Knowledge-based recommendations for cocktail recipes

Sigurd Sippel
Hamburg University of Applied Sciences, Department of Computer Science,
Berliner Tor 7, 20099 Hamburg
sigurd.sippel@haw-hamburg.de

February 10, 2015

1 Introduction

The cocktail recommended by a bartender in a bar has to be appropriate for its ambience to the preferences of individual guests. The success of the bar depends on how appropriate the recommendation is. If the guest asks the bartender for a recommendation, the bartender then acts as a gatekeeper [PGK11, p. 105] for the cocktails for him because the bartender is limited by the given circumstances. The recommendation made by the bartender is precise, since he knows what to do. However, though he probably knows a large number of drinks, only a few are on his mind at any one point of time.

A bartender's recommendation serves as a metaphor for a knowledge-based recommendation approach, a thesis outline of which is considered in this paper. The cognitive processes of human beings and machines are different; so, only the performances and not the process will be validated. The aim is to make the machine better than the bartender.

A knowledge-based recommendation is based on features such as the ingredients, the preparation, and the glassware. Implicit personalization is modeled with the help of an exemplary favorite. This example is used to recommend cocktails which are close to this example.

The main question is: Does a knowledge-based distance function have sufficient precision for a cocktail recommendation? Recommendations for a specific domain — in this case cocktails — can only be validated by domain experts such as bartenders. A threshold for sufficient precision is that the recommendation is acceptable to domain experts.

[Sip14] describes the contextual and personalization aspects of recommender systems and the data analysis as a KDD process. This includes constructions of distance functions and data mining methods such as clustering. In particular, the relationship to the domain cocktails — such as cocktail balances — is revealed. This paper considers methods for developing cocktail recommendation systems and for validating their precision. General conditions, advantages and risks are part of every methodic step.

The section 2 describes the target data structure with the KDD process as its guidelines. The section 3 presents an approach for feature extraction with basic categories. Based on the categories, section 4 works with distances on the target structure and an approach to identify cross-linkages. The section 5 considers preprocessing of cocktail books. The section 6 includes a validation approach with domain experts. The last section contains the conclusions and future work.

2 Target structure

A recommendation connects an item, such as a cocktail, to a user. The aim is to satisfy him. That is why the center of data in a recommender system comprises users and items [RRSK10, p. 10]. From the perspective of a personalization process, there are two challenges [NM09, p. 7:1]. The first is the cold start problem, which says that a user profile has to exist and contain information such as preferences. The second is the overload problem, which says that a recommender system has to know how it connects the profile to the items. With such kinds of user data, it is possible to use collaborative filtering [RRSK10, p. 12], but without user data it is necessary to know something about the items to make a recommendation. This is a knowledge-based recommendation, which uses only the similarity between items. The user has to know the example of a favorite to get a similar, though not identical, item as a recommendation.

From the KDD perspective (Figure 1), the available data is the starting point. There are various sources for cocktail recipes. Books, both historical and new ones; magazines; blogs; and online cocktail databases. From the technical point of view, the cocktail database is the easiest thing. But there are no open interfaces and often, meta data is missing, such as author names, time stamps or description texts. [kin14] is an example, which presents recipes from some historic books, but the connection between recipes and books — including meta data — is missing. Only in a community-driven website, such as [coc14], there are often strongly similar recipes that have an excessive rate of sweetness or cream. Quality management is, unfortunately, missing.

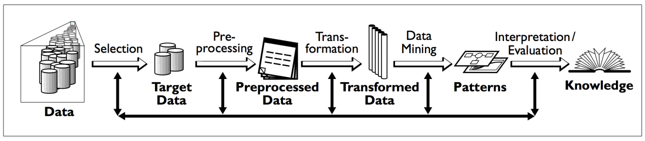


Figure 1: KDD process [FPSS96a, p. 29 Figure 1]

In the absence of an interface, the websites have to be parsed, so it is a better alternative to parse books or blogs directly. Meta information is available, content quality is controllable, and the authors are known. On the one hand, from the perspective of users, meta information is needed to classify recipes; and from the technical perspective, it helps to identify relationships in time and space. On the other hand, there is much unnecessary information, such as an introduction, or page numbers. A target structure is needed, which would show which information would have to be found in the preprocessing.

Cocktail books primarily contain recipes. Such a recipe contains a title, the names of the ingredients, and a list of ingredients. The ingredients are mostly described in terms of quantities, which can either be concrete units of measurement, or only proportions. Depending on whether the authors are US or British, the units can be imperial ones, US customary measurement systems, or metric. There is also additional information about the ways of preparation (whether to shake or stir the cocktail) and which glassware would be useful.

Manhattan Cocktail
(1882 Harry Johnson, Bartenders Manual p. 162)
1 dash of gum syrup, very carefully;
1 dash of bitters (orange bitters);
1 dash of curacao, if required;
1/2 wine glass of whiskey;
1/2 wine glass of sweet vermouth;
stir up well; strain into a fancy cocktail glass;

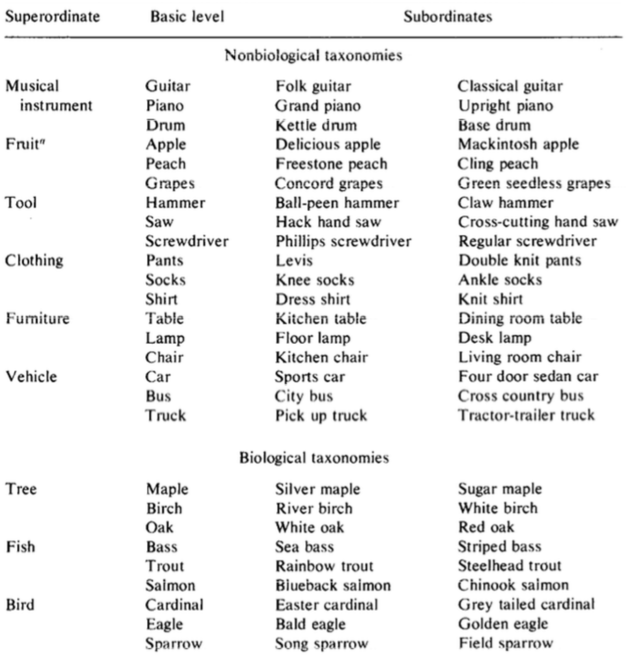
The main aspects of a target structure are features, which are useful for finding patterns in the recipe collection and information for presenting the extracted knowledge in a form that is readable to humans [FPSS96b, p. 39]. A list of ingredients with quantities, units, methods of preparation and the necessary glassware are useful for finding out similar recipes, since they describe the structure of a recipe. Further meta information, such as author or publication date, are useful for saying something about trends, such as alcohol strength related to time. A presentation that is readable to humans needs information such as title, the original names of ingredients, and also meta information. But this is not part of a recommendation. Data which is needed for identifying the similarity of two cocktails is considered in the next few chapters.

