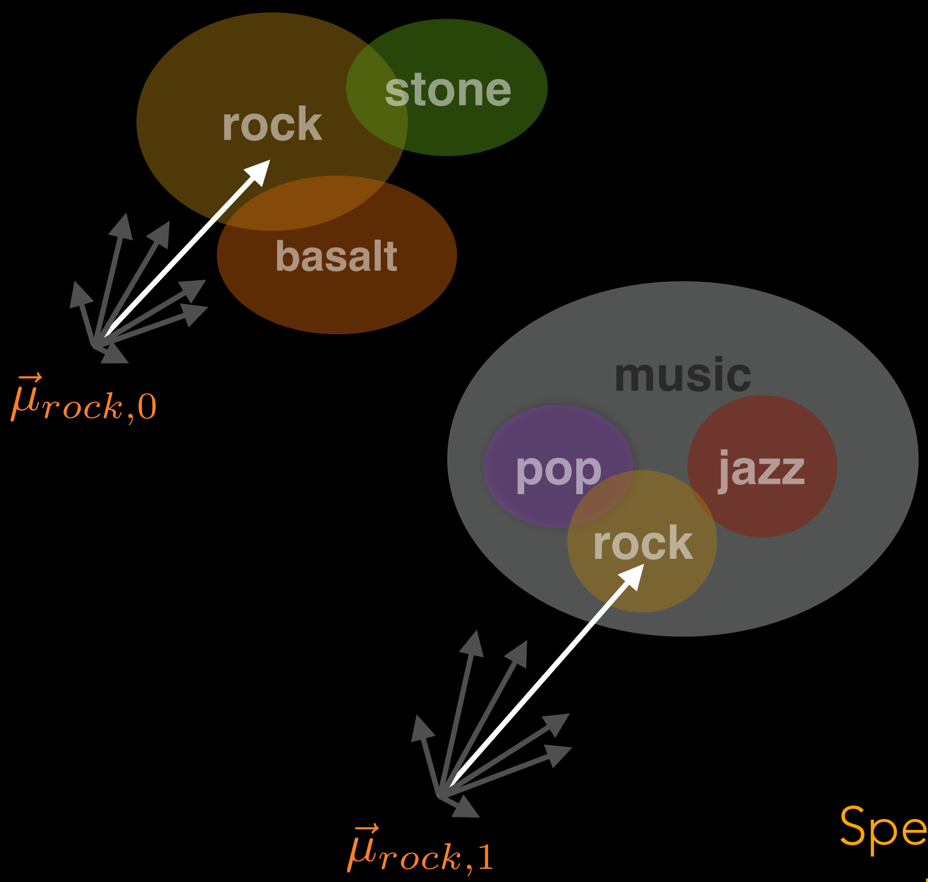


character-based probabilistic embeddings

- High semantic quality for rare words via root sharing

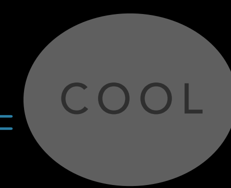
	w2gm	FastText	PFT
Spearman Correlation on RareWord dataset	0.43	0.48	**0.49**

PROBABILISTIC FASTTEXT



- Able to estimate distributions of unseen words

$$L[\text{“cool”}] =$$



$$L[\text{“coolz”}] = \ ?$$

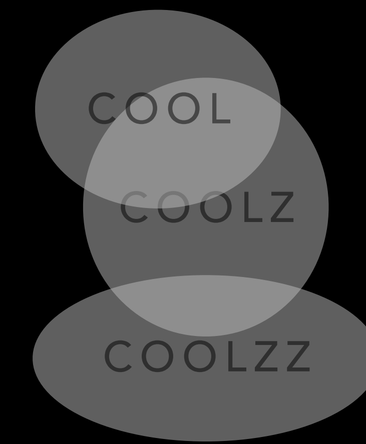
$$L[\text{“coolzz”}] = \ ?$$

dictionary-based embeddings

$$f(\text{“cool”}) =$$

$$f(\text{“coolz”}) =$$

$$f(\text{“coolzz”}) =$$



character-based probabilistic embeddings

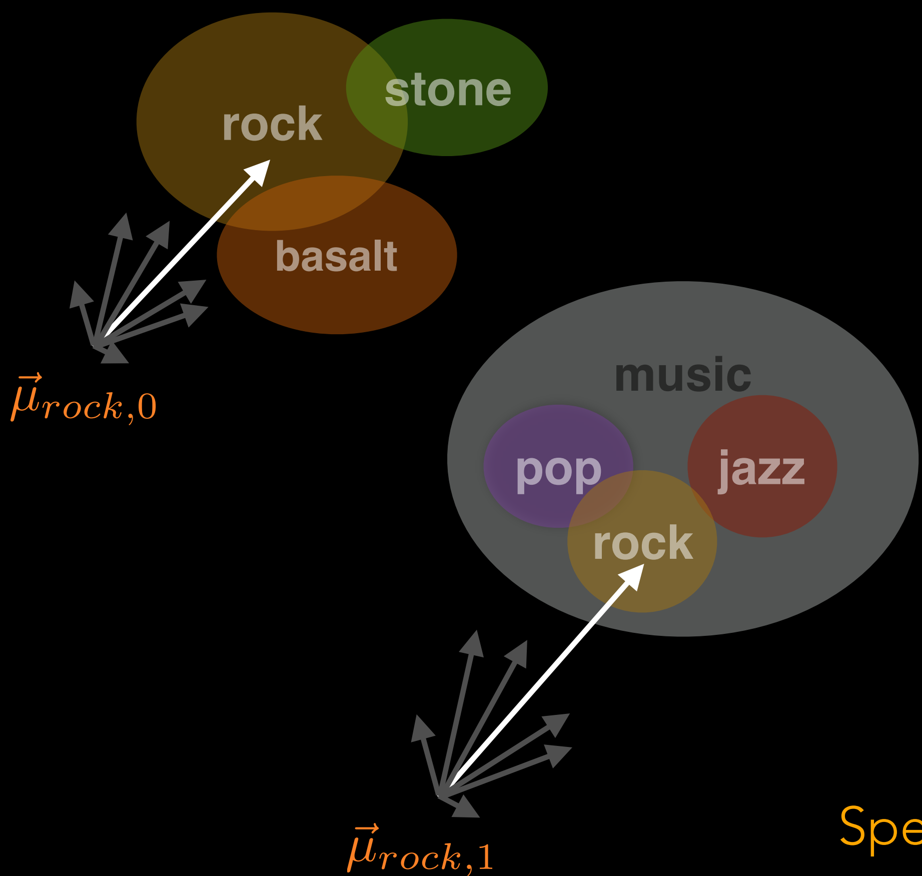
- High semantic quality for rare words via root sharing

	w2gm	FastText	PFT
Spearman Correlation on RareWord dataset	0.43	0.48	**0.49**

- disentangled meanings

Word	Component	Nearest neighbors (cosine similarity)
rock	0	rocks:0, rocky:0, mudrock:0, rockscape:0
rock	1	punk:0, punk-rock:0, indie:0, pop-rock:0

PROBABILISTIC FASTTEXT



- Able to estimate distributions of unseen words

$L[\text{“cool”}] =$ COOL	$f(\text{“cool”}) =$ COOL
$L[\text{“coolz”}] = \ ?$	$f(\text{“coolz”}) =$ COOLZ
$L[\text{“coolzz”}] = \ ?$	$f(\text{“coolzz”}) =$ COOLZZ
dictionary-based embeddings	**character-based** probabilistic embeddings

- High semantic quality for rare words via root sharing

Spearman Correlation on RareWord dataset	w2gm	FastText	PFT
	0.43	0.48	**0.49**

- disentangled meanings

Word	Component	Nearest neighbors (cosine similarity)
rock	0	rocks:0, rocky:0, mudrock:0, rockscape:0
rock	1	punk:0, punk-rock:0, indie:0, pop-rock:0

- Applicable to foreign languages without any changes in model hyperparameters!

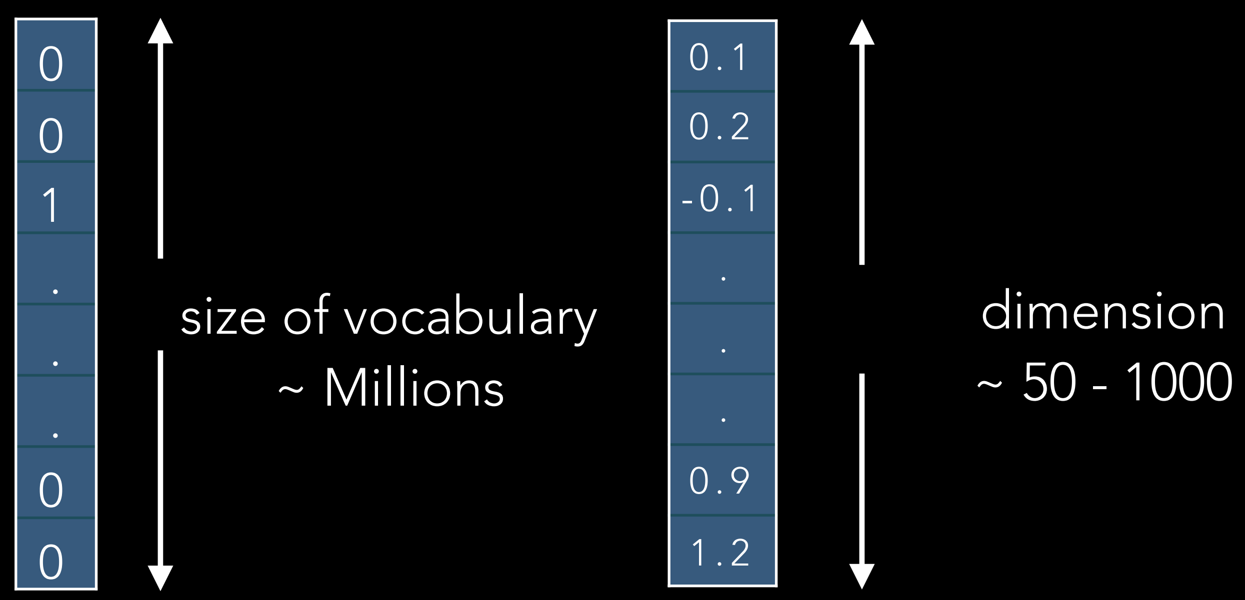
Word	Component / Meaning	Nearest neighbors (English Translation)
secondo	0 / 2nd	Secondo (2nd), terzo (3rd) , quinto (5th), primo (first)
secondo	1 / according to	conformit (compliance), attenendosi (following), cui (which)

