

# Version Spaces Task

Thor Galle

November 2015

## 1 Problem description

A start-up internet firm connects takeaway restaurants to hungry customers through an online platform. People can browse their website to find and order a meal that suits their appetite.

The firm keeps a database with information about the transactions conducted through the platform. Per order of a particular customer, it tracks among others the type of food ordered, the cost, the restaurant ordered from and the time that was needed to prepare the dish. A day after the order, it prompts the customer for a review score via email.

The platform faces competition, so it tries to differentiate itself by providing a new service: based on the order history of a customer and the accompanying reviews, it will suggest future orders. Their software systems engineer decides to use the Version Spaces algorithm to learn the taste of a customer.

For simplification, some assumptions are made:

- An order is limited to one food unit.
- The cuisine hierarchy contains only the most important categories, specific dishes are implicitly generalised to these, e.g. *Penne all'Arrabbiata* is a Pasta and a *Big Mac* is a Hamburger.
- We assume that the client always gives feedback, and that he considers all factors in an order to form his review score. These are: the meal itself, the meal cost, the restaurant proximity (distance he had to travel) and the preparation time (wait time).<sup>1</sup>
- If the review score was 3.5/5 or above, we assume the order was appreciated. Otherwise, it might have been acceptable, but we only want to generate recommendations based on really successful orders.

We consider a student whose order history is as follows:

Food	Restaurant	Cost	Preparation Time	Review Score (/5)
Nachos	Veracruz	€10	2h	1
Pizza Pepperoni	Sole Mio	€9	1h	3.5
Big Mac	McDonald's	€2	5min	3
Bami Chicken	Golden Garden	€4	30min	2.5
Pasta Pesto	Sole Mio	€7	15min	3.5
Fries w/ meat stew	De Brug	€13	1h	5

---

<sup>1</sup>The VS algorithm may output an unrealistic concept, this is only a guideline for recommendations.

<sup>2</sup>One could argue that the kebab came from Turkey, therefore it is half European.

