基于场感知分解机的五笔输入法

李泽南,刘汉明*,胡珍珍,黎　姿,司马燊,郭　港
(赣南师范大学 数学与计算机科学学院,江西 赣州 341000)

摘　要:计算机技术在中国的普及,使得人们大量使用计算机输入文本,从而大大减少了汉字的书写。加上拼音等易用的汉字输入法占据主导地位,使人们对熟悉的字变得生疏,“提笔忘字”非常普遍。五笔等字形编码的汉字输入法体现了汉字的书写,可有效减少“提笔忘字”等现象,但易用性不高。研究把推荐系统中的场感知分解机与传统的五笔输入法相结合,解决了稀疏特征问题,并根据用户的历史数据,预测用户需求同时推送最可能的候选汉字,提高了第一候选字词推荐准确率,降低了使用难度。实验表明,该五笔输入法具有稳健的“推荐”能力,第一候选字词推荐准确率达到 98.91%,显著优于现有输入法,并且准确率可随用户对字词使用次数的增加而提高,达到了改善用户体验、增加用户粘性的目的。

关键词:场感知分解机;五笔输入法;推荐系统;提笔忘字;易用性
中图分类号:TP391.1　　**文献标识码**:A　　**文章编号**:1673-629X(2023)08-0165-07
doi:10.3969/j.issn.1673-629X.2023.08.024

Wubi Input Method Based on Field-aware Factorization Machines

LI Ze-nan,LIU Han-ming*,HU Zhen-zhen,LI Zi,SIMA Shen,GUO Gang
(School of Mathematics and Computer Science,Gannan Normal University,Ganzhou 341000,China)

Abstract:The popularization of computer technology in China has led to a large number of people using computers to input text,which has greatly reduced the writing of Chinese characters. In addition,the easy-to-use input methods such as Pinyin have dominated,making people unfamiliar with many characters,which lead to a problem is Character amnesia. The Chinese character input methods with glyph encoding such as Wubi embody the writing of the characters,which can effectively alleviate the problem,but they are not easy to use. The field-aware factorization machine is combined with the Wubi input method to solve the sparse feature problem. Based on the history user data,the proposed method can push the best candidate Chinese characters to the user by predicting their demand to improve the recommendation accuracy of the first candidate and to reduce the difficulty using Wubi. The experiments show that the proposed method has a robust recommendation with accuracy of 98.91% for the first candidate,being greatly better than that of the existing methods,and the accuracy increases while the selection of the words increase,improving user experience and prompting their stickiness.

Key words:field-aware factorization machines;Wubi input method;recommendation system;character amnesia;easy-to-use

0　引　言

20 世纪 80 年代初期,随着汉字编码的发明,出现了中文输入法。中文输入一般可以分为键盘、手写、语音等三种输入方式[1],其中键盘输入需要对汉字二次编码,具有输入不受环境制约、对系统性能要求低、响应速度快等特点,是桌面计算机系统的主流输入方式[2]。在汉字输入法的发展过程中,出现了大量的汉字编码方案,分为以音为主、以形为主、音形结合三类[3]。目前常用的汉字编码方案有拼音、双拼、五笔、笔画等[4]。

以音为主的汉字编码以拼音输入法为代表,入门门槛低,通过大容量词库、模糊音支持、用户词自动添加、热词自动更新等一系列功能,拼音输入法取得了极大的成功。特别是手机等无实体键盘设备上应用广泛,截至 2020 年底,使用拼音类手机输入法用户规模达到 7.55 亿人,随着互联网的影响深化下沉,用户规模进一步扩大[5]。杨新涛等研究了基于深度学习的拼音输入法,希望通过深度学习技术使汉字输入更准确更高效[6]。拼音输入法重码率过高,特别生僻字和单字的速度远远落后于五笔输入法[7];在线词库越来越大导致备选字词切换速度慢,输入法软件本身也越来越复杂使得占用的系统资源越来越多。

收稿日期:2022-09-28　　**修回日期**:2023-02-07
基金项目:江西省高校人文社会科学研究项目(YY17101);江西省教育科学规划课题(21YB169)
作者简介:李泽南(1995-),男,硕士研究生,研究方向为机器学习;通信作者:刘汉明(1970-),男,博士,教授,CCF 会员(I8207M),研究方向为数据挖掘与机器学习。

以形为主的输入方案体现了汉字的书写、重码率低、文字输入效率高。王永民[8]通过长达五年的研究，在 1983 年发明了根据笔画和字形特征对汉字进行编码的五笔字型输入法。李亭骞等人提出的 E 码汉字输入法，根据汉字字形首尾形状与键盘上的英文字母存在相似的特点实现汉字的输入，降低了用户记忆字根的难度。以形为主的输入法在初期过于专注降低重码率，导致编码方案要么过于复杂，如五笔输入法需要用户记忆字根；要么码长较长，如笔画输入法[9]。但五笔输入法需要熟悉五笔字根表，入门门槛较高，随着计算机的普及，更多的用户需要操作简单的输入法[10]。加上五笔输入法自出现以来基本上没有太大的改进，使得拼音输入法逐渐占据了如今的主导地位。