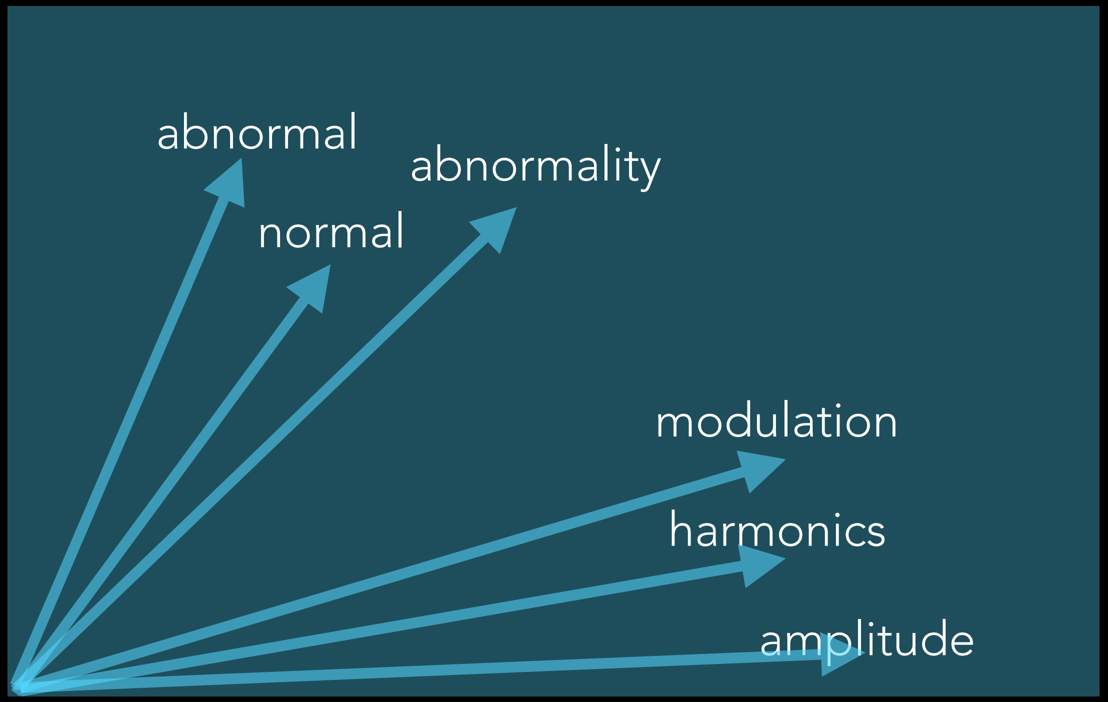
VECTOR EMBEDDINGS & FASTTEXT

WORD EMBEDDINGS

one-hot vector

dense representation



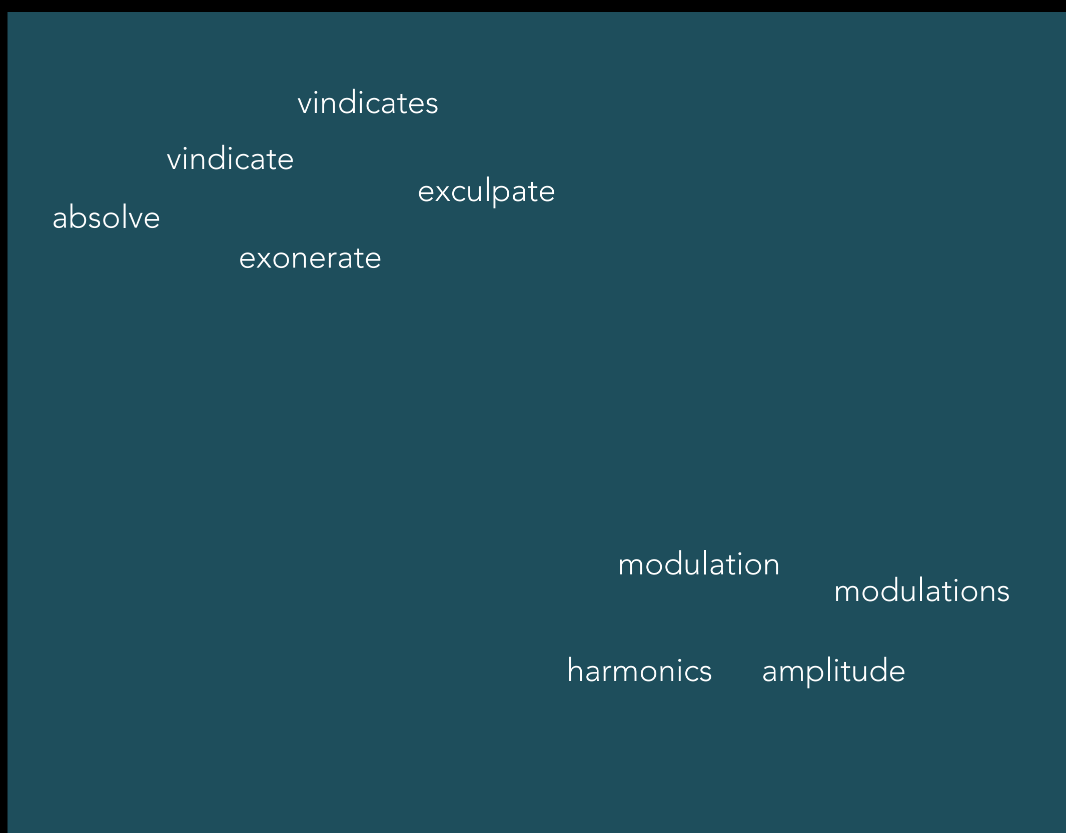


- word2vec (Mikolov et al., 2013)
- GloVe (Pennington et al., 2014)

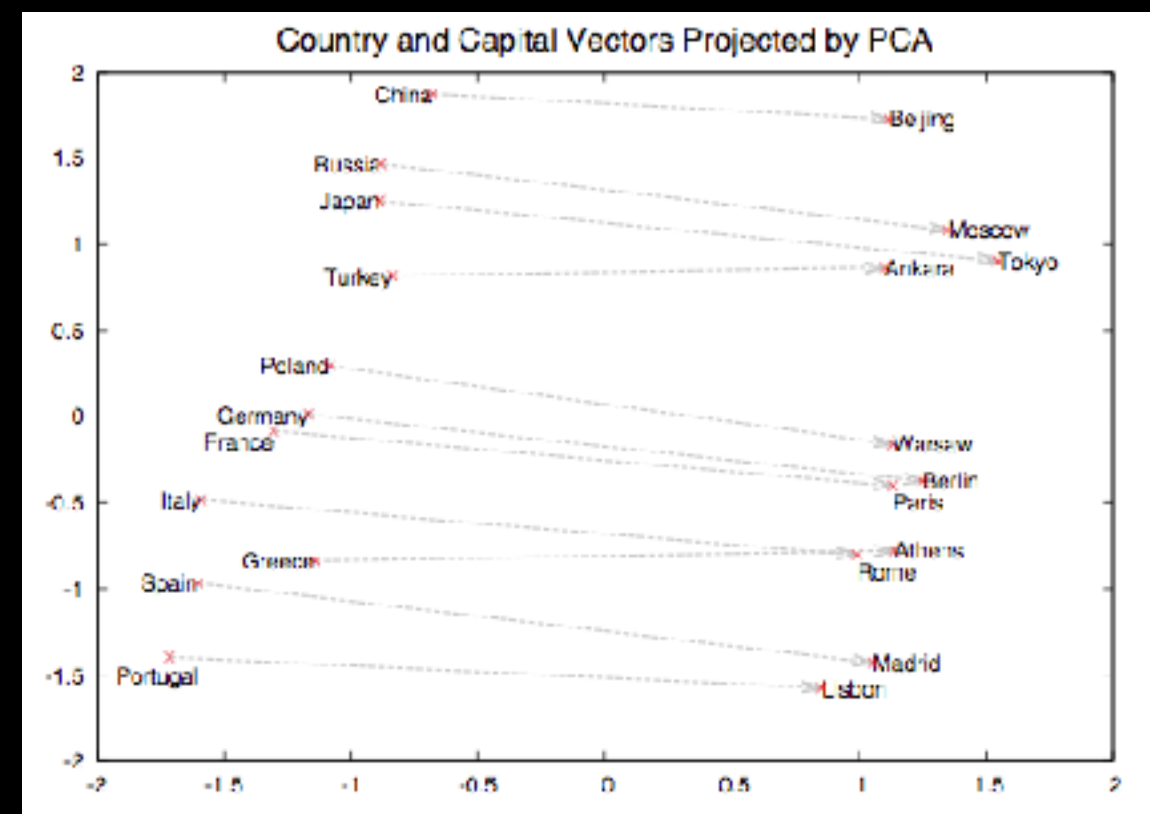
vectors

DENSE REPRESENTATION OF WORDS

Meaningful nearest neighbors



Relationship deduction
from vector arithmetic



i.e.

