



HAL
open science

Systèmes de saisie de texte pour les personnes avec une déficience motrice

Jean-Yves Antoine, Mathieu Raynal

► **To cite this version:**

Jean-Yves Antoine, Mathieu Raynal. Systèmes de saisie de texte pour les personnes avec une déficience motrice. 2022. hal-03661895

HAL Id: hal-03661895

<https://hal.science/hal-03661895>

Submitted on 8 May 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Systemes de saisie de texte pour les personnes avec une déficience motrice : Comment les systemes de prédiction linguistique contribuent à améliorer les performances de saisie?

Jean-Yves Antoine¹ , Mathieu Raynal²

¹Université de Tours Tours, Jean-Yves.Antoine@univ-tours.fr

²Université de Toulouse - IRIT, mathieu.raynal@irit.fr

Introduction

Avec l'utilisation croissante de dispositifs numériques (ordinateurs de bureau, portables, smartphones, tablettes, etc.), la saisie numérique de texte ou de données est devenue une tâche prépondérante dans notre quotidien, que ce soit pour travailler (rédaction de documents) ou pour communiquer (mails, réseaux sociaux, etc.). Pour cela, nous avons pris l'habitude d'utiliser un clavier. Lorsque le dispositif que nous utilisons ne dispose pas de clavier physique, ce dernier est généralement remplacé par un clavier dit logiciel : c'est-à-dire un clavier affiché à l'écran avec lequel l'utilisateur peut interagir soit au travers d'un écran tactile soit au moyen d'un dispositif de pointage.

Pour les personnes ayant une déficience motrice des membres supérieurs, le seul moyen de saisir sur un appareil informatique est le clavier logiciel qu'ils peuvent manipuler avec un dispositif de pointage adapté à leurs capacités motrices (joystick, système de suivi du regard, etc.). Pour les personnes ayant une très faible motricité, il est également possible de remplacer le dispositif de pointage par un système de défilement automatique où le curseur se déplace de touche en touche et que l'utilisateur peut arrêter sur le caractère souhaité en utilisant un contacteur.

Dans ces différentes situations de handicap, la saisie sur clavier logiciel se révèle bien plus lente que ce qu'elle peut être avec un clavier physique. Cependant, pour augmenter la vitesse de saisie, les claviers logiciels ont généralement recours à des systèmes de prédiction linguistique pour faciliter l'accès aux caractères suivants ou proposer les mots à venir. Dans cet article, nous dressons un état des lieux des travaux existants sur l'intégration des systèmes linguistiques dans les claviers logiciels et la manière dont les résultats issus de ces systèmes sont exploités par le clavier logiciel.

1 Couplage d'un système de prédiction à un système de saisie de texte

1.1 Utilisation de la prédiction de mots

La technique la plus couramment utilisée pour accélérer la saisie de texte est de présenter une liste des mots les plus probables que l'utilisateur est en train de saisir. Si le mot que l'utilisateur souhaite saisir se trouve dans la liste, il peut alors le sélectionner et ainsi lui éviter de taper la fin du mot au clavier. Si le mot désiré n'est pas dans la liste, l'utilisateur continue d'entrer les caractères au clavier et le système met à jour la liste des mots en fonction du préfixe saisi. L'objectif est de prédire le mot que l'utilisateur souhaite saisir. Comme annoncé dans [9], les listes de prédiction de mots visent à accélérer la vitesse de saisie de texte en réduisant le nombre de clics nécessaire pour saisir le même texte sur un clavier ordinaire. Cela peut également réduire le nombre d'erreurs que l'utilisateur peut commettre au cours de sa saisie.

