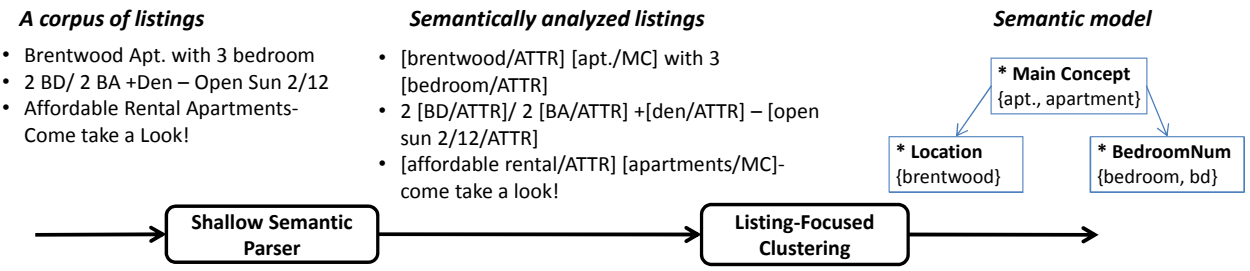

Figure 3: Steps of our system: The parser tokenizes listings (multiword detection) and then identifies main concepts and attributes (marked as MC and ATTR). The clustering module then clusters tokens of the same type. The tags (such as Location, BedroomNum) are included to help understand the figure. They are not produced by the system.

1. If a bigram (e.g., *top floor*) in a listing frequently appears as either a single or dashed token (e.g., *TopFloor* or *top-floor*) in other listings, then the bigram is regarded as a multiword.

2. For each bigram, w_1 w_2 (excluding symbols and numbers), if the conditional probability of the bigram given either w_1 or w_2 (i.e., p($w_1 w_2$ | w_1(*or* w_2)) is high (over 0.75 in our system)), the bigram is considered as a candidate multiword. This rule tests the tendency of two tokens appearing together when either one appears.

However, this test alone is insufficient, as it often generates coarse-grained results – e.g., *baseball glove*, *softball glove*, and *Hi-Def TV* [2]. To prevent this problem, for each w_2, we measure the entropy over the distribution of the tokens in the w_1 position. Our intuition is that high variability in the w_1 position (i.e., high entropy) indicates that the multiword is likely a breakable phrase. Hence, those candidates with high entropy are removed.

SPP repeatedly applies the above rules to acquire multiwords of arbitrary length. In our implementation, we limit multiword detection up to four-gram.

3.2.2 Main Concept Identification

SSP then identifies the main concepts (*mc_words*) and their attributes (*attrs*) to produce a partial semantic model. This process is guided by the observation that main concepts tend to appear as head nouns in a listing and attributes as the modifiers of these head nouns (see the examples in Fig. 3).

[2]Even though these examples are legitimate multiwords, they overlook useful information such as baseball and softball are types of gloves and Hi-Def is an attribute of TV.

Algorithm 1 describes the discovery process of *mc_words* and *attrs*. First, SSP initializes *attrs* with tokens that are likely to be a modifier (line 2), by choosing tokens that frequently appear as the object of a preposition within the corpus – e.g., *for rent*, *with washer and dryer*, *for baseball*.

SSP then iteratively performs two steps – PARSE and EXPANDMODEL (lines 3 $\sim$ 6) – in a bootstrap manner (see Fig. 4). PARSE tags the noun tokens in each listing as either head nouns or modifiers. Specifically, PARSE first assesses if a listing is "hard" to parse (line 10) based on two criteria – (1) the listing contains a long sequence of nouns (seven words or longer in our system) without any prepositions (e.g., *worth shutout series 12" womens fastpitch softball fielders glove s0120 lefty*); and (2) the majority of these nouns do not appear in *mc_words* and *attrs* (e.g., over 70% in our system). The listings meeting these criteria are generally difficult to recognize the head noun without any semantic knowledge. PARSE will revisit these listings in the next round as more *mc_words* and *attrs* are identified.

If a listing does not meet these criteria, PARSE tags nouns appearing in *mc_words* and *attrs* as head nouns and modifiers respectively (line 11). If this step fails to recognize a head noun, a heuristic is used to identify the head noun – it identifies the first noun phrase by finding a sequence of nouns/adjectives/numbers, and then tags as the head noun the last noun in the phrase that is not tagged as a modifier (line 13). For example, in the first listing of Fig. 3, *brentwood apts.* is the first noun phrase that meets the condition above; and hence *apt.* is tagged as the head noun. The remaining untagged nouns in the listing are tagged as modifiers (line 15).