

ПОСТРОЕНИЕ ИЕРАРХИЧЕСКОГО МЕНЮ ДЛЯ МИНИМИЗАЦИИ ВРЕМЕНИ ПОИСКА

Губко М.В., Даниленко А.И.

(ИПУ РАН, Москва)

mgoubko@mail.ru, danilenko.alexander@gmail.com

Рассмотрена задача построения оптимального иерархического меню. Для различных стратегий поведения пользователей построены унифицированные модели с применением математической теории оптимизации иерархических структур. Вычислены параметры оптимальной иерархии – каждая панель меню должна содержать одинаковое число элементов, закон изменения относительных популярностей элементов одинаков.

Ключевые слова: иерархические меню, оптимизация иерархии, модели навигации по меню, эффективность поиска.

Введение

Иерархические меню являются популярным методом организации команд и каталогов объектов. Удобство меню, как правило, определяется средним временем доступа к искомому элементу. Таким образом, задача оптимизации состоит в поиске структуры меню, минимизирующей среднее время одной сессии.

Активная разработка формальных методов оптимизации структуры меню началась в 80-х годах прошлого века. Ряд публикаций посвящен изучению различных принципов организации меню, экспериментальным исследованиям в области поведения пользователей в меню [2], а также теоретическим моделям оптимальных иерархических меню [3].

Тем не менее, до сих пор не существует единой методики, описывающей принципы построения меню для различных условий и принципов поведения пользователей.

1. Задача оптимизации иерархического меню

Рассмотрим множество элементов $w \in N$ (например, команд или ссылок для web-каталогов), которые необходимо разместить в меню. За $\mu(w)$ обозначим популярность элемента w (вероятность того, что этот элемент требуется пользователю).

При построении меню множество элементов N разбивается на категории, определяющие панели меню. Каждая категория (т.е. панель меню) характеризуется набором элементов $s \subseteq N$, доступных в ней. Дальнейшее объединение категорий в более общие формирует древовидную структуру меню. Листьями дерева являются конечные элементы из множества N , а остальные вершины соответствуют категориям.

Как правило, считается, что пользователь ищет один элемент нижнего уровня [2, 3]. Время одной сессии тогда складывается из времени, затраченного пользователем на поиск требуемого элемента в каждой категории, которую он проходит.

Для каждой категории $s \subseteq N$ можно определить ее популярность как сумму популярностей входящих в нее элементов $\mu_s = \sum_{w \in s} \mu(w)$. Тогда среднее время одной пользовательской сессии в иерархии H определяется суммой $T = \sum_{s \in H} \mu_s t_s$.

В общей теории оптимизации иерархий доказано [1], что для рассматриваемых моделей оптимальной структурой меню является однородное дерево. В однородном дереве каждая категория имеет одинаковое количество подкатегорий k (также называемое *шириной иерархии*) с популярностями, разбитыми в одной и той же пропорции y_1, \dots, y_k . Таким образом, задача оптимизации структуры меню сводится к поиску параметров однородного дерева, минимизирующих среднее время сессии.

2. Модели навигации в меню

Рассмотрим две базовые стратегии поведения пользователя в меню. При поведении, называемом «read all» [3], пользователь делает выбор только после прочтения всех представленных вариантов. Время, затраченное пользователем на поиск и выбор

любого из вариантов $t_s = t_{resp} + t_{load}k + t_{read}k + t_{click}$, где t_{resp} – время ответа сервера в секундах, t_{load} – среднее время загрузки одного элемента меню, t_{read} – среднее время прочтения элемента, t_{click} – среднее время, требуемое на подведение курсора и выбор элемента. Оптимально симметричное меню с шириной иерархии, определяемой решением уравнения $\ln k = 1 + \frac{t_{resp} + t_{load}k + t_{click}}{t_{read}k}$.

Если взять параметры, типичные для web-каталогов, $t_{resp} = 2$ с, $t_{load} = 0.02$ с, $t_{read} = 1$ с, $t_{click} = 1$ с, то оптимально симметричное дерево с числом элементов в каждой категории $k = 5$.

