Distributional Gastronomics

Jonathan Oberländer and Laura Bostan
@L3viathan2142 @sarnthil
Trento University

Distributional Gastronomics

The distributional gastronomical hypothesis: Similar ingredients will occur in similar contexts.

We present a distributional semantic model for recipes by building a representation of food ingredients and recipes in terms of co-occurrence vectors, recording their distributional pattern in a recipe corpus. We build a space for ingredients and two spaces for recipes.

Obtaining a Vector Space from Recipes

We utilize the recipe database crawled by **Open Recipes**. The list of ingredients per recipe plus its name was obtained, everything else discarded.
We then unify ingredient names (cleaning artifacts and getting rid of units and amounts), to reduce the number of types from over 400,000 to just 6514.

Number of recipes	172893
Number of ingredients (tokens in corpus)	1689892
Number of ingredients (types in corpus)	412858
Number of ingredients (types after unifying)	6514

In every recipe, we count each pair of ingredients as co-occurring, and build a large 6514-dimensional matrix.

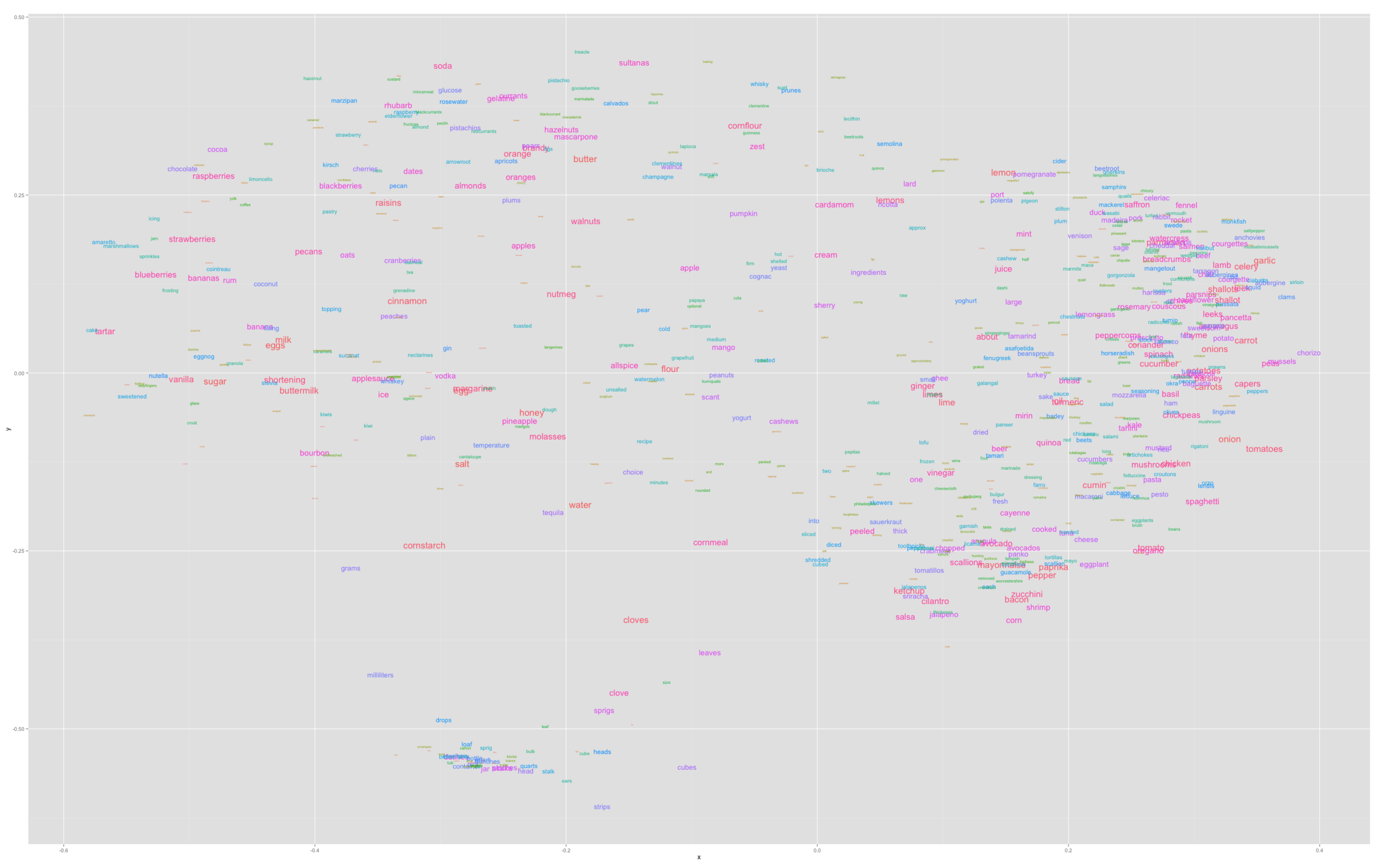
	tomatoes	basil	mozzarella	olive oil
tomatoes	1562	21	6	711
basil	21	434	13	239
mozzarella	6	13	149	95
olive oil	711	239	95	27659