音形结合的输入法试图结合汉字的“音”与“形”，以期解决拼音码的重码率高和形码难记的不足，如苗文音形编码[11]，但其要求拼音准确且仍需用户记忆字形码。

拼音输入法虽然入门简单，使用者初期的使用体验效果优，但五笔输入法整体上仍存在优势，在报社等需要专业性文字录入工作的场合仍大规模使用。特别地，计算机时代导致手写汉字的机会大大减少，使人们对熟悉的字变得生疏，许多原本会写的字变得只会读，“提笔忘字”变得越来越常见[12]，严重地影响了中华文化的传承。五笔等字形编码汉字输入法体现了汉字的书写，对减少“提笔忘字”等现象，促进中华文化传承具有重要意义。

近年来，机器学习取得的长足的发展，但机器学习用于汉字输入法的研究较少。杨新涛等提出了基于深度学习的拼音输入法[6]，但深度学习算法复杂、对计算机硬件要求高、数据训练中存在过拟合[13]，从而影响汉字输入的速度。推荐系统根据用户的历史记录，向用户推荐感兴趣的事务，研究结合场感知分解机[14](Field-aware Factorization Machine，FFM)推荐算法提出了一种基于 FFM 的五笔输入法(Wubi based FFM，WB-FFM)。该方法根据用户以往的数据，处理汉字数据解决稀疏特征问题，预测用户的需求向用户推送候选汉字，以期进一步提高五笔汉字输入的效率，改善用户体验，增加用户粘性，为保证中华文化的传承载体不会退化甚至消失，对中华文化传承也具有重要意义

实验表明，WB-FFM 输入法具有稳健的“推荐”能力，第一候选字词推荐准确率达到 98.91%，优于现有典型的输入法。

1 FFM 推荐系统

为解决稀疏特征和特征组合的问题，Y. Juan 等提出了 FFM 算法，它是 FM(Factorization Machine)[15]模型的改进版，以更好地适应稀疏特征。常用汉字 2 000 多个，对汉字输入来说是稀疏特征问题。

1.1 FM

FM 旨在解决诸如推荐系统等面临的稀疏数据下的特征组合问题。假设数据有 n 个特征，x_i 是第 i 个特征值，x_ix_j 表示 x_i 和 x_j 的组合 $(x_i,x_j\neq 0)$，ω_0、ω_i、ω_{ij} 是模型参数，则二阶多项式的模型为：

$$y(x)=\omega_0+\sum_{i=1}^{n}\omega_i x_i+\sum_{i=1}^{n}\sum_{j=i+1}^{n}\omega_{ij}x_ix_j \tag{1}$$

在数据稀疏的情况下，因为 $x_i,x_j\neq 0$ 的样本不足，导致参数 ω_{ij} 的训练十分困难。

矩阵分解可有效解决参数 ω_{ij} 的训练问题。设 ω_{ij} 组成的矩阵为 $\boldsymbol{W}$，分解得 $\boldsymbol{W}=\boldsymbol{V}^T\boldsymbol{V}$，那么，$\omega_{ij}$ 可以看作第 i、j 维特征的隐向量之积 $<v_i,v_j>$，得 FM 模型。

$$y(x)=\omega_0+\sum_{i=1}^{n}\omega_i x_i+\sum_{i=1}^{n}\sum_{j=i+1}^{n}<v_i,v_j>x_ix_j \tag{2}$$

其中，二次项

$$\sum_{i=1}^{n}\sum_{j=i+1}^{n}<v_i,v_j>x_ix_j=\frac{1}{2}\sum_{f=1}^{k}\left(\left(\sum_{i=1}^{n}\nu_{i,f}x_i\right)^2-\sum_{i=1}^{n}v_{i,f}^2x_i^2\right) \tag{3}$$

其中，$v_{i,f}$ 是第 i 个变量的第 f 个因子，$k\ll n$ 是超参数，由用户指定。这样，FM 的复杂度可由原来的 $O(kn^2)$ 降为 $O(kn)$。

1.2 FMM

Y. Juan 等借鉴“场”[16]的概念提出的 FFM 把相同性质的特征归为一个“场”，同一个“场”的特征单独 One-Hot 编码。在 FFM 中，每一维特征 x_i，对特征 $x_j(j\neq i)$ 的“场”f_j，都有一个隐向量 v_{i,f_j}，FFM 模型为：

$$y(x)=\omega_0+\sum_{i=1}^{n}\omega_i x_i+\sum_{i=1}^{n}\sum_{j=i+1}^{n}(v_{i,f_j},v_{j,f_i})x_ix_j \tag{4}$$