3 Extraction with background knowledge

Cocktail recipes primarily contain lists of ingredients, but also information about methods of preparation and the necessary glassware. From the semantic point of view, two ingredients that are not equal can share some properties. These properties have to be known to say something about the distance between two ingredients. For instance, *rye* and *bourbon* are special types of *american whiskey*, so they share properties like origin, manufacturing, barrel-aging, and color.

All organisms classify their environments [RMG$^+$76, p. 382]. In the world, huge quantities of information reach a person. These are called stimuli, which everyone has to process. A category assigns a name to a group of things that share important properties. A category is not arbitrary, since all things in the world depend on one another, though the strength of dependency can vary significantly. A *gin* and *whiskey* are both different kinds of *spirits*, but a *gin* not very similar to a *whiskey*. Both are unrelated to a *cocktail glass*. Of course, these can be associated with each other, but ingredients and drinking glasses belong to entirely different classes.

A taxonomy contains categories of the same class, which are organized as a root tree. The size of a category subtree depends on the level of abstraction. The category *spirits* contains *gin* and *whiskey*, but *gin* does not contain *whiskey*. The term *spirits* is an abstraction for the other ones. The level of abstraction which carries the most information is called the basic level [RMG^{+}76, p. 383].



Superordinate	Basic level	Subordinates	
		Nonbiological taxonomies	
Musical instrument	Guitar	Folk guitar	Classical guitar
	Piano	Grand piano	Upright piano
	Drum	Kettle drum	Base drum
Fruit"	Apple	Delicious apple	Mackintosh apple
	Peach	Freestone peach	Cling peach
	Grapes	Concord grapes	Green seedless grapes
Tool	Hammer	Ball-peen hammer	Claw hammer
	Saw	Hack hand saw	Cross-cutting hand saw
	Screwdriver	Phillips screwdriver	Regular screwdriver
Clothing	Pants	Levis	Double knit pants
	Socks	Knee socks	Ankle socks
	Shirt	Dress shirt	Knit shirt
Furniture	Table	Kitchen table	Dining room table
	Lamp	Floor lamp	Desk lamp
	Chair	Kitchen chair	Living room chair
Vehicle	Car	Sports car	Four door sedan car
	Bus	City bus	Cross country bus
	Truck	Pick up truck	Tractor-trailer truck
		Biological taxonomies	
Tree	Maple	Silver maple	Sugar maple
	Birch	River birch	White birch
	Oak	White oak	Red oak
Fish	Bass	Sea bass	Striped bass
	Trout	Rainbow trout	Steelhead trout
	Salmon	Blueback salmon	Chinook salmon
Bird	Cardinal	Easter cardinal	Grey tailed cardinal
	Eagle	Bald eagle	Golden eagle
	Sparrow	Song sparrow	Field sparrow

Figure 2: Classification with basic level object [RMG^{+}76, p. 388 Table 1]

A category, which has many implicit properties, helps say something about it. A large number of categories with small discriminations presents a very detailed perspective of the class. A basic category, such as a *car* (Figure 2), combines two categories; it is not too abstract, such as a *vehicle*, and not too detailed, such as a *sports car*. A basic category is a category on the basic abstraction level. It can be imagined as a picture and is probably tangible [RMG^{+}76, p. 406]. The more abstract category is called a superordinate and the more concrete category is the subordinate [RMG^{+}76, p. 385]. The tree depth is not limited, and so, for every subordinate, a refinement is possible [RMG^{+}76, p. 432].

In a new domain, the categories have to be detected. People do not find correlations where there is nothing; they can only find less than what there is [RMG^{+}76, p. 430]. One method of detecting this is to ask people what they see in a picture — and ask them which pictures they would put together under one category [RMG^{+}76, p. 416]. The first results are the basic object and the last are the superordinates. In the domain of cocktails, a randomly chosen person could categorize the picture of a bottle of gin as a bottle of some kind of alcohol. Experts can change the results because their knowledge has many more special properties [RMG^{+}76, p. 430]. But experts are most focused [RMG^{+}76, p. 432], such in one basic category of ingredient, which affects the richness of the details in the result. In this case, the categories are not balanced; the relationships differ in accordance with their relevance.

In a study of airplane classification (Figure 3) with people with and without expert knowledge, the recognition in accordance with superordinates was very similar, but on a basic level, experts' recognition was greater in accordance with superordinates. So experts are needed, but the balance of the results has to be investigated.

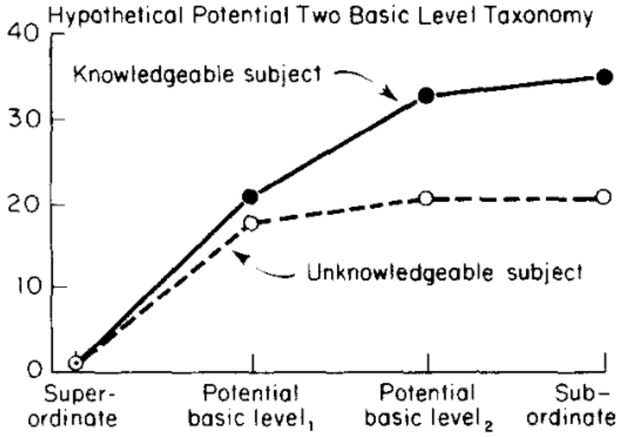


Figure 3: Airplane classification [RMG^{+}76, p. 431 Fig 4]

Now, *whiskey* is highly classified by country of origin, such as *Ireland* or the *US*, and then it is classified into ingredient-based categories, such as *bourbon* and *rye* for the *US*. *Gin* does not have such a detailed official classification for subordi-