After reweighting with *Positive Pointwise Mutual Information (PPMI)* and reducing the number of dimensions to 20 using *Singular Value Decomposition (SVD)*, we obtain an ingredient space containing 6514 rows.

$$\mathrm{PPMI}(a,b) = \max\left(0, \log\frac{P(a,b)}{P(a)P(b)}\right)$$

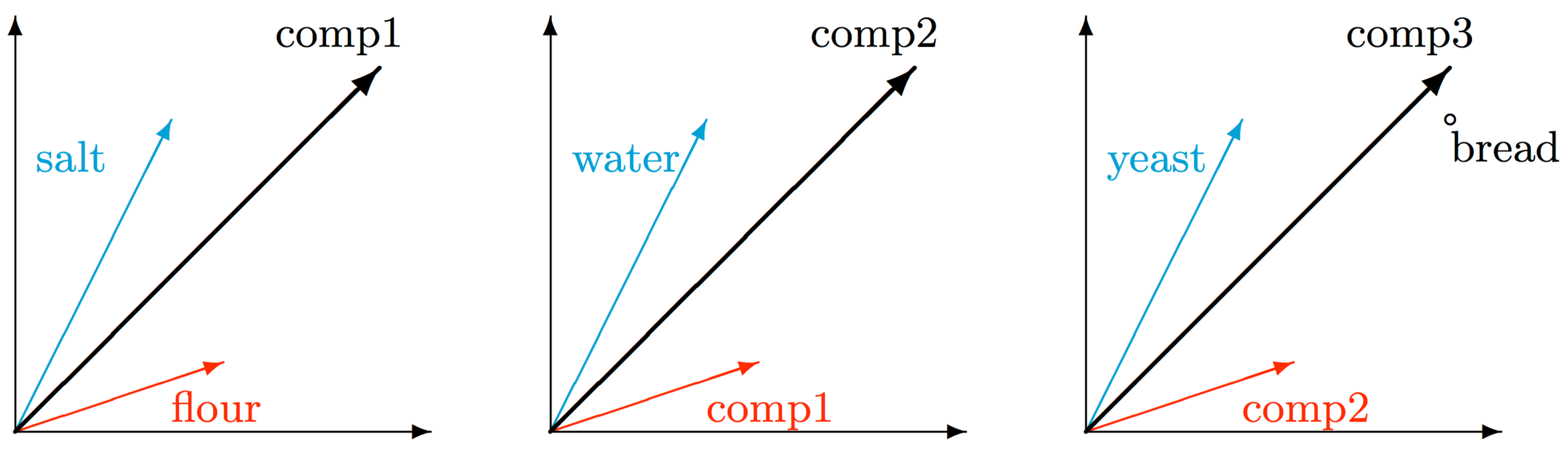
Using the same context, we also construct a recipe space (**BasicRecipes**) counting a recipe as co-occurring with each of its ingredients, in the same way, to obtain a compatible space.

2D projection



Composition

We can compose a given list of ingredients by summing up the vectors individually. In this way, we constructed a second recipe space (**ComposedRecipes**). Next, we can find similar recipe vectors to the one constructed by adding ingredient vectors.