Les listes de mots peuvent être utilisées avec tout type de système de saisie de texte et quelle que soit la modalité d'interaction utilisée. Par exemple, le système Sibylle [23] est un système basé sur le défilement d'un curseur qui se déplace automatiquement d'une touche à une autre. La liste de mots proposés est intégrée au système de manière à ce que le curseur passe également par chaque touche liée aux différents mots. Le système HandIAS [13] est quand à lui un clavier logiciel augmenté d'une liste de mots prédits. Cette liste est

positionnée de manière fixe à gauche du clavier. Le clavier PoBox [12] a lui été conçu pour une utilisation sur appareil mobile. Le clavier est affiché sur l'écran tactile de l'appareil et l'utilisateur peut utiliser ce clavier en utilisant un stylet. Ce clavier a comme originalité de proposer la liste de mots toujours au plus près de la dernière touche saisie par l'utilisateur. Cette technique a l'avantage de réduire la distance entre la saisie de l'utilisateur et la liste de mot, mais présente l'inconvénient de cacher une partie du clavier lorsque la liste est affichée.

Dans tous ces systèmes, l'utilisateur ne peut sélectionner dans la liste qu'un mot entier. Par conséquent, tant que le mot désiré n'est pas dans la liste, l'utilisateur est obligé de continuer de saisir le préfixe de son mot sur le clavier pour que le système de prédiction mette à jour la liste des mots et ce jusqu'à ce que le mot souhaité soit affiché dans la liste. Dans ce cas de figure, l'utilisateur perd du temps à saisir son mot, du fait de la recherche visuelle du mot dans la liste qui n'abouti pas à la saisie du mot dans la liste. Pour remédier à ce problème, le système WordTree [4] propose une liste de mot où l'utilisateur peut sélectionner qu'une partie du mot qu'il souhaite saisir. Cette technique permet ainsi à l'utilisateur de sélectionner une chaîne de caractères dans la liste sans que celle-ci soit un mot complet. Ainsi cela réduit le nombre d'actions à réaliser pour saisir le mot.

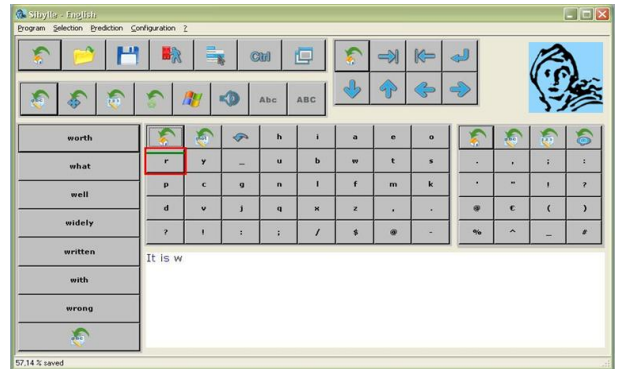


Figure 1 – Sibylle avec le pavé central qui permet de saisir les caractères alphabétiques, et la liste de mots prédits à gauche. Le curseur, rectangle rouge, se déplace automatiquement de touche en touche

1.2 Utilisation de la prédiction de caractères

Les systèmes de saisie utilisant les résultats des algorithmes de prédiction de caractères sont moins répandus que ceux basés sur la prédiction de mots. Ils sont surtout utilisés dans le cas des claviers logiciels utilisant le défilement automatique du curseur. Dans ce cas précis, le curseur se déplace de touche en touche à intervalle régulier, généralement de gauche à droite et de bas en haut sur le clavier. Après chaque caractère saisi, le curseur est toujours repositionné dans le coin supérieur gauche du clavier. L'utilisateur doit alors attendre que le curseur arrive sur le caractère souhaité pour le valider au moyen d'un contacteur. L'algorithme de prédiction de caractères est donc utilisé pour ré-agencer les caractères sur le clavier en fonction de leur probabilité d'être le prochain caractère à saisir. Ainsi, les caractères les plus probables sont positionnés au plus près du coin supérieur gauche pour limiter le nombre de déplacements que devra effectuer le curseur.