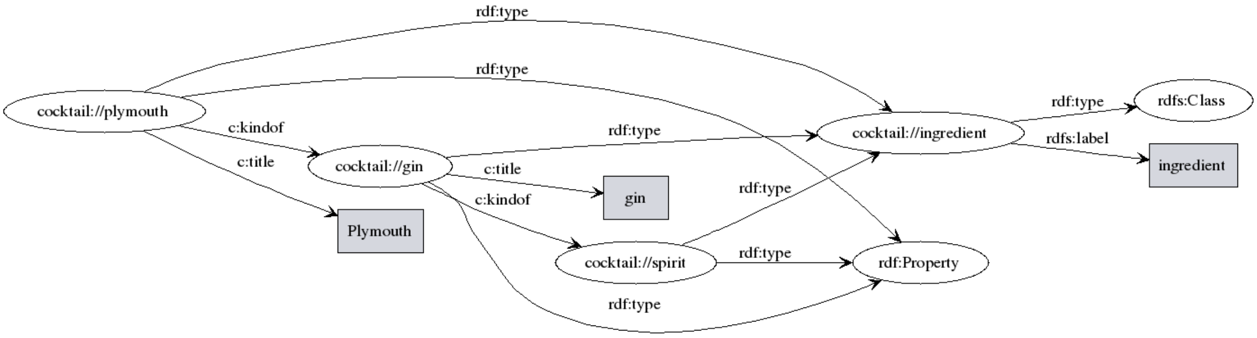


Figure 4: Categorization in RDF

nates. The relations are not completely equal to any relations with another parent category. Experts know that but cannot change this situation. Coloring the basic categories and weighting their subordinates are methods to balance knowledge, but the reasons are not objective.

Independent of experts, there are some basic objects, which are obviously a kind of superordinate, and there are basic objects that are not. From the perspective of gin experts, *London dry gin* is the most widely propagated kind of gin. Most gin products are subordinates of this. There are also very special gins, such as ones which are barrel-aged in peaty whiskey casks. Both are basic objects, but *London dry gin* has the most common properties in the category. It is a prototype of this category [RMG$^+$76, p. 433]. It represents the center of the category.

4 Ontology-based distances

Categories are simple ontologies. A popular ontology is Resource Description Framework (RDF) [RDF15], which is based on XML or Turtle [CLS01, S. 7]. It is a domain-independent description language, which can connect content. It can also separate content into several classes.

The RDF example Figure 4 includes a superordinate *spirits*, a basic category *gin* and a subordinate *Plymouth*. The RDF model contains a set of triples $(resource, property,\ atomic\ values)$. Instead of atomic values, such as labels or titles, there could also be other triples. This nested definition is used to model trees [CLS01, S. 10]. Every property can have a URI for ensuring a unique address. The property describes the edge which connects the left with the right one. There are predefined properties. Every resource has a type, which is referred to a class, such as *ingredient*. The self-defined property *kindof* allows one to model sub-categories.

In order to calculate the contextual distance between two recipes, components of the recipe — such as ingredients — have to be classified under categories. This is the transformation in KDD: from an unknown entity to a known one. In the absence of a classification, it is only possible to state whether the name of an ingredient name is the same as that of another. Categories identify similar properties.

The concrete product *Plymouth* is a kind of *gin*; they are not equal, but very similar. The difference is that *Plymouth*, as a subcategory, possesses some special characteristics, such as additional herbs, which are not in common with the parent category, but is a prototype of the parent category. If *Plymouth* were a very special gin, the similarity would decline.

Negroni	Negroni
3 cl gin	3 cl Plymouth
3 cl Campari	3 cl Campari
3 cl vermouth	3 cl Carpano
orange zest	orange zest

These two recipes are very similar, though the one on the left is a very abstract Negroni recipe, while the one on the right is a more concrete recipe, because it contains concrete products of *gin* and *vermouth*. In an ingredient with ontology as background knowledge, there are paths (Equation 1) of categories for each category, which represent super classes of it [CBC08, p. 562]. In this examples, the numbers of $steps(plymouth, gin)$ needed to get a common category is 1. Two concrete gins a, b needs two steps.

$$\begin{aligned} path(c) = c :: kindof(c) \\ :: kindof(kindof(c)) \\ :: kindof(kindof(kindof(c)))\ ...\ :: Nil \end{aligned} \tag{1}$$

These steps represent a distance. The interpretation of distance can be scaled (Equation 2). The maximum distance is scaled to 1 and the minimum distance is scaled to 0. If there are no steps, then the distance $steps = 0 \rightarrow distance(0)$. But steps are a kind of qualitative information, which is not scalable.

$$\frac{steps(x)}{maxsteps} \tag{2}$$

The interpretation has to be subjective. If the interpretation is that all steps are equal to another, then the high step sizes in detailed ontologies will imply distances close to 0. Assuming that the ontology contains enough information, the basic categories, such as *gin*, are not missing, and all concrete products, which are necessary, are categorized under the basic categories, the following interpretation is possible: the first step is very similar, but next step has to be a bigger step. The interpretation function would have to be maximized to 1. This interpretation ensures that the steps always have the same effect on a distance.

For comparing this approach, the path can be limited to $size = 2$, so that no long path would be possible. So the minimal distance (Equation 3) for one step is a controlled value. The limit hides extractable knowledge, but the non-linear interpretation could be tested against this.

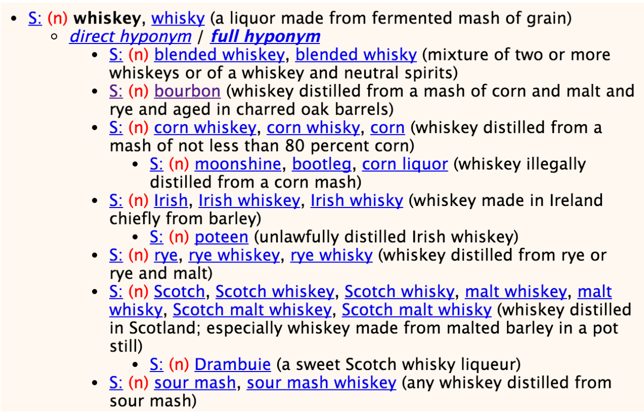
$$min.\ distance = 1\ step/(2 * limit) = 0.5 \tag{3}$$

This approach can be extended with colored categories, which have colors for categories such as *superordinate*, *basic*, and *subordinate*. Thus, the sum of all subordinates can be scaled to one step of a subordinate to a basic category.