China - Beijing ~ Japan - Tokyo

Mikolov 2013

CHAR-MODEL: SUBWORD REPRESENTATION

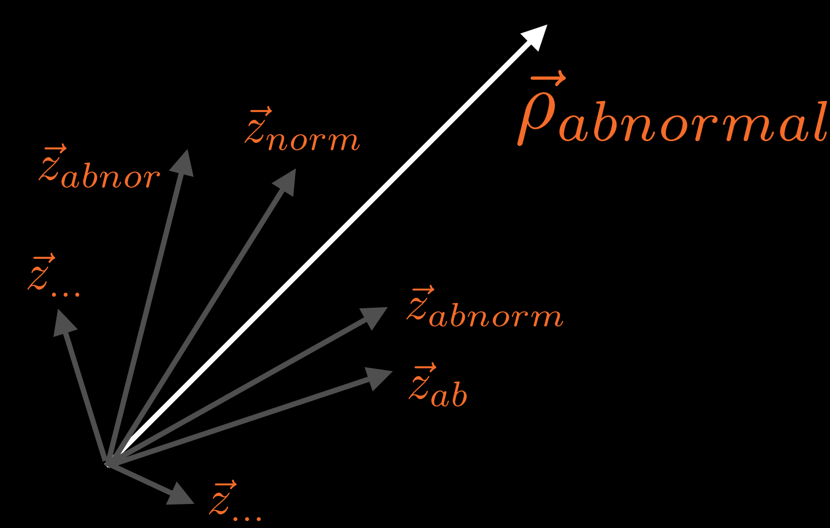
$$\vec{\rho}_w = \frac{1}{|NG_w| + 1}\left(\vec{v}_w + \sum_{g \in NG_w} \vec{z}_g\right)$$

FastText (P Bojanowski, 2017)

- representation = average of n-gram vectors
- automatic semantic extraction of stems/prefixes/suffices

w = <abnormal>

$$\text{N-grams}(w) \ni \{\langle ab, abn, \ldots, \langle abn, abnor, \ldots, \}$$



CHAR-MODEL: SUBWORD REPRESENTATION

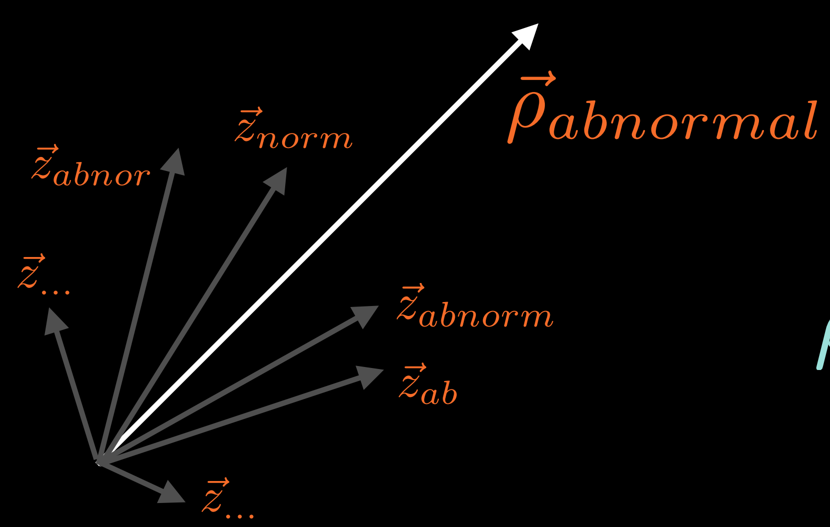
$$\vec{\rho}_w = \frac{1}{|NG_w| + 1}\left(\vec{v}_w + \sum_{g \in NG_w} \vec{z}_g\right)$$

FastText (P Bojanowski, 2017)

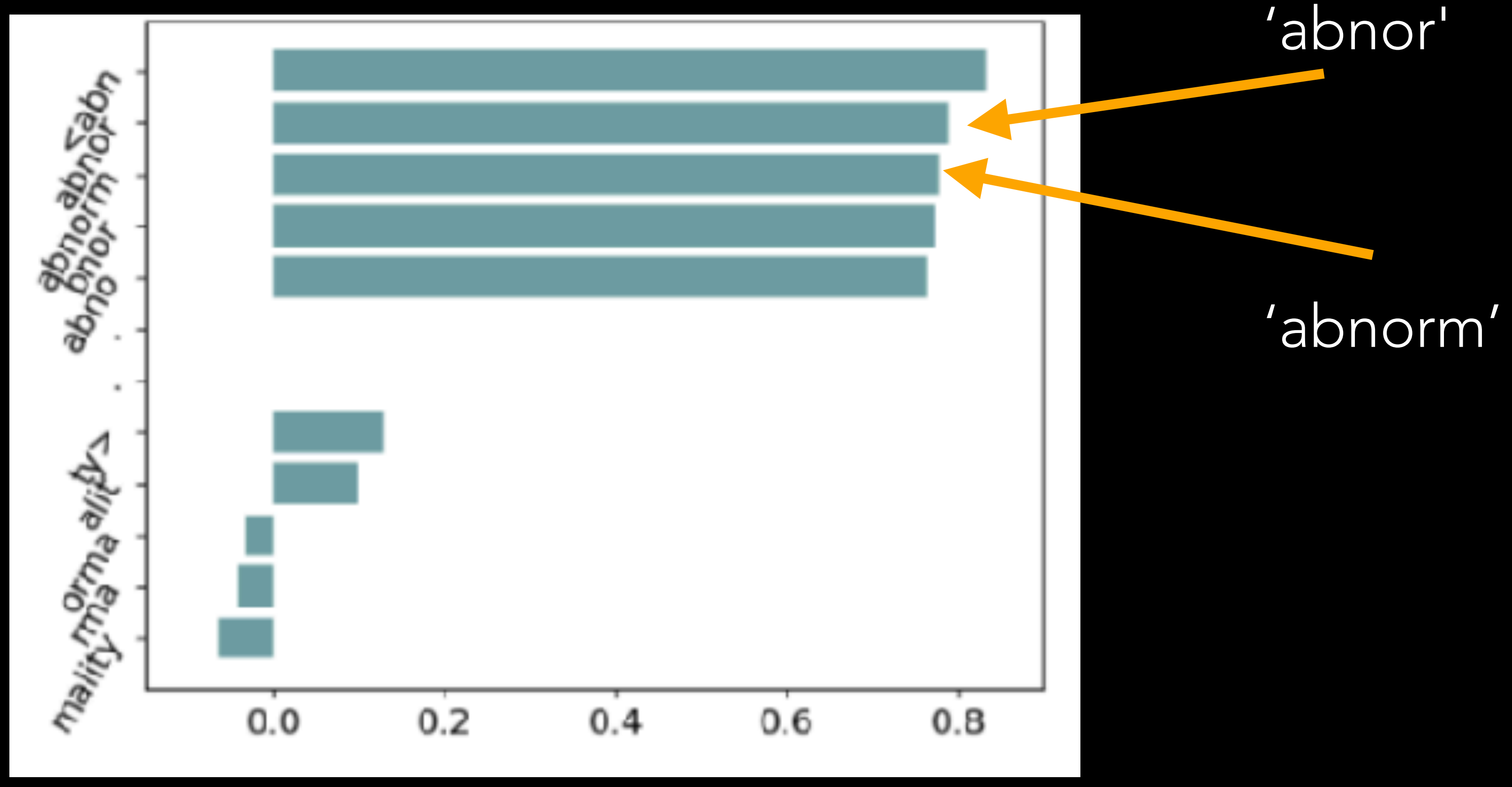
- representation = average of n-gram vectors
- automatic semantic extraction of stems/prefixes/suffices

w = <abnormal>

$$\text{N-grams}(w) \ni \{\langle ab, abn, \ldots, \langle abn, abnor, \ldots, \}$$