Dans le cas des claviers logiciels utilisés avec un dispositif de pointage pour déplacer un pointeur, le ré-agencement des caractères n'est pas envisageable car, à la différence du défilement automatique, le pointeur n'est pas repositionné après chaque caractère. Un ré-agencement des caractères imposerait à l'utilisateur un temps de recherche visuel trop important entre la saisie de chaque caractère. L'utilisateur ne pourrait alors plus anticiper les déplacements à effectuer pour aller d'un caractère à un autre. Ce problème n'est pas présent avec un clavier à défilement automatique car l'utilisateur est dépendant du défilement automatique du curseur et ne peut donc pas anticiper la saisie des caractères. Ce temps de défilement peut également lui permettre de regarder la nouvelle disposition sans que cela n'ait d'incidence sur le temps de saisie. Néanmoins, il a été montré avec le système SlideKey [18] que même dans le cadre d'un défilement automatique du curseur, si le système propose un mécanisme de pré-visualisation des résultats à venir, l'utilisateur pourrait valider plus rapidement le caractère suivant lorsque celui-ci se trouve être le plus probable. Le ré-agencement des caractères a montré son efficacité dans un cas spécifique de clavier logiciel manipulé avec un pointeur. Il s'agit des claviers dit « ambigus », tel que le clavier type T9, où plusieurs caractères sont présents sur une même touche. Les

différents caractères d'une même touche sont alors accessibles, selon leur position sur la touche, par un, deux ou trois clic(s) sur la touche concernée. Le ré-agencement des caractères est alors bénéfique car d'une part, l'utilisateur n'a que les 3 ou 4 caractères de la touche concernée à regarder ; et d'autre part, le caractère le plus probable sera alors placé en première position sur la touche et sera donc accessible en un seul clic. Cette technique réduit donc le nombre de clic à effectuer et permet ainsi à l'utilisateur de gagner du temps dans sa saisie [21].

Les résultats des algorithmes de prédiction de caractères ont toutefois fait l'objet d'étude sur les claviers logiciels manipulés avec un dispositif de pointage. Magnien et al. [11] ont notamment utilisé la prédiction de caractères pour mettre en évidence les caractères qui ont le plus de chance d'être saisis. Ce système est à destination des utilisateurs qui découvrent une disposition nouvelle de caractère. L'objectif du système est alors d'attirer l'attention de l'utilisateur vers les caractères qui ont le plus de chance d'être saisis.

Mais l'intérêt principal de la prédiction de caractères est de faciliter la phase de sélection du caractère souhaité. D'après la loi de Fitts [6], le temps nécessaire pour pointer une cible est dépendant de la distance pour atteindre cette cible et de la taille de cette dernière. Cette loi est notamment utilisée pour prédire les performances théoriques que pourrait avoir un utilisateur avec un clavier donné [20], et a été utilisée à de nombreuses reprises pour proposer des claviers avec des dispositions de caractères optimisées pour une saisie au doigt ou avec un pointeur (voir [15] pour une proposition de disposition de caractères optimisée pour le français). D'après la loi de Fitts, il est donc possible d'améliorer la tâche de pointage de 2 manières : soit en réduisant la distance à parcourir, soit en augmentant la taille de la cible. Le but est donc d'utiliser les résultats de la prédiction de caractères pour soit diminuer la distance à parcourir pour atteindre le caractère souhaité, soit augmenter la taille des touches contenant les caractères les plus probables. Le système le plus connu qui utilise la prédiction de caractères est le clavier Dasher [24] : tous les caractères sont affichés verticalement sur la droite de l'écran, et défilent de droite à gauche. Pour sélectionner un caractère, l'utilisateur déplace un pointeur sur l'axe vertical au centre de l'écran. Au fur et à mesure de la sélection des caractères, les caractères apparaissent à nouveau sur la droite du clavier. Les résultats de l'algorithme de prédiction de caractères permet de modifier la taille des touches en fonction de la probabilité de saisir le caractère associé à celle-ci. Les caractères les plus probables sont donc sur des touches plus grandes et donc plus facilement sélectionnables. Cependant le défilement automatique de droite à gauche des caractères tout au long de la saisie rend ce système fatiguant pour l'utilisateur car il nécessite une attention importante tout au long de la saisie avec ce dispositif. Les claviers BigKey [1] et FloodKey modifient également la taille des touches en fonction de la probabilités d'être saisi des caractères qui y sont associés. Cependant, les touches agrandies par le système BigKey recouvrent en partie les touches voisines. Ceci peut alors engendrer une gêne lors de la sélection de ces touches. De même, les modifications de forme et de taille des touches réalisées par le clavier FloodKey sont apparues trop perturbantes pour les utilisateurs [3]. Le système SpreadKey [14] a l'avantage de conserver la forme initiale du clavier tout au long de la saisie. En revanche, un caractère peut se « propager » sur les touches qui l'entourent lorsque sa probabilité d'être saisi est bien plus importante que celle des caractères des touches voisines. Dans le cas où un caractère est propagé sur les touches voisines, celui-ci peut être saisi sur sa touche d'origine, mais également sur les touches sur lesquelles il s'est propagé. Cependant, chaque caractère reste présent et accessible sur sa touche d'origine quelle que soit sa probabilité d'être saisi. Dans ce cas, le caractère initial est affiché en second plan dans le coin supérieur gauche de la touche et est sélectionnable via un tap long ou un geste du pointeur en direction du coin supérieur gauche de la touche. Par conséquent, même si la disposition des caractères changeait, l'utilisateur pouvait quand même effectuer le mouvement qu'il avait anticipé, et réajuster la fin du déplacement du pointeur si le caractère souhaité était propagé sur les touches voisines.