A detailed ontology is important, but not every value is useful: superordinates, such as *spirits*, have a low validity [RMG+76, p. 385]. With it, the distance of *gin* and *absinthe* has a lower distance than $max = 1$. Different categories are categorized under one category that is not even imaginable — a superordinate — so it is better to hide superordinates for *kindof* operations.

The ontology have to know the ingredients and synonyms to find semantic similarities, but the world is always greater than one ontology. This is an important risk for the precision of the distance. There are huge ontologies, which contain indiscriminate categories. WorldNet (Figure 5) is one which also includes cocktail ingredients. WorldNet contains words with types of words, such as nouns (n), synonyms, and hyponyms (subcategories) [Mil95, p. 40]. But there are missing categories, too, such as *Japanese whiskey*.

Specialized databases, such as e-commerce databases of ingredient shops, contain more products and these are, perhaps, categorized. However, these databases have to be available and integrated into the ontology. Manual optimizing is necessary to precisely extract these features.

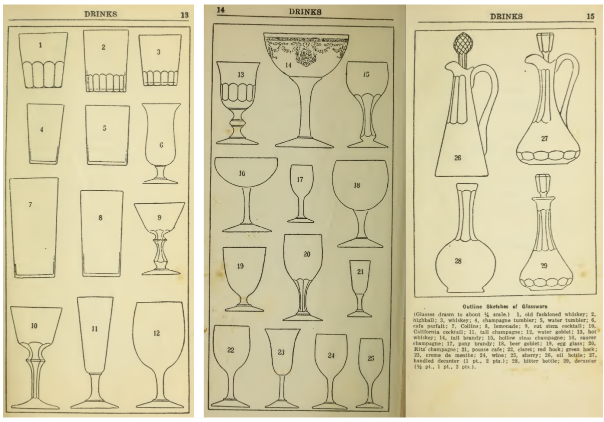


- S: (n) **whiskey**, whisky (a liquor made from fermented mash of grain)
  - *direct hyponym* / ***full hyponym***
    - S: (n) blended whiskey, blended whisky (mixture of two or more whiskeys or of a whiskey and neutral spirits)
    - S: (n) bourbon (whiskey distilled from a mash of corn and malt and rye and aged in charred oak barrels)
    - S: (n) corn whiskey, corn whisky, corn (whiskey distilled from a mash of not less than 80 percent corn)
      - S: (n) moonshine, bootleg, corn liquor (whiskey illegally distilled from a corn mash)
    - S: (n) Irish, Irish whiskey, Irish whisky (whiskey made in Ireland chiefly from barley)
      - S: (n) poteen (unlawfully distilled Irish whiskey)
    - S: (n) rye, rye whiskey, rye whisky (whiskey distilled from rye or rye and malt)
    - S: (n) Scotch, Scotch whiskey, Scotch whisky, malt whiskey, malt whisky, Scotch malt whiskey, Scotch malt whisky (whiskey distilled in Scotland; especially whiskey made from malted barley in a pot still)
      - S: (n) Drambuie (a sweet Scotch whisky liqueur)
    - S: (n) sour mash, sour mash whiskey (any whiskey distilled from sour mash)

Figure 5: WorldNet ontology [Wor15]

Preparations and glassware are very similar to ingredients, but the complexity is lower. Instead of a list of ingredients, there is only one preparation and one glassware item, which can be found in the ontology. Both need their own classes because, for example, no ingredient is mixed with a preparation. Preparation contains basically the terms *stir* and *strain* and, as a subcategory of *strain*, probably *build*, which means to stir directly in the glass that is used for drinking. That is only a practical difference. In the ontology, *build* is only necessary as a synonym of *strain* to prevent a too detailed ontology.

Glassware is another taxonomy. There are many names for many kinds of glasses (Figure 6), but they can be manually classified into a small number of raw figures, such as *highballs*, *tumblers*, *ballons*, *goblets*, or *cocktail glasses* (little bowls). The ontology contains only a few categories, but a lot of synonyms. This is important because behind a name, there could be a long story, but from the perspective of similarity, only the figure is important. Glasses are the most replacable part of a recipe. A missing synonym — or a name that is not unique — make the distance imprecise.



Outline Sketches of Glassware

(Glasses drawn to about ¼ scale.) 1, old fashioned whiskey; 2, highball; 3, whiskey; 4, champagne tumbler; 5, water tumbler; 6, cafe parfait; 7, Collins; 8, lemonade; 9, cut stem cocktail; 10, California cocktail; 11, tall champagne; 12, water goblet; 13, hot whiskey; 14, tall brandy; 15, hollow stem champagne; 16, saucer champagne; 17, pony brandy; 18, beer goblet; 19, egg glass; 20, Ritz champagne; 21, pousse cafe; 22, claret; red hock; green hock; 23, creme de menthe; 24, wine; 25, sherry; 26, oil bottle; 27, handled decanter (1 pt., 2 pts.); 28, bitter bottle; 29, decanter (½ pt., 1 pt., 2 pts.).

Figure 6: Glassware illustrations, p.13-15 in Drinks, Jacques Straub (1914)

4.1 Quantity-based weighting

If a cocktail recipe is considered as a simple kind of recipe, then the ingredients are the most important part of the recipe [KLSL12, p. 2]. In a Negroni recipe, an *orange zest* is less important than 6 *cl gin*, so a quantity-based weighting is needed. The quantities have to be comparable. For recognition of the measurement units, another taxonomy is necessary. It needs a factor which is usable to convert to a standard unit, such as *cl*, to convert a unit, such as *ounce*, easily (Equation 4).

$$1\ ounce = 3\ cl \tag{4}$$

These units are quantitative units, which are scalable. Another kind of units comprises qualitative units, such as *dash* or *piece*, which are not scalable. A standard factor — that means 1 *dash* = 1 *cl* — is a simple solution, but the intensity of the two recipes is not equal. A dash of *absinthe* changes a recipe, but a dash of *sugar syrup* does not do that. An intensity score in the ingredient ontology is able to differentiate, but this score is a manually and subjectively additional information. That is a limit of an ontology.

Nevertheless, the sum of all ingredients with a quantitative unit, represents the absolute cocktail size, so it is possible to calculate the ratio of one ingredient, which represents the weight of this ingredient.