$\vec{\rho}_w \cdot \vec{z}$

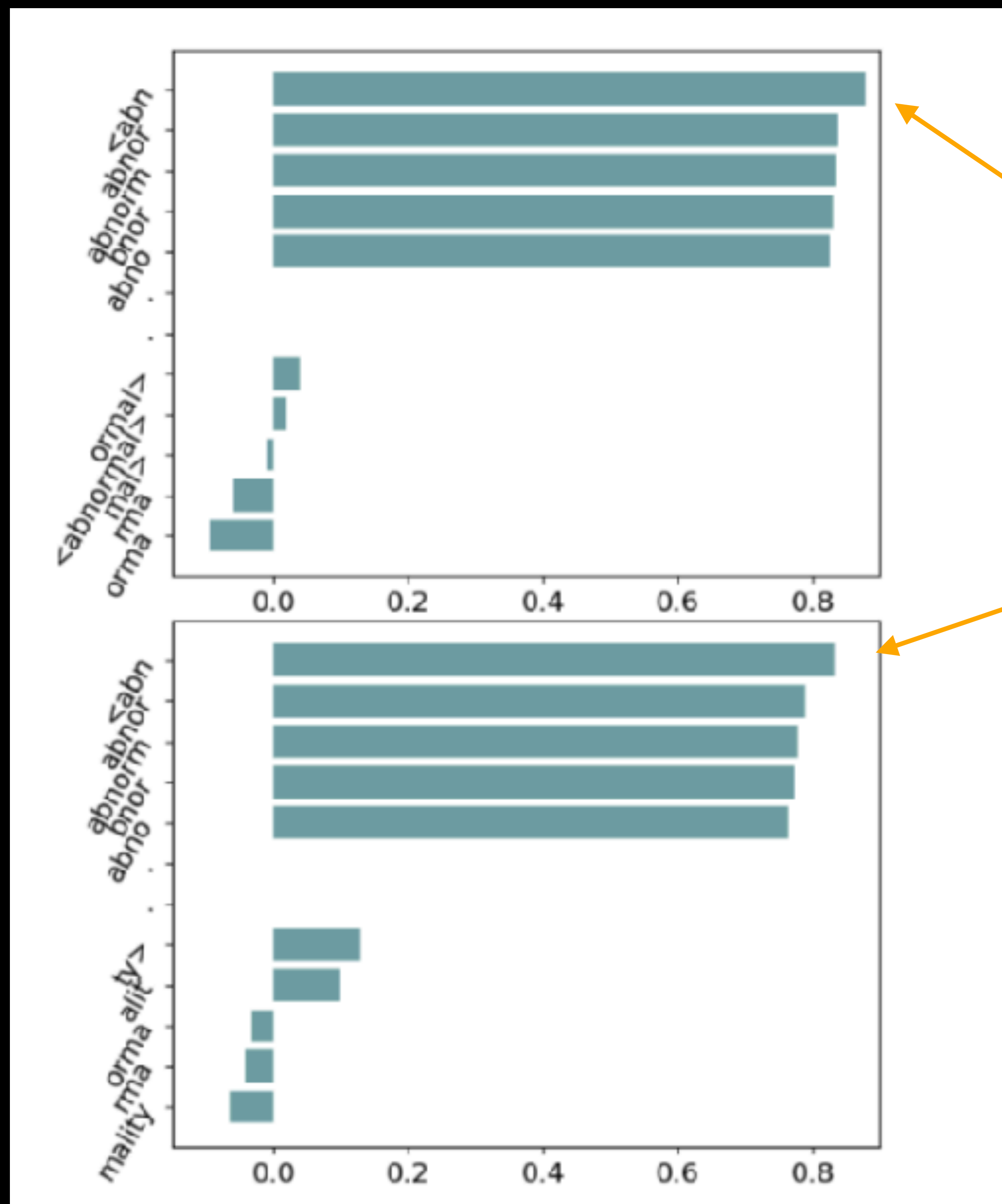


cosine similarity between vector and n-gram vectors

SUBWORD CONTRIBUTION TO OVERALL SEMANTICS

abnormal

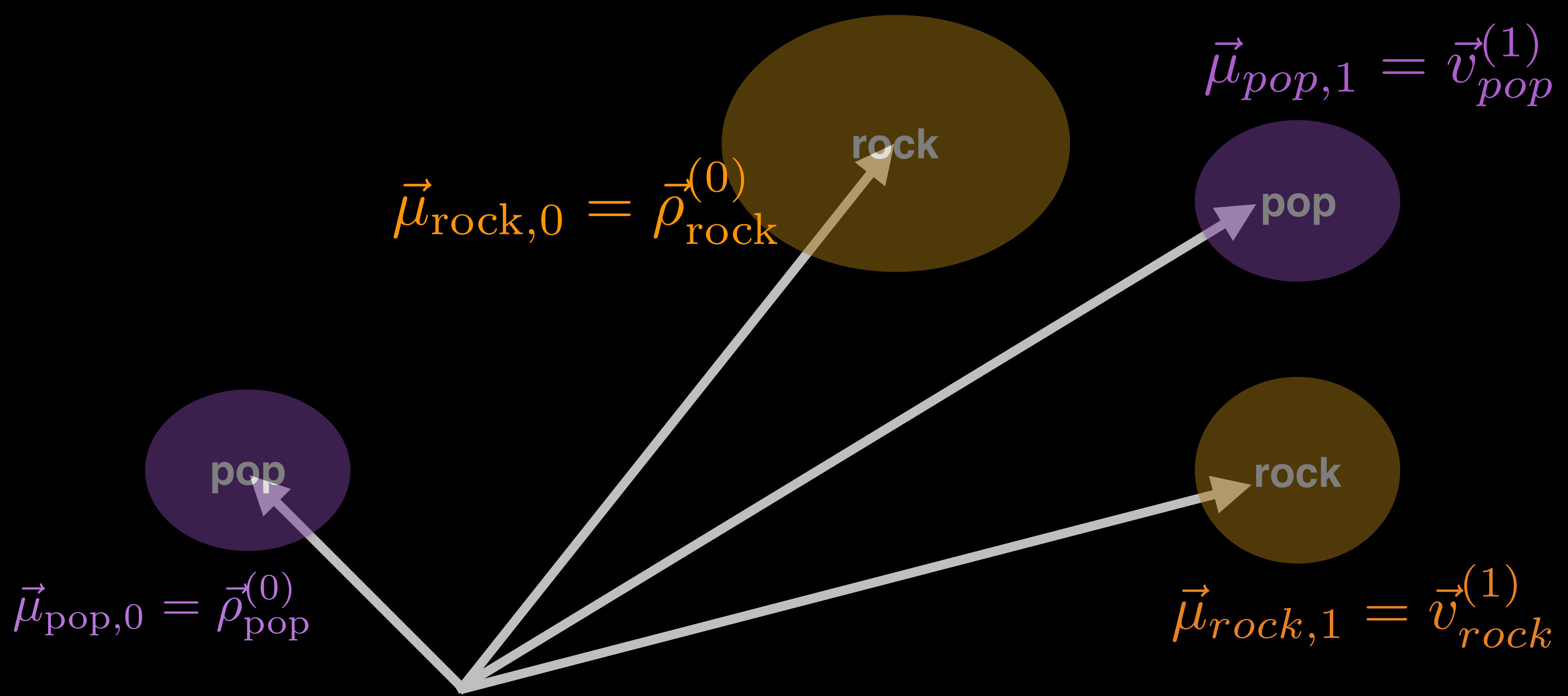
abnormality



- Similar n-grams with high contribution
- Similar words have similar semantics

cosine similarity between n-gram vectors and mean vectors

FASTTEXT WITH WORD2GM



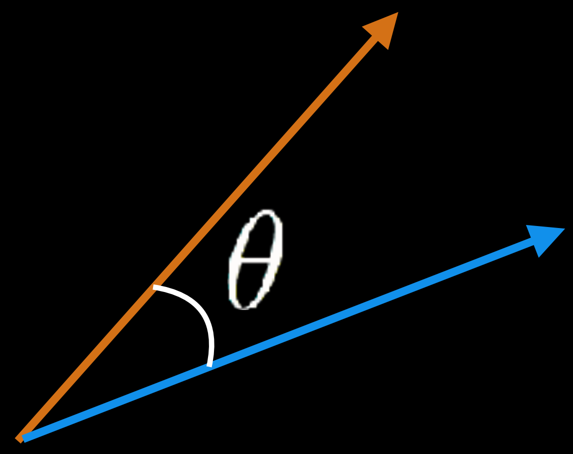
$$\rho^{(j)}_{w,i} = \frac{1}{|NG_w| + 1}\left(\vec{v}^{(j)}_w + \sum_{g \in NG_w} \vec{z}^{(j)}_g\right)$$

- Augment Gaussian mixture representation with character-structure (FastText)
- Promote independence: using dictionary-level vectors for other components

SIMILARITY SCORE (ENERGY) BETWEEN DISTRIBUTIONS

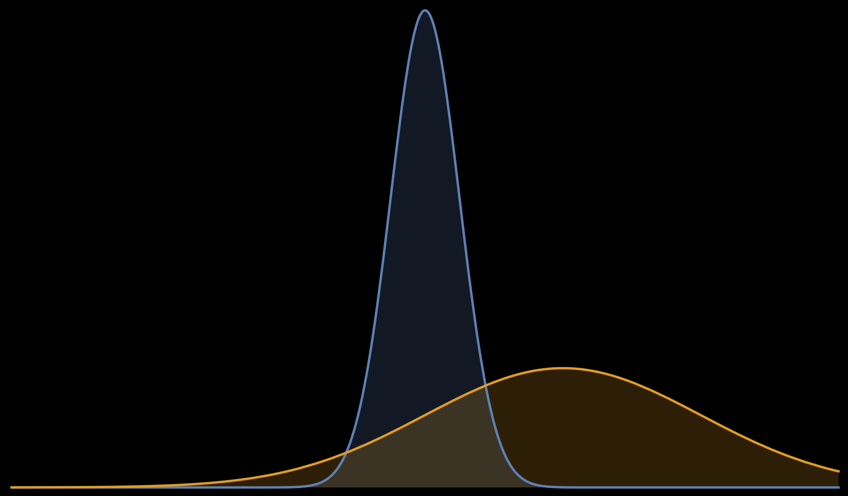
vector space

$$s(u,v) = \langle \vec{u}, \vec{v} \rangle = \vec{u} \cdot \vec{v}$$



function space

$$s(u,v) = \langle u, v \rangle_{L_2} = \int u(x)v(x)\ dx$$



ENERGY OF TWO GAUSSIAN MIXTURES

$$f(x) = \sum_{i=1}^{K} p_i \mathcal{N}(x; \vec{\mu}_{f,i}, \Sigma_{f,i}), \quad g(x) = \sum_{i=1}^{K} q_i \mathcal{N}(x; \vec{\mu}_{g,i}, \Sigma_{g,i})$$