若 f 是“场”的个数，则 FFM 的参数个数为 nfk。对每个隐向量，只需要学习它的“场”的效应，使得 $k_{\mathrm{FFM}}\ll k_{\mathrm{FM}}$，从而进一步降低了算法复杂度。

1.3 FFM 的优化

在 FFM 领域中，LIBFFM 作为一个广泛使用的分解机库，利用随机梯度下降(SGD)优化。

随机梯度下降算法(Stochastic Gradient Descent，SGD)[17]源于 1951 年 Robbins 和 Monro 提出的随机逼近，最初应用于模式识别[18]和神经网络[19]。这种方法在迭代过程中随机选择一个或几个样本的梯度来替代总体梯度，从而大大降低了计算复杂度。1958 年 Rosenblatt 等研制出的感知机采用了随机梯度下降法的思想，即每轮随机选取一个样本，求其对应损失函数的梯度，再基于给定的步长更新参数。1986 年

Rumelhart 等分析了多层神经网络的误差反向传播算法,该算法每次按顺序或随机选取一个样本来更新参数,它实际上是小批量梯度下降法的一个特例。近年来,随着深度学习的迅速兴起,随机梯度下降算法已成为求解大规模机器学习优化问题的一类主流方法[20]。

SGD 在每轮更新参数时,仅随机抽取一个样本计算其梯度,并以此梯度为全局梯度的估计值。SGD 的参数更新公式为:

$$w_{t+1} = w_t - \alpha_t \nabla L_{i_t}(w_t) \tag{5}$$

其中,α_t 为第 t 轮迭代的学习率,用于调整参数更新的幅度。为防止学习率过大而错过最优解,常将其设置为一个递减的序列。$i_t \in \{1,2,\cdots,n\}$ 表示第 t 轮迭代中按均匀分布随机抽取的样本序号。

FFM 模型使用带 L2 正则项的 logistic loss 作为损失函数,采用 SGD 来优化损失函数,选取单个样本简化损失函数,公式为:

$$L = \log(1 + \exp\{-y_i \varphi_{\mathrm{FFM}}(w,x_i)\}) + \frac{\lambda}{2} \| w \|^2 \tag{6}$$

每次迭代时选取一个样本数据点 (y,x),对式(6)中 ω_{j_1,f_2} 和 ω_{j_2,f_1} 求偏导得:

$$g_{j_1,f_2} = \nabla_{w_{j_1,f_2}} f(w) = \lambda \cdot w_{j_1,f_2} + k \cdot w_{j_2,f_1} x_{j_1} x_{j_2} \tag{7}$$

$$g_{j_2,f_1} = \nabla_{w_{j_2,f_1}} f(w) = \lambda \cdot w_{j_2,f_1} + k \cdot w_{j_1,f_2} x_{j_1} x_{j_2} \tag{8}$$

其中,k 为:

$$k = \frac{\partial \log(1 + \exp(-y\varphi_{\mathrm{FFM}}(w,x)))}{\partial \varphi_{\mathrm{FFM}}(w,x)} = \frac{-y}{1 + \exp(y\varphi_{\mathrm{FFM}}(w,x))} \tag{9}$$

加入学习率提升 SGD 的训练效率,通过 Adagrad 算法自动调整学习率。此时,SGD(公式(5))的更新公式为:

$$w_{t+1,j} = w_{t,j} - \frac{\alpha}{\sqrt{G_{t,jj} + \epsilon}} g_{t,j} \tag{10}$$

$$G_{t,jj} = \sum_{j=1}^{t} g_{t,j}^2 \tag{11}$$

其中,$g_{t,j}$ 为第 t 轮第 j 个参数的梯度,ϵ 是平滑项,避免分母为 0,式(11)的 $G_{t,jj}$ 对角矩阵,对角线的值 j 是参数 w_j 的平方和,随着迭代次数的进行,参数进行累加,学习率逐渐减小。此时需要更新 G_{j_1,f_2} 与 G_{j_2,f_1},更新公式为:

$$G_{j_1,f_2} = G_{j_1,f_2} + (g_{j_1,f_2})^2 \tag{12}$$

$$G_{j_2,f_1} = G_{j_2,f_1} + (g_{j_2,f_1})^2 \tag{13}$$

最后更新模型参数为:

$$w_{j_1,f_2} = w_{j_1,f_2} - \frac{\alpha}{\sqrt{G_{j_1,f_2} + \epsilon}} * g_{j_1,f_2} \tag{14}$$

$$w_{j_2,f_1} = w_{j_2,f_1} - \frac{\alpha}{\sqrt{G_{j_2,f_1} + \epsilon}} * g_{j_2,f_1} \tag{15}$$