There are also recipes without measurement units; these recipes contains only ratios. Ratios are usable directly as the ingredient weight. In the real world, the recipe has to be filled in a glass with a definite size. This ingredient distance is independent of absolute size; the cocktail glass implicitly represents the size of a cocktail. A *highball* is much taller than *cocktail glass*. Of course, it is not an exact specification, but it does say something about the dimension.

4.2 Cross-connections with balance

If recommendations are made by example, a user who knows such examples is required. This is an implicit form of personalization, since the example is user-specific. The drawback is that the recommendation will only yield those obvious results that the user already knows. This is a kind of filter bubble [Par11]. The recommendations would have to yield more than those obvious results; it needs to have cross-connections.

The similarity between two recipes, which only have differences in their choice of another kind of subordinate, such as another gin product, can be recognized with the help of categories. Another property shared by two recipes is the abstract idea. A Negroni, for instance, is a old classic cocktail. It was created a long time ago and, since then, it has been adapted, and will be adapted again in future as well.

classic	adaption
3 cl gin	3.5 cl mezcal
3 cl Campari	2 cl Gran Classico
3 cl vermouth	3.5 cl Carpano
orange zest	

At first glance, the recipes are not very similar. The distance $d(vermouth, carpano)$ is the only one which is smaller than 1 (= no similarity). The item *orange zest* does not exist in the adaption. Again, *gin* and *mezcal* share only their superordinate *spirits*, *Campari* and *Grand Classico* share their superordinate *liquor*. Superordinates are hidden. The last ones are also *bitter*, but at this point, a *vermouth* is also *bitter*. Its validity is too low. In the structure, measured with

$distance(classic, adaption)$, the *classic* and the *adaption* are not very similar.

$$balance = (c(alcohol), c(sugar), c(acid), c(water))) \tag{5}$$

Based on the nutritional food balance [KF10], such as meat or fish ratio, a cocktail balance (Equation 5) is also possible. With an extended ontology of ingredients, which contains the ratio of characteristic categories, such as of sweet, sour, water and alcohol, a cocktail balance can be computed. It is the sum of all ingredient ratios for each category.

The balances of the chosen examples (Equation 6) are very similar. Of course, since there is no minimum or maximum value available, the interpretation is not precise. A minimum and a maximum value of each characteristic category from a huge database with a large variation of recipes is useful for obtaining a better interpretation.

$$\begin{aligned} classic = \{&3\ cl\ gin\ (0.47\ alcohol, 0.53\ water), \\ &3\ cl\ Campari\ (0.25\ alcohol, 0.12\ sugar, 0.63\ water), \\ &3\ cl\ vermouth\ (0.18\ alcohol, 0.12\ sugar, 0.7\ water)\} \\ adaption = \{&3\ cl\ mezcal\ (0.4\ alcohol, 0.6\ water), \\ &2\ cl\ Gran\ Classico\ (0.28\ alcohol, 0.15\ sugar, 0.57\ water), \\ &3.5\ cl\ Carpano\ (0.18\ alcohol, 0.14\ sugar, 0.68\ water)\} \\ balance(classic) &= (0.281, 0.094, 0, 0.62) \\ balance(adaption) &= (0.307, 0.08, 0, 0.643) \end{aligned} \tag{6}$$

Ontologies need to contain information about the balance of each ingredient, but such information is not always available. Spirits usually do not contain sugar and sourness, but rum sometimes does contain sugar. In contrast to alcohol strength, it is not necessary to declare the sugar ratio. If this information is declared, it could be missing in the ontology as well.

Default logics [Rei80] use default values to reach conclusions based on what is known. If it is only known, for instance, that x is a kind of *spirits*, it will use the value of spirits (Equation 7). The M predicate says that it can be assumed [Rei80, p. 82], which means that there is enough information to draw this conclusion. But this is not always true in the real world; it is only as good as one's own knowledge.

$$\frac{\text{ingredient(x): M kindof(x,spirits)}}{c(alcohol){=}0.4 \wedge c(sugar){=}c(sour){=}0 \wedge c(water){=}0.6} \tag{7}$$

If the world that is modeled is changed and the sugar ratio of x is also known, the conclusion contains the sugar ratio of x and beyond that, the rest of the known information (Equation 8).

$$\frac{\text{ingredient(x): M kind of(x,spirits)} \wedge \text{c(sugar)} = 0.08}{c(alcohol){=}0.40 \wedge c(sugar){=}0.08 \wedge c(sour){=}0 \wedge c(water){=}0.6} \tag{8}$$

4.3 Testing with preprocessed examples

In order to test the feature extraction, it does not make sense to start with a big cocktail book. Examples of manually preprocessed recipes from such cocktail books helps to prepare simple tests, since any error in preprocessing is hidden. The recipes are presented in a readable format, such as XML. Each recipe is separated into ingredients, preparations and glassware; the ingredients are also separated into name, quantity value and unit. Every single part can used to extract features. Every missing value can be added, step by step, to the ontology. In the next step, simple distances, such as ingredient distance, preparation distance, glassware distance, balance distance and their combination, can be tested. The following tests are based on assumptions that have to be validated; the combined distance is called distance.

If the distances work in a small number of examples, the the collection of examples is extendable. In order to have recommendations based on examples of favorites, a result is expected that contains similar, but not nearly equal recipes. This requires clusters that represent recipes that are nearly equal. Recipes in neighboring clusters can serve as potential recommendations. The intuitive distances of humans [JMF99, p. 268] are not as accurate as automatically computed ones. Nevertheless, human experts are the threshold for this recommendation.

In order to test distance, collections of nearly equal recipes show whether the distances work out as expected. The distances have to be near to 0. Also, such a collection has to have a high distance from an different one. These distances have to be close to 1. Historic cocktails, such as a *Negroni*, are called classics, because they have been isolated from each other for a long time. There cannot be two different classics that are nearly the same. Of course, the determination of classics is not always obvious.

In order to collect nearly equal recipes within a single classic, a minimum distance is required that is clearly higher than 0 to all other collections within a single different classic. In fact, these classic collections are hand-made clusters. If that does not occur, one of the classic choices would be unwise, or the distance would be unwise. At this point, the expert would have to decide which result is acceptable.