L'autre solution consiste à réduire la distance à parcourir pour atteindre le caractère suivant. Le système KeyGlass [16] propose 4 touches supplémentaires après chaque caractère saisi par l'utilisateur (cf. figure 2). Les 4 caractères les plus probables sont associés à ces nouvelles touches. Les évaluations de ce système ont mis en évidence que les utilisateurs utilisaient bien ces touches supplémentaires (dans 96% des cas où le caractère est proposé sur une des touches supplémentaires), et leur utilisation a bien permis de réduire les distances

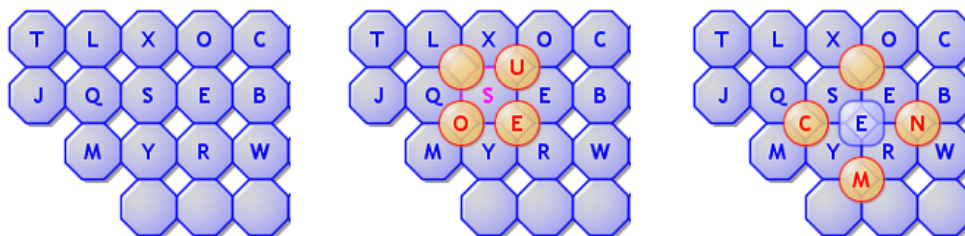


Figure 2 – Clavier pour la version 2 des KeyGlasses : A. clavier simple ; B. KeyGlasses qui apparaissent après la saisie du 's' ; C. KeyGlasses qui apparaissent après la saisie du 'e' qui était placé sur une KeyGlass. Extrait de [17]

de 53,8%. En revanche, cette diminution des distances parcourues n'a pas suffi pour augmenter la vitesse de saisie de texte (augmentation du temps pour saisir un mot de 20,1%). Le fait de leur proposer des caractères supplémentaires les obligent à prendre connaissance de ces nouveaux caractères, et par conséquent les coupent dans leur dynamique, leur anticipation impactant alors négativement leur performance. Cependant, le système a été apprécié des utilisateurs car il permet de minimiser les distances à parcourir et par conséquent réduit la fatigue liée à l'utilisation du clavier logiciel.