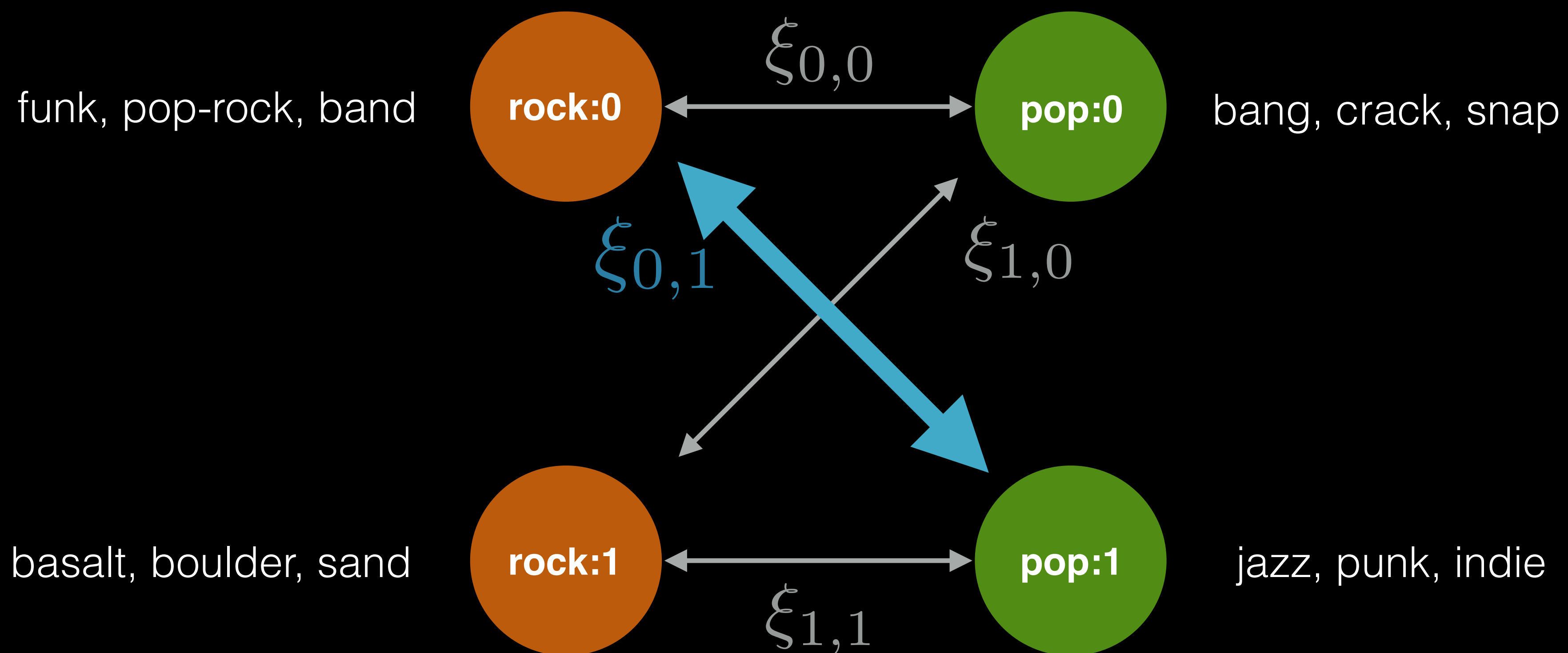
$$\langle f, g \rangle_{L_2} = \sum_{j=1}^{K} \sum_{i=1}^{K} p_i q_j e^{\xi_{i,j}}$$

closed form!

total energy = weighted sum of pairwise partial energies

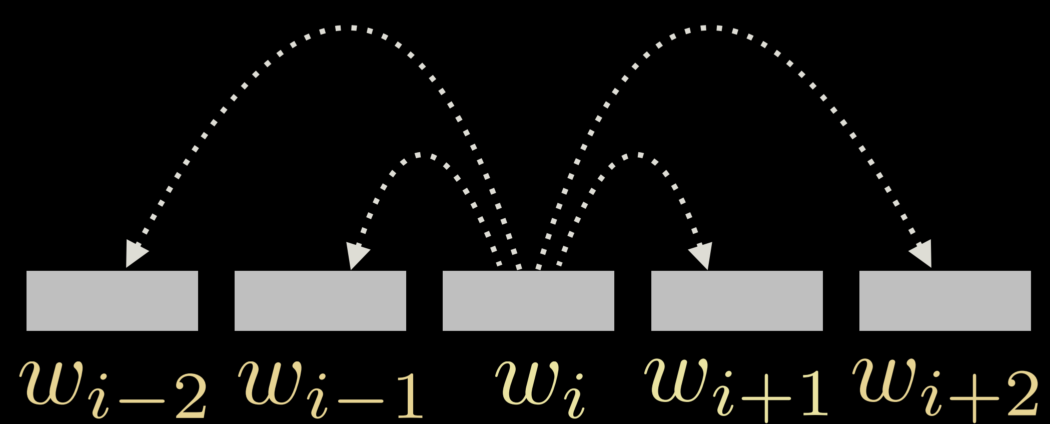
$$\xi_{i,j} = -\frac{\alpha}{2} ||\mu_{f,i} - \mu_{g,i}||^2$$

simplified partial energy



WORD SAMPLING

I like that $\boxed{\text{rock}}$ band



Dataset: ukWac + WackyPedia
(3.5 billion tokens)

LOSS FUNCTION

Energy-based Max Margin

word: w	context word: c	
rock	band	high E(w,c)

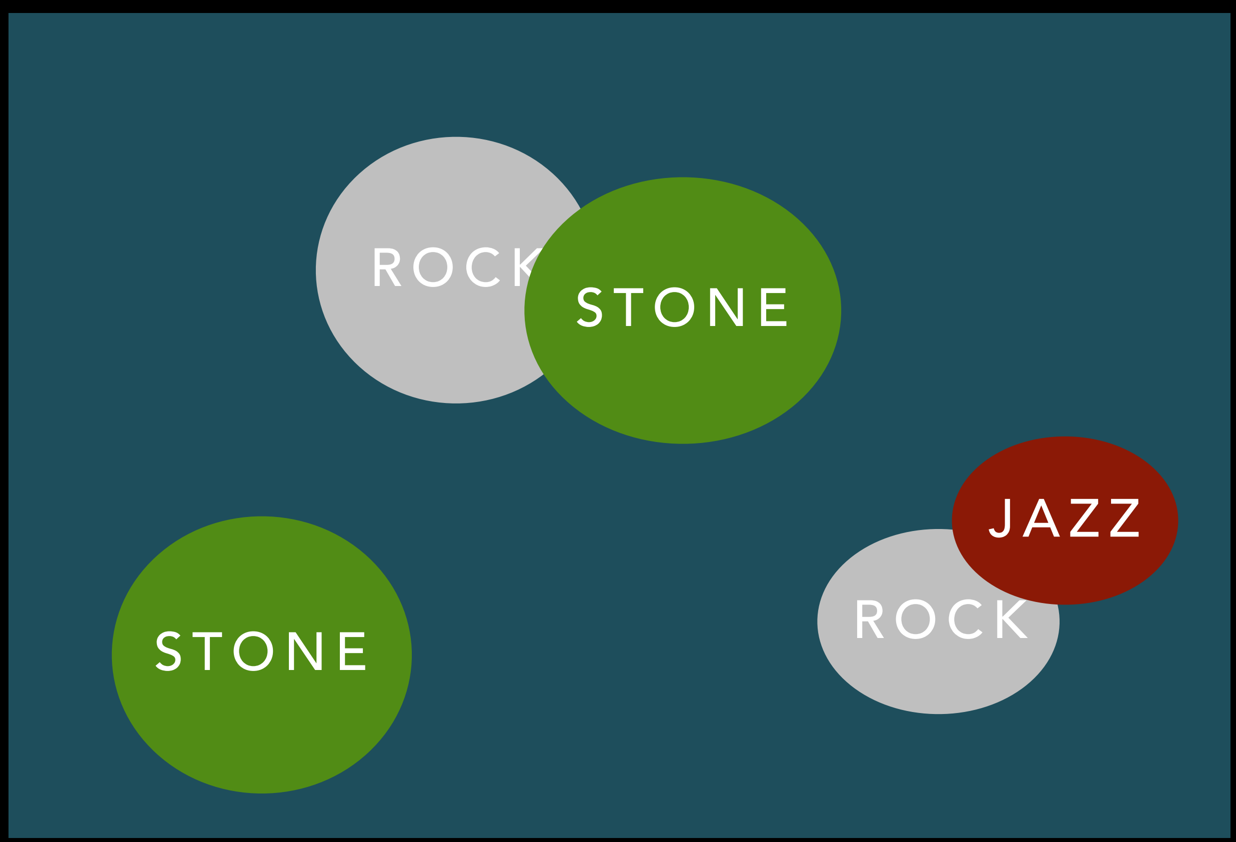
word: w	negative context: c′	
rock	dog	low E(w,c′)

Minimize the objective

$$L(w, c, c') = \max(0, m - \log E(w, c) + \log E(w, c'))$$

MULTIMODAL REPRESENTATION - MIXTURE OF GAUSSIANS

$$\vec{\rho}_w = \frac{1}{|NG_w| + 1} \left(\vec{v}_w + \sum_{g \in NG_w} \vec{z}_g \right)$$



Model parameters:

dictionary vectors

$$\{\{v_i^w\}_{i=1}^{i=K}\}_w$$

char n-gram vectors

$$\{z_g\}$$

Model hyperparameters:

$$\alpha, m$$

(covariance scale, margin)