The test can be extended with a collection of adaptions for each classic, which have a balance distance close to 0 to the classic. The combined distance have to be bigger than each combination in the classic cluster and smaller than the other classic collections. The hand-made clusters have to be detected on the basis of these clustering methods. Unpredicted clusters represent new correlations that have not been predicted by experts, or have errors in their distances.

5 Preprocessing

Historical recipe books are chosen as the sources of preprocessing. Several examples in PDF format are available [His15]. Each of these books contains content such as title pages, introduction, index, explanation of preparations, lists of ingredients, glassware, a lot of recipes, and a few pages at the end. Since PDF is a display-oriented format, it contains physical structures, such as lines or columns, instead of logical structures, such as headings, sections or recipes [GTL+11, p. 11]. These PDFs are scanned books, saved as images. Also, there is a lot of errors in the content, since these texts were added with the help of optical character recognition (OCR).

The preprocessing overview contains several steps (Figure 7); the extraction of the physical structure recognizes characters, such as words and lines. The detection of the global typography provides information about headers, footers, page body and/or fonts. When combined, these help in separation, because similar content probably has similar typography.

At the page level, the page element labeling classifies a piece of content under a logical element, such as a heading. Usually, a recipe starts with a heading, which contains the name of the recipe. The detection of the reading order finds a logical order of content, which, in PDF format, is only ordered in a technical way and positioned on the page. Associating figures, such as images, with their captions, is necessary, because a caption helps the reader understand the figure. But in these OCR PDFs, figures are rare. They are also unimportant, since there are no recipes there.

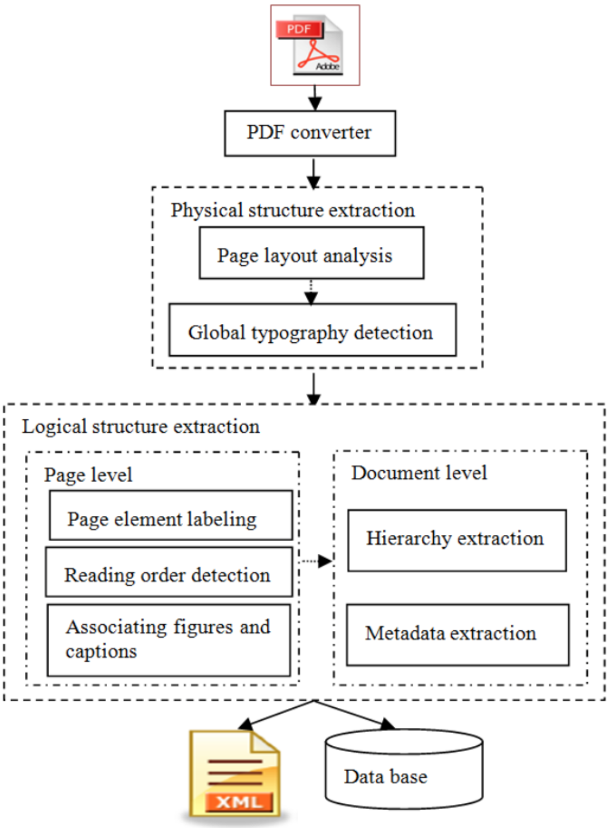


Figure 7: Preprocessing overview [GTL+11, p. 13 Figure 1]

On the basis of the structures that are found, such as headers on the page level, book hierarchies can be extracted. In connection with meta data, such as title, publisher and copyright, this is an interesting aspect for visualization of knowledge. In this case, the page level is important, as it requires domain-specific knowledge [GTL+11, p. 11]. Without any idea of what a recipe is, it cannot be recognized.

The next example contains two recipes without any preprocessing. These follow the following rules: titles are in capitals; for titles, ingredients and preparations/glassware, only a single line is used. All lines end with a point. A title is an exception. Ingredients have a quantity as a fraction; a blank space follows, and then the name comes. A single character that is not a number makes no sense. With these rules, it can be parsed; many other recipes of this book can also be parsed. But these rules are strongly dependent on this book. These rules work only if one knows that these are recipes and there is no one such an index. There is a risk in preprocessing that the recipes cannot be found.

> (1937 W. J. Tarling, CAFE ROYAL p. 127)
> MANHATTAN, DRY
> 1/2 French Vermouth.
> 1/2 Rye or Bourbon Whisky.
> Stir and strain into cocktail glass.
> MANHATTAN, MEDIUM
> f
> 1/3 Rye or Bourbon Whisky.
> 1/3 French Vermouth.
> 1/3 Italian Vermouth.
> Stir and strain into cocktail glass, with cherry. A dash of Angostura can be added if desired.

The target structure does not need a grammatical context; it requires single words, such as names of ingredients or their quantities. The removal of the stop words and the use of stamming are necessary to make the raw text cleaner. Stamming is used to find base forms, such as *stirred* → *stir*.

This ontology is not only usable for extracting features, it can also be used to find relevant words in the raw book. If the frequency of names of preparations, glassware, measurement units and ingredients are very high, there will be a recipe. This acts as a clustering approach for separating recipes. The extraction of the recipe position is important for success in preprocessing because the next few possible steps are very few. With a method of elimination, all the known names can be mapped; only the unknown data could cause errors.

Another approach is to find common spellings with specific rules. In particular, detailed information about an ingredient is placed in brackets, such as alcohol strength, ingredient category or a product recommendation (Equation 9). Such rules reduce ambiguity in information about ingredients, because from perspective of single words there are two ingredients.

$$\begin{aligned} gin\ (Plymouth) &\rightarrow (Plymouth \rightarrow gin) \\ Plymouth, 41\%\ (LondonDryGin) & \\ &\rightarrow (Plymouth \rightarrow gin) \\ Plymouth\ (Gin) &\rightarrow (Plymouth \rightarrow gin) \end{aligned} \tag{9}$$

In remembering an existing feature extraction process, a bottom-up approach can be used to carry out tests, step by step. The recognition of different spellings is the simplest problem. Recipes in XML structure, which have more complex spellings of ingredients, or a combined spelling of preparations and glassware, such as in the examples of the Manhattan recipes, can be preprocessed, and their features extracted. In the next step, single recipes can be preprocessed, probably with more and more unknown characters or words. In the following step, a collection of recipes, and in the last step, a complete book, can be preprocessed. If automatic preprocessing steps fail, manually preprocessing is always possible.