2　WB-FFM 输入法

FFM 模型对特征数较多且稀疏问题有很好的适应性,其根据历史点击率(Click-Through Rate,CTR)来提高向用户推荐的准确性,汉字输入法在存在重码的情况下通过候选窗口向用户提供字词选择,且候选字词也是高维、稀疏的。结合这些特点,实现的基于 FFM 的五笔输入法,利用用户选择候选词的历史记录向用户推荐最可能的字词,提高了输入效率和用户体验。

2.1　训练集

把 FFM 用于五笔输入推荐,首要的问题是如何得到训练集,通过提取微软五笔输入法(86 版)的词库来达到这一目的。该词库包含了各字词的编码、用户选择次数和编码长度,共有 529 882 个字词。相对于编码,编码长度特征冗余,这里从数据集中去除该特征。

2.2　“场”的构建

显然,对训练集利用 One-Hot[21] 重构特征后,其特征量相当大。根据训练集的特点,构建 3 个“场”(见表 1):

- 字词。采用 One-Hot 构造特征;
- 编码。采用 One-Hot 构造特征;
- 选择次数,即用户输入某字、词的次数。考虑到如果对其重构特征,需要对特征值离散化,不但会大大增加特征数量,而且会影响表示精度,所以这里不重构特征(即 1 个特征)。

表 1　训练集的“场”

场	字词	编码	选择次数
示例	工	a	19 829
	工	aaa	

2.3　实　现

由于“场”的存在,需要把重构特征后的数据转化为“场标识:特征标识:值”格式(见表 2)。当特征是离散型时,“值”固定为 1,否则是归一化后的字词选择次数。

表 2　特征与“场”对应标识

场	场标识	特征	特征标识
字词	1	字词=工	1
编码	2	编码=a	2
选择次数	3	编码=aaa	3
		选择次数	4

SGD 训练 FFM 模型见算法 1。

算法 1:SGD 训练 FFM

#分别是训练样本集、验证样本集和训练参数设置

输入:(tr,va,pa)

输出:model,Loss(损失函数)

```
#特征数( tr. n )、场数( tr. m )和参数( pa )
model = init( tr. n,tr. m,pa)
R_tr = 1,R_va = 1
#归一化的 pa. norm 为真,计算训练和验证样本的系数
if pa. norm then
R_tr = norm( tr),R_va = norm( va)
end if
for it = 1,…,pa. itr do
#数据迭代,若新参数为真则打乱训练顺序
if pa. rand then
tr. X = shuffle( tr. X)
end if
for i = 1,…,tr. l do
#计算单个样本的 FFM 输出 φ
φ = calcΦ( tr. X[ i],R_tr[ i],model)
eφ = exp{ - tr. Y[ i] * φ}
#计算样本的训练误差
L_tr = L_tr + log{1 + eφ}
#单个样本的损失函数计算梯度 g_Φ
g_Φ = - tr. Y[ i] * eφ/(1 + eφ)
#再根据梯度更新 model 参数
model = update( tr. X[ i],R_tr[ i],model,g_Φ)
end for
#验证样本,计算样本的 FFM 输出并验证误差
for i = 1,…,va. l do
φ = calcΦ( va. X[ i],R_va[ i],model)
L_va = L_va + log{1 + exp{ - va. Y[ i] * φ}}
end for
end for
```

训练好的模型用于 WB-FFM 输入法(算法 2)。

```
算法 2:基于 FFM 的五笔输入法
输入:编码 D
输出:用户选择的候选字词 Z
#检查字词库,在字词库中匹配相应编码的字词
#获取用户库中匹配字词 HC(点击次数)
if HC > 0 then
  获取用户库的 HC
else
  HC=1//字词点击次数为 0,HC 取值默认为 1
end if
#获取字词数据,构建 FFM 数据并预测候选字词
SelectJdates( D)
#对候选字词进行排序, arr 字词的相关数据
bubbleSort( arr)
#用户点击相应的字词
if Z then
  #对应用户库的字词点击次数累加+1
HC + +
else
  HC=1 #结束候选
end if
#WB-FFM 清除候选字词
return 重新输入 D
```

3 实验结果与分析

实验采用 FM、MF[22] 作为模型训练时的对比算法。FM 采用 LIBFM[23] 方法实现,LIBFM 是一个广泛使用的推荐系统矩阵分解库,支持 SGD 等多种优化方法,这里采用 SGD,与 FFM 一致。MF 采用 LIBMF[24] 方法实现,LIBMF 是一个用于潜在空间使用两个矩阵的积来逼近一个不完整矩阵的开源工具库;WB-FFM 的 FFM 采用开源工具 LIBFFM 实现。实验采用逻辑损失对模型进行性能评价。