Enfin, le clavier sémantique se base sur le principe du pointage sémantique pour améliorer la vitesse de saisie. Le pointage sémantique [5] consiste à moduler le rapport entre le déplacement à l'écran et celui dans l'espace physique en fonction de ce qui se trouve sous le pointeur. S'il n'y a pas d'éléments pertinents sous le pointeur, le déplacement du pointeur est accéléré, et inversement, si le pointeur se trouve sur un objet d'intérêt, sa vitesse de déplacement sera réduite. L'objectif est de faciliter le déplacement d'une cible à une autre et d'améliorer la précision de pointage d'un objet. Ce principe a été repris pour le clavier sémantique en donnant un poids à chaque touche en fonction du caractère qui y est associé et de sa probabilité d'être saisi. La vitesse de déplacement du pointeur sur le clavier est alors liée au poids de chaque touche : plus la touche a un faible poids, plus le pointeur passe rapidement dessus, et plus la touche a un poids élevé, plus le pointeur se déplace lentement sur la touche. L'objectif est ainsi de passer rapidement sur les touches qui ont peu de chance d'être sélectionnées et de faciliter le pointage sur celles dont le caractère a une forte probabilité d'être saisi. Le pointage sémantique a l'avantage de ne pas modifier l'apparence du clavier. Ainsi, par rapport aux autres solutions proposées précédemment, l'utilisateur n'est pas perturbé par les changements dynamiques effectués sur le clavier.

2 Les systèmes de prédiction linguistique

Que l'on parle de prédiction de caractères ou de prédiction de mots, le rôle de la prédiction linguistique est le même : estimer la probabilité d'occurrence d'un item langagier en fonction de ceux déjà saisis par l'utilisateur. Cette prédiction fait appel à un modèle de langage qui est estimé, par apprentissage automatique, sur de très grands corpus.

2.1 Modèles de prédiction

Dans le domaine de l'aide technique au handicap comme pour la prédiction sur smartphone, on utilise généralement des modèles stochastiques de type N-gram compilés sous la forme de transducteurs à états finis [4]. Une fois entraîné, le modèle fournit ainsi la probabilité de l'item linguistique i_k connaissant les $(n-1)$ items de même nature qui le précèdent dans le texte saisi : $Pr(i_k | i_{k-1} \dots i_{k-(n-1)})$.

Dans le cas de la prédiction de lettres, l'utilisation d'un pentagramme ($N=5$) est déjà suffisante. Un étude sur le français a par exemple montré qu'ainsi, la lettre recherchée sur un clavier dynamique à 65 caractères permet de présenter la lettre attendue en position moyenne de 2,9 (i.e. ce caractère est en moyenne toujours dans les trois premiers), la maximum théorique de prédiction étant de 2,7[19]. Les modèles de prédiction de lettres sont très peu gourmands en terme d'occupation mémoire et de temps de prédiction, du fait du nombre limité de caractères. A l'opposé, la prédiction de mots fonctionne sur des dictionnaires de plusieurs dizaines de milliers de mots, ce qui pose le problème de la dimensionnalité des modèles. Le recours à des techniques de compression mais aussi de repliement (*backoff*) pour lutter contre l'éparpillement des données d'apprentissage permet néanmoins d'utiliser là encore des 5-grams de manière efficace. Notons toutefois que les gains de performances en prédiction entre un 3-gram et un 5-gram restent modérés. L'utilisation de techniques d'apprentissage profond pour la prédiction de mots sur dispositifs mobiles faire également face au problème de l'obésité des modèles construits. Une première solution peut être de contourner cette difficulté en déportant les calculs sur un serveur distant, voire de faire appel à des modèles distribués sur le *cloud* [8]. Cette approche est peu appropriée à l'aide au handicap, où la prédiction doit pouvoir opérer en toutes circonstances, en particulier hors ligne. La compression de réseaux récurrents de type LSTM peut toutefois permettre le développement d'une prédiction neuronale efficace implantée sur dispositif mobile [25]. L'absence de jeux d'évaluation partagés rend cependant difficile l'évaluation de l'apport de l'apprentissage profond sur la prédiction de texte : les niveaux performances rapportées dans le contexte de l'aide à la saisie de texte sont solides mais restent dans la gamme de résultats observés en prédiction statistique (N-gram). Le temps de calcul rapportés restent par ailleurs de l'ordre du double ou du triple de ceux observés avec des N-grams, sans être a priori pénalisants pour l'utilisateur (de l'ordre de 5 ms par mot).