6 Validation by domain experts

From the perspective of clusters, the distribution of clusters and cluster sizes [SZ15, p. 1251] helps one obtain a feel of the diversity of collected data, but a recommender system has to be related to the user. Experiments without users cannot validate a recommender system. Experiments require feedback mechanisms, which can be used to obtain precise measurements.

Such feedback is very important for testing the validity. Clicking behavior [ANH13, p. 168] on the list of recommendation results will indicate which recommendation is being watched. User ratings [LWL14, p. 101] show how satisfied a user is with an item. In both solutions, a relation ship with the initial favorite example is missing. Results which serve to attract attention in an emotional way, or when the user is hungry for knowledge, have a low validity in such feedback. Recordings of the uses of the recommendation, such as video recommendations [BMCMB+10], have greater validity. If the user watches a long, full-length video, then that is an important piece of information for the user profile. However, there is no relationship with a favorite example.

Such recommendations are domain-dependent, which have to be made in accordance with expert knowledge [SG11, p. 3]. In order to know how precise a recommendation is, it is necessary to ask a domain expert. A recommender system and three experts, who are isolated from each other, can make one recommendation each for a single example; if all recommend the same, then it is a precise recommendation. Experts have different areas of interest, [McD83, p. 105], and so, their focuses are different, which affects the recommendation. Such a validation have to fail. If the result of each recommendation would have to be acceptable only to each expert, then every expert could have a different opinion, but could agree.

A result have to be comparable to a competitive solution [SG11, p. 3]: A validation needs a hypothesis, such as the recommender systems, which would be better than recommendations by bartenders. There are controlled variables, such as a static testing set and variables, which are focused on the test. The last is the generalization power, which shows how stable the conclusions are in different contexts.

There are three types of experiments [SG11, p. 10]: Offline experiments with a static testing set and feedback, such as by domain experts, can be used to test whether an expected adaption is in the example-based recommendation. In a user study, the expert would use the recommender system directly; it results in feedback about the use case, understandability, and the expected results for the testing set. A user study is only a qualitative measurement. It has no statistical significance. Online evaluations are used by real users, such as bartenders, for real tasks, such as for guests who are seeking recommendations. In such cases, direct feedback is missing, and so a common online evaluation or user study is not possible. It is not that only domain experts use the recommender system; therefore, only experts are able to rate a long-term study.

In case offline experiments are chosen: A specific group of domain experts — such as bartenders or bloggers — will be shown a set of recipe pairs. The first is the example and the second the potential recommendation. The domain experts are able to rate the validity of the recommendation with a numeric scale (Equation 10).

$$\left[\underset{\text{(unacceptable)}}{-3}, -2, -1, \underset{\text{(appropriate)}}{0}, 1, 2, \underset{\text{(obviously)}}{3}\right] \tag{10}$$

Domain experts have different kinds of backgrounds and experience. Some may be working as bartenders, others may be connoisseurs during their leisure time. Bartenders would be more focused on well-known and easily made drinks, while connoisseurs would focus more on experimental drinks. Again, experiences with respect to time or variety could be very different. Therefore, it is necessary to test them with hand-made pairs. Both types of pairs contain appropriate, obvious and unacceptable ones. Assuming that enough domain experts can be motivated and enough usable ratings are taken, the results show the precision of the chosen distance function, in accordance with the experts' knowledge.

7 Conclusion and future work

There are three main challenges associated with this approach. The first is the knowledge stored in an ontology; it is the key for a cocktail recommendation. Therefore, the quality of ontology needs to be the focus in every step of its development. Distance functions cannot be a miracle. Missing or imprecise information entail the major risks for this approach, since unrecognized data is lost data.

The second challenge relates to the preprocessing, with which the selected target data — the cocktail books — has to be understood. These books contain information for the target data structures: The recipes. If the preprocessing does not work, huge volumes of data cannot be processed. It needs expensive manual preprocessing.

The third challenge is the validation. If a recommendation is not carried out in accordance with expert knowledge, then the recommendation would be useless. From the perspective of development, it is not sufficient to think that this is precise enough; the recommendation would have to compete in the real world. This challenge will depend on the motivation of the experts who will provide support with their knowledge. They will have to understand how this approach can affect.

Future work will contain these three challenges for answering the primary question: Whether or not a knowledge-based distance function will have enough precision. Assuming that the precision is good enough, the next challenge is personalization. An online recommendation service collects user data, which implies that the cold start problem is solved. Users and items are connected; so, a collaborative filtering approach for improved precision would come into range.

References

[ANH13] Al-Nazer, Ahmed ; Helmy, Tarek: Semantic Query-manipulation and Personalized Retrieval of Health, Food and Nutrition Information. In: *Procedia Computer Science* 19 (2013), Nr. 0, 163 - 170. http://dx.doi.org/10.1016/j.procs.2013.06.026. – DOI 10.1016/j.procs.2013.06.026. – ISSN 1877–0509. – The 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013), the 3rd International Conference on Sustainable Energy Information Technology (SEIT-2013)

[BMCMB$^+$10] Barragáns-Martínez, Ana B. ; Costa-Montenegro, Enrique ; Burguillo, Juan C. ; Rey-López, Marta ; Mikic-Fonte, Fernando A. ; Peleteiro, Ana: A hybrid content-based and item-based collaborative filtering approach to recommend {TV} programs enhanced with singular value decomposition. In: *Information Sciences* 180 (2010), Nr. 22, 4290 - 4311. http://dx.doi.org/10.1016/j.ins.2010.07.024. – DOI 10.1016/j.ins.2010.07.024. – ISSN 0020–0255

[CBC08] Cantador, Iván ; Bellogín, Alejandro ; Castells, Pablo: Ontology-Based Personalised and Context-Aware Recommendations of News Items. In: *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*. Washington, DC, USA : IEEE Computer Society, 2008 (WI-IAT '08). – ISBN 978–0–7695–3496–1, 562–565

[CLS01] Candan, K. S. ; Liu, Huan ; Suvarna, Reshma: Resource Description Framework: Metadata and Its Applications. In: *SIGKDD Explor. Newsl.* 3 (2001), Juli, Nr. 1, 6–19. http://dx.doi.org/10.1145/507533.507536. – DOI 10.1145/507533.507536. – ISSN 1931–0145

[coc14] *cocktaildatenbank*. Online visit (15.12.2014) cocktaildatenbank.de : Website, 2014