另外,为测试 WB-FFM 输入法的性能,还选择了 QQ、微软、极点、陈桥、搜狗和王码等 6 种常用五笔输入法以及 QQ 和搜狗两种常用拼音输入法作为对比。

3.1 训练集

这里使用微软五笔输入法 86 版的字库(节 3.1),既作为 WB-FFM、FM 和 LIBMF 的训练集,也作为 WB-FFM 的字库。经过特征重构后,共得到 729 288 个特征(见表 3)。

表 3 数据集特征数

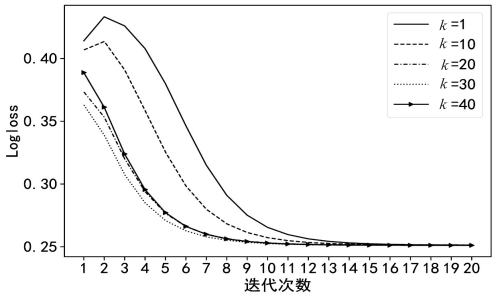
特征	特征数
字词	529 882
编码	199 405
选择次数	1

3.2 模型构建

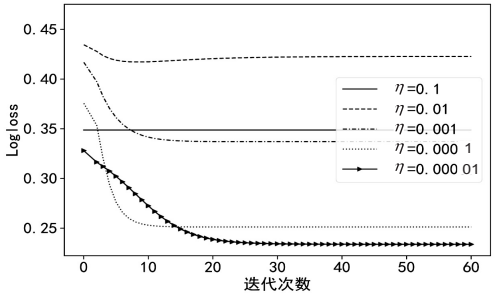
主要通过实验的方法优化 FM、MF 和 FFM 模型的参数。实验基于 i7-7700HQ@2.80 GHz CPU、16 GB 内存、Windows10 系统,C 语言编程。算法需要调整的参数主要有模型的迭代次数(t)、学习率(η)、场/因素个数(k)、惩罚因子(λ)等。实验对不同算法最优化:首先选取一个参数 a 作为优化对象,其余参数设为默认值;然后,在 a 的范围内(算法不同,范围可能不同,以该算法最佳范围为准)均匀取 5 个值对 a 进行优化;接着,固定 a 为最优值,优化第二个参数,以此类推,优化完所有参数(见图 1 ~ 图 3)。根据优化后的参数,测试了三种模型的性能(见表 4、图 4)。

表 4 不同模型的性能比较

模型	最优参数	时间/s	Logloss
FM	$\eta = 1 \times 10^{-5}$, $k = 30, t = 55$	81.00	0.23
MF	$\eta = 0.1, \lambda = 1 \times 10^{-3}$, $k = 20, t = 100$	33.54	0.11
FFM	$\eta = 0.1, \lambda = 1 \times 10^{-5}$, $k = 16, t = 30$	50.00	0.001 4