## 1.1 Concept hierarchies

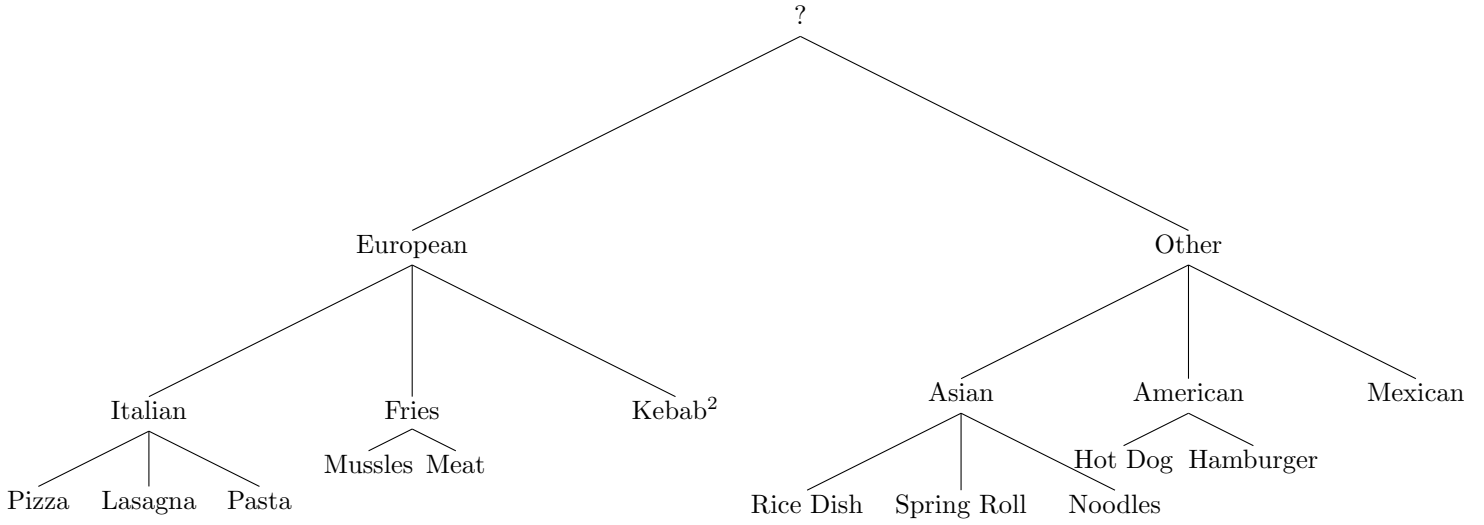


Figure 1: Cuisine hierarchy

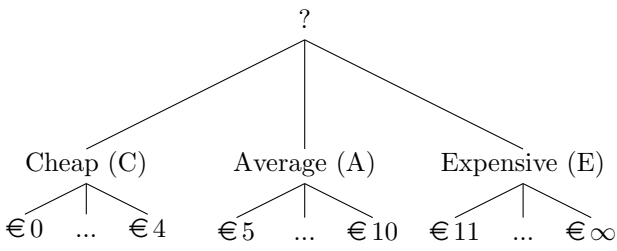


Figure 2: Cost hierarchy

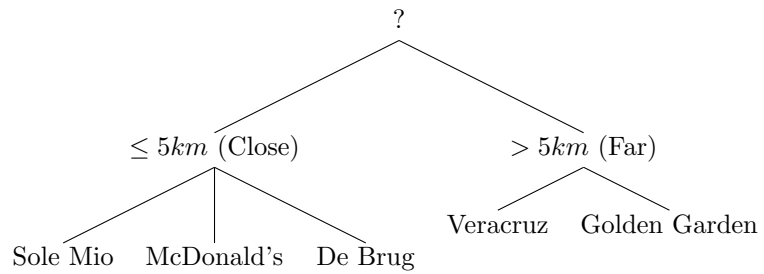


Figure 3: Restaurant proximity hierarchy

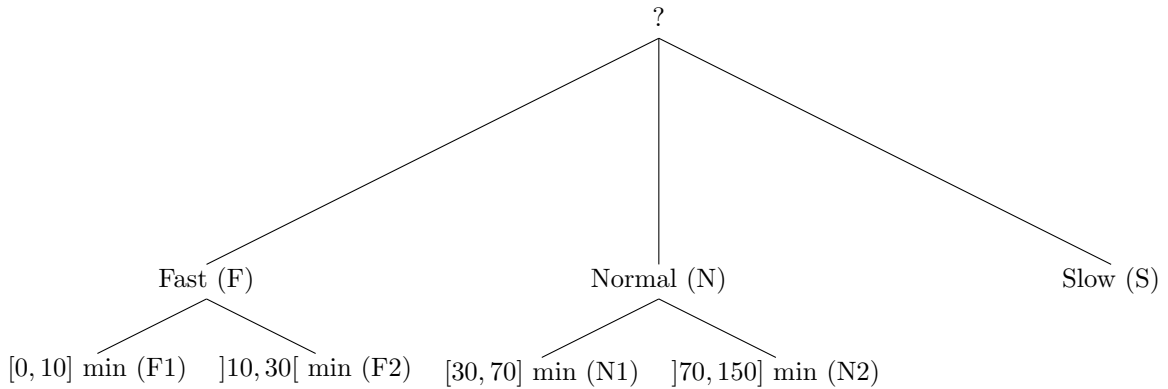
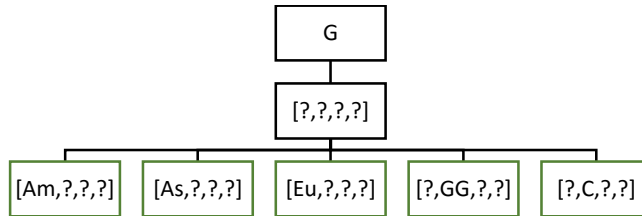


Figure 4: Preparation time hierarchy

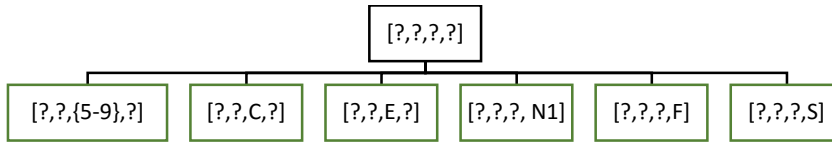
Legend:

- **Yellow outline:** minimal specialization that can't be added because it does not generalize the specific model. (only shown extensively for ex. 3)
- **Green outline:** minimal specialization / generalization that can be added.
- **Blue outline:** can't be added because it is a specialization of some other general hypothesis.
- **Red outline:** is pruned because it is not covering a positive example

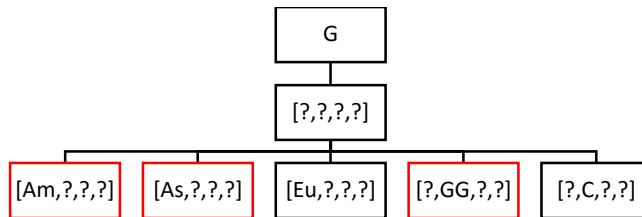
Example 1: [Mx, VC, 10, N2, -]



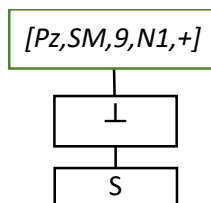
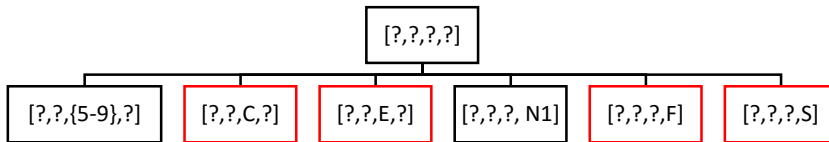
Continued:



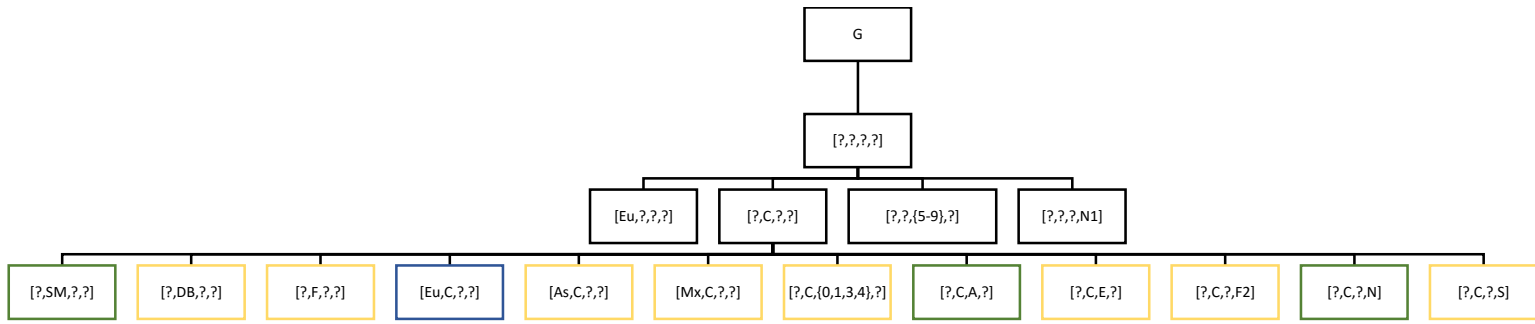
Example 2: [Pz, SM, 9, N1, +]



Continued:

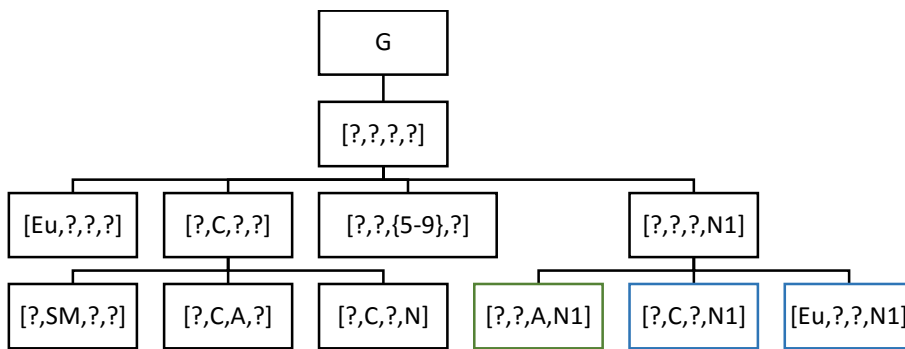


Example 3: [Hm, MD, 2, F1, -]



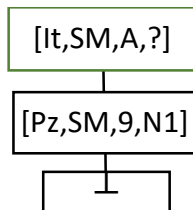
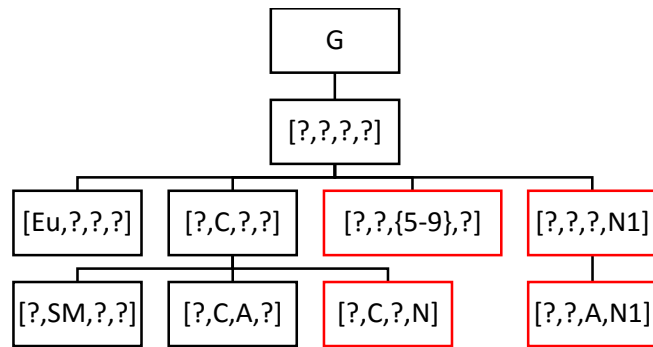
(nothing changes in S)

Example 4: [Nd, GG, 4, N1, -]

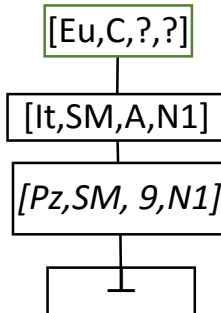
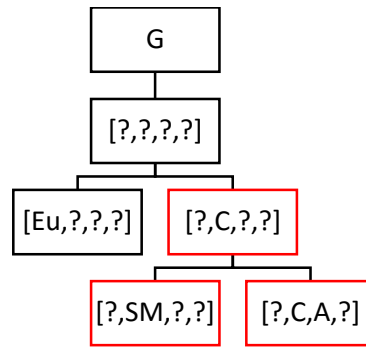


(nothing changes in S)

Example 5: [Ps, SM, 7, F2, +]



Example 6: [Mt, DB, 13, N1, +]



We have not converged to a concept. Eventually the G space was not really useful.

However, we can still to an extent deduce the taste of this particular student:

If he would order a Döner Kebab in Sole Mio, where he already had pleasant experiences and if it cost 4 euro and was ready in four minutes ([Kb, SM, 4, N1, +]), this will be classified as positive because it is covered by the only S hypothesis. If it cost 20 euro and took 1h to make he will, according to this model, still like it.

If he would try a Spring Roll at Golden Garden (eg [SR,GG,13,N1]), this will be classified negative because it is not covered by any hypothesis in G.

In short, he likes European food and doesn't care about the price.