2.2 Évaluation et optimisation de la prédiction de mots

Deux métriques principales sont utilisées pour évaluation l'aide à la saisie de texte. La première est la vitesse de saisie (par mot ou par caractère), qui fournit une évaluation direct de l'aide apportée par le couple modèle prédiction-clavier logiciel. Lorsque l'on cherche à évaluer l'impact de la prédiction seule, la métrique standard est le taux d'économie de saisie (*KSR* pour *Keystroke Saving Rate*). Celui-ci mesure le pourcentage moyen de saisies de caractères (ou d'appuis sur le clavier logiciel) évitées lorsque l'on utilise les propositions correctes de la prédiction pour compléter la saisie. Cette métrique n'est pas totalement indépendante de la disposition du clavier. En particulier, elle dépend du nombre d'hypothèse lexicales présentées à l'utilisateur. Le plus souvent, la mesure retenue est le KSR_5 qui correspond à une liste de prédiction de 5 mots. De nombreux études ont en effet montré que pour $N=5$, les courbes de *KSR* commencent à adopter un comportement asymptotique. L'aide à la saisie de texte pour personnes handicapées pouvant concerner des personnes aux capacités motrices et langagières extrêmement variés, il n'existe malheureusement pas de jeu de données de référence représentatif qui serait partagé par toute la communauté. Chaque système étant évalué sur un jeu de test spécifique, il est le plus souvent difficile de comparer les mérites de chaque modèle de prédiction. Toutefois, on note que les systèmes de niveau état de l'art présentent des KSR_5 autour de 0,6 pour des langues telles que l'anglais ou le français [25, 22]. Les langues agglutinatives et/ou à forte morphologie flexionnelle (allemand par exemple) sont plus délicates à prédire et donnent parfois lieu à des traitements additionnels spécifiques. Sur ces langues, un KSR_5 approchant les 0.5 constitue déjà un bon niveau de performance [22]. Dans tous les cas, on observe ainsi que la prédiction de mots peut éviter plus de la moitié des saisies. Toutes ces études montrent par ailleurs l'importance de développer des modèles de prédiction propre à l'utilisateur en sus de l'apprentissage standard effectué sur de grands jeux de connaissance génériques. Cette adaptation dynamique se fait de manière active au fil des saisies utilisateurs. L'intégration d'une dimension sémantique explicite, sous forme de plongements de mots ou d'analyse sémantique latente, permet également une focalisation thématique de la prédiction. S'il peut être apprécié qualitativement par l'utilisateur, son impact sur le KSR_5 est toutefois plus limité. Enfin, il est essentiel de noter que le score *KSR* constitue avant tout une estimation théorique de

l'aide que peut apporter la prédiction, puisque son calcul suppose que l'utilisateur choisit systématiquement une prédiction lorsqu'elle est correcte. Les études en situation réelle d'usage montrent que c'est rarement le cas, l'utilisateur manquant le plus souvent une bonne prédiction affichée à l'écran. Un des verrous scientifiques centraux de l'aide à la saisie de texte n'est ainsi pas l'amélioration de la prédiction, qui atteint déjà des niveaux de performances très acceptables, mais celle du couplage ergonomique entre prédiction et clavier logiciel. Un autre enjeu oublié est certainement celui du couplage entre prédiction et correction orthographique.

