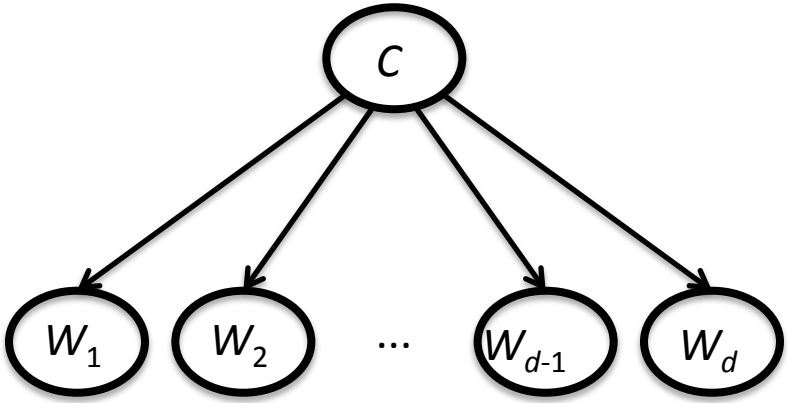
Modèle bayésien naïf multinomial

- Réseau bayésien: **modèle bayésien naïf multinomial**



- En général la **probabilité conjointe** d'un document $[W_1,\ldots,W_d]$ ayant d mots et de sa catégorie C:

$$P([W_1,\ldots,W_d], C) = P(C) \prod_i P(W_i \mid C)$$

Apprentissage du modèle

- Comment obtient-on les distributions $P(C)$ et $P(W_i \mid C)$?
  - on les obtient à partir de vraies données
  - on choisit $P(C)$ et $P(W_i \mid C)$ pour quelles reflètent les statistiques de ces données

- Soit un **corpus**, c.-à-d. un ensemble de T documents $\{ (D_t, c_t) \}$
  - chaque document D_t est une liste de mots $[w^t_1,...,w^t_d]$ de taille variable
  - c_t est la catégorie de D_t

$P(C=c)$ = (nb. de documents de la catégorie c) / (nb. de documents total)
$= |\{ t \mid c_t = c\}| \; / \; T$

$$P(W_i = w \mid C=c) = \frac{\text{nb. de fois que } w \text{ apparaît dans les documents de la catégorie } c}{\text{nb. de mots total dans les documents de la catégorie } c}$$

$$= \frac{\sum_{t \mid ct=c} \text{freq}(w, D_t)}{\sum_{t \mid ct=c} |D_t|}$$

Lissage du modèle

- Selon la formule pour $P(W_i = w \mid C=c)$, un mot w aura une probabilité de 0 s'il n'apparaît jamais dans notre corpus

- Si un seul des $P(W_i = w \mid C=c) = 0$, alors tout $P(C=c,[w_1,...,w_d]) = 0$!
  - les mots rares vont beaucoup faire varier $P(C=c,[w_1,...,w_d])$ en général

- Pour éviter cette instabilité, deux trucs afin de **lisser la distribution $P(w|c)$**
  - on détermine un **vocabulaire** V de taille fixe, et on associe les mots qui ne sont pas dans ce vocabulaire au **symbole OOV** (*out of vocabulary*)

  - **lissage δ**: on ajoute une constante δ au numérateur, pour chaque mot

$$P(W_i = w \mid C=c) = \frac{\delta + \sum_{t \mid ct=c} \text{freq}(w, D_t)}{\delta\,(|V|+1) + \sum_{t \mid ct=c} |D_t|}$$

Lissage du modèle

- Exemple: soit le vocabulaire

  V = { « Perceptron », « , », « un », « apprentissage »}

- La phrase

  « Perceptron, un algorithme d'apprentissage. »

  sera représentée par la liste de mots

  [« Perceptron », « , », « un », « OOV », « OOV », « apprentissage », « OOV »]

  w_1 w_2 w_3 w_4 w_5 w_6 w_7

- Les statistiques sont calculées à partir de cette représentation
  - on pourrait aussi enlever les mots « OOV » et les ignorer

Exemple

- Si, parmi tous les intra des années dernières (corpus de 426 mots)
  - « Perceptron » apparaît 0 fois
  - « , » apparaît 15 fois
  - « un » apparaît 10 fois
  - « apprentissage » apparaît 1 fois
  - « OOV » (tous les autres mots) apparaissent 400 fois

- Si on utilisait δ = 1, alors
  - $P($« Perceptron » $\mid C=intra) = (1 + 0) / (1 (4+1) + 426)$ $= 1 / 431$
  - $P($« , » $\mid C=intra) = (1 + 15) / (1 (4+1) + 426)$ $= 16 / 431$
  - $P($« un » $\mid C=intra) = (1 + 10) / (1 (4+1) + 426)$ $= 11 / 431$
  - $P($« apprentissage » $\mid C=intra) = (1 + 1) / (1 (4+1) + 426)$ $= 2 / 431$
  - $P($« OOV » $\mid C=intra) = (1 + 400) / (1 (4+1) + 426)$ $= 401 / 431$

somme à 1

Prétraitement des données

- Comment choisir *V*
  - ne garder que **les mots les plus fréquents** (ex.: apparaissent au moins 10 fois)
  - **ne pas garder les mots trop communs**
    - » ne pas inclure la ponctuation
    - » ne pas inclure les déterminants (« un », « des », etc.)
    - » ne pas inclure les conjonction (« mais », « ou », etc.)
    - » ne pas inclure les pronoms (« je », « tu », etc.)
    - » ne pas inclure les verbes communs (« être », « avoir », « faire », etc.)
    - » etc.
  - utiliser une **forme normalisée des mots** (fusion de mots différents en un seul)
    - » **enlever les majuscules** (« Perceptron » → « perceptron »)
    - » **lemmatiser** les mots (« marchons » → « marcher »,
      « suis » → « être », « est » → « être »)
- Il n'y a pas de recette universelle, le meilleure choix de *V* varie d'une application à l'autre