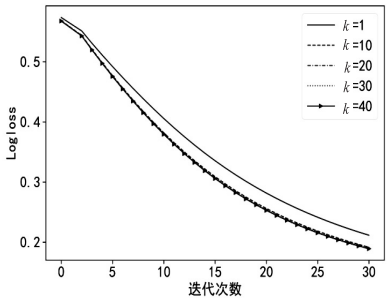


(a) η 默认,调整 k

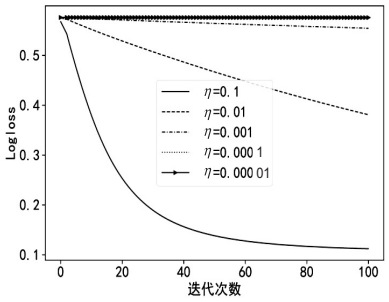


(b) k =30,调整 η

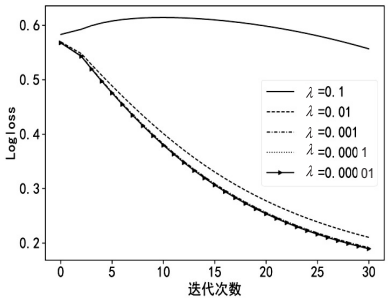
图 1　FM 损失



(a) η , λ 默认,调整 k

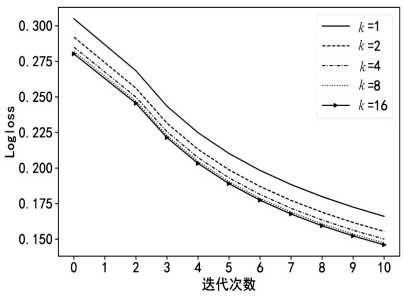


(b)λ 默认, k =20,调整 η

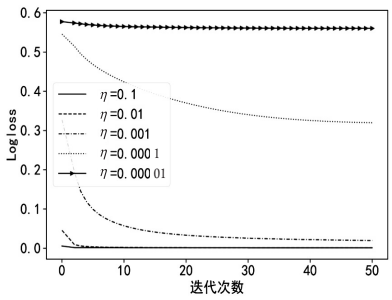


(c) k =20, η =0.1,调整 λ

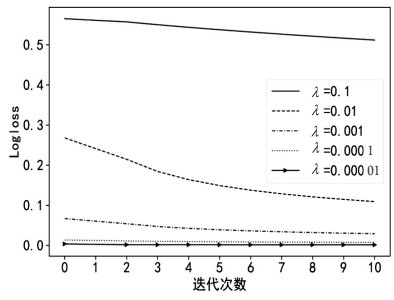
图 2　MF 损失



(a) η , λ 默认,调整 k



(b) λ 默认, k =16,调整 η



(c) k =20, η =0.1,调整 λ

图 3　FFM 损失

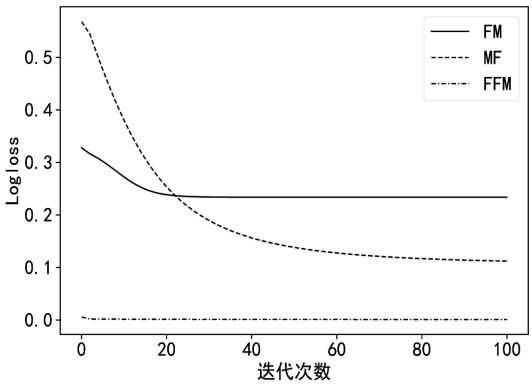


图 4　不同模型的损失

表 4 显示,MF 速度最快,这是因为该模型相比于 FM 和 FFM 来说,算法复杂度更低。另外,尽管 FFM 模型复杂度高于 FM,但它有更小的 k (表 4)和更快的收敛速度(图 4),使得它的算法时间明显小于 FM。表 4 和图 4 还显示,FFM 的对数损失明显小于 FM 和 MF,体现了该模型的优越性。

3.3　现有输入法比较

实验随机从已发表的文献中选取科技、体育、农业、旅游、医疗、生态环境、航天科技、非文化遗产、商贸、法律共 10 段不同类型的文字,每段文字数量 200 ~ 300 字,测试包括 WB-FFM 在内的 9 种输入法的性能。作对照的极点、QQ 五笔、搜狗五笔、陈桥、QQ 拼音和搜狗拼音 6 种输入法具有按输入次数排序、按最近输入排序或有重码时被选择的候选项自动调位至首位等“推荐”功能,测试前将它们的这些功能开启。测试时每种输入法按以上顺序连续输入 10 段文字,并统计候选字词推荐到第一位的准确率(见表 5、图 5)。

表 5　第一候选平均准确率　　%

QQ 五笔	微软	极点	陈桥	搜狗五笔	王码	WB-FFM	QQ 拼音	搜狗拼音
98.11	92.47	98.78	95.56	99.44	99.65	98.91	61.03	60.45

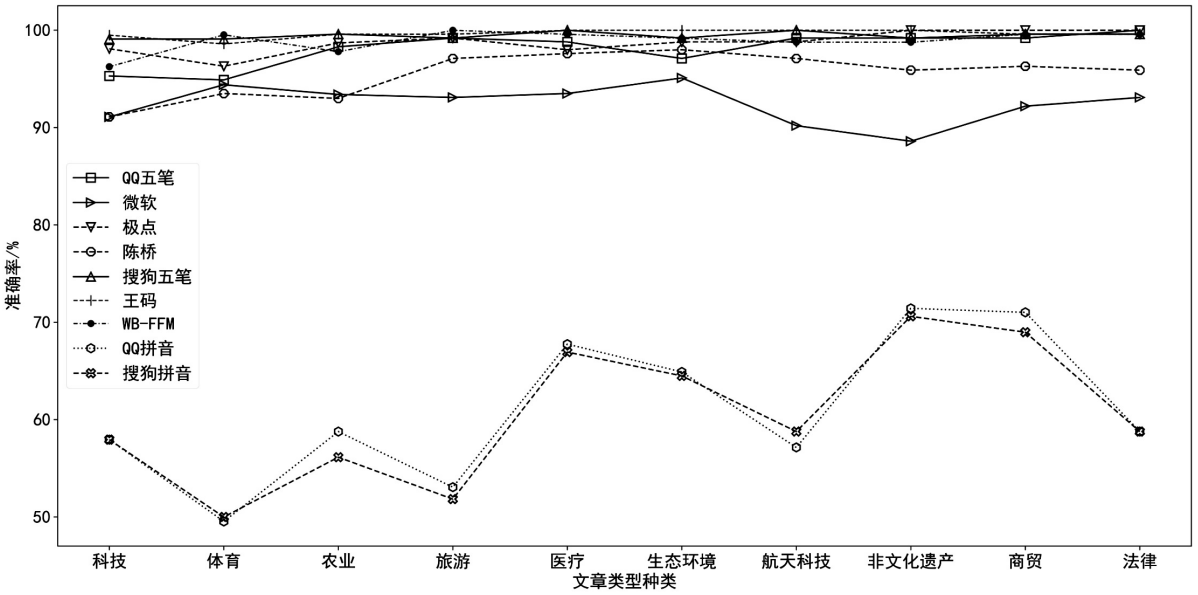


图 5 第一候选准确率

表 6 输入法第一候选准确率线性拟合

	QQ 五笔	极点	陈桥	搜狗五笔	WB-FFM	QQ 拼音	搜狗拼音
斜率	0.46	0.30	0.47	0.05	0.19	1.40	1.40
方差	3.12	1.32	5.15	0.12	1.26	55.39	49.81