2.3 Prédiction et correction intégrée

La correction orthographique est une question importante pour l'aide à la saisie de texte, à la fois parce que les utilisateurs sont soucieux de la qualité orthographique des textes qu'ils produisent, mais aussi parce que de nombreuses pathologies s'accompagnent de troubles langagiers associés. Les études menées en situation réelle d'usage montrent généralement une amélioration sensible de la qualité orthographique des saisies avec l'emploi d'une prédiction de mots. Souvent l'utilisateur va avoir recours à la prédiction non pas tant pour accélérer la saisie de texte, mais pour sélectionner un mot dont il espère qu'il soit ainsi orthographiquement correct. Une étude longitudinale sur 12 mois (non encore publiée) réalisée par le laboratoire LIFAT avec l'hôpital de Garches et le centre de rééducation de Kerpape dans le cadre du projet Predict4All (fondation Benetot) va prochainement confirmer cette observation. Toutefois, la prédiction peut se retrouver complètement perdue lorsque le texte déjà saisi comporte de trop nombreuses fautes de frappe. Le recours à une correction orthographique intégrée à la prédiction semble alors une nécessité. Malheureusement, on ne dispose pas à l'heure actuelle de correcteurs suffisamment fiables pour répondre à ce besoin. Une étude [2] menée sur des productions d'enfants paralysés cérébraux ne présentant pas de troubles langagiers a montré que les correcteurs grand public peinaient à corriger la moitié des erreurs présentes. Sur des productions de personnes dyslexiques, ce sont seulement 20% qui ont pu être corrigées. Dans les autres cas, l'erreur était soit mal corrigée, soit non détectées. Ce problème des non détection concerne majoritairement les erreurs non lexicales, où l'erreur orthographique se traduit pas un autre mot de la langue, comme dans l'exemple : *je *mangue une pomme*. Que la personne souffre ou non de trouble dyslexique, ce sont ainsi près de 40% des erreurs qui ne sont pas identifiées. Les modèles actuels de correction, qu'ils procèdent à une correction hors contexte basée sur une distance d'édition, ou à une correction contextuelle à base d'ensembles de confusion [7], ne sont intrinsèquement pas à même de détecter ce type d'erreurs très fréquentes, même chez le grand public. A notre connaissance, l'application de techniques neuronales n'a jamais été abordée dans le domaine de l'aide à la saisie pour personnes handicapées. Il s'agit donc d'une problématique largement ignorée (voir tout de même [10] dans un contexte proche), qui limite pourtant parfois cruellement l'assistance apporté par la prédiction.

REFERENCES

- [1] Al Faraj, K. ; Mojahid, M. ; Vigouroux, N. : Bigkey : A virtual keyboard for mobile devices. Dans International Conference on Human-Computer Interaction, 3–10. Springer, 2009.
- [2] Antoine, J.Y. ; Crochetet, M. ; Arbizu, C. ; Lopez, E. ; Pouplin, S. ; Besnier, A. ; Thebaud, M. : Ma copie adore le vélo : analyse des besoins réels en correction orthographique sur un corpus de dictées d'enfants. Dans TALN 2019. Toulouse, France, 2019. <https://hal.archives-ouvertes.fr/hal-02375246>.
- [3] Aulagner, G. ; François, R. ; Martin, B. ; Michel, D. ; Raynal, M. : Floodkey : increasing software keyboard keys by reducing needless ones without occultation. Dans les actes de la 10^e WSEAS international conference on Applied computer science, 412–417. World Scientific and Engineering Academy and Society (WSEAS), 2010.
- [4] Badr, G. ; Raynal, M. : WordTree : Results of a word prediction system presented thanks to a tree. Dans actes de International Conference on Universal Access in Human-Computer Interaction (LNCS 5616), 463–471. Springer, Berlin, 2009.