TRAINING - ILLUSTRATION

Model parameters:

dictionary vectors

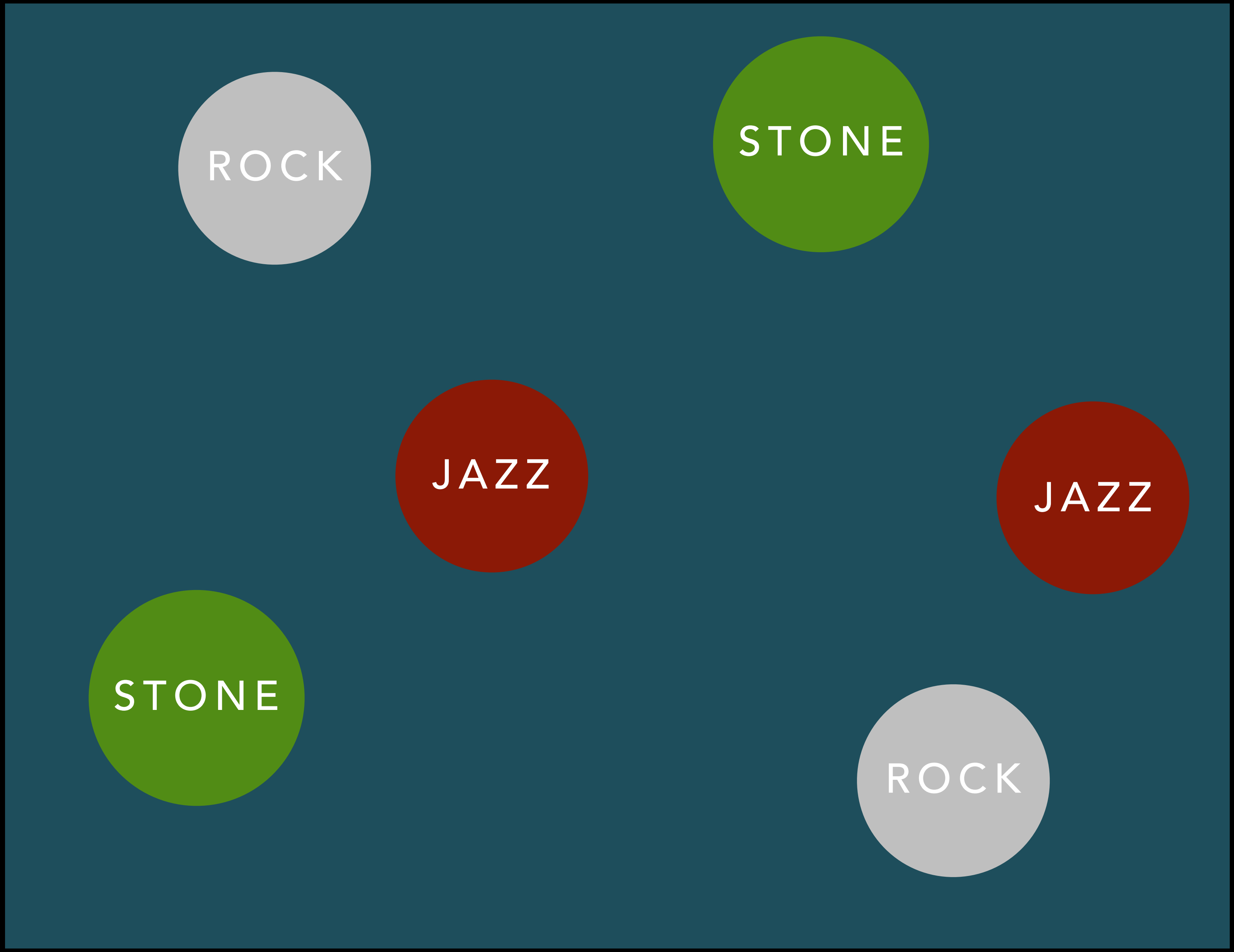
$$\{\{v_i^w\}_{i=1}^{i=K}\}_w$$

char n-gram vectors

$$\{z_g\}$$

Train with max margin objective using minibatch SGD (AdaGrad)

Mixture of Gaussians



TRAINING - ILLUSTRATION

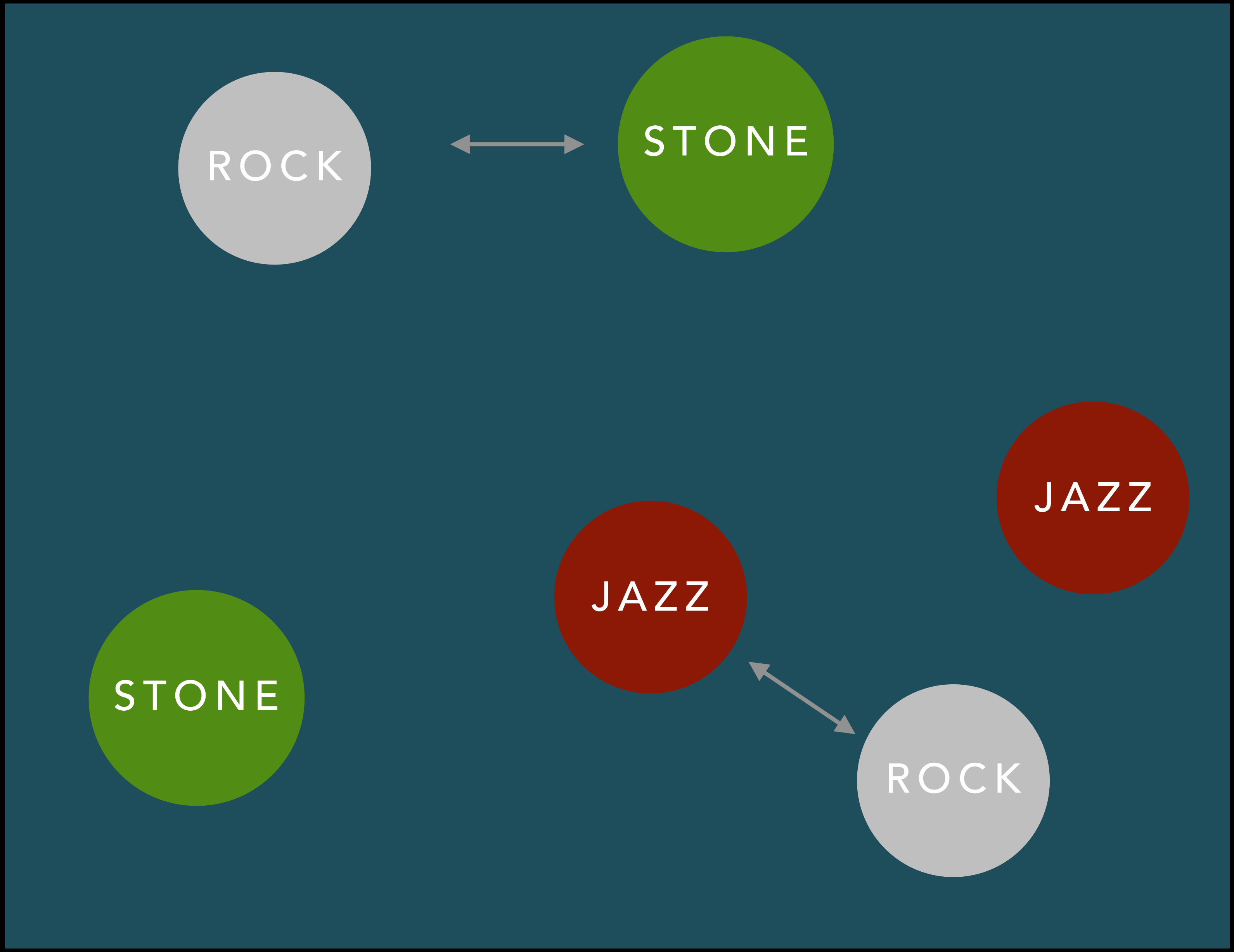
Model parameters:

dictionary vectors
$$\{\{v_i^w\}_{i=1}^{i=K}\}_w$$

char n-gram vectors
$$\{z_g\}$$

Train with max margin objective using minibatch SGD (AdaGrad)