表 5 显示,尽管王码输入法没有“推荐”功能,但它的第一候选平均准确率却最高,这是因为 10 段内容来源于公开发表的文献,属于较常用的文字,该输入法会优先推荐常用字(类似的原因,搜狗也获得了较高的平均准确率)。从图 5 中可以看出,王码输入法的准确率曲线几乎平直,这是没有“推荐”功能的表现。微软输入法也没有“推荐”功能,但它的曲线出现了很大的波动,这应该是它对汉字的“理解”远不如王码所造成的。其它 7 种输入法由于具有“推荐”功能,图 5 显示它们的准确率逐步上升,且 QQ 五笔、极点、搜狗五笔和 WB-FFM 最终“收敛”到了与王码基本相同的准确率。表 5 和图 5 还显示,五笔输入法第一候选准确率明显高于拼音输入法,这是由于拼音输入法的重码率显著高于五笔输入法所造成的。另外,还统计了 QQ 和搜狗两种拼音输入法第一候选的平均码长,分别为 4.62 和 4.76(对于词组的码长以平均计,如,“zg”为“中国”,则码长为 1),也高于五笔不高于 4 的码长。

为了进一步考察 7 种具有“推荐”功能的输入法的“推荐”特性,把图 5 作直线拟合,并计算对应的斜率和方差(见表 6)。表 6 显示 WB-FFM 的斜率和方差都比较小。较小的斜率说明它的“推荐”比较温和,较小的方差意味着算法稳健性较高,可减小过拟合风险。需要说明的是,尽管搜狗五笔的斜率和方差最小,但它的斜率几乎为零,会导致“推荐”收敛过慢甚至不收敛。

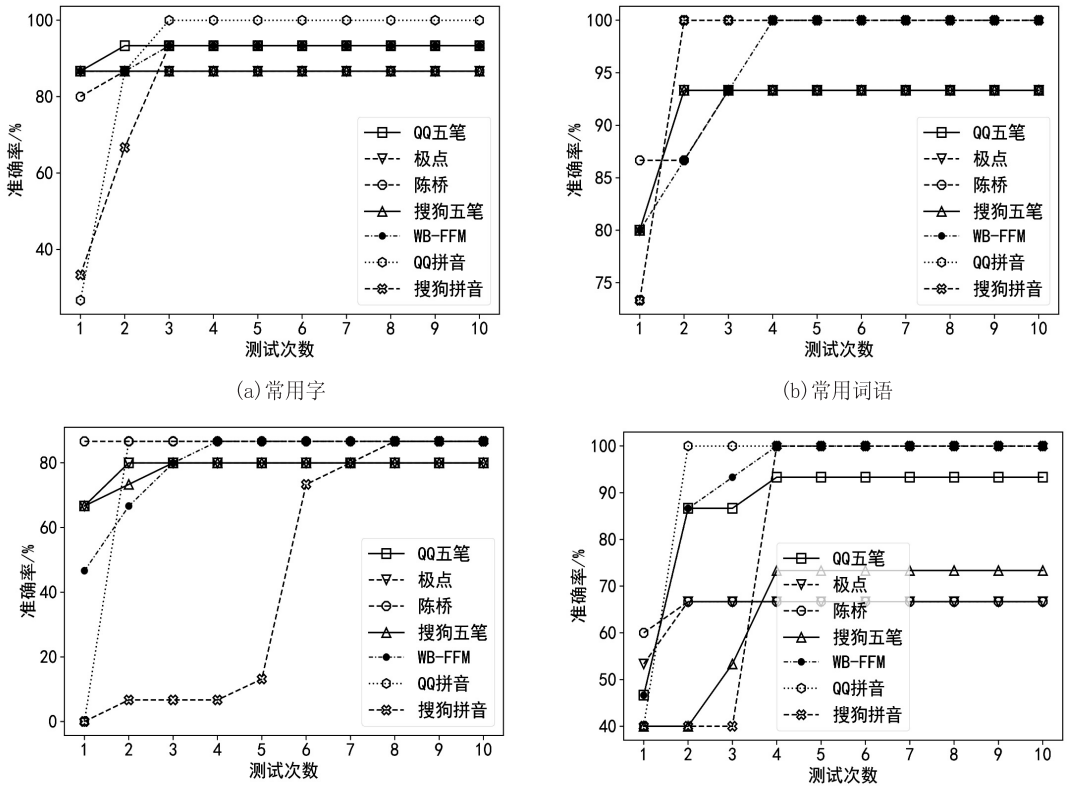
实验还选取了常用、生僻等共 15 个不同类型的字和词以进一步探索 7 种具有“推荐”功能的输入法的“推荐”能力。每组字、词连续重复输入 10 次,并统计各输入法第一候字词选准确率(见表 7、图 6)。之所以使用生僻字和词组,是考虑到这类字和词在各输入法的历史记录基本为零。