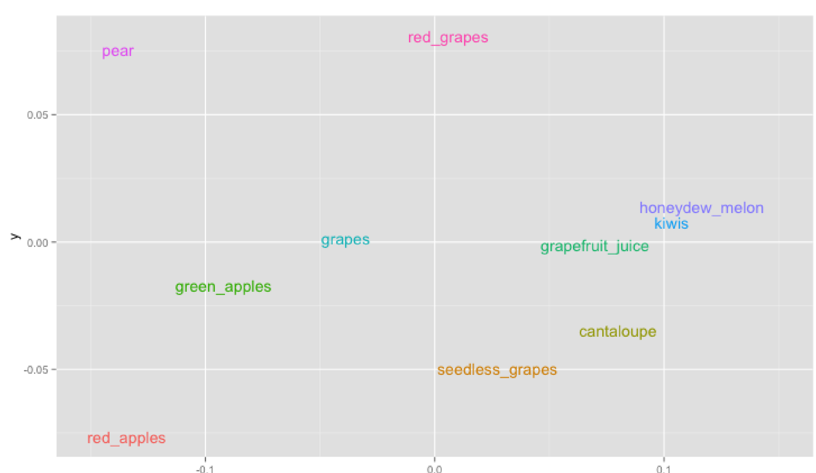
$$\overrightarrow{bread} \approx \overrightarrow{flour} + \overrightarrow{salt} + \overrightarrow{water} + \overrightarrow{yeast}$$

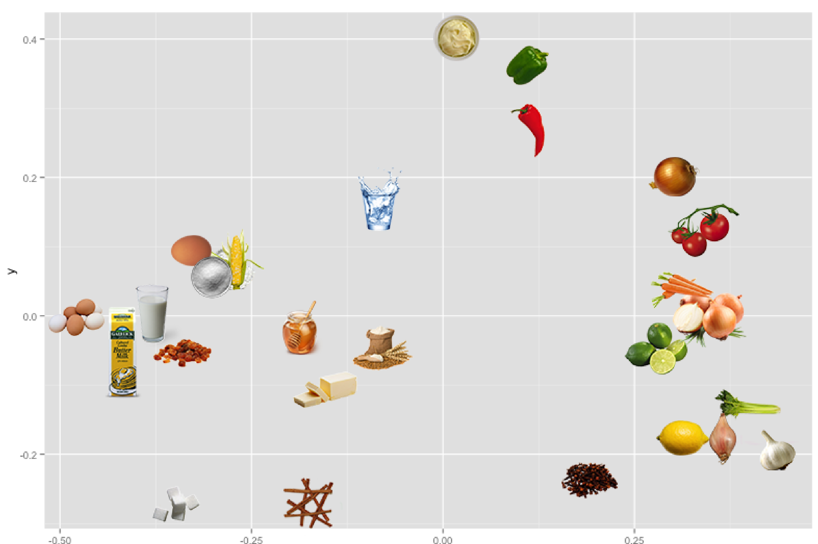
Results & Trying Evaluation

Subjectively evaluated, the resulting spaces seem to produce good results, when looking at the nearest neighbors of ingredients. As a similarity measure, we use the *cosine distance*:

$$cos(u,v) = \frac{< u,v >}{\sqrt{||u||\,||v||}}$$

Input ingredient	Nearest neighbors
flour	warm water, egg yolk, nutmeg
milk	melted butter, butter or margarine, eggs
mozzarella	basil, pasta, freshly grated parmesan
blueberries	frozen mixed berries, peaches, strawberries





Evaluation through **WordNet** was tried, but no significant link between our similarity and *Path distance* or *JCN distance* could be found. This is not very surprising, since WordNet's hierarchy is unrelated to how ingredients are used in cooking. For example, while *butter* and *margerine* are often used interchangably, they are very different things.

Two scripts were created to experiment with our semantic gastrospaces:

```
$ python nearestNeighbor.py 5
> honey
Nearest neighbors: maple syrup (0.894713303936), agave nectar (0.876596677544),
sliced almonds (0.864462421642), sunflower seeds (0.859026663599), dried cherrie
s (0.858817235489)
```

```
$ python makeMeARecipe.py 5
Enter ingredients, enter when done
> water
> orange juice
> basil
>
Nearest neighbors: citrus balsamic salmon (0.950769192732), cantaloupe-basil gra
nita (0.94599751641), balsamic glazed salmon (0.944641273507), ribeyes (0.943482
021276), strawberry and chicken pasta salad (0.942513766126)
```

Future Work

- Additional cleaning & unification
- Composition weighting based on ingredient amounts
- Experimenting with other composition methods, e.g. Multiplicative
- Finding better methods of automatic evaluation
- Make use of additional information in recipe data, e.g. cooking process
- Twitter-Bot or App for Make-Me-A-Recipe

Barcelona, August 3rd-14th
ESSLLI 2015
27th European Summer School in Logic, Language and Information