Mixture of Gaussians



TRAINING - ILLUSTRATION

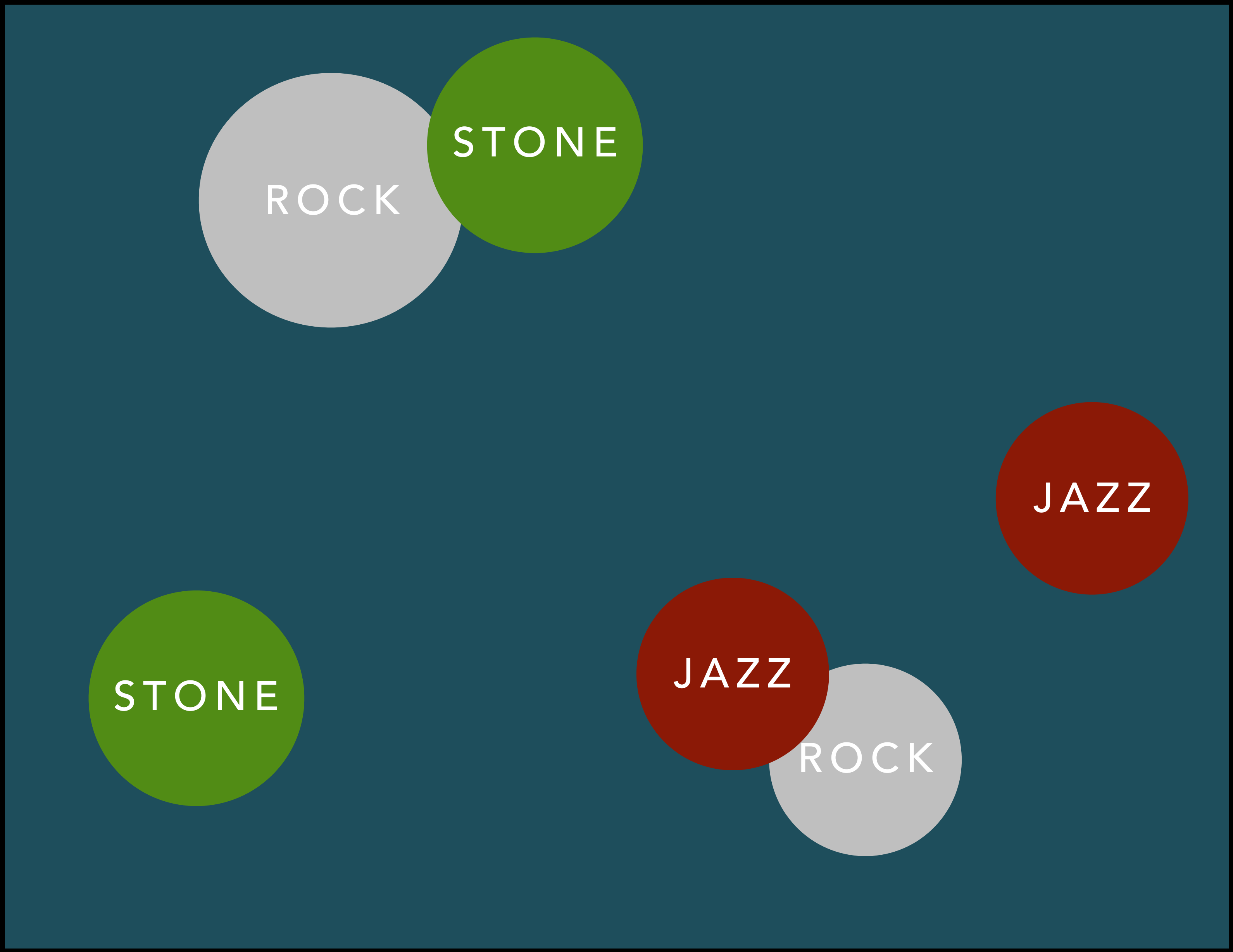
Model parameters:

dictionary vectors
$\{\{v_i^w\}_{i=1}^{i=K}\}_w$

char n-gram vectors
$\{z_g\}$

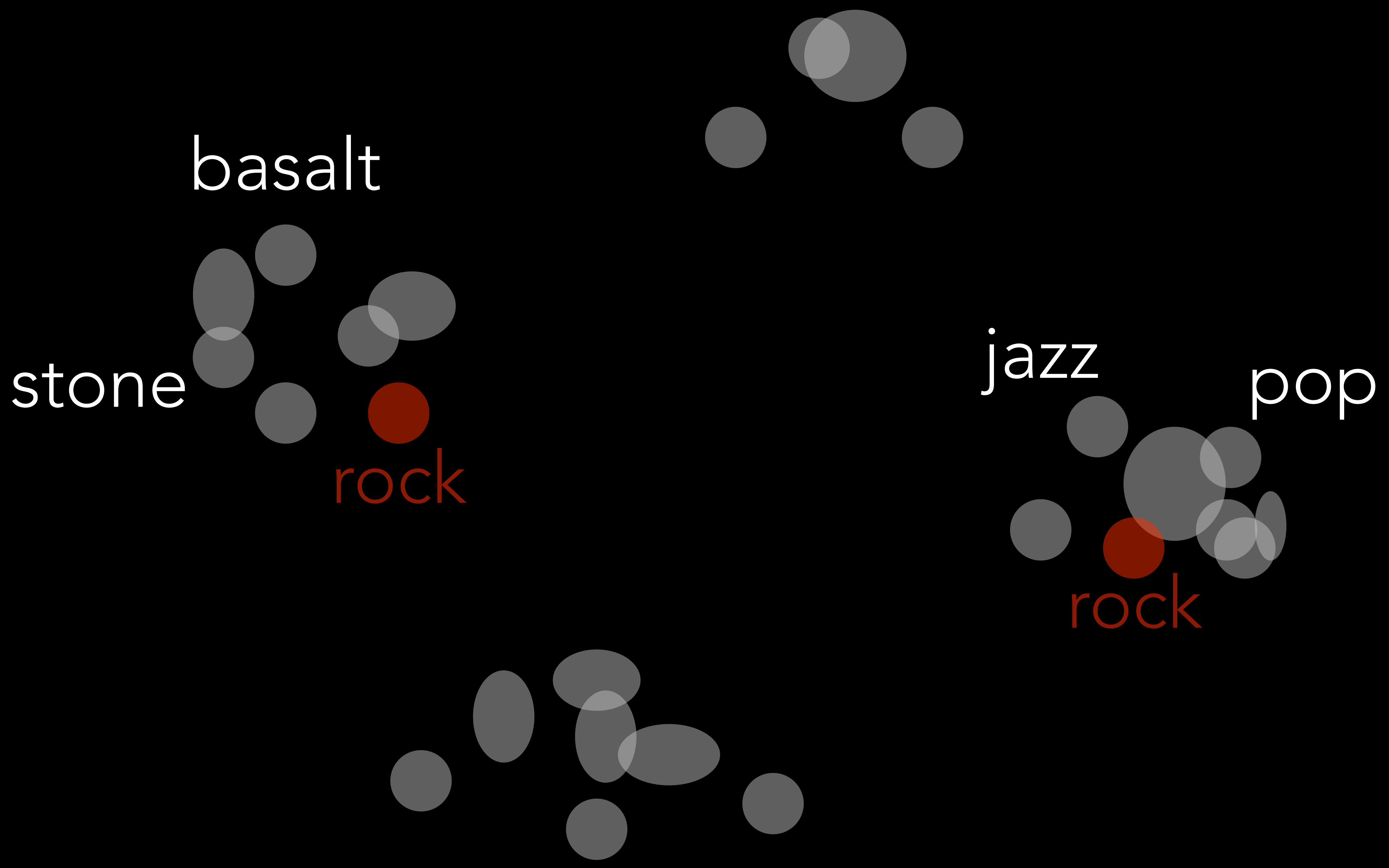
Train with max margin objective using minibatch SGD (AdaGrad)

Mixture of Gaussians



EVALUATION

QUALITATIVE EVALUATION - NEAREST NEIGHBORS



NEAREST NEIGHBORS

PFT-GM

Word	Gaussian Mixture Component	Nearest neighbors (cosine similarity)
rock	0	rocks:0, rocky:0, mudrock:0, rockscape:0, boulders:0 , coutcrops:0
rock	1	punk:0, punk-rock:0, indie:0, pop-rock:0, pop-punk:0, indie-rock:0, band:1
bank	0	banks:0, banker:0, bankers:0, bankcard:0, Citibank:0, debits:0
bank	1	banks:1, river:0, riverbank:0, embanking:0, banks:0, confluence:1
star	0	stars:0, stellar:0, nebula:0, starspot:0, stars.:0, stellas:0, constellation:1
star	1	stars:1, star-star:0, 5-stars:0, movie-star:0, mega-star:0, super-star:0

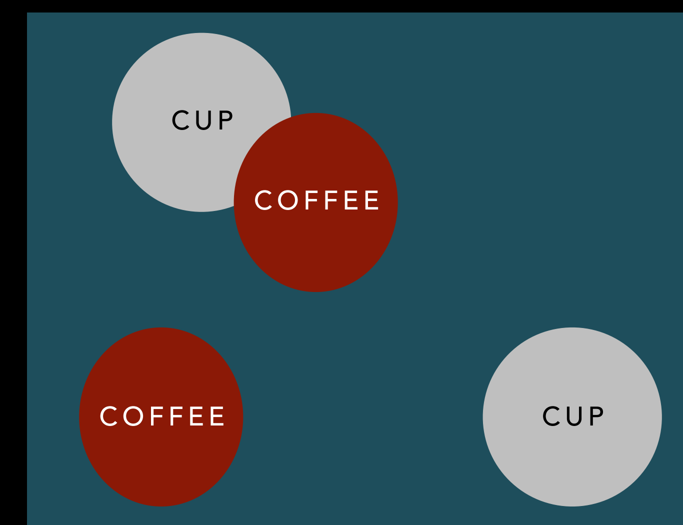
FastText

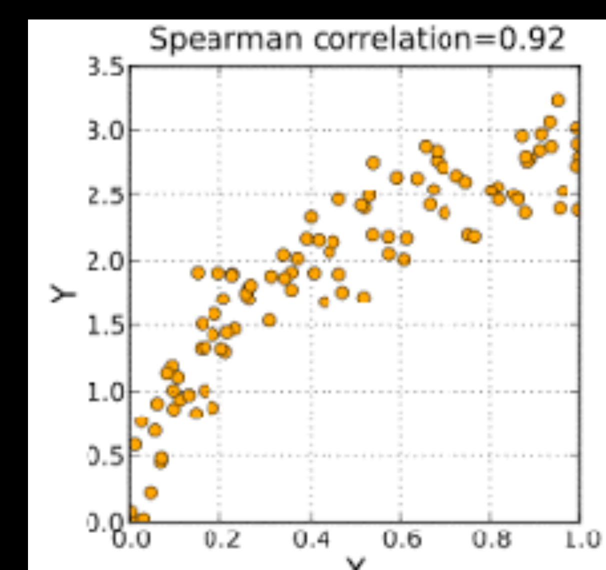
Word	Nearest neighbors (cosine similarity)
rock	rock-y, rockn, rock-, rock-funk, rock/, lava-rock, nu-rock, rock-pop, rock/ice, coral-rock
bank	bank-, bank/, bank-account, bank., banky, bank-to-bank, banking, Bank, bank/cash, banks.**
star	movie-stars, star-planet, G-star, star-dust, big-star, starsailor, 31-star, star-lit, Star, starsign