При альтернативном поведении («read until» [3]) пользователь просматривает элементы последовательно и, как только встречает искомый, выбирает его. Тогда время, затраченное в одной панели меню $t_s = t_{resp} + t_{load}k + t_{read} \sum_{i=1}^k i \cdot y_i + t_{click}$. Для таких пользователей оптимальна несимметричная структура с пропорцией $y_i = a(k)^{A(k)+i}$, где $A(k) = (t_{resp} + t_{load}k + t_{click}) / t_{read}$, $a(k)$ – решение уравнения $\sum_{i=1}^k a^{A(k)+i} = 1$. Оптимальное число элементов в меню $k = \arg \min_k \frac{1}{\ln(1/a(k))}$.

Для параметров, приведенных выше, численный расчет показывает, что оптимальная ширина иерархии $k = 13$, а соответствующая пропорция $x_1 \approx 0.27$, $x_2 \approx 0.20$, $x_3 \approx 0.15$, ..., $x_{13} \approx 0.007$.

2.1 УЧЕТ ОШИБОК ПОЛЬЗОВАТЕЛЯ

Предположим, что пользователи могут совершать ошибки при выборе элементов в категории с вероятностью p . Когда пользователь ошибается, он затрачивает на это дополнительно время. Рассмотрим систему, в которой доля α пользователей придерживается стратегии «read all», а остальные $(1 - \alpha)$ пользователей – стратегии «read until». На рис. 1 приведена зависимость оптимальной ширины иерархии k от вероятности ошибки p для различных распределений типов пользователей α .

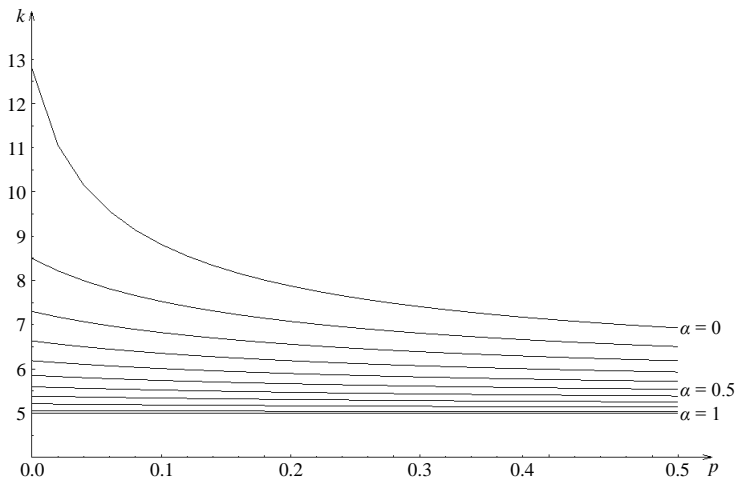


Рис. 1. Зависимость оптимальной ширины иерархии от вероятности ошибки для различных типов пользователей

Рис. 1 иллюстрирует полученные аналитически результаты. Если все пользователи придерживаются стратегии «read all» ($\alpha = 1$), оптимальная структура меню не зависит от вероятности ошибки. При наличии пользователей со стратегией «read until», с увеличением вероятности ошибки ширина оптимального дерева уменьшается. При этом влияние ошибки на ширину иерархии тем больше, чем выше процент таких пользователей.

Литература

1. ГУБКО М.В. *Математические модели оптимизации иерархических структур*. М.: ЛЕНАНД, 2006.
2. HOLLINK V., VAN SOMEREN M. *Optimal link categorization for minimal retrieval effort*. // Proceedings of Sixth Dutch-Belgian Information Retrieval Workshop, 2006. P. 65 – 72.
3. LEE E., MACGREGOR J. *Minimizing user search time in menu retrieval systems*. Human Factors, 27(2), 1985. P. 157 – 162.