[FPSS96a] Fayyad, Usama ; Piatetsky-Shapiro, Gregory ; Smyth, Padhraic: The KDD Process for Extracting Useful Knowledge from Volumes of Data. In: *Commun. ACM* 39 (1996), November, Nr. 11, 27–34. http://dx.doi.org/10.1145/240455.240464. – DOI 10.1145/240455.240464. – ISSN 0001–0782

[FPSS96b] Fayyad, Usama M. ; Piatetsky-Shapiro, Gregory ; Smyth, Padhraic: Advances in Knowledge Discovery and Data Mining. Version: 1996. http://dl.acm.org/citation.cfm?id=257938.257942. Menlo Park, CA, USA : American Association for Artificial Intelligence, 1996. – ISBN 0–262–56097–6, Kapitel From Data Mining to Knowledge Discovery: An Overview, 1–34

[GTL$^+$11] Gao, Liangcai ; Tang, Zhi ; Lin, Xiaofan ; Liu, Ying ; Qiu, Ruiheng ; Wang, Yongtao: Structure Extraction from PDF-based Book Documents. In: *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*. New York, NY, USA : ACM, 2011 (JCDL '11). – ISBN 978–1–4503–0744–4, 11–20

[His15] *Historic cocktail book collection*. Online visit (16.01.2015) http://www.euvs.org/en/collection/books : Website, 2015

[JMF99] Jain, A. K. ; Murty, M. N. ; Flynn, P. J.: Data Clustering: A Review. In: *ACM Comput. Surv.* 31 (1999), September, Nr. 3, 264–323. http://dx.doi.org/10.1145/331499.331504. – DOI 10.1145/331499.331504. – ISSN 0360–0300

[KF10] Karikome, Shihono ; Fujii, Atsushi: A System for Supporting Dietary Habits: Planning Menus and Visualizing Nutritional Intake Balance. In: *Proceedings of the 4th*

International Conference on Uniquitous Information Management and Communication. New York, NY, USA : ACM, 2010 (ICUIMC '10). – ISBN 978–1–60558–893–3, 56:1–56:6

[kin14] *kindredcocktails.* Online visit (15.12.2014) kindredcocktails.com : Website, 2014

[KLSL12] Kuo, Fang-Fei ; Li, Cheng-Te ; Shan, Man-Kwan ; Lee, Suh-Yin: Intelligent Menu Planning: Recommending Set of Recipes by Ingredients. In: *Proceedings of the ACM Multimedia 2012 Workshop on Multimedia for Cooking and Eating Activities.* New York, NY, USA : ACM, 2012 (CEA '12). – ISBN 978–1–4503–1592–0, 1–6

[LWL14] Li, Xin ; Wang, Mengyue ; Liang, T.-P.: A multi-theoretical kernel-based approach to social network-based recommendation. In: *Decision Support Systems* 65 (2014), Nr. 0, 95 - 104. `http://dx.doi.org/10.1016/j.dss.2014.05.006`. – DOI 10.1016/j.dss.2014.05.006. – ISSN 0167–9236. – Crowdsourcing and Social Networks Analysis

[McD83] McDermott, John: Extracting Knowledge from Expert Systems. In: *Proceedings of the Eighth International Joint Conference on Artificial Intelligence - Volume 1.* San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1983 (IJCAI'83), 100–107

[Mil95] Miller, George A.: WordNet: A Lexical Database for English. In: *Commun. ACM* 38 (1995), November, Nr. 11, 39–41. `http://dx.doi.org/10.1145/219717.219748`. – DOI 10.1145/219717.219748. – ISSN 0001–0782

[NM09] Noor, Salma ; Martinez, Kirk: Using Social Data As Context for Making Recommendations: An Ontology Based Approach. In: *Proceedings of the 1st Workshop on Context, Information and Ontologies.* New York, NY, USA : ACM, 2009 (CIAO '09). – ISBN 978–1–60558–528–4, 7:1–7:8

[Par11] Pariser, Eli: *The filter bubble: What the Internet is hiding from you.* Penguin UK, 2011

[PGK11] Pinxteren, Youri van ; Geleijnse, Gijs ; Kamsteeg, Paul: Deriving a Recipe Similarity Measure for Recommending Healthful Meals. In: *Proceedings of the 16th International Conference on Intelligent User Interfaces.* New York, NY, USA : ACM, 2011 (IUI '11). – ISBN 978–1–4503–0419–1, 105–114

[RDF15] *RDF - Resource Description Framework.* Online visit (06.01.2015) www.w3.org/RDF : Website, 2015

[Rei80] Reiter, Raymond: A logic for default reasoning. In: *Artificial intelligence* 13 (1980), Nr. 1, S. 81–132

[RMG$^+$76] Rosch, Eleanor ; Mervis, Carolyn B. ; Gray, Wayne D. ; Johnson, David M. ; Boyes-Braem, Penny: Basic objects in natural categories. In: *Cognitive Psychology* 8 (1976), Nr. 3, 382 - 439. `http://dx.doi.org/10.1016/0010-0285(76)90013-X`. – DOI 10.1016/0010–0285(76)90013–X. – ISSN 0010–0285

[RRSK10] Ricci, Francesco ; Rokach, Lior ; Shapira, Bracha ; Kantor, Paul B.: *Recommender Systems Handbook.* 1st. New York, NY, USA : Springer-Verlag New York, Inc., 2010. – ISBN 0387858199, 9780387858197

[SG11] Shani, Guy ; Gunawardana, Asela: Evaluating recommendation systems. In: *Recommender systems handbook.* Springer, 2011, S. 257–297

[Sip14] Sippel, Sigurd: Recommendations for cocktail recipes. (2014). http://users.informatik.haw-hamburg.de/~ubicomp/projekte/master2014-aw2/sippel/bericht.pdf

[SZ15] Santos, Tiago R. ; Zárate, Luis E.: Categorical data clustering: What similarity measure to recommend? In: *Expert Systems with Applications* 42 (2015), Nr. 3, 1247 - 1260. http://dx.doi.org/10.1016/j.eswa.2014.09.012. – DOI 10.1016/j.eswa.2014.09.012. – ISSN 0957–4174

[Wor15] *Worldnet ontology example.* Online visit (16.01.2015) http://wordnetweb.princeton.edu/perl/webwn?s=whiskey&i=1&h=1000#c : Website, 2015