- [5] Blanch, R. ; Guiard, Y. ; Beaudouin-Lafon, M. : Semantic pointing : improving target acquisition with control-display ratio adaptation. Dans Proceedings of the SIGCHI conference on Human factors in computing systems, 519–526, 2004.
- [6] Fitts, P.M. : The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6), 381, 1954.
- [7] Golding, A.R. ; Roth, D. : A winnow-based approach to context-sensitive spelling correction. *Machine learning*, 34(1), 107–130, 1999.
- [8] Hard, A. ; Rao, K. ; Mathews, R. ; Ramaswamy, S. ; Beaufays, F. ; Augenstein, S. ; Eichner, H. ; Kiddon, C. ; Ramage, D. : Federated learning for mobile keyboard prediction. arXiv preprint arXiv :1811.03604, 2018.
- [9] Koester, H.H. ; Levine, S. : Effect of a word prediction feature on user performance. *Augmentative and alternative communication*, 12(3), 155–168, 1996.
- [10] Li, A.Q. ; Sbattella, L. ; Tedesco, R. : Polispell : an adaptive spellchecker and predictor for people with dyslexia. Dans *International Conference on User Modeling, Adaptation, and Personalization*, 302–309. Springer, 2013.
- [11] Magnien, L. ; Bouraoui, J.L. ; Vigouroux, N. : Mobile text input with soft keyboards : optimization by means of visual clues. Dans *actes de International Conference on Mobile Human-Computer Interaction*, 337–341. Springer, 2004.
- [12] Masui, T. : POBox : An efficient text input method for handheld and ubiquitous computers. Dans *actes de International Symposium on Handheld and Ubiquitous Computing*, 289–300. Springer, Berlin, 1999.
- [13] Maurel, D. ; Le Pévédic, B. : The syntactic prediction with token automata : Application to handias system. *Theoretical computer science*, 267(1-2), 121–129, 2001.
- [14] Merlin, B. ; Raynal, M. : SpreadKey : Increasing software keyboard key by recycling needless ones. Dans *actes de 10th European conference for the advancement of assistive technology in europe (AAATE 2009)*, 138–143. IOS Press, Amsterdam, 2009.
- [15] Raynal, M. : Claviers GAG : claviers logiciels optimisés pour la saisie de texte au stylet. Dans *les actes de la 18^e Conference on l'Interaction Homme-Machine*, 3–10, 2006.
- [16] Raynal, M. : Le système keyglass : Système d'ajout dynamique de touches sur clavier logiciel. *Traitement Automatique des Langues, Communication assistée*, 48(2), 97–121, 2007.
- [17] Raynal, M. : Keyglasses : semi-transparent keys on soft keyboard. Dans *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility*, 347–349, 2014.
- [18] Raynal, M. ; Martin, B. : Slidekey : Impact of in-depth previews for a predictive text entry method. Dans *International Conference on Computers Helping People with Special Needs*, 363–370. Springer, 2020.
- [19] Schadle, I. ; Antoine, J.Y. ; Le Pévédic, B. ; Poirier, F. : Sybilette, prédiction de lettres pour la communication augmentée, *revue d'interaction homme-machine*. RIHM, *Revue d'Interaction Homme-Machine*, 3(2), 115–133, 2002.
- [20] Soukoreff, W. ; Mackenzie, S. : Theoretical upper and lower bounds on typing speed using a stylus and a soft keyboard. *Behaviour & Information Technology*, 14(6), 370–379, 1995.
- [21] Vigouroux, N. ; Vella, F. ; Truillet, P. ; Raynal, M. : Evaluation of aac for text input by two groups of subjects : Able-bodied subjects and disabled motor subjects. Dans *actes de 8th ERCIM Workshop, User Interface for All*, Vienne, Autriche, 28–29. Springer, Berlin, 2004.
- [22] Wandmacher, T. : Adaptive word prediction and its application in an assistive communication system. Ph.D. thesis, Université de Tours, 2008.

- [23] Wandmacher, T. ; Antoine, J.Y. ; Poirier, F. ; Départe, J.P. : Sibylle, an assistive communication system adapting to the context and its user. *ACM Transactions on Accessible Computing (TACCESS)*, 1(1), 1–30, 2008.
- [24] Ward, D.J. ; Blackwell, A.F. ; MacKay, D.J. : Dasher : A data entry interface using continuous gestures and language models. Dans *actes de 13th annual ACM symposium on User interface software and technology – UIST'00*, 129–137. ACM, New York, 2000.
- [25] Yu, S. ; Kulkarni, N. ; Lee, H. ; Kim, J. : On-device neural language model based word prediction. Dans *Proceedings of the 27th International Conference on Computational Linguistics : System Demonstrations*, 128–131, 2018.