表 7 输入法第一候选字词最终准确率 %

输入法	常用字	常用词语	生僻字	生僻词语
QQ 五笔	93.33	93.33	80.00	93.33
极点	86.67	93.33	80.00	66.67
陈桥	86.67	93.33	86.67	66.67
搜狗五笔	86.67	93.33	80.00	73.33
WB-FFM	93.33	100.00	86.67	100.00
QQ 拼音	100.00	100.00	86.67	100.00
搜狗拼音	93.33	100.00	86.67	100.00

表 7 显示,无论常用或生僻字词,WB-FFM 的最终准确率位列第二,仅常用字低于 QQ 拼音,综合考虑图 5 和表 6,该输入法整体“推荐”能力优于现有方法。另外,图 6 显示了各输入法“推荐”准确率随测试次数的增加而增加,符合推荐算法会利用历史记录的思想。在对常用字的“推荐”上(图 6(a)),WB-FFM 并无特别之处,这是因为一般的汉字输入法在常用字的处理上都比较成熟。但图 6(b)~(d)中,WB-FFM 的准确率的提升比较稳健,明显优于其它输入法,说明其“推

荐”稳健性优于其它输入法。



(a)常用字　(b)常用词语　(c)生僻字　(d)生僻词语

图 6　第一候选字词准确率随测试次数的变化

4　结束语

输入法是人们使用计算机的最基本需求。如今,人们手写汉字的机会大大减少,对熟悉的字变得生疏,加上拼音等易用的汉字输入法的广泛使用,使得“提笔忘字”等现象越来越严重。五笔字型输入法可较好地表征汉字字型,对改善人们“提笔忘字”有一定的帮助。研究把 FFM 推荐算法应用到五笔字型输入法,以期提高第一候选字词的推荐性能,降低五笔输入法的使用难度,增加用户的使用粘性。

实验表明,提出的结合 FFM 算法的五笔输入法 WB-FFM 的第一候选字词的推荐准确率和推荐稳健性均高于现有输入法,验证了推荐算法在输入中的应用优势。WB-FFM 良好的推荐能力和较短的码长,增加了五笔输入法的易用性,但与流行的拼音输入法相比,其较高的入门门槛还有待于今后进一步探索。

参考文献:

[1] 佐建明,李景泉,程晓佳. 基于汉字部件的汉字字形输入法和排检法应用研究[C]//2019 年中国索引学会年会暨学术研讨会论文集. 成都:复旦大学出版社,2019:35-48.

[2] 孟　真. 基于深度强化学习的中文拼音输入法[D]. 上海:上海交通大学,2019.

[3] 庄　航. 基于深度学习的中文词表示学习技术研究[D]. 合肥:中国科学技术大学,2018.

[4] 戴　露. 基于语言模型的拼音输入法研究与实现[D]. 武汉:华中科技大学,2019.

[5] 上海艾瑞市场咨询有限公司. 2020-2021 年中国第三方手机输入法行业年度研究报告[EB/OL]. (2021-03-16)[2022-10-21]. https://report.iimedia.cn/repo1-0/39380.html.

[6] 杨新涛. 基于深度学习模型的输入法研究[D]. 哈尔滨:哈尔滨工业大学,2018.

[7] 张大奎. 用户输入行为在中文分词中的应用研究[D]. 北京:北京理工大学,2018.

[8] 王永民. 汉字编码研究用 40 年巡礼[J]. 中国发明与专利,2018,15(12):6-14.

[9] 张　琪. 基于和弦原理的二键笔画虚拟键盘设计与可用性研究[D]. 杭州:浙江理工大学,2019.

[10] KOMIYA K, NAKAJIMA T. A new Japanese input method for virtual reality applications[C]//20th international conference on human-computer interaction. Las Vegas: HCI, 2018:43-55.

[11] 莫礼平,周恺卿. 方块苗文键盘输入音形编码的优化[J]. 吉林大学学报:工学版,2019,49(2):656-663.

[12] 杨　丹,刘汉明. 计算机汉字输入法对中国文化传承的影响[J]. 考试周刊,2019(31):137-138.

[13] MARCUS G. Deep learning: a critical appraisal[J]. arXiv: 1801.00631,2018.

(下转第 179 页)