QUANTITATIVE EVALUATION

WORD PAIR		HUMAN SCORE	EMBEDDING SIMILARITY
CUP	COFFEE	6.58	S(CUP, COFFEE) = 0.7
CUP	SUBSTANCE	1.92	S(CUP, SUBSTANCE) = 0.2
STOCK	MARKET	8.08	S(STOCK, MARKET) = 0.9
STOCK	PHONE	1.62	S(STOCK, PHONE) = 0.05
KING	QUEEN	8.58	S(KING, QUEEN) = 0.8
KING	CABBAGE	0.23	S(KING, CABBAGE) = 0.2

s(cup, coffee) = similarity between 'cup' and 'coffee'





Spearman correlation coefficient

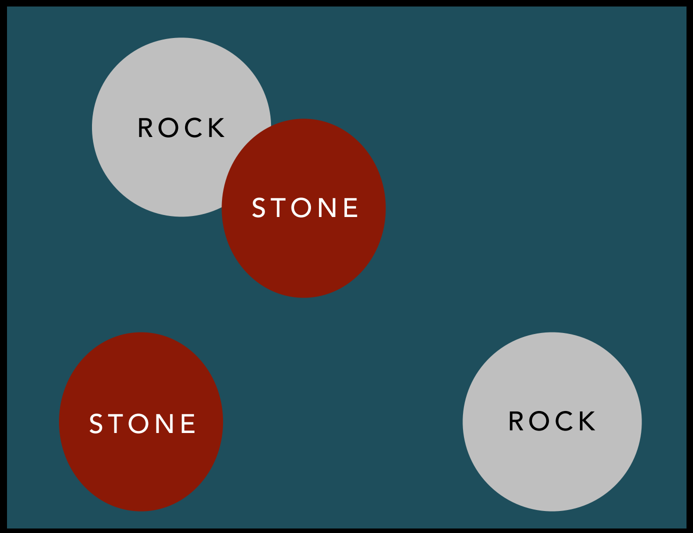
0: no correlation

1: perfect correlation

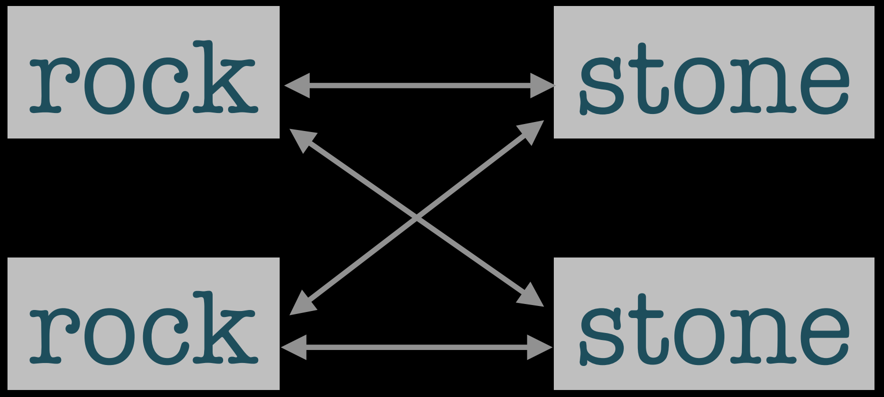
SIMILARITY METRIC

s(rock, stone)

Expected Likelihood



$$\int f_{\text{rock}}(x) g_{\text{stone}}(x) dx$$



Pairwise Maximum Cosine Similarity

$$\max_{i,j} \langle \vec{\mu}_{\text{rock},i}, \vec{\mu}_{\text{stone},j} \rangle$$

SPEARMAN CORRELATIONS

WORD SIM DATASETS	FASTTEXT	W2GM	PFT-GM
SL-999	38.03	39.62	39.60
WS-353	78.88	79.38	76.11
MEN-3K	76.37	78.76	79.65
MC-30	81.20	84.58	80.93
RG-65	79.98	80.95	79.81
YP-130	53.33	47.12	54.93
MT-287	67.93	69.65	69.44
MT-771	66.89	70.36	69.68
RW-2K (RAREWORD)	48.09	42.73	49.36
AVG.	49.28	49.54	**51.10**

- PFT performs much better on RareWord dataset compared to w2gm, even slightly better than FastText
- Based on the average spearman correlation, PFT-GM performs the best.
- First multi-sense models that achieve high scores on RareWord

COMPARISON WITH OTHER MULTI-PROTOTYPE EMBEDDINGS

Model	Dim	$\rho \times 100$
HUANG AVGSIM	50	62.8
TIAN MAXSIM	50	63.6
W2GM MAXSIM	50	62.7
NEELAKANTAN AVGSIM	50	64.2
PFT-GM MAXSIM	50	63.7
CHEN-M AVGSIM	200	66.2
W2GM MAXSIM	200	65.5
NEELAKANTAN AVGSIM	300	**67.2**
W2GM MAXSIM	300	66.5
PFT-GM MAXSIM	300	**67.2**

Table 3: Spearman's Correlation $\rho \times 100$ on word similarity dataset SCWS.

- PFT performs better than other multi-prototype embeddings on SCWS, a benchmark for word similarity with multiple meanings.

FOREIGN LANGUAGE EMBEDDINGS

Word	Meaning	Nearest Neighbors
(IT) *secondo*	2nd	Secondo (2nd), terzo (3rd) , quinto (5th), primo (first), quarto (4th), ultimo (last)
(IT) *secondo*	according to	conformit (compliance), attenendosi (following), cui (which), conformemente (accordance with)
(IT) *porta*	lead, bring	portano (lead), conduce (leads), portano, porter, portando (bring), costringe (forces)
(IT) *porta*	door	porte (doors), finestrella (window), finestra (window), portone (doorway), serratura (door lock)
(FR) *voile*	veil	voiles (veil), voiler (veil), voilent (veil), voilement, foulard (scarf), voils (veils), voilant (veiling)
(FR) *voile*	sail	catamaran (catamaran), driveur (driver), nautiques (water), Voile (sail), driveurs (drivers)
(FR) *temps*	weather	brouillard (fog), orageuses (stormy), nuageux (cloudy)
(FR) *temps*	time	mi-temps (half-time), partiel (partial), Temps (time), annualis (annualized), horaires (schedule)
(FR) *voler*	steal	envoler (fly), voleuse (thief), cambrioler (burgle), voleur (thief), violer (violate), picoler (tipple)
(FR) *voler*	fly	airs (air), vol (flight), volent (fly), envoler (flying), atterrir (land)

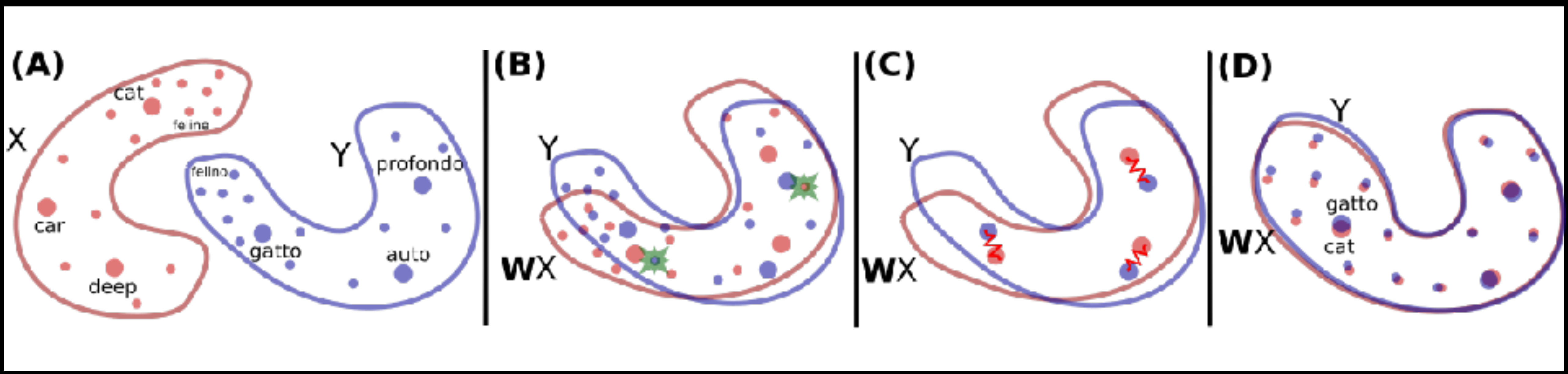
Table 5: Nearest neighbors of polysemies based on our foreign language PFT-GM models.

Lang.	Evaluation	FASTTEXT	w2g	w2gm	pft-g	pft-gm
FR	WS353	38.2	16.73	20.09	41.0	**41.3**
DE	GUR350	70	65.01	69.26	77.6	**78.2**
	GUR65	81	74.94	76.89	81.8	**85.2**
IT	WS353	57.1	56.02	61.09	60.2	**62.5**
	SL-999	29.3	29.44	**34.91**	29.3	33.7

Table 4: Word similarity evaluation on foreign languages.

FUTURE WORK: MULTI-LINGUAL EMBEDDINGS

Literature: align embeddings of many languages after training (Conneau, 2018)



Use disentangled embeddings to disambiguate alignment

CONCLUSION

- Elegant representation of semantics using multimodal distributions
- Suitable modeling words with multiple meanings
- Model words as character levels
  - Better semantics for rare words
  - Able to estimate semantics of unseen words